



FastReport .NET Kullanıcı Kılavuzu (WinForms, Mono, WPF, Avalonia, ASP.NET)

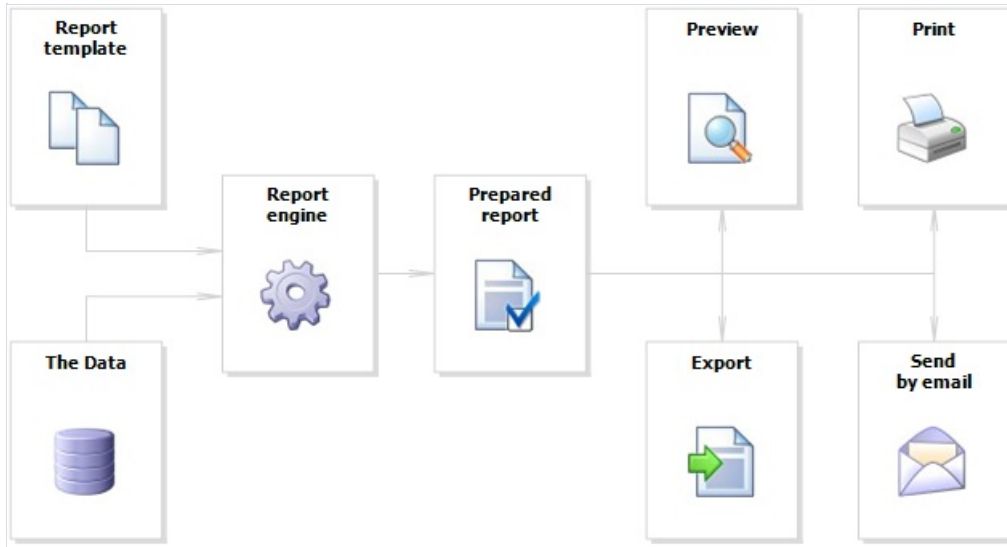
Sürüm 2026.1
© 2008-2025 Fast Reports Inc.

Fundamentals

In this chapter we will learn the principles of working with a report in the FastReport. We will also take a close look at report elements such as report pages, bands, and report objects.

Rapor

Raporla çalışma süreci aşağıdaki gibi özetlenebilir:



Rapor şablonu (bundan böyle "rapor" olarak anılacaktır) tasarımcıda gördüğümüz yapıdır. Raporlar, .FRX uzantılı dosyalarda saklanır. Rapor, tasarımcı kullanılarak veya programatik olarak oluşturulabilir.

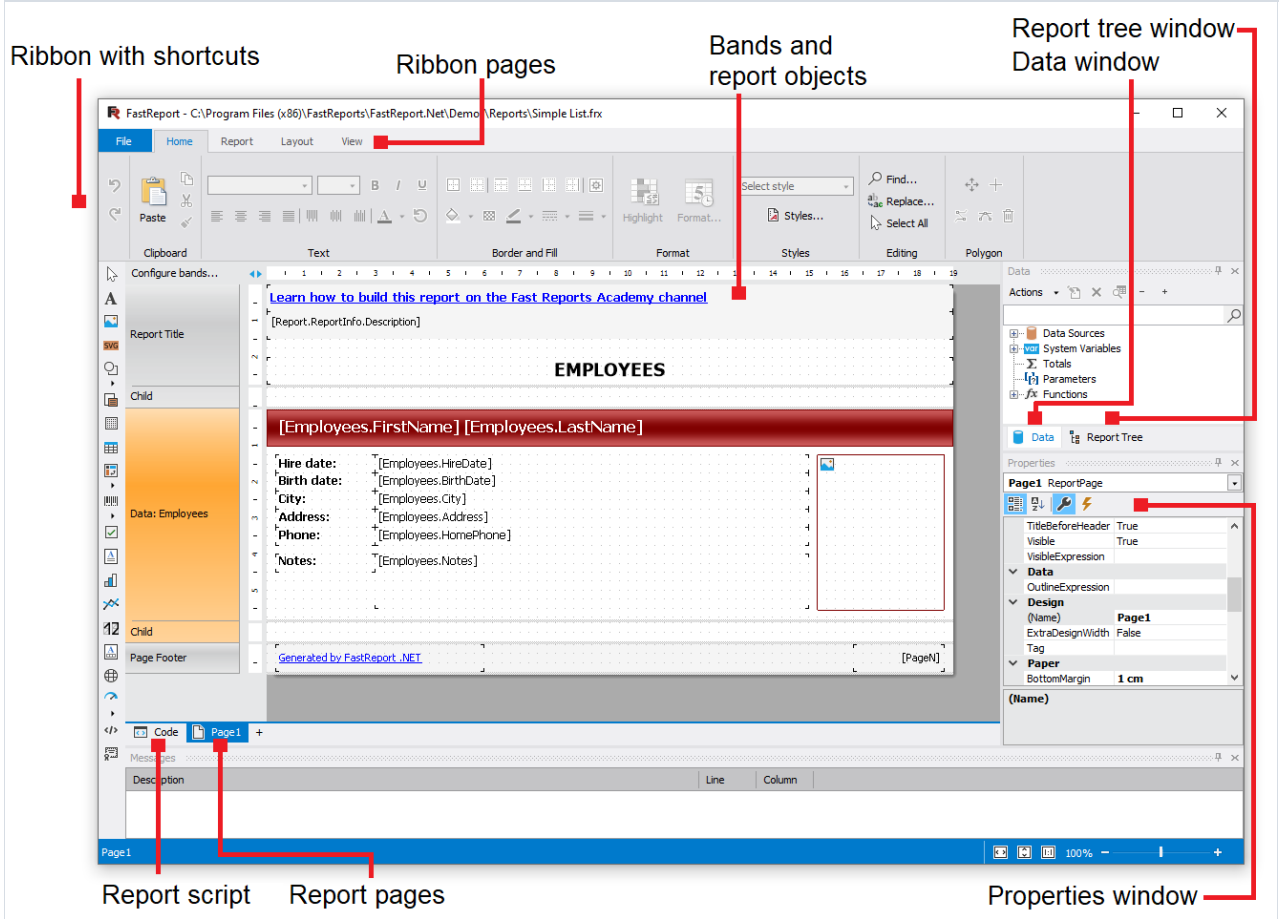
Veriler herhangi bir türde olabilir: bunlar program içinde tanımlanmış veriler veya örneğin MS SQL gibi bir veritabanı yönetim sisteminden alınan veriler olabilir. FastReport ayrıca, iş mantığı nesneleriyle (bundan böyle "iş nesneleri" olarak anılacaktır) de çalışabilir.

Hazır rapor, önizleme penceresinde gördüğümüz çıktıdır. Hazır rapor görüntülenebilir, yazdırılabilir, desteklenen formatlardan birinde (.doc, .xls, .pdf ve diğerleri) kaydedilebilir veya e-posta ile gönderilebilir.

Rapor Tasarımcısı

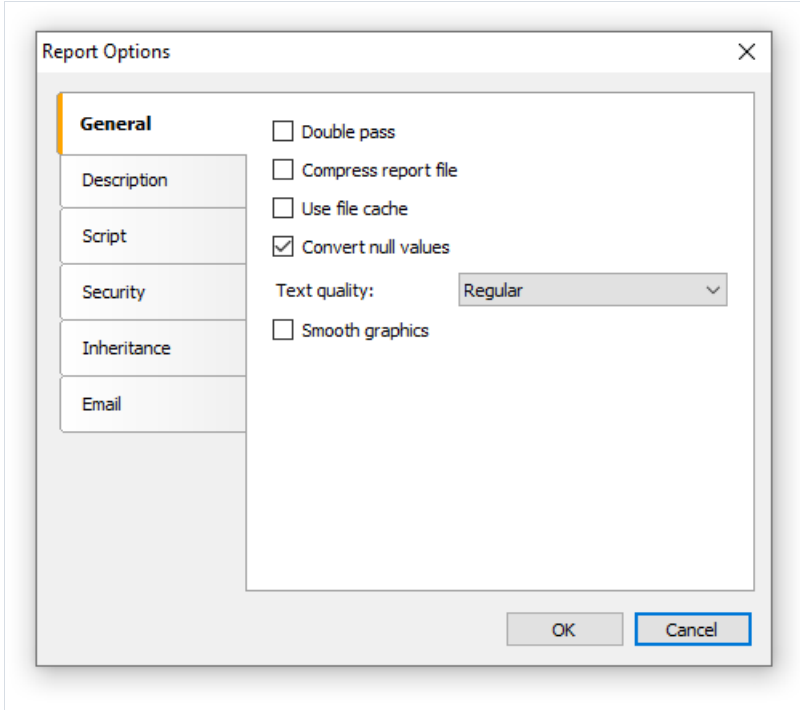
Rapor şablonu oluşturmak için rapor tasarımcısı kullanılır. Tasarımcı, kullanıcıya raporun dış görünümünü geliştirmek için kullanışlı araçlar sunar ve anında önizleme yapma imkânı verir.

Rapor tasarımcısı, FastReport'un ayrılmaz bir parçasıdır ve geliştirme ortamına (örneğin, MS Visual Studio) bağlı değildir. Eğer bir uygulama geliştiriciyseniz, tasarımcıyı kendi uygulamalarınıza entegre edebilirsiniz. Bu, kullanıcılarınıza mevcut raporları değiştirme veya yeni raporlar oluşturma imkânı tanır.



Rapor Ayarları

Rapor ayarları penceresi, menüden "**Rapor | Ayarlar...**" seçeneğiyle açılır. Karşınıza birkaç sekmeden oluşan bir diyalog penceresi çıkar:



- "Genel" sekmesinde raporun aşağıdaki ayarlarını yönetebilirsiniz:
- "Çift geçiş" parametresi, rapor için iki geçişli işlem yapılmasına olanak tanır. Bu, örneğin raporda toplam sayfa sayısına ilişkin bilgi yazdırmak istiyorsanız gereklidir.
- "Rapor dosyasını sıkıştır" parametresi, raporu sıkıştırılmış biçimde kaydetmeye olanak tanır. Sıkıştırma işlemi ZIP algoritmasıyla yapılır, bu nedenle içeriği herhangi bir arşiv programıyla kolayca çıkarabilirsiniz.
- "Dosya ön belleğini kullan" parametresi, rapor oluşturulurken belleğin verimli kullanılmasını sağlar. Çok sayfalı raporlar oluşturuyorsanız bu parametreyi kullanın.
- "Null değerleri dönüştür" parametresi, veri alanlarındaki null (boş) değerlerin varsayılan değerlere (0, boş string, `false` gibi) dönüştürülmesini sağlar.
- "Metin kalitesi" parametresi, rapordaki metnin ekranda nasıl görüntüleneceğini belirler. Bu ayar yazdırmayı etkilemez.
- "Yumuşak grafik çizimi" parametresi, grafik nesneleri (çizgiler, çerçeveler, resimler) çizilirken kenarların yumuşatılmasını sağlar (anti-aliasing).

"Açıklama" sekmesinde raporun açıklamasını belirtebilirsiniz. Bu ayarlar isteğe bağlıdır ve sadece bilgilendirme amaçlıdır.

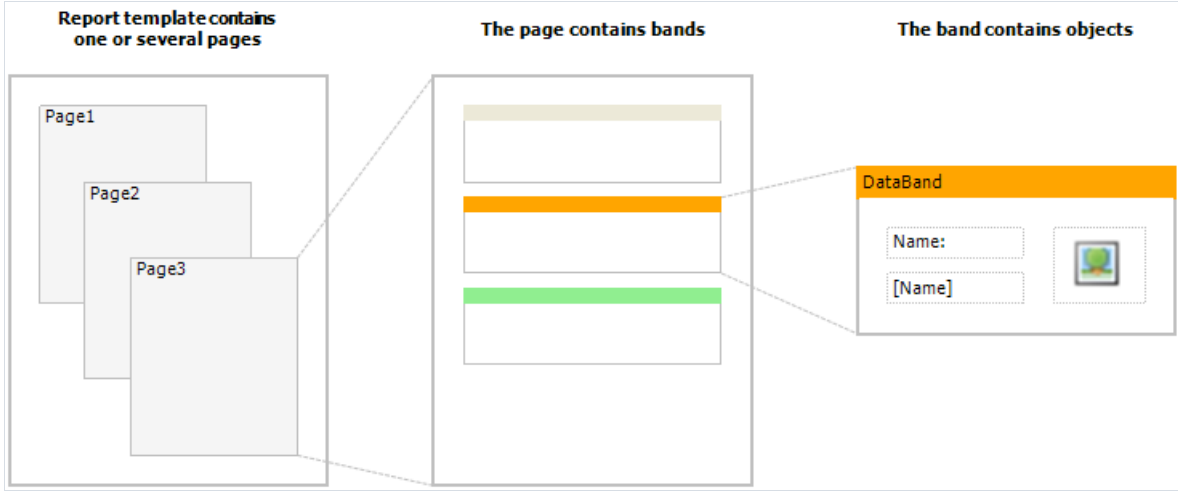
"Betik" sekmesinde geçerli rapor için kullanılacak betik dili seçilir. Betiklerle çalışma detayları "**Betik**" bölümünde açıklanmıştır.

"Güvenlik" sekmesinde rapor açılırken sorulacak bir parola belirleyebilirsiniz. Parola belirlenmiş bir rapor şifreli olarak saklanır, bu yüzden parolayı unutmayın! Unutulursa raporu geri getirmek neredeyse imkânsızdır.

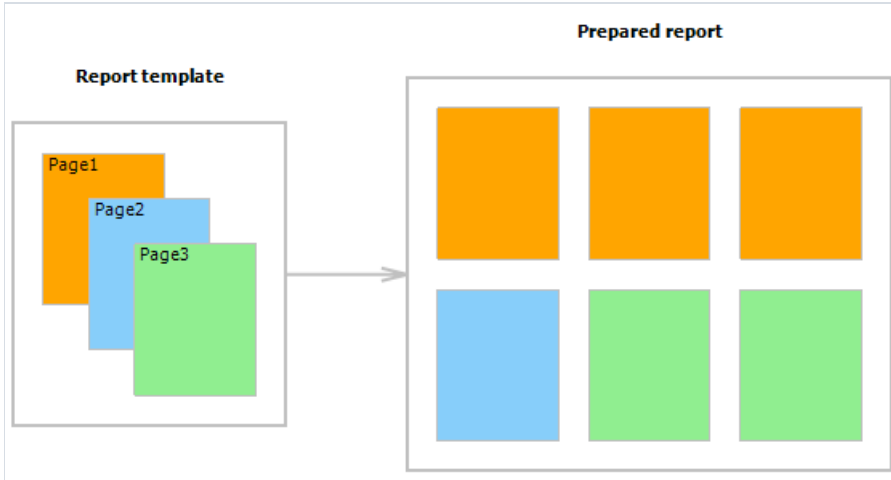
"Kalıtım" sekmesinde raporun kalıtım (miras alma) ayarlarını yönetebilirsiniz. Bu özellik daha sonra ayrıntılı olarak ele alınacaktır.

Rapor Sayfaları

Rapor genellikle bir veya birkaç sayfadan oluşur. Rapor sayfası ise, bantlar (İngilizce: band – şerit) içerir. Bu bantlar üzerinde "Metin", "Resim" gibi rapor nesneleri yer alır:



FastReport'un özelliği, rapor şablonunun birden fazla sayfadan oluşabilmesidir. Örneğin, bir kapak sayfası ve veri sayfası içeren bir şablon oluşturabilirsiniz. Böyle bir rapor oluşturulduğunda, önce ilk sayfa, ardından ikinci sayfa vb. yazdırılır. Şablonun her sayfası, içerdiği verilere bağlı olarak bir veya birden fazla hazır rapor sayfası üretebilir:

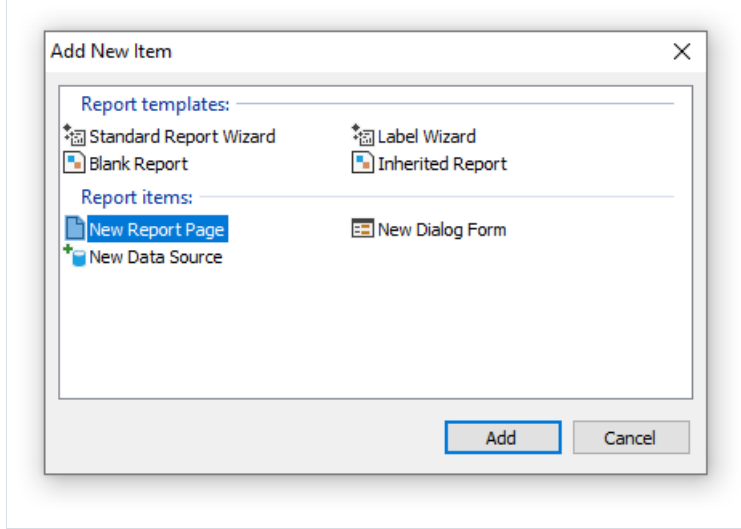


Rapor sayfaları, alt raporlarla (subreport) çalışırken de kullanılır. Diğer rapor oluşturucuların aksine, FastReport'ta alt raporlar, ayrı bir dosyada değil, şablonun ayrı bir sayfasında saklanır.

Bunun yanı sıra, şablon rapor sayfalarının dışında bir veya daha fazla diyalog formu da içerebilir. Diyalog formları, rapor oluşturulmadan önce parametrelerin sorgulanması için kullanılabilir. Diyalogların kullanımı hakkında daha ayrıntılı bilgi, "Diyalog Formları" bölümünde ele alınacaktır.

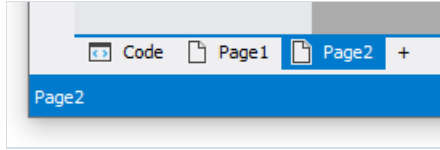
Sayfa Yönetimi

Yeni bir rapor oluşturduğunuzda, zaten birkaç bandı olan bir sayfa içerir. Yeni bir sayfa eklemek için düğmesine tıklayın. Ayrıca, düğmesine tıklayarak ve pencerede "Yeni Rapor Sayfası"nı seçerek de bir sayfa ekleyebilirsiniz.



Benzer şekilde, diyalog formları rapora eklenebilir. Bunun için düğmesini kullanın.

Şablon sayfaları, tasarımcıda sekmeler olarak görüntülenir:



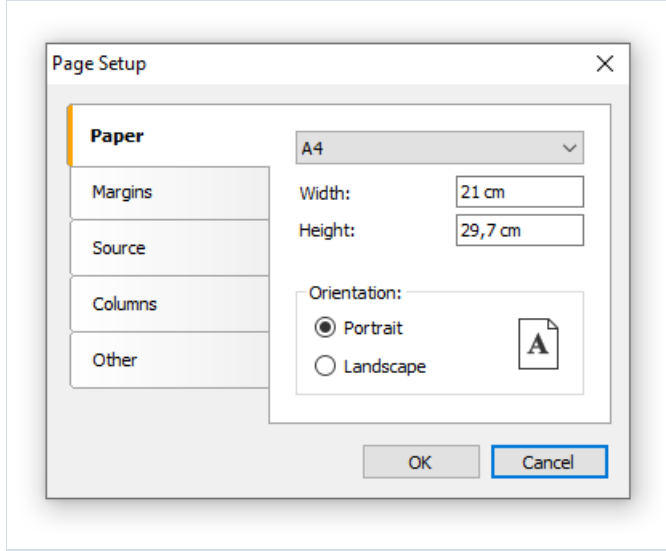
İlk sekme rapor kodudur. Ne taşınabilir ne de silinebilir.

Gerekli sayfaya geçmek için sekmesine tıklamanız yeterlidir. Sayfaların sırasını değiştirmek fare yardımıyla yapılabilir. Bunun için sekmeye sol tıklayın ve fareyi bırakmadan sekmeyi istediğiniz konuma sürükleyin.

Bir sayfayı silmek için düğmesine tıklayın. Rapor yalnızca bir sayfadan oluşuyorsa, bu düğme aktif değildir.

Sayfa Özellikleri

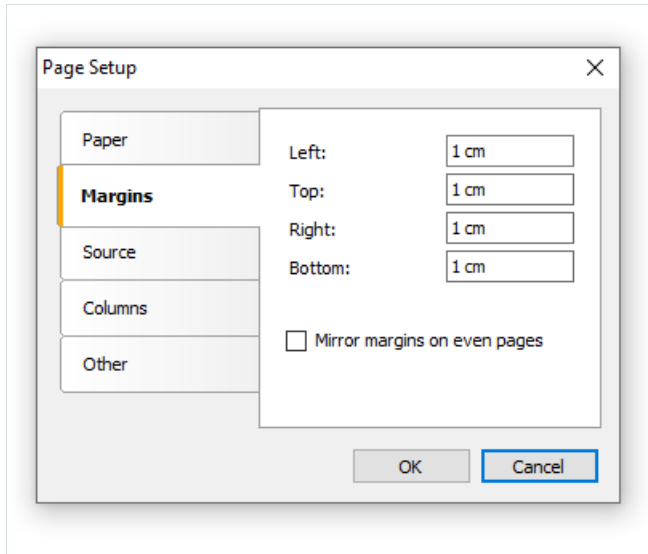
Her rapor sayfası, kâğıt boyutu, yönlendirme (yatay veya dikey), kenar boşlukları, sütunlar, kâğıt kaynağı ve diğer ayarlar gibi kendine özgü ayarlara sahip olabilir. Rapor şablonu, farklı yönlendirme ve kâğıt boyutlarına sahip birden fazla sayfa içerebilir. Sayfa ayarları penceresi, bir düğme yardımıyla veya menüden **"Dosya | Sayfa Ayarları..."** seçilerek açılabilir. Bu pencere, soldaki listeden seçilebilen birkaç ayar grubunu içerir:



"Kâğıt" grubu, sayfa boyutunu ve yönlendirmesini belirlemeyi sağlar. Açılır listeden bir boyut değeri seçilebilir. Bu liste, mevcut yazıcı tarafından desteklenen tüm kâğıt boyutlarını içerir.

Geçerli yazıcı ayarları, **"Dosya | Yazıcı Ayarları..."** menüsünden yapılandırılabilir.

"Kenar Boşlukları" grubu, sayfa kenar boşluklarını belirlemeye yarar. Ayrıca, çıktıdaki çift sayfalar için sol ve sağ kenar boşluklarının yer değiştirmesi gerektiği de burada belirtilebilir:

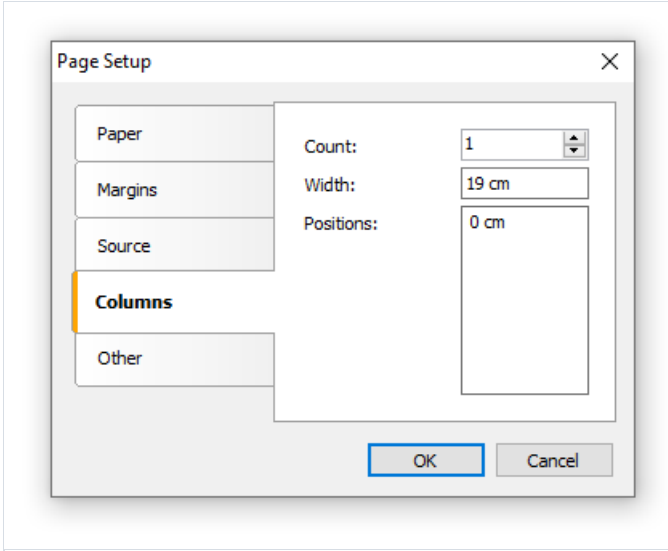


"Kâğıt Kaynağı" grubu, kâğıdın hangi kaynaktan alınacağını seçmenizi sağlar. İlk sayfa ve diğer sayfalar için farklı kaynaklar belirlemek mümkündür.

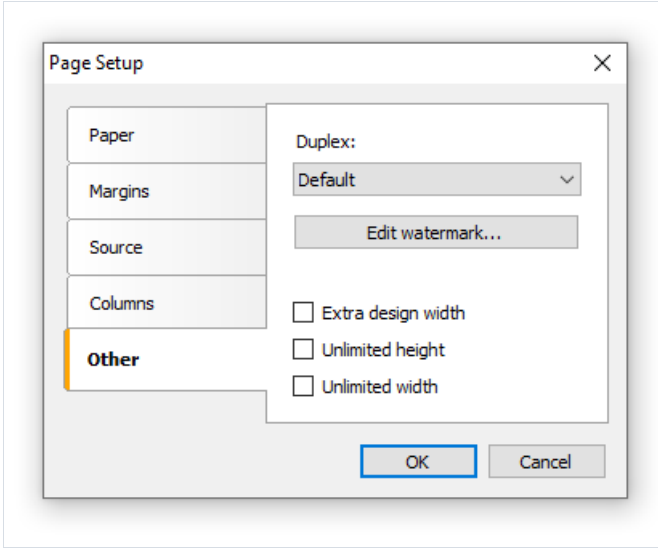
Kâğıt kaynağı, doğrudan yazdırma penceresinde, **"Yazdır"** ekranından da seçilebilir:

"Sütunlar" grubu, çok sütunlu raporlar için sütun parametrelerinin ayarlanmasına olanak tanır. Sütun sayısı

girilmeli ve isteğe bağlı olarak her sütunun genişliği ve konumu düzenlenebilir.



"Diğer" grubu, sayfa için bazı yardımcı özellikleri ayarlamanıza imkân tanır. Eğer yazıcınız destekliyorsa, çift taraflı yazdırma (duplex) modu belirtilebilir. Ayrıca, çıktı sayfalarında yazdırılacak filigran da burada ayarlanabilir.



"Tasarımcıda geniş sayfa" seçeneği, tasarımcı modunda sayfa genişliğini artırmanıza olanak tanır. Bu, özellikle genişleyebilen ve sayfalara bölünebilen "Tablo" ve "Matris" nesneleriyle çalışırken yararlıdır.

Çift taraflı yazdırma modu, yazdırmadan hemen önce, **"Yazdır"** penceresinden seçilebilir.

Bantlar

Bant – rapor sayfasına doğrudan yerleştirilen ve "Metin", "Resim" gibi diğer nesneleri içeren bir kapsayıcı nesnedir.

FastReport'ta toplam 13 bant türü bulunmaktadır. Türüne bağlı olarak, bant raporun belirli bir yerinde yazdırılır:

Bant	Yazılır
Rapor Başlığı	Rapor başlığı, raporun en başında yalnızca bir kez basılır. Yazdırma sırası, "Sayfa Başlığı" bandından önce veya sonra olacak şekilde, sayfa özelliği TitleBeforeHeader aracılığıyla seçilebilir. Bu özellik, "Özellikler" adlı hizmet penceresi kullanılarak değiştirilebilir. Varsayılan ayar true'dur; yani, rapor başlığı, sayfa başlığından önce basılır.
Rapor altbilgisi	Rapor altbilgisi, raporun sonunda, son veri satırından sonra ancak "Sayfa Altbilgisi" bandından önce bir kez basılır.
Sayfa başlığı	Sayfa başlığı, raporun her sayfasının üst kısmında basılır.
Sayfa altbilgisi	Her rapor sayfasının altında basılır.
Sütun başlığı	Bu bant, çok sütunlu rapor yazdırıldığında (sayfa ayarlarında sütun sayısı > 1 olarak belirtildiğinde) kullanılır. Sayfa başlığından sonra, her sütunun üst kısmında basılır.
Sütun altbilgisi	Her sütunun altında, sayfa altbilgisinden önce basılır.
Veriler	Bu bant, veri kaynağına bağlanır ve veri kaynağındaki satır sayısı kadar basılır.
Veri başlığı	Bu bant, "Veriler" bandına bağlanır ve ilk veri satırından önce basılır.
Veri altbilgisi	Bu bant, "Veriler" bandına bağlanır ve son veri satırından sonra basılır.
Grup başlığı	Gruplama koşulunun değeri değiştiğinde, her grubun başında basılır.
Grup altbilgisi	Her grubun sonunda basılır.
Çocuk	Bu bant, herhangi bir banda, hatta başka bir çocuk banta da bağlanabilir. Kendi ebeveyninin hemen ardından basılır.
Arka plan	Her rapor sayfasında arka plan olarak basılır.

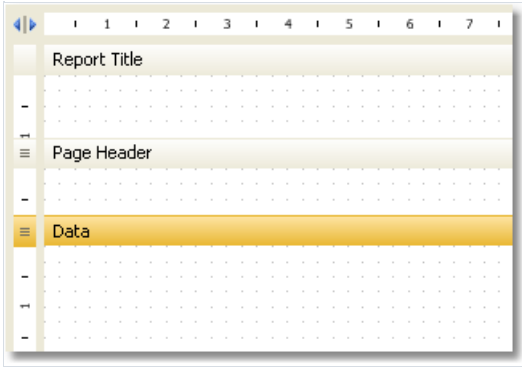
Tasarımcıda Bantların Gösterimi

Tasarımcıda bir bant, dikdörtgen bir alan şeklinde görünür. Bir bant, birçok diğer rapor nesnesi gibi, bir kenarlık ve dolguya sahip olabilir (varsayılan olarak devre dışıdır). Bunun dışında, bir bantta bir ızgara görüntülenir. Izgara modunu ayarlamak için "View|Options..." menüsüne gidin ve "Report page" seçeneğini seçin. Izgara, ayrıca "View" menüsünde de etkinleştirilebilir veya devre dışı bırakılabilir.

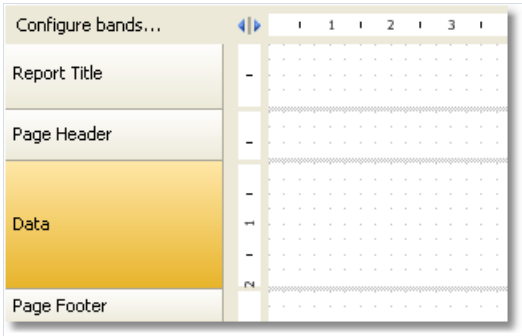
Bir bantın yüksekliğini üç şekilde ayarlayabilirsiniz:

- Fare işaretçisini bantın altına yerleştirin. İmleç şekli "yatay ayırıcı"ya dönüşecek ve bantın boyutunu değiştirebileceksiniz.
- Bant tutamacını sol cetvelde sürükleyin.
- Bantın Yükseklik (**Height**) özelliğini ayarlamak için "Özellikler" penceresini kullanın.

Tasarımcı, her zaman geçiş yapabileceğiniz iki bant görüntüleme moduna sahiptir. İlk modda, her bantın bir başlığı vardır, bu başlık bantın adı ve hakkında yararlı bilgileri içerir (örneğin, bağlandığı veri kaynağının adı).



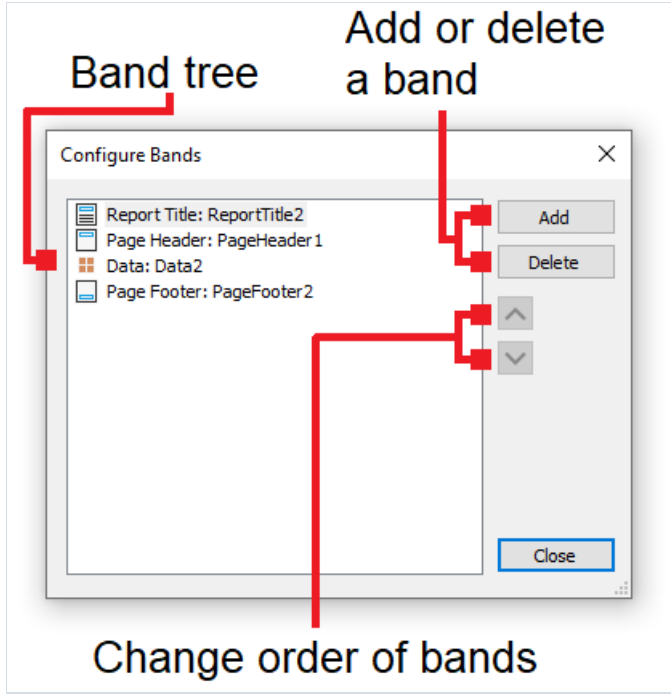
İkinci modda, bantın bir başlığı yoktur. Bunun yerine, pencerenin sol tarafında bantların yapısı görüntülenir. Bu mod, raporun yapısını kolayca anlamanıza yardımcı olur, özellikle de raporu siz oluşturmadıysanız.



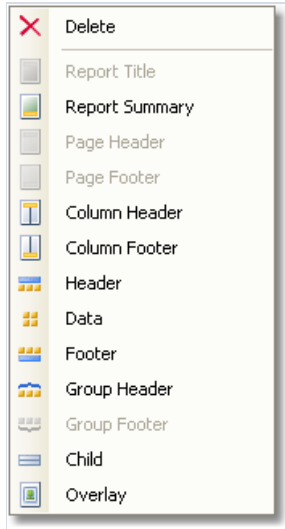
Bu modlar arasında geçiş yapabilirsiniz, bunu butona  tıklayarak yapabilirsiniz.

Bantları Yapılandırma

Bantları, "Configure Bands" penceresinden yapılandırabilirsiniz. Bu pencereye "Report|Configure Bands..." menüsünden veya bantlar ağacının üzerindeki "Configure bands" düğmesi yardımıyla ulaşabilirsiniz.

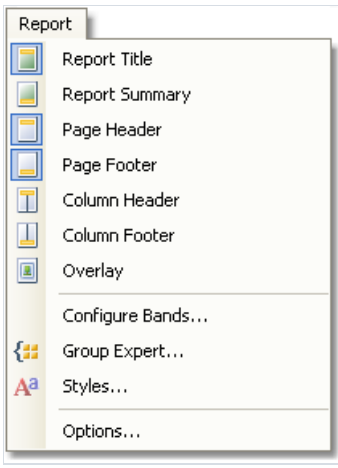


Bu pencerede rapora bant eklemek, bantları silmek veya sıralarını değiştirmek mümkündür. Bir bant eklemek için "Add" düğmesine tıklayın veya bant ağacına sağ tıklayın. Açılan bağlam menüsünde bantların listesi görüntülenecektir. Eklenemeyen bantlar soluk (pasif) şekilde gösterilir.



"Add" işlemi, bantlar ağacında hangi bandın seçildiğine bağlıdır. Örneğin, "Data Header" ve "Data Footer" bantlarını eklemek yalnızca "Data" bandı önceden seçilmişse mümkündür.

Ayrıca bazı bantları yapılandırmanın başka bir yolu da vardır. Bu işlem, "Report" menüsünden yapılabilir:



Bir bandı silmek için, onu seçin ve Delete (Sil) tuşuna basın.

Bantları yapılandırırken, FastReport yanlış bir rapor şablonunun oluşmasına yol açacak işlemleri yapmanıza izin vermez. Örneğin, gruba bağlı olan "Data" bandını silemezsiniz; bunun için önce grubun silinmesi gerekir. Bir diğer örnek, "Data" bandını silerken, başlık ve altbilgisi otomatik olarak silinir. Ayrıca, eğer bir band sayfadaki tek bant ise, o bandı silmek mümkün değildir.

Bantların Yazdırılma Sırası

Sayfada birkaç bant yerleştirilmiş. Peki, FastReport hazırlanan raporu nasıl oluşturacak? Aşağıdaki örneğe bakalım:

Report Title	-	Report Title
Page Header	-	Page Header
Data: Employees	-	Data
Report Summary	-	Report Summary
Page Footer	-	Page Footer

"Report Title" bandı ilk olarak yazdırılacaktır. Ardından "Page Header" bandı hemen yazdırılacaktır. Daha sonra, "Data" bandı, bağlandığı veri kaynağındaki satır sayısı kadar yazdırılacaktır. "Data" bandının tüm satırları yazdırıldıktan sonra, "Report Summary" bandı yazdırılacak ve sayfanın altında "Page Footer" bandı yer alacaktır. Raporun yazdırılması bu şekilde sonlanacaktır. Hazırlanan rapor aşağıdaki gibi görünecektir:

Report Title	Page Header
Page Header	Data
Data	Data
Data	Report Summary
Data	
Data	
Data	
Data	
Data	
Data	
Data	
Page Footer	Page Footer

Yazdırma işlemi sırasında, FastReport, mevcut sayfada bandın yazdırılabilmesi için yeterli alan olup olmadığını kontrol eder. Eğer yeterli alan yoksa, aşağıdaki işlemler gerçekleşir:

- Sayfa altbilgisi yazdırılır;
- Yeni bir sayfa eklenir;
- Sayfa başlığı yazdırılır;
- Önceki sayfaya sığmayan band yazdırılmaya devam edilir.

Bant Özellikleri

Her bant, baskı sürecini etkileyen birkaç faydalı özelliğe sahiptir. Bu özellikleri, bantın bağlam menüsünü kullanarak ayarlayabilirsiniz. Bunun için, diğer nesneler tarafından işgal edilmemiş bantın boş bir alanına sağ tıklayın. Ayrıca, klasik görüntüleme modu etkinse bant başlığına veya aksi halde bant yapısına tıklayabilirsiniz. Bir diğer yöntem ise, bantı seçip "Özellikler" penceresinde ilgili özelliği değiştirmektir.

Özellik	Açıklama
Büyüyebilir, Küçülebilir (CanGrow, CanShrink)	Bu özellikler, bant üzerine yerleştirilen nesnelerin boyutuna bağlı olarak bantın büyüüp küçülebileceğini belirler. Bu özelliğin kullanımı hakkında daha fazla bilgiyi "Rapor Oluşturma" bölümünde bulabilirsiniz.
Bölünebilir (CanBreak)	Bu özellik etkinleştirilirse ve sayfada bantı yazdırmak için yeterli alan yoksa, bant içeriğinin bir kısmını mevcut alana yazdırmaya, yani "bölmeye" çalışılır. Bu özelliğin kullanımı hakkında daha fazla bilgiyi "Rapor Oluşturma" bölümünde bulabilirsiniz.
Yeni sayfa oluştur (StartNewPage)	Bu özelliğe sahip bir bantın yazdırılması yeni bir sayfadan başlar. Genellikle bu özellik, grupları yazdırırken kullanılır; bu durumda her grup yeni bir sayfada yazdırılır.
Sayfanın alt kısmına yazdır (PrintOnBottom)	Bu özelliğe sahip bant, sayfanın en altında, "Sayfa Altbilgisi" bandından önce yazdırılır. Bu, bazı belgelerin yazdırılmasında, sayfanın alt kısmına toplam tutarı yazdırmanın gerektiği durumlarda yararlı olabilir.
Her sayfada tekrar (RepeatOnEveryPage)	Bu özellik, veri ve grup başlıkları ile altbilgisi bantlarında bulunur. Veri yazdırma devam ettiği sürece, bu tür bir bant her yeni sayfanın üstünde yazdırılır. Bu özelliğin kullanımı hakkında daha fazla bilgiyi "Rapor Oluşturma" bölümünde bulabilirsiniz.

Rapor Nesneleri

Raporda geniş bir nesne yelpazesi kullanılabilir:

İkon	Ad	Açıklama
	Metin (TextObject)	Bir veya birden fazla satır metin gösterir.
	Resim (PictureObject)	Resim gösterir.
	"SVG" (SVGObject)	Vektörel resmi gösterir.
	Çizgi (LineObject)	Çizgi gösterir. Çizgi dikey, yatay veya çapraz olabilir.
	Şekil (ShapeObject)	Dikdörtgen, elips, üçgen gibi geometrik şekillerden birini gösterir.
	Kırık çizgi (PolyLineObject)	Farklı sayıda köşe ve segment eğriliği desteğiyle kırık çizgi gösterir.
	Çokgen (PolygonObject)	Farklı sayıda köşe ve segment eğriliği desteğiyle çokgen gösterir.
	Biçimlendirilmiş metin (RichObject)	RTF formatında biçimlendirilmiş metni gösterir.
	Barkod (BarcodeObject)	Barkod gösterir.
	Onay Kutusu (CheckBoxObject)	İki durumu olan – "açık" veya "kapalı" – bir onay kutusu gösterir.
	Tablo (TableObject)	Satırlar, sütunlar ve hücrelerden oluşan tabloyu gösterir.
	Matris (MatrixObject)	Matris (ayrıca özet tablo ya da çapraz tablo olarak da bilinir) gösterir.
	Grafik (MSChartObject)	Grafiği gösterir.
	Posta Kodu (ZipCodeObject)	Posta kodunu gösterir.
	Hücrelerdeki Metin (CellularTextObject)	Her karakterin ayrı bir hücrede yazdırıldığı metni gösterir.
	Dijital İmza (DigitalSignatureObject)	Elektronik imza için kullanılabilecek görünür bir alanı gösterir.

Nesneler hem bilgi görüntülemek için ("Metin" nesnesi) hem de bir raporu tasarlamak için ("Resim", "Çizgi", "Şekil" nesneleri) kullanılabilir. "Tablo" ve "Matris" gibi karmaşık nesneler, diğer basit nesneleri içerebilir.

Genel Nesne Özellikleri

Tüm rapor nesneleri, tek bir temel sınıf olan (`ReportComponentBase`) sınıfından türetilir ve ortak bir özellik kümesine sahiptir. Her bir nesneyi incelemenden önce, bu özelliklere göz atalım.

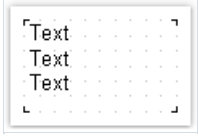
Özelliklerin değerlerini "Özellikler" penceresi aracılığıyla değiştirebilirsiniz. Mümkün olduğunda (örneğin, çerçeve ve dolgu işlemlerinde) araç çubuklarını veya nesnenin bağlam menüsünü kullanmak daha uygundur.

Özellik	Açıklama
Koordinatlar ve Boyutlar (Left, Top, Width, Height)	FastReport'taki rapor nesnesi bir dikdörtgendir. Onun koordinatları (özellikler Left, Top) ve boyutları (özellikler Width, Height) vardır.
Çapa (Anchor)	Bu özellik, nesnenin bulunduğu konteynerin boyutları değiştiğinde, nesnenin konumu ve/veya boyutlarının nasıl değişeceğini belirler. Çapa kullanılarak, nesnenin konteynerle senkronize bir şekilde genişlemesi veya yer değiştirmesi sağlanabilir. Bu özelliğin kullanımı hakkında daha fazla bilgiyi "Nesnelerin Otomatik Yükseklik Ayarı" bölümünde okuyabilirsiniz.
"Yerleştirme" (Dock)	Bu özellik, nesnenin konteynerin hangi kenarına yerleştirileceğini belirler. Bu özelliğin kullanımı hakkında daha fazla bilgiyi "Nesnelerin Otomatik Yükseklik Ayarı" bölümünde okuyabilirsiniz.
Çerçeve ve Dolgu (Border, Fill)	Bu özellikler, sırasıyla çerçeve ve dolgu ayarlarını içerir. Bunları araç çubuklarını kullanarak değiştirebilirsiniz.
"Büyüyebilir", "Daralabilir" (CanGrow, CanShrink)	Bu özellikler, nesnenin tüm metni barındıracak şekilde yüksekliğini otomatik olarak ayarlamasına olanak tanır. Bu özelliklerin kullanımı hakkında daha fazla bilgiyi "Nesnelerin Otomatik Yükseklik Ayarı" bölümünde okuyabilirsiniz.
"Kaydırma" (ShiftMode)	Bu özellik etkinleştirildiğinde, üstünde genişleyen veya daralan bir nesne varsa, nesne aşağıya veya yukarıya kaydırılır. Bu özelliğin kullanımı hakkında daha fazla bilgiyi "Nesnelerin Otomatik Yükseklik Ayarı" bölümünde okuyabilirsiniz.
"Aşağıya Doğru Büyü" (GrowToBottom)	Bu özelliğe sahip nesne, baskı sırasında bantın alt sınırına kadar uzar. Bu özelliğin kullanımı hakkında daha fazla bilgiyi "Nesnelerin Otomatik Yükseklik Ayarı" bölümünde okuyabilirsiniz.
"Kırılabilir" (CanBreak)	Bu özellik, "Metin" ve "Biçimlendirilmiş Metin" adlı metin nesnelerinde bulunur. Nesnenin içeriğinin parçalara ayrılıp ayrılmayacağını belirler. Bu, nesnenin bulunduğu bantta benzer bir özelliğin etkin olması durumunda gerçekleştirilebilir.
"Yazdırılacak..." (PrintOn)	Bu özellik, nesnenin hangi sayfalarda yazdırılabileceğini belirler. Bu özelliğin kullanımı hakkında daha fazla bilgiyi "Çok Sayfalı 'Buket' Raporu" bölümünde okuyabilirsiniz.
İmleç (Cursor)	Bu özellik, nesnenin üzerine gelindiğinde fare işaretçisinin görünümünü belirler. Özellik yalnızca ön izleme penceresinde çalışır.
"Görünür" (Visible)	Bu özellik, nesnenin raporda görüntülenip görüntülenmeyeceğini belirler. Görünmeyen nesne, ön izleme penceresinde görüntülenmez ve yazıcıda basılmaz.

"Yazdırılabilir" (Printable)	Özellik, nesnenin yazıcıda yazdırılıp yazdırılmayacağını belirler. Eğer bu özellik devre dışı bırakılırsa, nesne ön izleme penceresinde görüntülenecektir ancak baskıya gönderilmeyecektir.
Hiperbağlantı (Hyperlink)	Bu özellik, rapor nesnesinin etkileşimli hale getirilmesini sağlar. Hiperbağlantılarla ilgili ayrıntılar, "Etkileşimli Raporlar" bölümünde ele alınacaktır.
"Yer imi" (Bookmark)	Bu özellik, "Hiperbağlantı" özelliği ile birlikte kullanılır. Herhangi bir ifade içerebilir. İfade rapor çalıştırılırken hesaplanır ve değeri yer imi adı olarak kullanılır.
"Kısıtlamalar" (Restrictions)	Bu özellik, nesne üzerinde belirli işlemlere kısıtlamalar getirmeyi sağlar. Varsayılan olarak bu özellik boştur, yani nesne üzerinde tüm işlemler gerçekleştirilebilir.
"Stil" (Style)	Bu özelliğe bir stil adı atanabilir. Bu durumda nesne, stilde belirtilen görünüme sahip olacaktır. Stil parametreleri değiştirildiğinde, nesnenin dış görünümü de buna bağlı olarak güncellenecektir.

"Metin" Nesnesi

"Metin" nesnesi, en sık çalışmanız gereken temel nesnedir. Şu şekilde görünür:

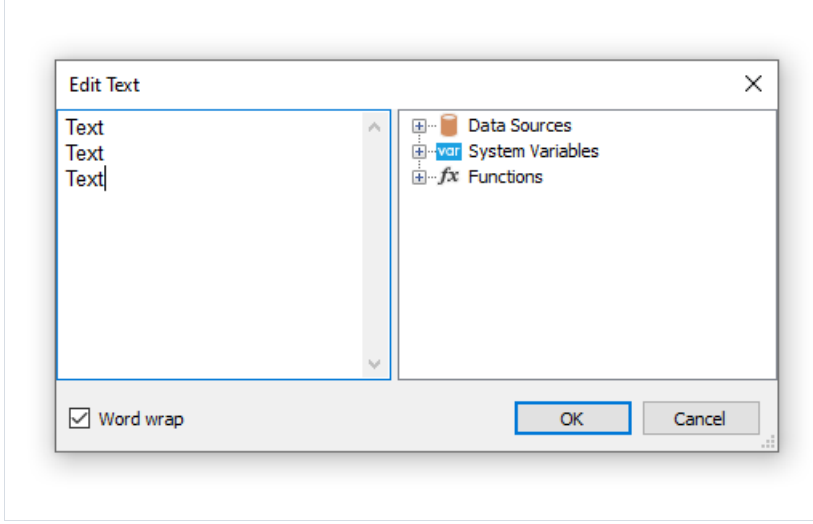


Bir veya birden fazla metin satırını:

- Veri kaynakları alanlarını;
- Sistem değişkenlerini, örneğin "Sayfa Numarası";
- Rapor parametrelerini;
- Toplam değerleri;
- İfadeleri;
- Yukarıda belirtilen öğelerin herhangi bir kombinasyonunu gösterebilir.

Metin Düzenleme

Bir nesnenin metnini düzenlemek için, üzerine çift tıklayın. Metin düzenleyici penceresi açılacaktır:



Düzenleyicinin sağ tarafında, metne eklenebilecek verilerin bulunduğu bir pencere yer alır. Bu veriler, fareyle sürükle-bırak (drag&drop) yöntemiyle metnin istenen yerine eklenebilir. Bir başka ekleme yöntemi ise, veriye çift tıklamaktır — bu durumda, veri imlecin bulunduğu konuma eklenir. \

Değişiklikleri kaydetmek ve düzenleyici penceresini kapatmak için "Tamam" düğmesine tıklayın ya da Ctrl+Enter tuş kombinasyonunu kullanın.

Metni düzenlemenin başka bir yolu da yerinde düzenlemedir (rapor sayfası üzerinde doğrudan). Bunun için "Metin" nesnesini seçin ve Enter tuşuna basın. Düzenlemeyi bitirmek için nesnenin dışına tıklayın veya Ctrl+Enter tuşlarına basın. Esc tuşu yapılan değişiklikleri iptal eder.

Yerinde düzenleme sırasında nesnenin boyutları fare ile değiştirilebilir.

İfade Görüntüleme

"Metin" nesnesi hem normal metin hem de ifadeler içerebilir. Hatta ifadeler metinle karışık halde nesnede bulunabilir. Örneğin:

Bugün [Date]

Böyle bir nesne yazdırıldığında, metinde yer alan tüm ifadeler hesaplanır. Metne ifadenin değeri yerleştirilir, örneğin:

Bugün 12 Eylül 2010

Görüldüğü gibi, ifadeler köşeli parantezler kullanılarak gösterilir. Bu, varsayılan olarak [,] değerini içeren **Brackets** özelliğiyle ayarlanır. Gerekirse farklı bir karakter çifti belirtilebilir, örneğin <,> veya <!,!>. Son durumda, metindeki ifadeler şu şekilde görünecektir:

Bugün <!Date!>

Ayrıca, ifadelerin işlenmesini tamamen devre dışı bırakmak mümkündür – bu, **AllowExpressions** özelliğiyle sağlanır. Bu durumda metin olduğu gibi görüntülenir.

Köşeli parantezler içinde, derleyici açısından geçerli olan herhangi bir ifade yer alabilir. İfadeler hakkında daha fazla bilgi için "[İfadeler](#)" bölümüne bakınız.

Örneğin, aşağıdaki metni içeren bir nesne:

2 * 2 = [2 * 2]

şu şekilde yazdırılır:

2 * 2 = 4

Yaygın bir hata, ifadenin köşeli parantezler dışında yazılmaya çalışılmasıdır. Unutulmamalıdır ki yalnızca köşeli parantez içindeki kısımlar ifade olarak değerlendirilip hesaplanır. Diğer metin olduğu gibi yazdırılır. Örneğin:

2 * 2 = [2] * [2]

Bu metin şu şekilde yazdırılır:

2 * 2 = 2 * 2

İfadeler içinde kendi köşeli parantezlerine sahip elemanlar bulunabilir (bkz. "Rapor verilerine başvurma" bölümü).
Örneğin sistem değişkenlerine başvurular böyle yapılır. Aşağıdaki örnekte bir "Metin" nesnesinin içeriği verilmiştir:

Sonraki sayfa: [[Page] + 1]

Metin, dıştaki köşeli parantezlerle belirtilen [Page] + 1 ifadesini içerir. Page , raporun mevcut sayfa numarasını döndüren bir sistem değişkenidir. Bu değişken kendi köşeli parantezleri içinde yer almalıdır. Bu köşeli parantezler, "Metin" nesnesinin Brackets özelliğine bakılmaksızın **mutlaka kare parantez** olmalıdır.

Teknik olarak, önceki örneklerde sistem değişkeni olan Date 'i yazdırmaya çalışırken çift köşeli parantez kullanmamız gerekirdi:

Bugün [[Date]]

Burada dış köşeli parantezler ifadeyi tanımlar, içteki parantezler ise sistem değişkenine başvuru için zorunludur. Ancak, eğer ifade yalnızca tek bir öge içeriyorsa, FastReport fazladan parantezi atlamaya izin verir.

Veri sütunlarını görüntüleme

Veri sütununu şu şekilde yazdırabilirsiniz:

[Datasource name.Column name]

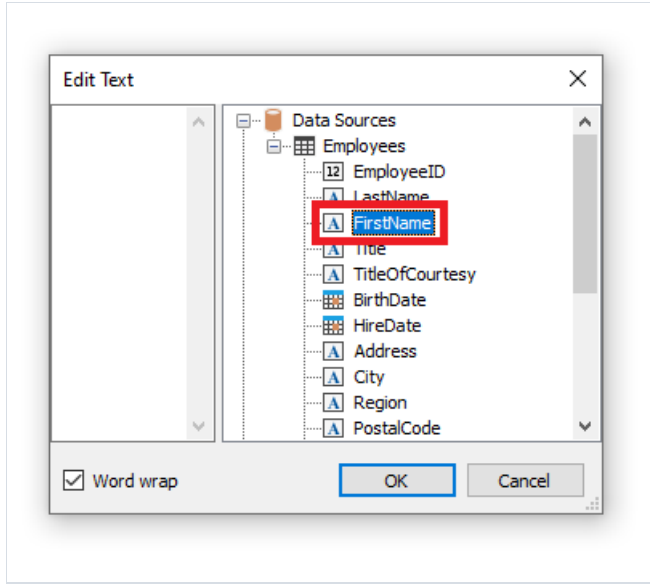
Gördüğünüz gibi, burada köşeli parantezler kullanılmıştır. Veri kaynağı adı ve veri sütunu adı nokta ile ayrılır. Örneğin:

[Employees.FirstName]

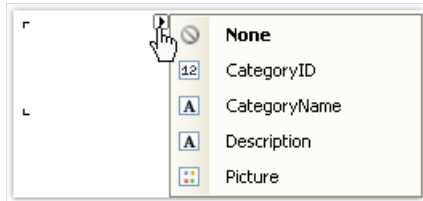
Veri sütunlarını kullanma hakkında daha fazla bilgi için "[İfadeler](#)" bölümüne bakın.

"Metin" nesnesine veri sütunu eklemenin birkaç yolu vardır.

1. "Metin" nesnesinin düzenleyicisinde veri sütununun adını manuel olarak yazarız. Bu yöntem en zahmetli olanıdır çünkü hata yapması kolaydır.
2. Nesnenin düzenleyicisinde gerekli veri sütununu seçeriz ve metne sürükleyip bırakırız:



3. Nesnenin sağ üst köşesindeki küçük düğmeye tıklayın ve listeden veri sütununu seçin:



4. "Veri" penceresinden bir veri sütununu rapor sayfasına sürükleyip bırakın. Bu durumda, sütuna bağlantı içeren bir "Metin" nesnesi oluşturulur.

Metin İçeriğinde HTML Etiketlerinin Desteklenmesi

Daha önce FastReport'ta "Metin" nesnesinde bazı basit HTML etiketlerinin kullanılmasına `HtmlTags` özelliği ile izin verilmekteydi. Ancak `HtmlTags` özelliği, `TextRenderType` adlı yeni bir özellik ile değiştirilmiştir. Bu yeni özellik, yalnızca HTML etiketlerinin işlenmesinden çok daha geniş bir işlevsellik sunmaktadır.

`TextRenderType` özelliği şu üç değerden birine sahip olabilir:

- **Default** – Etiketler dönüştürülmeden düz metin olarak görüntülenir.
- **HtmlTags** – HTML etiketleri uygulanır. Desteklenen etiketler sınırlıdır: `<b>`, `<i>`, `<u>`, `<strike>`, `<br>`, `<sub>`, `<sup>`, `<img>`
- **HtmlParagraph** – Satır aralığı, paragraf girintisi gibi ayarların yanında, `HtmlTags` ile aynı etiketleri destekler.
- **Inline** – Metin, basit biçimde görüntülenir. Yalnızca dahili kullanım içindir, harici kullanıma önerilmez.

Önceki `HtmlTags` özelliği tarafından desteklenen etiketlerden biri `<font>` idi. Ancak bu etiket artık eski kabul edilmektedir ve tüm tarayıcılarda desteklenmesi garanti edilmemektedir. Bu nedenle, belirli CSS stillerinin `<span>` etiketiyle `style` öznetiliği aracılığıyla uygulanmasını sağlayan yeni bir görselleştirme mekanizması geliştirilmiştir.

HTML Etiketlerinin İşlenme Modları

HtmlTags

Daha önce belirtildiği gibi, "Metin" nesnesi aşağıdaki HTML etiketlerini destekler:

- `<b>` – Kalın yazı:

Örnek:

```
<b>FastReport</b>
```

Sonuç:

FastReport

- `<i>` – İtalik yazı:

Örnek:

```
<i>FastReport</i>
```

Sonuç:

FastReport

- `<u>` – Altı çizili yazı:

- Örnek:

```
<u>FastReport</u>
```

Sonuç:

FastReport

- `<strike>` – Üstü çizili yazı:

Örnek:

```
<strike>FastReport</strike>
```

Sonuç:

FastReport

- `<br>` – Satır sonu:

Örnek:

```
Fast<br>Report
```

Sonuç:

**Fast
Report**

- `<sub>` – Alt simge:

Örnek:

```
Fast<sub>Report</sub>
```

Sonuç:

Fast^{Report}

- `<sup>` – Üst simge: Örnek:

```
Fast<sup>Report</sup>
```

Sonuç:

Fast^{Report}

- `<img>` – Metin içine görsel yerleştirme:

Bu etiket yalnızca `src` , `width` , `height` özniteliklerini destekler. Görseller, `http` , `https` veya `base64`

formatlarında metne eklenebilir. `http` ve `https` bağlantıları özellikle web raporlarında geçerlidir.

Örnek:

```

```

Sonuç:



HtmlParagraph

Bu mod, yeni bir HTML işleyici kullanır. `HtmlTags` ile aynı etiketleri desteklemekle birlikte, işleme biçimi farklıdır – özellikle `<sub>` ve `<sup>` etiketlerinde belirgin olur. Ayrıca, `<span>` etiketi de desteklenir.

`<span>` etiketi ile `style` özniteliği kullanılarak metin üzerinde CSS stil ayarları uygulanabilir. Bu sayede, artık modası geçmiş olan `<font>` etiketine gerek kalmadan yazı tipi boyutu, rengi, hizalama ve arka plan gibi biçimlendirmeler yapılabilir.

Kullanım örnekleri:

```
<span style="font-size:20pt;">FastReport</span>  
<span style="color:red;">Fast</span>Report  
<span style="font-family:Consolas;">FastReport</span>  
<span style="background-color:yellow;">FastReport</span>
```

Sonuç:



ParagraphFormat

Bu özellik `HtmlParagraph` ile birlikte kullanılır ve paragrafların görüntülenmesini ayarlar (satır aralığı, paragraf başı girintisi vb.). Aşağıdaki ayarları içerir:

- **FirstLineIndent** – İlk satır girintisi
- **LineSpacing** – Satırlar arası boşluk (santimetre cinsinden)
- **LineSpacingMultiple** – Önceki değeri çarpacek katsayı (sadece Multiple tipi için geçerlidir)
- **LineSpacingType** – Satır aralığı tipi:
 - **Single** – Tekli

- **At least** – En az
- **Exact** – Sabit
- **Multiple** – Katlı

Ayar örneği:

▼ ParagraphFormat	(ParagraphFormat)
FirstLineIndent	1,25 cm
LineSpacing	2,38 cm
LineSpacingMultiple	0,9
LineSpacingType	Multiple

Sonuç:

Lorem Ipsum is simply dummy text of the printing and typesetting industry.
 Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.
 It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

"Metin" nesnesinin özellikleri

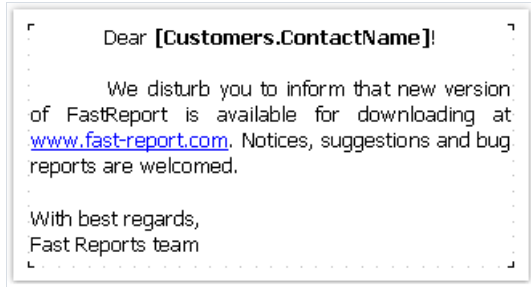
Özellik	Açıklama
Metin içindeki ifadeler (AllowExpressions)	Nesne metnindeki ifadelerin işlenmesini devre dışı bırakma. Varsayılan olarak etkindir.
Açı (Angle)	Metnin dönüş açısı
Otomatik küçültme (AutoShrink)	Tüm metni sığdırmak için yazı tipi boyutunun (Font.Size) veya genişliğinin (FontWidthRatio) otomatik olarak küçültülmesi.
Otomatik küçültme için minimum yazı boyutu (AutoShrinkMinSize)	AutoShrink özelliğinin değerine bağlı olarak, yazı tipi boyutunun veya FontWidthRatio özelliğinin en düşük değeri.
Otomatik genişlik (AutoWidth)	En uzun metin satırını kelime kaydırma olmadan tamamen sığdırmak için nesnenin genişliğinin otomatik olarak ayarlanması.
Ayraçlar (Brackets)	Nesne metninde ifadeleri belirtmek için kullanılan karakter çifti.
Şuraya kadar böl (BreakTo)	Bu özellik sayesinde iki "Metin" nesnesi arasında metnin "taşmasını" organize edebilirsiniz. Diyelim ki A ve B adında iki nesnemiz var. A nesnesi çok miktarda metin içeriyor ve bu metnin bir kısmı nesneye sığmıyor. Eğer A nesnesinin BreakTo özelliğinde B nesnesi belirtilirse, A nesnesine sığmayan metin B nesnesinde yazdırılır.
Kırpma (Clip)	Nesneye sığmayan metnin kırılması. Varsayılan olarak etkindir.
Yinelenenler (Duplicates)	Bu özellik, yinelenen değerlerin nasıl yazdırılacağını belirler. Bu özelliğin kullanımıyla ilgili daha fazla bilgiye "Biçimlendirme" bölümünden ulaşabilirsiniz.
İlk sekme uzaklığı (FirstTabOffset)	Bu özellik, ilk sekme karakterinin kaç piksel kaydırılacağını belirler.
Yazı tipi (Font)	Yazı tipi ayarları.
Yazı tipi en oranı (FontWidthRatio)	Yazı tipinin genişlik ölçekleme katsayısı. Varsayılan olarak bu özellik 1 değerindedir. Yazı tipinin genişliğini artırmak için 1'den büyük bir değer, azaltmak için ise 0 ile 1 arasında bir değer girin.
Değeri gizle (HideValue)	Belirtilen değere eşit olan ifadelerin değerlerini gizleme. Bu özelliğin kullanımıyla ilgili daha fazla bilgiye "Biçimlendirme" bölümünden ulaşabilirsiniz.
Sıfırları gizle (HideZeros)	İfade değerlerinden sıfır olanların gizlenmesi. Bu özelliğin kullanımıyla ilgili daha fazla bilgiye "Biçimlendirme" bölümünden ulaşabilirsiniz.
Koşullu biçimlendirme (Highlight)	Belirli koşullara bağlı olarak nesnenin dış görünümünün değiştirilmesi. Bu özelliğin kullanımıyla ilgili daha fazla bilgiye "Biçimlendirme" bölümünden ulaşabilirsiniz.

HTML etiketleri (HtmlTags)	Kullanımdan kaldırıldı, yerine TextRenderType kullanılıyor.
Metni hizalama (HorzAlign, VertAlign)	Bu özellikler, nesne içindeki metnin yatay ve dikey hizalamasını belirler.
Satır aralığı (LineHeight)	Piksel cinsinden ölçülen satır aralığı. Varsayılan değer = 0 olup, bu durumda varsayılan satır aralığı kullanılır.
Birleştirme modu (MergeMode)	**Yan yana bulunan ve aynı içeriğe sahip metin nesnelerinin birleştirilmesi. Bu özellik 2 değer alabilir: - Horizontal – yatay birleştirme; - Vertical – dikey birleştirme. Bu değerler ayrı ayrı ya da birlikte kullanılabilir. Önemli not: Nesneler birbirine bitişik olmalıdır.**
Null değeri (NullValue)	Null değeri yerine görüntülenecek metin dizesi. Bu özelliği kullanmak için "Rapor Ayarlar..." menüsünden "Null değerleri dönüştür" seçeneğinin işaretini kaldırmanız gerekir.
İç boşluk (Padding)	Metnin, nesnenin kenarlarından olan boşlukları (piksel cinsinden ölçülür).
Paragraf formatı (ParagraphFormat)	Paragrafların görüntülenmesiyle ilgili ayarlar (satır aralığı, paragraf girintisi). Bu özelliğin kullanımıyla ilgili daha fazla bilgiye "Metinde HTML etiketlerinin desteklenmesi" sayfasından ulaşabilirsiniz.
Paragraf ofseti (ParagraphOffset)	Piksel cinsinden ölçülen paragraf kaydırması. TextRenderType.HtmlParagraph için ParagraphFormat.FirstLineIndent özelliğini kullanın.
Sağdan sola yazım yönü (RightToLeft)	Metnin sağdan sola yazdırılması.
Sekme pozisyonları (TabPositions)	Sekme karakterlerinin konumlarını piksel cinsinden belirten küme. Negatif değerlerin bu özellik üzerinde etkisi yoktur.
Sekme genişliği (TabWidth)	Sekme karakterinin genişliğinin piksel cinsinden ayarlanması.
Metin (Text)	Nesnenin metni.
Metin doldurma rengi (TextFill)	Metin rengi. Bu özelliğin düzenleyicisi aracılığıyla mevcut dolgu renklerinden herhangi biri seçilebilir.
Metin dış çizgisi (TextOutline)	Metnin kenar çizgisi.
Metin işleme türü (TextRenderType)	Bu özellik, nesne metninde HTML etiketlerinin kullanılmasına olanak tanır. Bu özelliğin kullanımıyla ilgili daha fazla bilgiye "Metinde HTML etiketlerinin desteklenmesi" sayfasından ulaşabilirsiniz.
Metin kırpma (Trimming)	Bu özellik, nesne sınırlarını aşan metnin nasıl görüntüleneceğini belirler. Yalnızca "Sözcük kaydırma" özelliği devre dışı bırakıldığında kullanılır.
Alt çizgiler (Underlines)	Metnin her satırının altına çizilen alt çizgi çizgileri. Alt çizgi yalnızca üst kenara hizalanmış metinler için kullanılabilir.

Kelime kaydırma (WordWrap)	Bu özellik, metnin sözcük bazında satıra kaydırılıp kaydırılmayacağını belirler.
Wysiwyg	Metnin ekranda ve yazıcıda görüntülenmesi arasında en yüksek uyumu sağlamak amacıyla metin görüntüleme modunun değiştirilmesi. Bu mod, metni iki yana yaslama veya özel satır aralığı kullanıldığında dolaylı olarak etkinleştirilir.

"Rich Text" Objesi

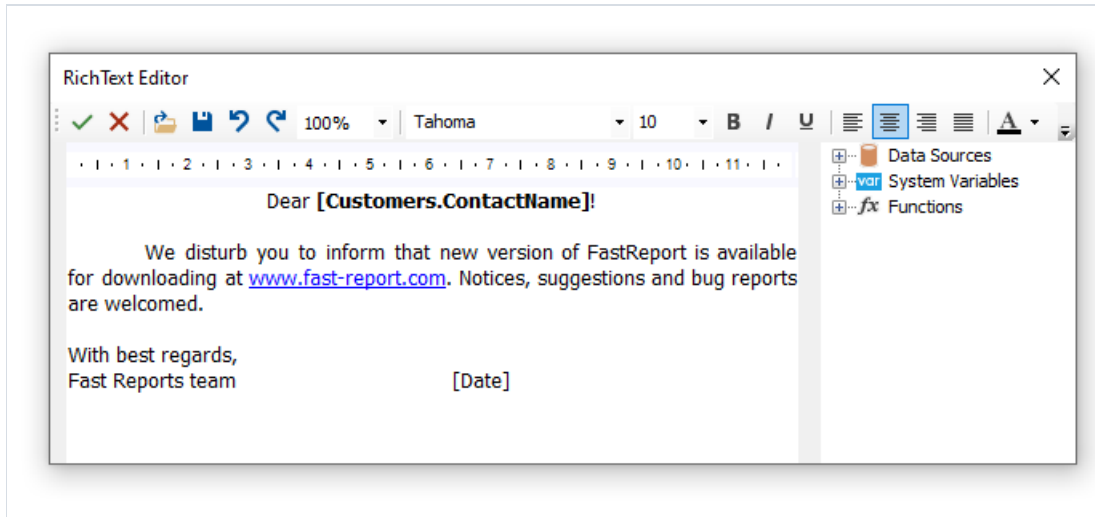
Bu obje, RTF formatında biçimlendirilmiş metin görüntülemeyi sağlar. Şu şekilde görünür:



Mümkünse, metni görüntülemek için "Metin" objesini kullanmaya çalışın. Rapor başka bir formata dışa aktarıldığında, "Biçimlendirilmiş Metin" objesi resim olarak dışa aktarılır.

Objeye yalnızca düz dolgu türünü destekler. Gradyanlı ve diğer dolgular desteklenmez.

Metni düzenlemek için objeye çift tıklayın:



Ayrıca, metin oluşturmak için Microsoft Word paketini kullanabilirsiniz. Metni oluşturduktan sonra, RTF formatında kaydedin. Ardından obje düzenleyicisini çağırın ve RTF dosyasından metni yüklemek için düğmeye tıklayın.

Objeye, Microsoft Word'ün tüm özelliklerini desteklemez.

Objeye, veritabanı alanlarından verileri iki şekilde bağlayabilir:

- Objeye metnine ifadeler ekleyebilirsiniz. Bunlar, "Metin" objesinde olduğu gibi işlenir. Objeye metnine bir alana bağlantı ekleyin.
- Veritabanı alanının içeriğini tamamen yazdırmak için `DataColumn` özelliğini kullanın.

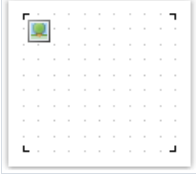
Objenin şu özellikleri vardır:

Özellik	Açıklama
"Metindeki İfadeler" (AllowExpressions)	Objeye metnindeki ifadelerin işlenmesini devre dışı bırakma yeteneği. Varsayılan olarak etkindir.

"Parantezler" (Brackets)	Obje metnindeki ifadeleri belirtmek için kullanılan simgeler çifti.
"Veri Alanı" (DataColumn)	Obje metninin yüklendiği veri alanı.
"Metin" (Text)	Obje metni RTF formatında.
"İçerik Boşluğu" (Padding).	Metnin, objenin kenarlarından olan iç boşlukları, piksel cinsinden ölçülür.

"Resim" Nesnesi

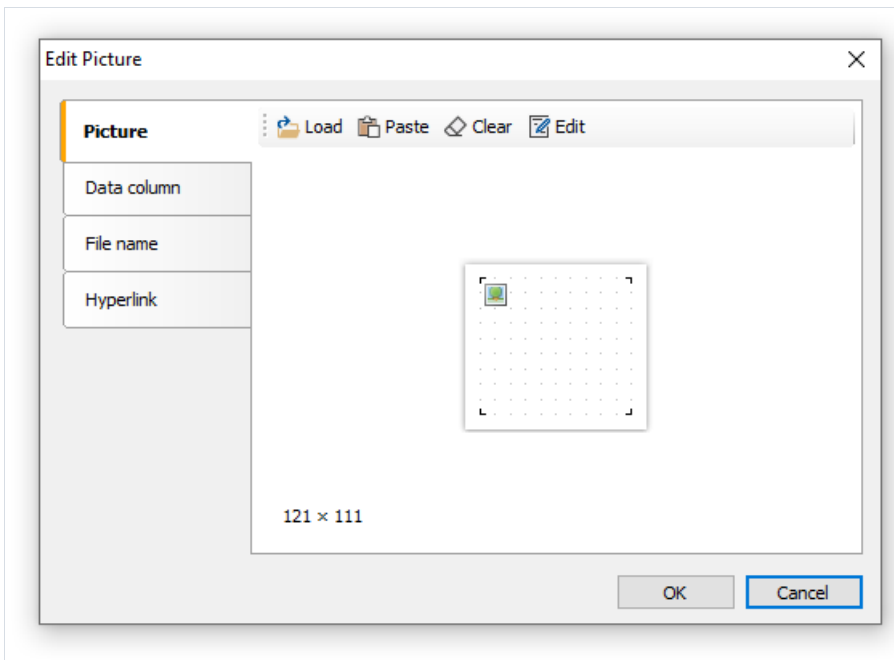
"Resim" nesnesi, rapora firma logosu, çalışan fotoğrafı veya herhangi bir grafik bilgi eklemek için kullanılabilir. Nesne, BMP, PNG, JPG, GIF, TIFF, ICO, EMF, WMF formatlarındaki grafikleri görüntüleyebilir. Aşağıdaki gibi görünmektedir:



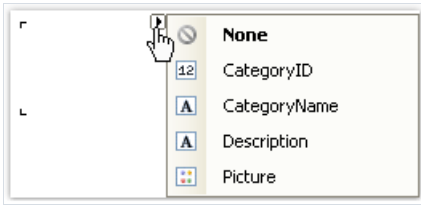
Nesne, aşağıdaki veri kaynaklarından gelen verileri gösterebilir:

Kaynak	Açıklama
Resim dosyası	Resim dosyadan yüklenir ve raporun içinde saklanır. Resim, Image özelliğinde depolanır.
Veri alanı	Resim, veri kaynağı alanından yüklenir. Alan adı, DataColumn özelliğinde saklanır.
Dosya adı	Resim, belirtilen ada sahip dosyadan yüklenir. Dosya adı, ImageLocation özelliğinde saklanır. Resim, raporun içinde depolanmaz. Raporla birlikte resim dosyasının da dağıtılması gerekmektedir.
URL	Resim, rapor her oluşturulduğunda internetten yüklenir. Resim, raporun içinde depolanmaz. Adres, ImageLocation özelliğinde saklanır.

Bir resim için veri kaynaklarından birini seçmek için nesneye çift tıklayın. "**Resim**" nesnesinin düzenleyicisini göreceksiniz:



Bir nesneyi veri alanına hızlıca bağlamak için nesnenin sağ üst köşesindeki küçük oka tıklayın ve açılan listeden gerekli alanı seçin:

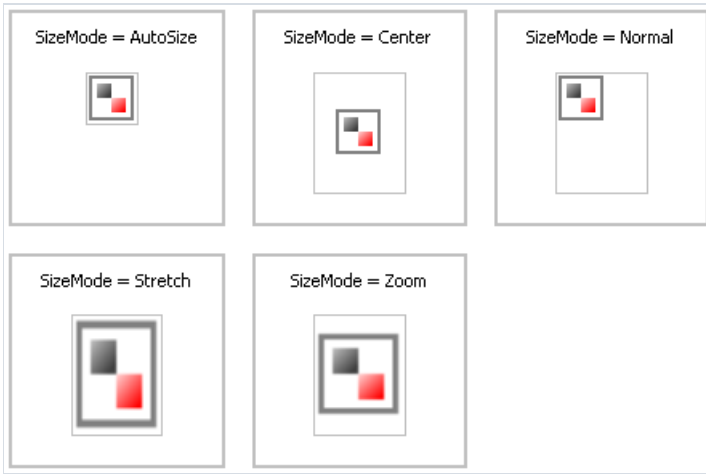


Ayrıca, gerekli alanı "**Veriler**" penceresinden rapor sayfasına fare ile sürükleyerek (drag & drop) bağlayabilirsiniz. Bu durumda, seçilen alana referans içeren bir "**Resim**" nesnesi oluşturulur.

Nesnenin bağlam menüsünde resmin görüntülenme modunu ayarlayabilirsiniz:

- **Otomatik Boyut (AutoSize)** – nesne, resmin boyutlarına uyum sağlar.
- **Ortala (CenterImage)** – resim, nesnenin içinde ortalıdır.
- **Normal (Normal)** – resim, ölçeklendirilmeden nesnenin sol üst köşesinde görüntülenir.
- **Genişlet (StretchImage)** – resim, nesnenin boyutlarına kadar genişletilir, ancak oranlar korunmaz.
- **Yakınlaştır (Zoom)** – resim, nesnenin boyutlarına ölçeklenir ve oranlar korunur.

Modlar arasındaki farklar aşağıdaki resimde gösterilmektedir:



Nesne aşağıdaki özelliklere sahiptir:

Özellik	Açıklama
Dönme Açısı (Angle)	Resmin döndürüleceği açı. Bu özelliğin alabileceği değerler: 0, 90, 180, 270 .
Görüntüleme Modu (SizeMode)	Resmin görüntüleme modu.
Şeffaflık (Transparency)	Resmin şeffaflık derecesi. Bu özellik 0 ile 1 arasında bir değer alabilir. 0 (varsayılan) değeri, resmin tamamen opak olduğunu belirtir.
Şeffaf Renk (TransparentColor)	Resim görüntülenirken şeffaf olacak renk.
Resim (Image)	Asıl resim.
Veri Alanı (DataColumn)	Resmin yükleneceği veri alanı.
Resmin Konumu (ImageLocation)	Bu özellik bir dosya adı veya URL içerebilir. Resim, rapor oluşturulurken belirtilen konumdan yüklenecektir.

Dolgu Boşluğu (Padding)	Bu özellik, resmin nesnenin kenarlarından piksel cinsinden olan boşluklarını ayarlamaya olanak tanır.
Hata Resmini Göster (ShowErrorImage)	Resim boş olduğunda " Resim Yok " simgesini gösterir. Bu özellik, resim internetten yüklendiğinde kullanışlıdır.

"SVG" Objesi

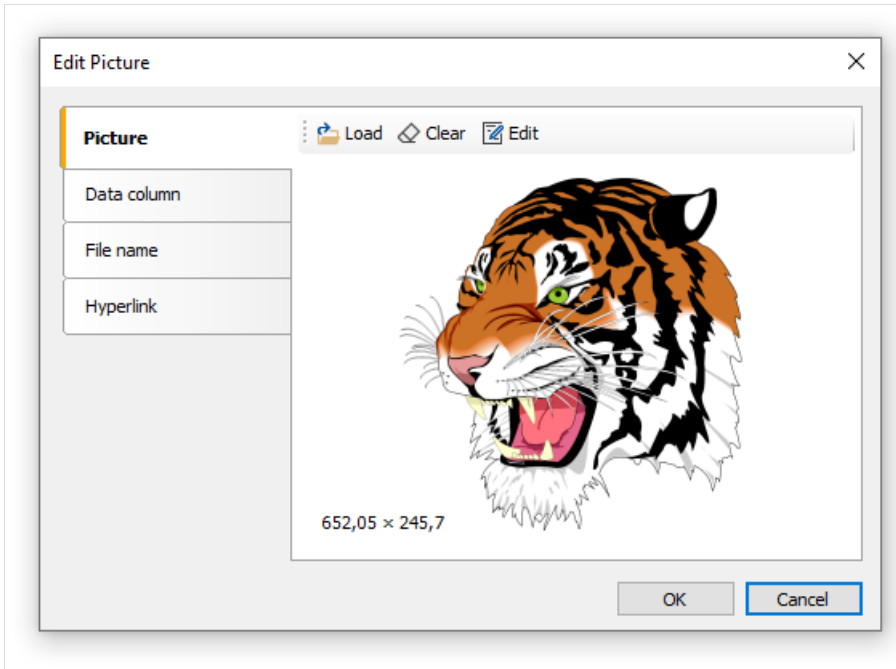
Bu obje, SVG formatındaki vektör görsellerini görüntülemek için tasarlanmıştır. Böyle bir görselin örneği:



SVG görseli eklemek için, raporunuza "SVG" objesini eklemek için aşağıdaki nesne panelindeki düğmeyi kullanın:

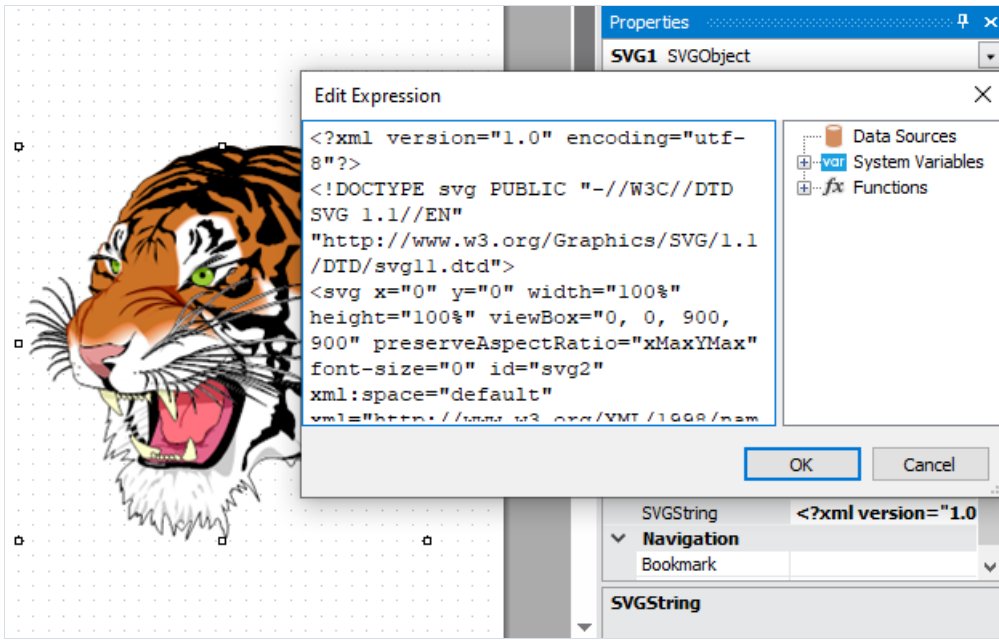


Sonrasında objeye iki kez tıklayın veya bağlam menüsünden "Düzenle" seçeneğini seçin. Bu, "Resim" objesinin düzenleyicisine benzer bir düzenleyici açacaktır.



Bu düzenleyicide, objeyle gösterilecek görseli belirleyebilirsiniz. Görseli rapor şablonuna dahil edebilirsiniz (bu, "Resim" sekmesindeki "Aç" butonuyla yapılır), veritabanından bir SVG görseli seçebilirsiniz (Veri Alanı sekmesi), harici bir SVG dosyasına bağlantı kurabilirsiniz (Dosya Adı sekmesi, bu durumda dosyayı şablonla birlikte dağıtmanız gerekir), veya bir URL üzerinden alınacak görseli belirleyebilirsiniz (Bağlantı sekmesi).

Eğer görseli şablona dahil ederseniz, bu görselin metin temsili "SVGString" özelliğinde saklanacaktır. Aşağıdaki ekran görüntüsünde bu tür bir metin temsili bir parçası gösterilmektedir:

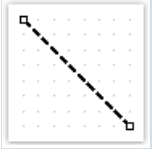


SVG objesi, "Resim" objesinin boyutlandırma modlarıyla aynı boyutlandırma modlarını destekler: Otomatik Boyutlandırma (AutoSize), Merkezle (CenterImage), Normal (Normal), Genişlet (StretchImage), Ölçekle (Zoom).

Bu modların ayrıntılı açıklaması "Resim" objesinin açıklamasında verilmiştir.

"Çizgi" Nesnesi

"Çizgi" nesnesi yatay, dikey veya çapraz bir çizgi görüntüleyebilir. Çizgiler, rapor düzenleme amacıyla kullanılabilir. Nesne aşağıdaki gibi görünür:



Mümkün olan durumlarda, çizgiler yerine nesne çerçevelerini kullanın. Bu, raporu sadeleştirerek gereksiz nesneleri ortadan kaldırır ve ayrıca raporun farklı formatlara aktarılması sırasında oluşabilecek olası sorunları önler.

FastReport tasarımcısı, çizgi çizmek için kullanışlı araçlar sunar. Raporunuza bir çizgi eklemek için araç çubuğundaki **"Nesneler"** panelinde bulunan düğmeye tıklayın ve açılan menüden **"Çizgi"** veya **"Çapraz Çizgi"** nesnesini seçin. Çizginin başlayacağı noktaya fare imlecini yerleştirin. Ardından, fare sol tuşuna basılı tutarak çizgiyi çizin. Çizimi tamamladıktan sonra yeni bir çizgi daha çizebilirsiniz. Tüm çizgiler çizildiğinde, araç çubuğundaki **"Nesneler"** düğmesine tekrar tıklayın.

Normal çizgi ile çapraz çizgi arasındaki fark, normal çizginin yalnızca yatay veya dikey olarak çizilebilmesidir.

"Çizgi" nesnesi için **"Çift Çizgi"** stilini seçmek mümkün değildir. Bu stil, araç çubuğundaki **"Çerçeve ve Dolgu"** açılır listesinden seçilebilse de yalnızca nesne çerçevesine uygulanabilir.

"Çizgi" nesnesinin aşağıdaki özellikleri vardır:

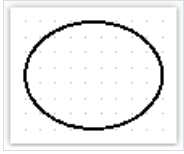
özellik	açıklama
"diyagonal" (Diagonal)	Özellik, çizginin diyagonal olup olmadığını belirler. Bu özelliği etkinleştirerek normal bir çizgiyi diyagonal hale getirebilirsiniz.
Çizgi uçları (StartCap, EndCap)	Bu özellikler, çizgi için aşağıdaki uç tiplerinden birini belirlemenizi sağlar: • Elips • Dikdörtgen • Baklava (Romb) • Ok Uç kısmının boyutları (genişlik ve yükseklik), uç kısmına ait Width ve Height özellikleri ile belirlenir. Çizginin her iki ucu için ayrı uç kısmı ayarlanabilir.

"Figür" Nesnesi

"Figür" Nesnesi "Figür" nesnesi raporu biçimlendirmek için kullanılır. Aşağıdaki geometrik şekillerden birini görüntülemenize olanak tanır:

- Dikdörtgen
- Yuvarlatılmış dikdörtgen
- Elips
- Üçgen
- Yamuk

Nesne şu şekilde görünür:



Rapora şekil eklemek için "Nesneler" araç çubuğundaki butona tıklayın ve listeden gerekli şekli seçin. Figür, diğer rapor nesneleri gibi dolgu ve çerçeveye sahip olabilir. "Metin" nesnesinden farklı olarak, çerçevenin ayrı çizgilerini yönetemezsiniz. Ayrıca "Çift Çizgi" çizgi stilini kullanamazsınız.

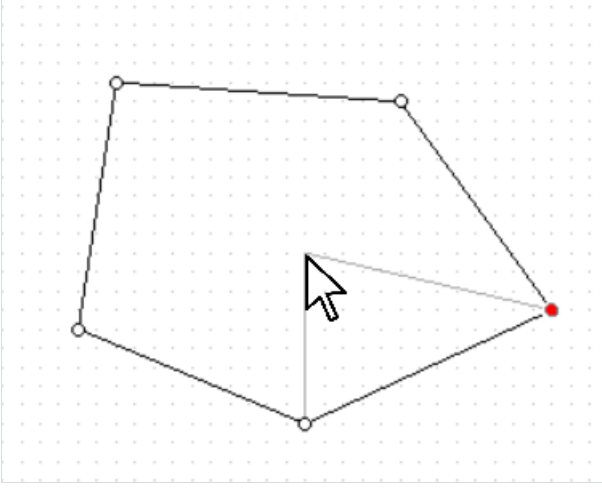
Mümkün olduğunda raporu biçimlendirirken nesnelerin çerçevelerini kullanın.

Nesnenin aşağıdaki özellikleri vardır:

Özellik	Açıklama
"Şekil Türü" (Shape)	Bu özellik, şekil türünü seçmenize olanak tanır.
"Kavis" (Curve)	Bu özellik, "Yuvarlatılmış Dikdörtgen" türündeki şekil için kavis belirlemenize olanak tanır.

"Kırık Çizgi" ve "Çokgen" Nesneleri

Bu nesneler, çokgenler gibi çeşitli şekilleri görüntülemek için kullanılır. Aynı kategoriye ait olup, Shape (Şekil) ve Line (Çizgi) nesneleriyle birlikte yer alır. Polygon (Çokgen) nesnesini yerleştirmek için, menüyü açın ve çokgenlerden birini seçin. Önceden yapılmış beşgen, altıgen, yedigen veya sekizgeni yerleştirebilirsiniz. Alternatif olarak, Polygon (Çokgen) seçeneğini seçerek şekli kendiniz çizebilirsiniz. Bir çokgenin nasıl görünebileceğine dair bir örnek:

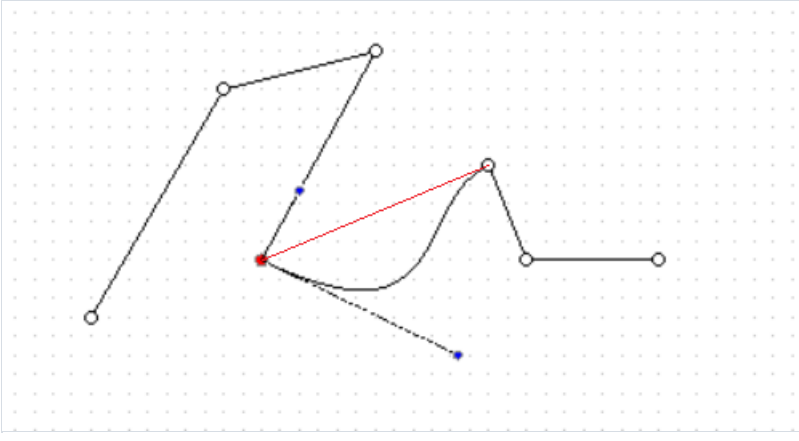


Resimdeki gri çizgiler, yeni bir nokta eklendikten sonra nesnenin nasıl görüneceğini göstermenize yardımcı olur.

İlk noktayı yerleştirdikten sonra, ek noktalar ekleyebilirsiniz. Çokgeni çizmeyi bitirdikten sonra, Escape tuşuna basın veya Düzenleme moduna geçin. Polyline (Polilin) ve Polygon (Çokgen) nesneleri için düzenleme modunun paneli, Ana sayfa sekmesinde bulunur.

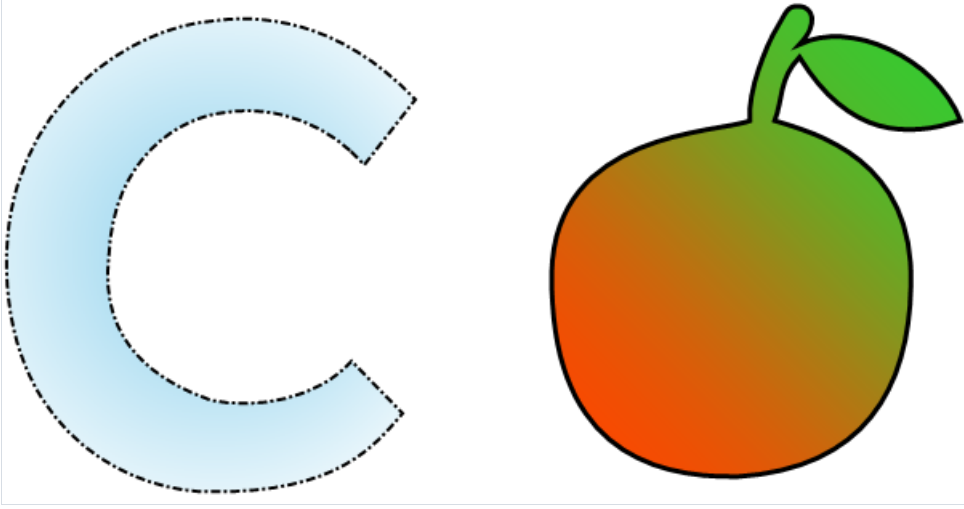
Polilin ve çokgen düzenleme modları:

Düğme	Mod	Açıklama
	Nokta Ekleme	Çokgen eklendikten hemen sonra etkinleşir. Bir nokta eklemek için bu modu seçin ve nesnenin noktalarından birini seçin. Daha sonra, noktayı yerleştirmek istediğiniz konuma gidin ve fare sol tuşuna tıklayın. Nokta, seçtiğiniz noktanın yanına eklenecektir.
	Taşıma ve Boyut Değiştirme	Bu modda, nesneyi tamamen taşıyabilir, ayrıca sıkıştırabilir veya uzatabilirsiniz.
	Nokta Silme	Bir çokgenden veya kırık çizgiden bir noktayı silmek için bu modu seçin ve silmek istediğiniz noktaya tıklayın. Ayrıca, başka bir modda noktayı seçip Delete tuşuna basarak da silebilirsiniz.
	Fare İmleci	Bu mod, daha önce eklenen nesnenin tüm noktalarını görüntülemenizi ve taşımanızı sağlar.
	Eğri Destek Noktası	Bu modda, seçilen noktaya bitişik olan çizgiye bir eğrilik noktası ekleyebilirsiniz. Bu çizgi, Bezier eğrisine dönüşecek ve eklenen nokta destek noktası rolünü üstlenecektir.



İşte eğrilik noktası belirlenmiş (mavi) kesik bir çizginin nasıl görüldüğü. Kırmızı çizgi, çizgi eğriye dönüştürülmeden önceki halini gösterir.

Her iki nesne de kenarlarını ayarlama yeteneğine sahiptir ve Polygon (Çokgen) nesnesi, diğer nesnelerle aynı dolgu modlarını destekler.



The "Barcode" object

The object displays barcodes in the report. It looks like this:



The Barcode object supports the following types of barcodes:

Two-dimensional

Barcode name	Length	Allowed symbols
PDF417	No fixed length	Any
Datamatrix	No fixed length	Any
QR Code	No fixed length	Any
Aztec	No fixed length	Any
MaxiCode	No fixed length	138 digits or 93 chars

EAN/UPC

Barcode name	Length	Allowed symbols
EAN-8	8	0-9
EAN-13	13	0-9
UPC-A	12	0-9
UPC-E0	6	0-9
UPC-E1	6	0-9

Post

Barcode name	Length	Allowed symbols
Deutsche Identcode	12	0-9
Deutsche Leitcode	14	0-9
PostNet	No fixed length	0-9

Barcode name	Length	Allowed symbols
Japan Post 4 State Code		
Intelligent Mail (USPS)	No fixed length	0-9, A-Z

GS1

Barcode name	Length	Allowed symbols
GS1-128	No fixed length	0-9, A-Z
GS1 DataBar Omnidirectional		
GS1 DataBar Limited		
GS1 DataBar Stacked		
GS1 DataBar Stacked Omnidirectional		
GS1 Datamatrix		

Others

Barcode name	Length	Allowed symbols
2 of 5 Interleaved	No fixed length	0-9
2 of 5 Industrial	No fixed length	0-9
2 of 5 Matrix	No fixed length	0-9
ITF-14	14	0-9
Codabar	No fixed length	0-9, -, \$, :, /, ., +
Code 128	No fixed length	128 ASCII chars
Code 39	No fixed length	0-9, A-Z, -, ., *, \$, /, +, %
Code 39 Extended	No fixed length	128 ASCII chars
Code 93	No fixed length	0-9, A-Z, -, ., *, \$, /, +, %
Code 93 Extended	No fixed length	128 ASCII chars
MSI	No fixed length	0-9
2-Digit Supplement	2	0-9
5-Digit Supplement	5	0-9

Barcode name	Length	Allowed symbols
Plessey	No fixed length	Hex digits (0-F)
Pharmacode	No fixed length	0-9

Barcode data in an object is of a string type. The string can contain any symbol, allowed for the chosen type of barcode. You can choose the type of barcode in the context menu of the "Barcode" object.

You can connect an object to data by using one of the following methods:

- set the barcode data in the `Text` property;
- bind the object to a data column using the `DataColumn` property;
- set the expression that returns the barcode data in the `Expression` property.

The Barcode object has the following properties:

Property	Description
Barcode	This property contains barcode-specific settings. Expand this property to set these settings.
Angle	This property determines the rotation of a barcode, in degrees. You can use one of the following values: 0, 90, 180, 270.
Zoom	This property allows to zoom a barcode. It is used along with the <code>AutoSize</code> property.
AutoSize	If this property is on, the object will stretch in order to display a whole barcode. If this property is off, the barcode will stretch to to object's bounds.
ShowText	This property determines whether it is necessary to show the human-readable text.
DataColumn	The data column which this object is bound to.
Expression	The expression that returns the barcode data.
Text	The barcode data.
Padding	The padding, in pixels.

The following properties are specific to the barcode type. To change them, select the barcode, go "Properties" window and expand the `Barcode` property.

Property	Description
WideBarRatio	This property is specific to all linear barcodes. It determines the wide-to-narrow bar ratio. For most of barcode types, the value for this property should be between 2 and 3.
CalcChecksum	This property is specific to all linear barcodes. It determines whether is necessary to calculate the check sum automatically. If this property is off, you should provide the check digit in the barcode data.

Property	Description
AutoEncode	This property is specific to the Code128 barcode. This code has three different encodings - A, B, C. You should either set the encoding explicitly in the barcode data, or set this property to true. In this case the encoding will be chosen automatically. Use the following control codes in the barcode data: &A; START A / CODE A &B; START B / CODE B &C; START C / CODE C &S; SHIFT &1; FNC1 &2; FNC2 &3; FNC3 &4; FNC4 If you set the <code>AutoEncode</code> property to <code>true</code>, all control codes will be ignored. Example of use the control codes: <code>&C;1234&B;ABC</code>
AspectRatio	This property is specific to the PDF417 barcode. It determines the height-to-width aspect ratio and is used to calculate the barcode size automatically (in case the <code>Columns</code> and <code>Rows</code> properties are not set).
CodePage	This property is specific to the PDF417 and Datamatrix barcode. It determines the code page which is used to convert non-ASCII chars. For example, the default windows codepage is 1251.
Columns, Rows	These properties are specific to the PDF417 barcode. They determine the number of columns and rows in a barcode. If both properties are set to 0, the size of the barcode will be calculated automatically. In this case the <code>AspectRatio</code> property is used as well.
CompactionMode	This property is specific to the PDF417 barcode. It determines the PDF417 data compaction mode.
ErrorCorrection	This property is specific to the PDF417 barcode. It determines the error correction level.
PixelSize	This property is specific to the PDF417 barcode. It determines the size of barcode element, in screen pixels. As a rule, the element's height should be greater than the element width by 3 times or more.
Encoding	This property is specific to the Datamatrix barcode. It determines the Datamatrix data encoding.
PixelSize	This property is specific to the Datamatrix barcode. It determines the size of barcode element, in pixels.
SymbolSize	This property is specific to the Datamatrix barcode. It determines the size of barcode symbol.

PDF417



Bu yaygın iki boyutlu barkod, büyük veri hacimlerini kodlamak için tasarlanmıştır. Kısaltması olan PDF, Portable Data File (Taşınabilir Veri Dosyası) ifadesinin baş harflerinden oluşur. 417 sayısı, 4 ve 17 rakamlarının toplamından oluşmuştur. Burada 4, dört çizgi ve dört boşluk anlamına gelirken, 17 ise bir kod sözcüğündeki modül sayısını temsil eder.

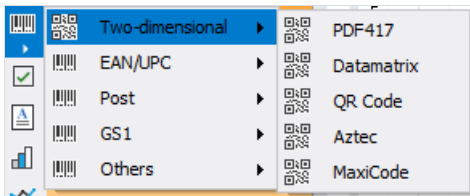
Daha önce belirtildiği gibi, PDF417, QR Code, MaxiCode ve Data Matrix gibi yaygın kullanılan barkod türlerinden biridir. Görsel olarak PDF417, klasik lineer barkod ile matris barkodun bir karışımı gibi görünse de aslında karmaşık bir lineer barkoddur. Kenarlarda klasik lineer barkodlarda olduğu gibi çizgiler bulunurken, ortadaki çizgiler üst üste sıralar halinde yerleştirilmiştir. Bu lineer barkod sıkıştırma yöntemi, kod satırlarının üst üste dizilimi ile gerçekleştirilir. Bu yapı sayesinde büyük hacimde veri saklanabilir: 3 ile 90 satır arasında değişen barkodda, 1859 alfabetik karakter veya 2725 sayısal karakter kodlanabilir.

Kodun zarar görmesine karşı koruma, barkodun %50'sine kadar hasarı kapatabilen bir yedeklilik içerir. Bu, oldukça yüksek bir değerdir, ancak bu durumda barkodun boyutları da artacaktır. Matris barkodları ile karşılaştırıldığında PDF417, aynı miktarda bilgiyi kodlamak için birkaç kat daha fazla yer kaplar ki bu durum, dezavantajlarından biri olarak değerlendirilebilir.

PDF417'nin uygulama alanı oldukça geniştir. Taşımacılık şirketleri tarafından yolcu biletleri ve kargo gönderileri üzerinde, posta gönderimlerinde, raporlama belgelerinde, çeşitli kimlik kartlarında, depo yönetiminde ve işaretleme ve tanımlama gereken birçok başka alanda kullanılır.

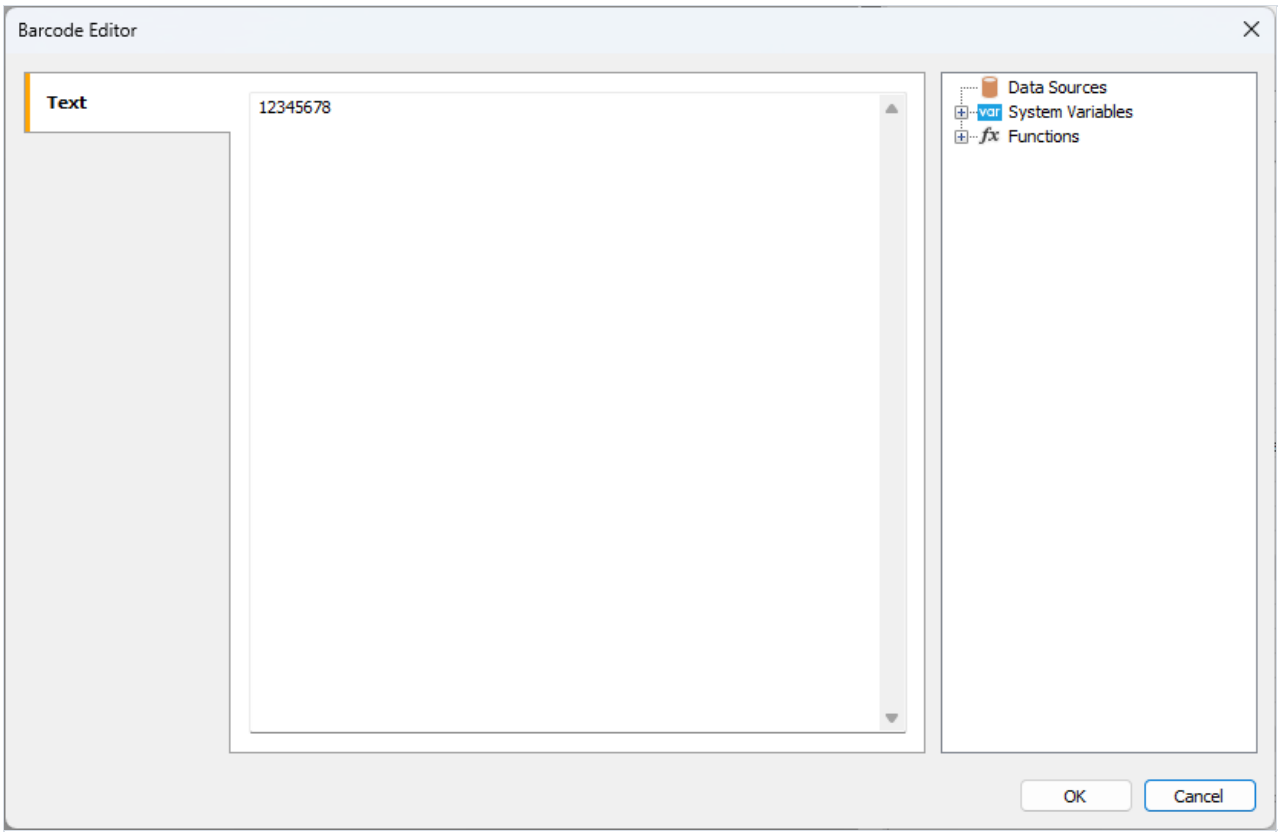
Bu barkodu okumak için, lineer barkodlar için kullanılan standart okuyuculardan biraz farklı olan lazer barkod okuyucuları kullanılır. Yukarıda belirtildiği gibi, kodun kenarlarında barkodun başlangıç ve bitişini belirten tipik lineer barkod çizgileri bulunur.

FastReport .NET içinde PDF417 barkodu oluşturmak için, rapor tasarımcı panelindeki bileşenlerden "Barkod" (Barcode) nesnesini seçiniz. Açılır listeden "İki Boyutlu" (2D) kategorisini seçip ardından PDF417'yi belirleyiniz:



Barkodu seçtikten sonra, rapor sayfasına yerleştiriniz.

Eklenen barkoda çift tıklanarak barkod düzenleyicisi açılır. Ayrıca eklenmiş nesnenin üzerine sağ fare tuşuyla tıklanarak açılan bağlam menüsündeki butona basılarak da düzenleyici açılabilir.



Eğer barkodun altındaki metni gizlemek gerekirse, ilgili barkodun özellikler panelinde "ShowText" özelliğini bulup, değerini "False" olarak ayarlayınız.

QR kodu

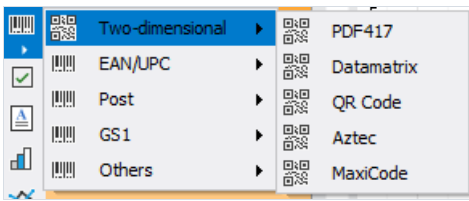
QR kodu, sayısal, alfa-sayısal ve ikili bilgileri saklamak için tasarlanmış iki boyutlu bir barkod türüdür.

Kamerayla QR kodunun doğru şekilde tanınması için, köşelerde ve görsel alanında özel işaretleyiciler kullanılır. Bu işaretleyiciler, kodun taranmasından sonra görüntünün normalleştirilmesini sağlar ve noktasal kodlamanın denetim toplamı doğrulaması ile ikili sayılara dönüştürülmesine imkân verir.

Alfa-sayısal kodlama kullanılarak bir QR kodunda en fazla **4296 karakter** saklanabilir.

Nesne

FastReport .NET ortamında QR kodu oluşturmak için, rapor tasarımcısında **barkod (Barcode)** bileşenini bileşenler panelinden seçin. Açılır listeden **“İki Boyutlu” (2D)** kategorisine geçin ve ardından **QR Code** seçeneğini tıklayın:

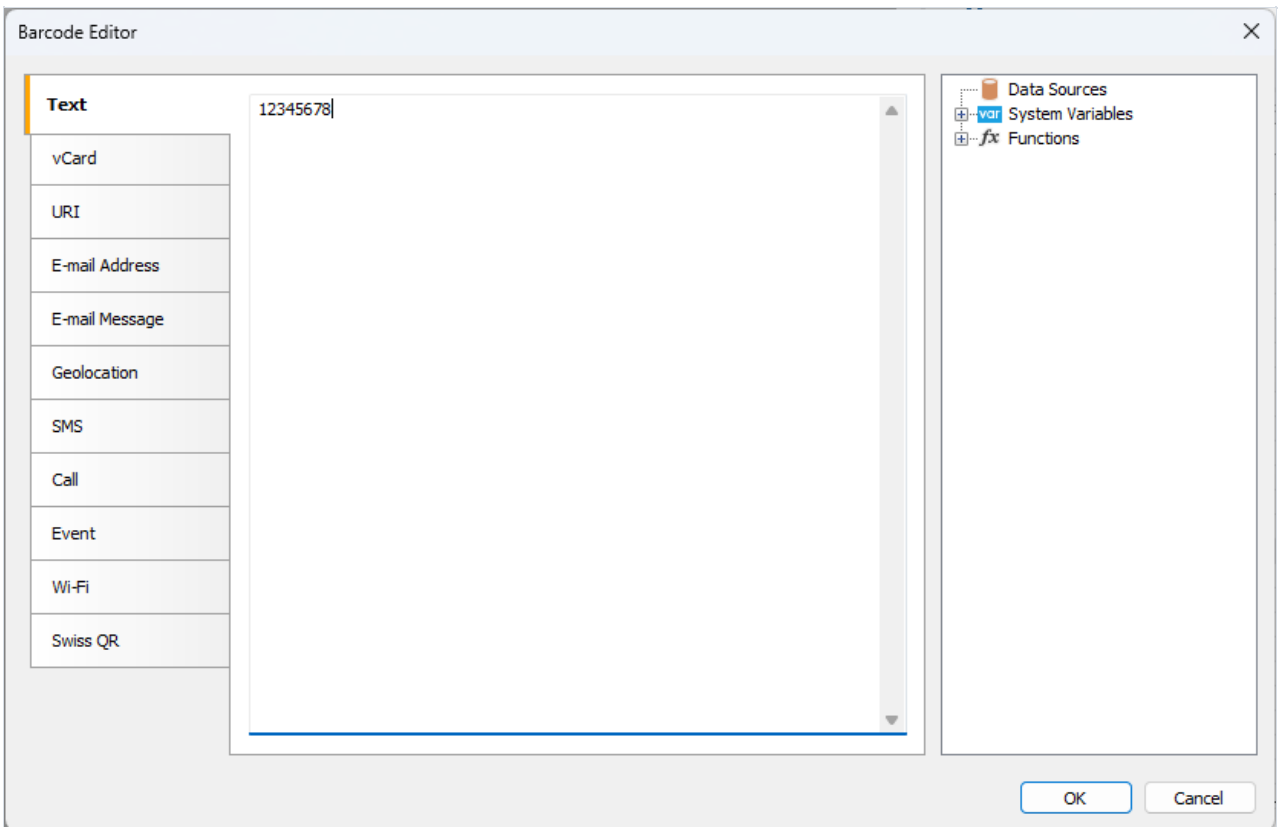


Barkod seçildikten sonra, onu rapor sayfasına yerleştirin.

Eklenen barkodu çift tıklayarak düzenleyici penceresini açabilirsiniz. Ayrıca, barkod nesnesine sağ tıklayarak açılan bağlam menüsünden **butona** tıklayarak da düzenleyiciye erişebilirsiniz.

Düzenleyici

QR kod düzenleyicisi, standart ifade düzenleyicisine benzer. Sağ tarafta veri, parametre ve fonksiyon ağaçları bulunur. Bu ağaçtan öğeleri sürükleyip metin düzenleyicisine bırakabilirsiniz:



İfade düzenleyicisinden temel farkı, pencerenin sol tarafında bulunan **sekmelerdir**. Bu sekmeler, QR kodunun içeriğinin türünü belirler.

Seçilen içeriğe bağlı olarak, düzenleyicide uygun **alanlar** görüntülenir.

The image shows a 'Barcode Editor' window. On the left is a sidebar with a list of content types: Text, vCard (highlighted), URI, E-mail Address, E-mail Message, Geolocation, SMS, Call, Event, Wi-Fi, and Swiss QR. The main area displays a form for a vCard. It has two columns of input fields. The first column contains: First name (Anne), Last name (Dodsworth), Title (Manager), Company, Website, E-mail (personal), and E-mail (business). The second column contains: Phone (mobile), Phone (personal), Phone (business) (71) 555-4444, Street (7 Houndstooth Rd.), Zip code, City, and Country. On the right side of the window, there is a 'Data Sources' panel with 'var System Variables' and 'fx Functions' options. At the bottom right are 'OK' and 'Cancel' buttons.

QR kodu içerik türleri

Düzenleyiciyle çalışırken, içerik metni (tablonun dördüncü sütunu) otomatik olarak oluşturulur.

Düzenleyicideki alanlar herhangi bir ifadeyi içerebilir; bu ifadeler veri kaynağından gelen alanlar da olabilir (bu alanlar sağdaki ağaçtan sürüklenerek eklenebilir).

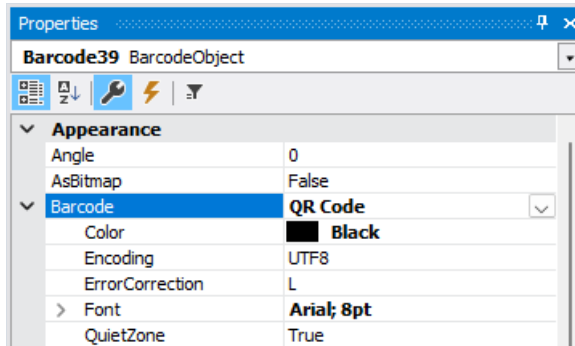
İçerik Türü	Açıklama	Görsel Örneği	İçerik Örneği
Text	Harf ve rakam biçiminde metin		12345678

vCard	Versitcard – elektronik bir kartvizit biçimidir. Aşağıdaki bilgileri içerebilir: • Soyadı; • Adı; • Görevi; • Şirket adı; • Şirketin web sitesi; • Kişisel e-posta adresi; • İş e-posta adresi; • Cep telefonu numarası; • Ev telefonu numarası; • İş telefonu numarası; • Adres; • Posta kodu; • Şehir; • Ülke. Yalnızca bazı alanlar doldurulmuş olabilir.		BEGIN:VCARD VERSION:2.1 FN:AnneDodsworth N:Dodsworth;Anne TITLE:Manager TEL;WORK;VOICE(71) 555-4444 ADR;;;7 HoundstoothRd.;;;; END:VCARD
URI	Tekdüzen Kaynak Tanımlayıcısı (URI). Dosya, belge, görsel, e-posta, web sitesi vb. öğelere bağlantı içeren bir dizedir.		http://www.fast-report.com/en/product/fast-report-net/
E-mail Address	E-posta Adresi		support@fast-report.com
E-mail Message	E-posta Metni		MATMSG:TO:support@fast-report.com;SUB:FastReport .NET question;BODY:Hello, I have a question about FastReport .NET.;;
Geolocation	Gerçek coğrafi konumu belirlemek için koordinatlar		geo:-50.737563,-79.490016,120
SMS	Metin Mesajı		SMSTO:(71) 555-4444:Hello, Dolly! I'm fine!
Call	Telefon Numarası		tel:(71) 555-4444
Event	Takvime eklenmek üzere bir etkinliktir. Tarih ve saat bilgilerine ek olarak metin mesajı da içerebilir.		BEGIN:VEVENT SUMMARY:Team Meeting DTSTART:20240420T090000Z DTEND:20240420T100000Z DESCRIPTION:Weekly team meeting to discuss project updates. END:VEVENT
Wi-Fi	Wi-Fi ağına bağlanmak için gerekli bilgiler		WIFI:T:WPA;S:HoneyPot;P:youarewelcome;H:true;

Swiss QR	Swiss Bill için ödeme bilgilerini içeren özel bir QR kodudur. Bu barkod hakkında daha fazla bilgiye dokümantasyonun başka bir bölümünde ulaşabilirsiniz.		SPC 0200 1 CH4431999123000889012 S Carl Ltd. Luber 16 123321 Berlin GE 50050.00 EUR S Sigmunt Shuld Lunglen 23 123322 Ferburg GE NON EPD
-----------------	--	---	--

Barkod Özellikleri

Şimdi QR kodunun özelliklerine bakalım. Bu özelliklere, nesne denetleyicisinde **Barcode** özelliği altında erişilebilir.



Özellik	Açıklama
Encoding	Barkod içeriğinin metin kodlamasıdır. Örneğin: UTF8, Windows_1251, CP_866 vb. Varsayılan değer: UTF8.

ErrorCorrection	Reed-Solomon kodu kullanılarak hata düzeltilmesi yapılır. Aşağıdaki değerleri alabilir: L (düşük – %7), M (orta – %15), Q (%25), H (yüksek – %30). Varsayılan değer: L 'dir.
QuietZone	QR kodunun etrafında beyaz bir kenarlık (çerçeve) olup olmadığını belirler. Varsayılan değer: True (etkin).
Color	Barkodun rengini belirler. Varsayılan renk: Siyah .

Kod görüntüsünün kısmen hasar görmesi veya üzerine başka bir görsel yerleştirilmesi durumunda verilerin doğru şekilde okunabilmesi için hata düzeltilmesi gereklidir.

Örneğin, yedeklilik seviyesi H (%30) olarak ayarlanmışsa, aşağıda gösterilen barkod sorunsuz şekilde okunabilir:



Barkodun altındaki metni gizlemek istiyorsanız, ilgili barkod nesnesinin özellikler penceresindeki ShowText (MetniGöster) özelliğini bulun ve değerini False olarak ayarlayın.

AutoSize (Otomatik Boyutlandırma) özelliği, barkodun boyutunun nesne boyutuna göre otomatik olarak ayarlanmasını sağlar. Eğer barkodun boyutunu manuel olarak (fareyle) değiştirmek istiyorsanız, bu özelliği devre dışı bırakmanız gerekir (False olarak ayarlayın). Bu durumda, oranları manuel olarak korumanız gerekecektir.

Swiss QR

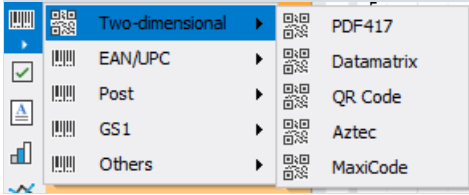
Swiss QR, QR kodlarının bir çeşididir. İsviçre’de, tıpkı dünyanın diğer yerlerinde olduğu gibi, QR kodlar yaygın olarak kullanılmaktadır. Ancak elektronik ödeme alanında bu kodların kullanılabilmesi için İsviçre’ye özgü bir QR kod çeşidinin oluşturulmasına karar verilmiştir. Artık İsviçre’deki tüm ödeme makbuzları ve faturalar, ortasında İsviçre haçı bulunan bir QR kod içermek zorundadır – bu, Swiss QR kodunun ayırt edici özelliğidir.

FastReport .NET, Swiss QR kodunu destekler. Kodun görünümü, ortasında İsviçre haçı bulunması dışında klasik QR kod ile neredeyse aynıdır:



Bu kod türünün amacı, ödeme bilgilerini şifrelemektir.

FastReport .NET’te Swiss QR oluşturmak için, rapor tasarımcısında bileşenler panelinden *Barkod (Barcode)* nesnesini seçin. Açılır listede "İki Boyutlu" kategorisine ve ardından "QR Code" seçeneğine gidin:



Barkod seçildikten sonra, sayfa üzerine yerleştirin. Eklenen barkoda çift tıklayarak ya da sağ tıklama ile açılan bağlam menüsünden editör simgesine tıklayarak *QR kod düzenleyicisini* açabilirsiniz.

Kod düzenleyicisinde *Swiss QR* sekmesine geçin. Bu sekme altında Swiss QR’a özel ayarlar gösterilir:

Barcode Editor

Text	IBAN	Type	Amount	Currency
vCard	CH4431999123000889012	Iban	50050	EUR
URI	Creditor Name: Carl Ltd. Street / Nr: Luber 16 Zip / City / Country: 123321 Berlin GE			
E-mail Address	Debitor Name: Sigmunt Shuld Street / Nr: Lunglen 23 Zip / City / Country: 123322 Ferburg GE			
E-mail Message	Additional information Reference: Type: NON Text type: Message: Bill Information: Alternative procedure 1: Alternative procedure 2:			
Geolocation				
SMS				
Call				
Event				
Wi-Fi				
Swiss QR				

OK Cancel

Parametreler:

- **Iban:**

İsviçre'de banka hesap numarası için IBAN (International Bank Account Number) standardı kullanılır. Bu, ISO 13616 altında kayıtlı uluslararası bir standarttır. Burada iki IBAN türünden biri seçilebilir: *Iban* veya *QR-Iban*. QR-Iban, QR referanslı ödemeler için kullanılır. QR-Iban da ISO 13616 standardına uygundur. Sisteme katılan her tüzel kuruluşa 30000 – 31999 aralığında bir tanımlayıcı (QR-IID) verilir ve bu tanımlayıcı QR-Iban'a dâhil edilir.

- **Creditor (Alacaklı)**

Fatura düzenleyenin adı ve adresi gibi bilgileri girin.

- **Reference (Referans)**

Ödeme yapan tarafından sağlanan ve alacaklının kullanacağı ödeme referans bilgisi.

- **Type (Tür)**

- **QRR** – QR Referans: Yalnızca rakamlardan oluşan, 26 karakter uzunluğunda İsviçre standardı.
- **SCOR** – Kreditor referansı: 5 ile 25 karakter arasında değişen uzunlukta uluslararası standart.
- **NON** – Referans alanı boş bırakılabilir.

- **Text Type (Metin Türü)**

- **QR-Reference** – QRR türü ile birlikte kullanılır.
- **ISO 13616** – SCOR türü ile birlikte kullanılır.

QR-Referans, daha önce kullanılan ISR referansının yerini alır. Eski kırmızı ve turuncu ödeme formlarından QR faturalara geçişi sağlar.

- **Debitor (Borçlu)**

Ödeme yapan kişi ya da kuruluşun adı ve adresi girilir.

- **Additional Information (Ek Bilgi)**

Fatura düzenleyici, ödeme yapacak kişiye yönelik yapılandırılmış ya da yapılandırılmamış ek bilgi girebilir.

- **Currency (Para Birimi)**

İsviçre ödeme sistemi kapsamında iki para birimi kullanılır: *Euro (EUR)* ve *İsviçre Frangı (CHF)*.

- **Alternative Procedure 1 ve Alternative Procedure 2**

Gelecekte, fatura düzenleyicilerinin banka havalesine alternatif yöntemler sunabilmesi için iki ayrı alan öngörülmüştür.

- **Amount (Tutar)**

Ödeme miktarı bu alanda girilir, ayırıcı olarak virgöl kullanılır.

Swiss QR kodu genellikle *Swiss bill* dokümanlarında kullanılır. Böyle bir Swiss fatura örneğini [Demos/Reports/](#) klasöründe bulabilirsiniz.

Aztec

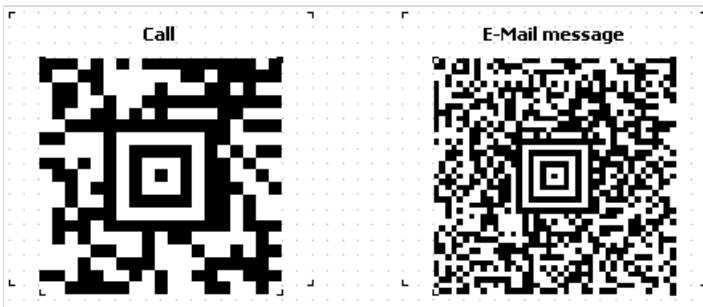
Aztec Code, QR kodunun bir türüdür. Aztekler, Orta Amerika yerlilerine ait bir kabile ismidir. Koda dikkatlice bakıldığında, merkezinde Aztek piramitlerinin üstten görünüşüne benzeyen kare bir alan fark edilir. Bu özel hedef alanı, kodun merkezini ve yönünü belirlemek için kullanılır.



Aztec Code, MaxiCode, SuperCode, CodeOne, DataMatrix, DotCode ve PDF417 gibi iki boyutlu barkodların en iyi özelliklerini bir araya getirir. Patentli olmasına rağmen, bu teknoloji kamu malı haline gelmiştir. Kodlama standardı ISO/IEC 24778:2008'de tanımlanmıştır.

Kodun boyutu, kodlanacak verinin miktarına göre değişir. Örneğin, minimum boyut olan 15x15 piksel, 6 baytlık yani 12 harf veya 13 rakamlık veri kodlayabilir. Maksimum boyut olan 151x151 piksel ise 1914 bayt, 3067 harf veya 3832 rakam barındırabilir.

Kodun iki görüntüleme biçimi bulunur: "Compact" (Kompakt), hedef sembolü iki kareden oluşur ve "Full-Range" (Tam Aralık), hedef sembolü üç kareden oluşur. Biçim seçimi, kodlanacak veri miktarına göre yapılır.



Diğer kodlama yöntemlerine göre önemli bir avantajı, herhangi bir açıdan okunabilir olmasıdır. Aynalı şekilde bile olsa, yön belirleme işaretleri sayesinde kolaylıkla okunabilir.

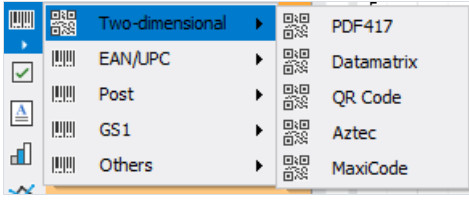
Kodun merkezindeki hedef kullanılarak, bozuk ya da şekli bozulmuş görüntülerden bile bilgi alınabilir. Reed-Solomon hata düzeltme algoritması sayesinde kısmen zarar görmüş kodlar dahi okunabilir. Bu amaçla kod içinde %5'ten %95'e kadar ayarlanabilir fazlalık payı bulunur, böylece yüksek hata toleransı sağlanır.

Katmanlı yapı sayesinde, veri alanı genişletilerek daha fazla bilgi saklanabilir.

Tüm bu avantajlar nedeniyle Aztec Code, hava ve demiryolu taşımacılığında elektronik biletleme gibi ulaşım ağlarında tercih edilmektedir. Bazı ülkelerde hükümet belgelerinde de kullanılır. Ayrıca yüksek yoğunluklu diğer kodlar gibi ticaret, lojistik, üretim ve eczacılık alanlarında popülerdir.

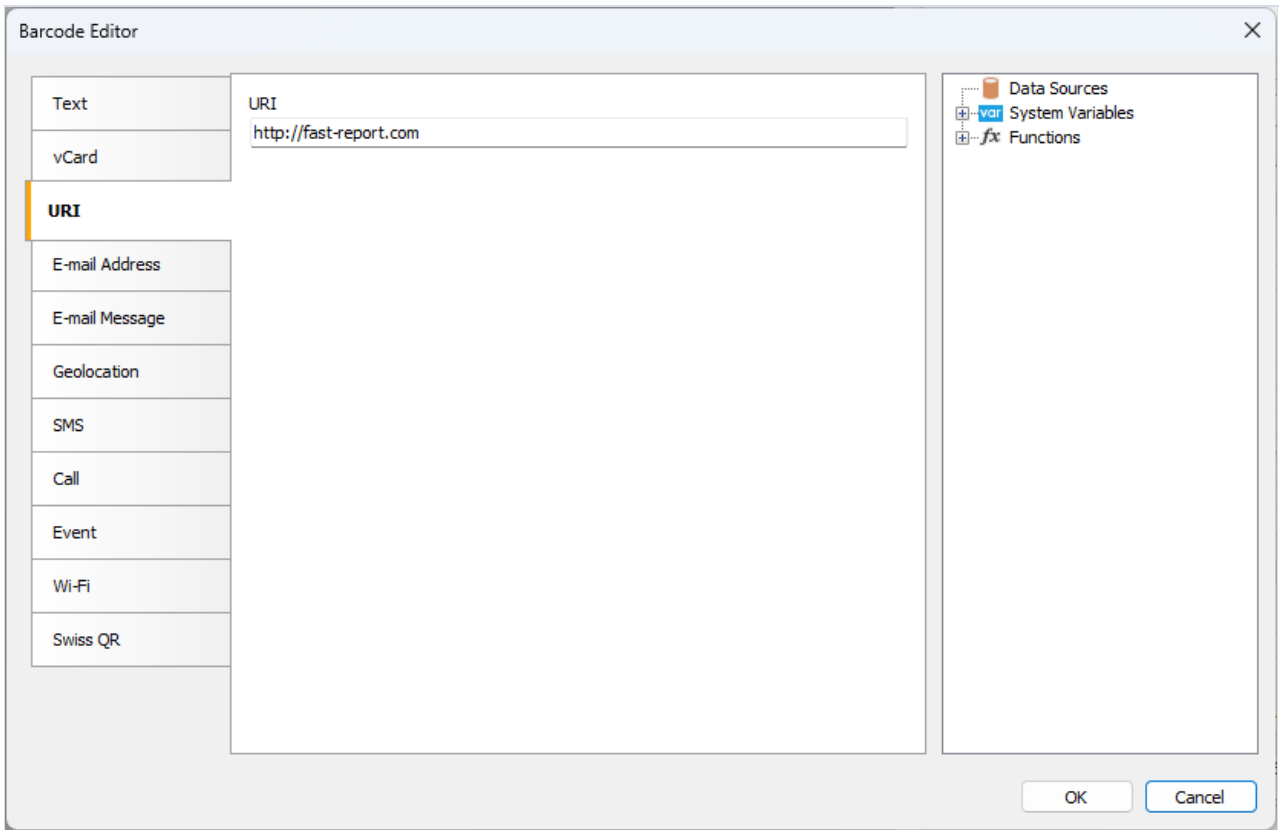
Aztec Code, QR koda kıyasla daha yüksek veri yoğunluğuna sahiptir ve çevresinde boş alan gerektirmez. Ayrıca minimum boyutu QR kodun 21x21 pikseline karşılık 15x15 pikseldir.

FastReport .NET programında Aztec Code barkodu oluşturmak için rapor tasarımcısında "Barkod" bileşenini seçin. Açılır menüden "İki Boyutlu" kategorisine gidin ve ardından Aztec'i seçin:



Barkodu seçtikten sonra rapor sayfasına yerleştirin. Eklenen barkoda çift tıklama ile düzenleyici açılır veya barkod nesnesinin sağ tuş menüsünden düzenleyiciye erişebilirsiniz.

Barkod düzenleyicisinde bilgi kodlamak için bir şablon seçilebilir. SwissQR hariç tüm şablonlar Aztec Code ile kullanılabilir. Örneğin, e-posta adresi:

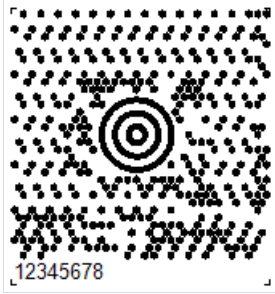


Sonuç olarak, oluşturulan barkod şu şekilde görünecektir:



MaxiCode

İki boyutlu MaxiCode barkodu, başka barkodlarla karıştırılması mümkün olmayan benzersiz bir tasarıma sahiptir. Tarama sırasında hassas konumlandırma için yuvarlak bir hedef noktası içerir. Kodlama elemanları, özel kombinasyonları oluşturan ve altıgen (bal peteği şeklinde) yapı içinde düzenlenmiş noktalar ile temsil edilir. Bu yapı, bilgi taşıyan 888 hücreden oluşmaktadır.



MaxiCode, 138 sayısal veya 93 alfabetik karakter kodlamasına olanak sağlar. Barkodun sabit boyutu 1,11 x 1,054 inçtir. Kod hasar gördüğünde, Reed-Solomon hata düzeltme algoritması sayesinde hataları düzeltme özelliğine sahiptir. Bu tür barkodların çoğu ambalajlarda bulunduğundan, hasara karşı dayanıklı olmaları gereklidir. Hata düzeltme özelliği, kodun 1/8'ine kadar olan hasarı tolere edebilir.

Her MaxiCode barkodu iki ana mesaj içerir: birincil ve ikincil mesajlar. Birincil mesajda posta kodu, ülke kodu ve sınıf bilgileri bulunur; ikincil mesaj ise adres bilgilerini içerir.

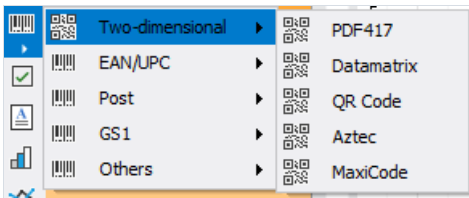
MaxiCode barkodlarını okumak için 2D barkod tarayıcısı gerekir, ideal olarak klavye emülasyonu yapan ve USB bağlantısından güç alan bir tarayıcı tercih edilir. Böylece harici bir güç kaynağına ihtiyaç duyulmaz.

MaxiCode'un temel uygulama alanları paket, palet ve benzeri ürünlerin etiketlenmesidir. Yük işleme sürecinde kritik öneme sahip bilgilerin her zaman erişilebilir olmasını sağlar.

MaxiCode, mevcut teslimat manifestoları sistemlerine entegre edilmek üzere geliştirilmiştir. Kompakt boyutu sayesinde, lineer barkodlar gibi daha düşük yoğunluktaki sembollerin yerine kullanılabilir.

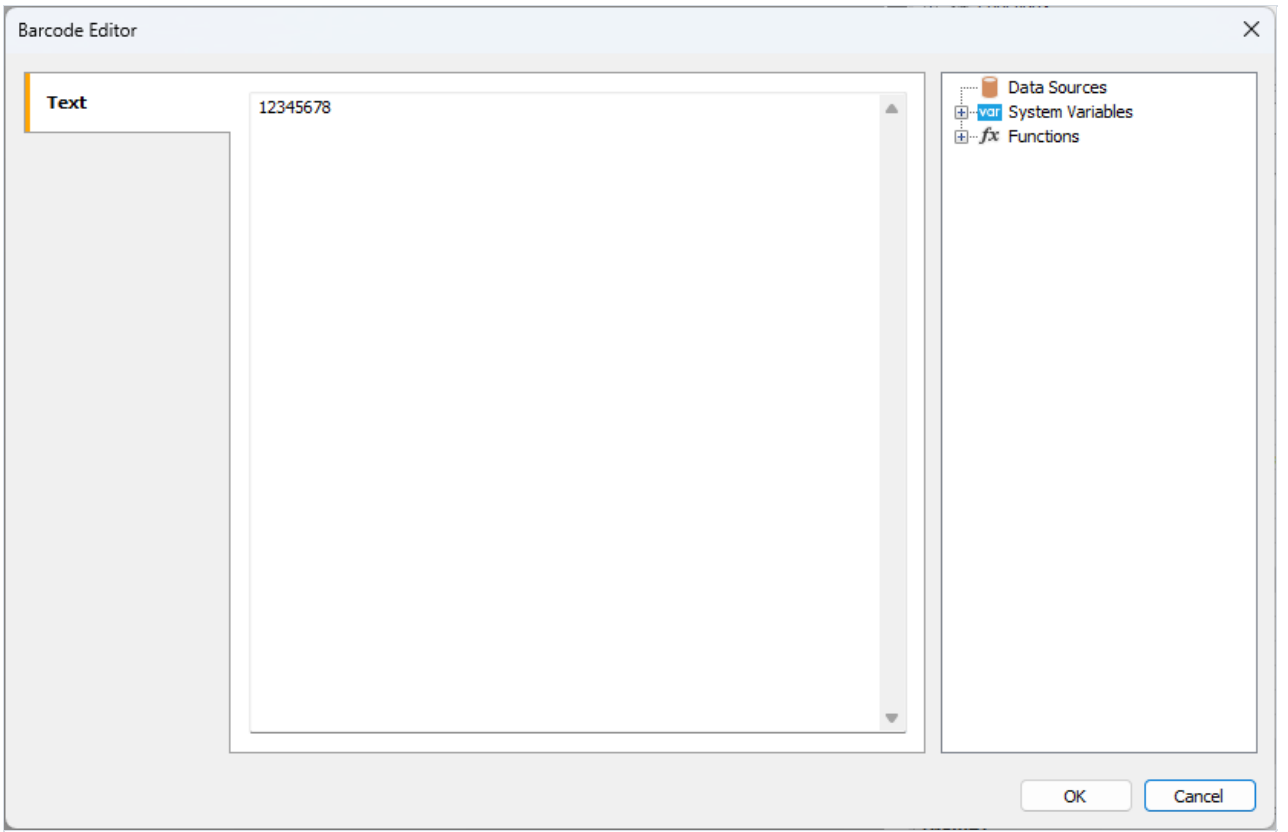
MaxiCode barkodunun önemli özelliklerinden biri, geniş görüş alanında hızlı okunabilme yeteneğidir. Bu sayede barkod, otomatik bilgi işleme sistemlerinde başarıyla kullanılabilir.

FastReport .NET ortamında MaxiCode barkodunu oluşturmak için rapor tasarımcısındaki bileşenler panelinde Barkod (Barcode) ögesini seçin. Açılan listeden "İki Boyutlu" kategorisine gidin ve MaxiCode'u seçin:



Barkodu seçtikten sonra rapor sayfasına yerleştirin. Eklenen barkoda çift tıklayarak düzenleyici açılır. Ayrıca eklenen nesne üzerinde sağ tıklayarak açılan bağlam menüsündeki düğmeye basarak da kod düzenleyici açılabilir.

Barkod düzenleyicide rakam ve harf içeren bilgiler girilebilir:



FastReport ortamındaki diğer barkodlar gibi, MaxiCode da kodun altında metin bilgisi ile veya metin bilgisi olmadan görüntülenebilir. Bu işlem, barkodun "ShowText" özelliğiyle kontrol edilir.

EAN-8

EAN-8 (Avrupa Madde Numarası) veya GTIN-8 olarak da bilinen kod, 8 rakamdan oluşan kısa bir barkoddur. Rakamlar, 4'er rakamdan oluşan 2 bloğa ayrılmıştır. İlk 2 rakam ürünün menşe ülkesini tanımlar, sonraki 5 rakam kodlanacak bilgiyi içerir ve son rakam ise verilerin doğruluğunu kontrol etmek amacıyla kullanılan bir kontrol basamağıdır.

Bu kod, EAN-13 temel alınarak boyutunun 8 rakama düşürülmesi ile oluşturulmuştur. Bu sayede, küçük ürün ambalajlarına barkod yerleştirme olanağı sağlanmıştır. Barkodda ülke tanımlamasının yer alması nedeniyle uluslararası bir öneme sahip olmuştur.

EAN-8'in kullanım alanları: ürün ve ekipmanların etiketlenmesi.

Her rakam, dikey çizgiler ve boşluklar ile temsil edilen 7 modül ile kodlanır. Boşluk 0, çizgi ise 1 olarak kabul edilirse, rakamların kodlaması şu şekildedir:

- 0 – 0001101;
- 1 – 0011001;
- 2 – 0010011;
- 3 – 0111101;
- 4 – 0100011;
- 5 – 0110001;
- 6 – 0101111;
- 7 – 0111011;
- 8 – 0110111;
- 9 – 0001011.

Bir örnekle görelim:

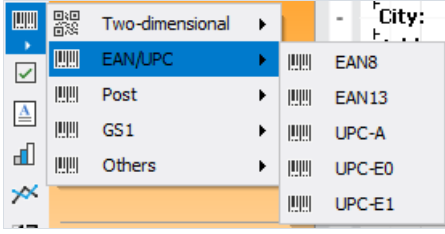


Kodun başlangıcı, 101 dizisiyle başlar ve ardından rakam "3"ün kodu gelir. Daha önce belirtildiği gibi, kod iki bölüme ayrılmıştır. Ayırıcı, başlangıç ve bitiş çizgileri gibi daha uzun çizgiler içerir ve 01010 koduyla ifade edilir.

Dikkatlice bakıldığında, ayırıcıdan sonra çizgilerin yukarıdaki listede yer alan rakamlara karşılık gelmediği görülebilir. Çünkü ikinci bölüm için farklı bir kodlama kullanılır:

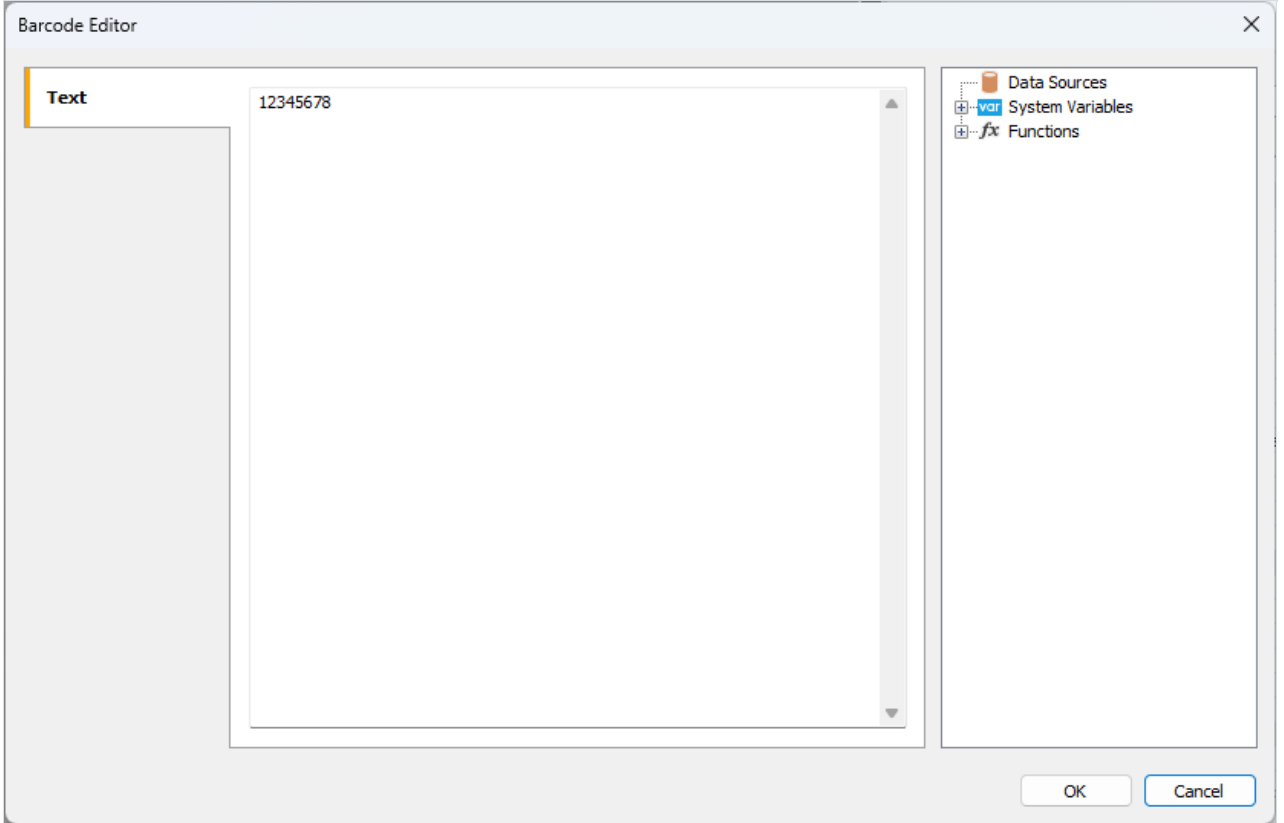
- 0 – 1110010;
- 1 – 1100110;
- 2 – 1101100;
- 3 – 1000010;
- 4 – 1011100;
- 5 – 1001110;
- 6 – 1010000;
- 7 – 1000100;
- 8 – 1001000;
- 9 – 1110100.

FastReport .NET'te EAN-8 barkodu oluşturmak için rapor tasarımcısının bileşenler panelinde bulunan Barkod (Barcode) nesnesini seçin. Açılır listeden "EAN/UPC" kategorisini seçin ve ardından EAN8'i işaretleyin:

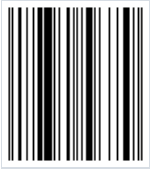


Barkodu seçtikten sonra rapor sayfasına yerleştirin.

Eklenen barkoda çift tıklayarak düzenleyici açılır. Ayrıca barkodun sağ tuşla açılan bağlam menüsündeki butona tıklayarak da düzenleyiciyi açabilirsiniz:



Barkodun altındaki metni gizlemek istiyorsanız, ilgili barkodun özellikler panelinde bulunan ShowText özelliğini bulun ve değerini False olarak ayarlayın:



EAN-13



EAN-13 (European Article Number) barkodu, lineer barkod sembolojileri arasında en yaygın olanıdır. Bu semboloji, 12 basamaklı rakamı kodlamak için kullanılır ve 13. basamak ise kodun bütünlüğünü doğrulamak amacıyla kullanılan kontrol rakamıdır. Yalnızca rakamların kodlanmasına izin verilir.

Kod yapısı şöyledir:

- İlk 2-3 basamak, ürünün üretici ülkesini belirtmek için ayrılmıştır.
- Ardından gelen 4-5 basamak, ürünün üreticisini belirtmek için ayrılmıştır.
- Kalan 3-5 basamak ise işletmedeki ürün numarasını temsil eder.

Daha önce belirtildiği gibi 13. rakam kontrol basamağıdır. Bu rakam, özel bir algoritma kullanılarak önceki 12 rakama dayanılarak otomatik olarak hesaplanır. Bu kontrol rakamı sayesinde barkodun doğru okunup okunmadığı kontrol edilebilir.

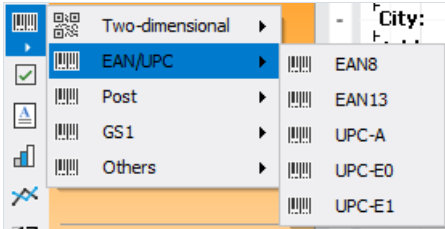
Kodun görüntüleme şekli incelendiğinde, kodun başlangıcında ve sonunda iki çizgiden oluşan barkod dizisi bulunur. Kod, çizgiler 1, boşluklar ise 0 olarak ikili (binary) biçimde ifade edilirse, bu barkod dizisi "101" şeklinde görülür. Ayrıca barkod, tam ortadan benzer bir barkod dizisi ile ikiye ayrılır.

Her rakam 7 modülden (çizgi ve boşluklardan) oluşur. Rakamlar için özel kod dizilerini içeren üç tablo mevcuttur. Kodun ilk ve ikinci yarısı, farklı tablo dizilerini kullanır.

Bu barkod hem soldan sağa hem de sağdan sola olmak üzere herhangi bir yönde okunabilir. Okuma kolaylığı ve basitliği nedeniyle perakende ticarete ürün etiketlemede hızla popüler hale gelmiştir.

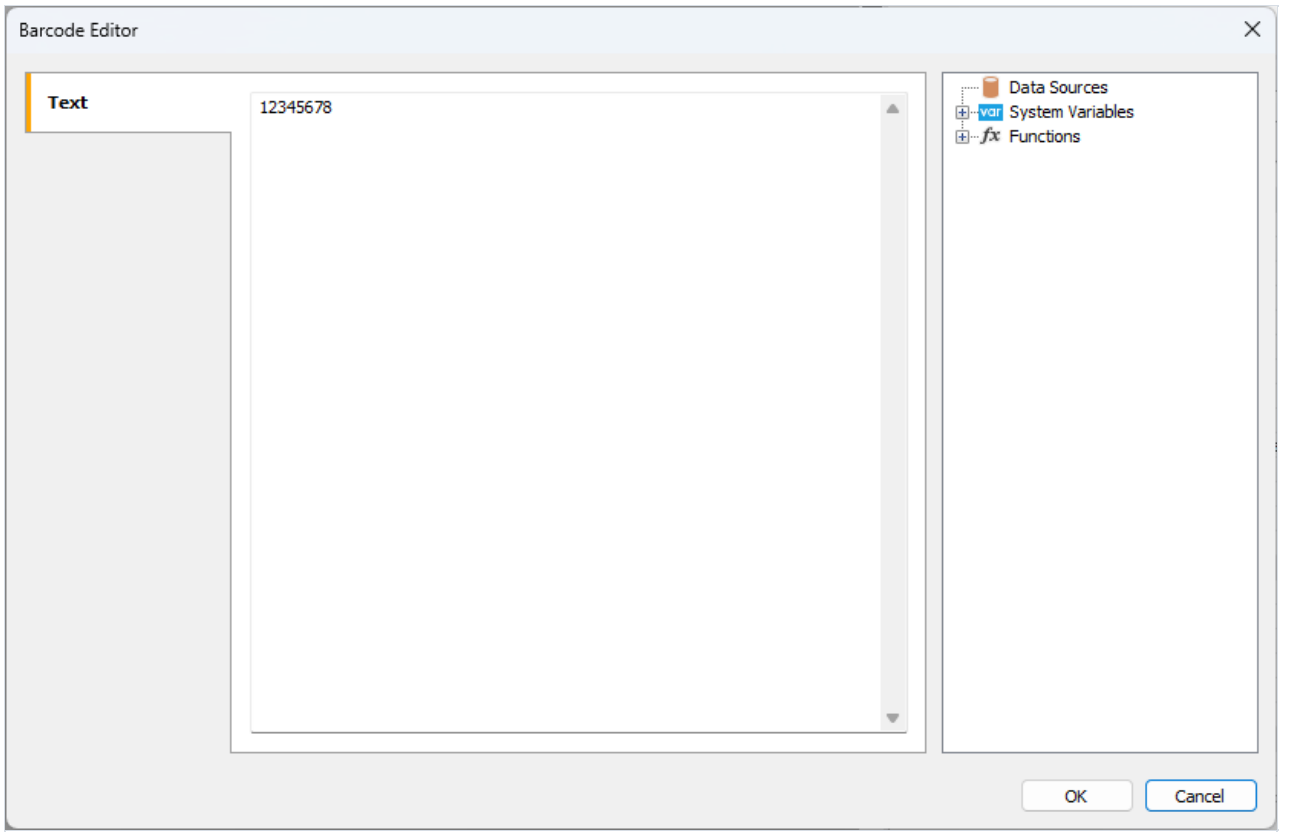
Bu barkodun olumsuz yönlerinden biri, düşük kapasitesi, yani kodlanabilir dizinin uzunluğunun kısıtlı olmasıdır.

FastReport .NET içerisinde EAN-13 barkodu oluşturmak için rapor tasarımcısının bileşenler panelinden Barkod (Barcode) nesnesini seçin. Açılır listeden "EAN/UPC" kategorisine ve ardından "EAN13" seçeneğine gidin:

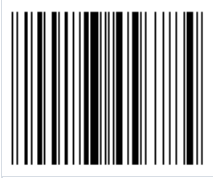


Barkodu seçtikten sonra rapor sayfasına yerleştirin.

Eklenen barkoda çift tıklandığında barkod düzenleyicisi açılır. Aynı düzenleyici, eklenen nesne üzerinde sağ fare tuşuyla çağrılan bağlam menüsündeki butona basılarak da açılabilir:



Eğer barkodun altındaki metnin görüntülenmesini istemiyorsanız, ilgili barkodun özellikler denetleyicisinde (Properties Inspector) ShowText özelliğini bulun ve değerini False olarak ayarlayın:



UPC-A



Evrensel Ürün Kodu (UPC), 12 karakterin kodlanmasına olanak tanıyan bir lineer barkoddur. 1973 yılında Uniform Grocery Product Code Council (Birleşik Market Ürün Kodu Konseyi) ile IBM tarafından ortaklaşa geliştirilmiştir. Yapısı ve amacı, yaygın olarak bilinen EAN-13 koduna benzerdir. UPC kodları Kuzey Amerika için tasarlanmıştır, EAN kodları ise Avrupa için kullanılmaktadır.

- Kodun başlangıcını belirten başlangıç karakteri;
- Ürün türünü belirten önek – 1 karakter;
- Üretici kodu – 5 karakter;
- Ürün kodu – 5 karakter;
- Kontrol basamağı – 1 karakter; önceki 11 rakamdan Modulo 10 formülü kullanılarak hesaplanır.
- Kodun sonunu belirten bitiş karakteri.

Koddan önce ve sonra yaklaşık 9 modüllük boş bir alan olmalıdır. Bu, barkodun tarayıcı tarafından doğru şekilde tanınmasını sağlamak için gereklidir.

Kod uzunluğu 12 karakterdir; bunların yalnızca 11'i kodlanmış veridir, geri kalan 1 karakter ise kontrol basamağıdır.

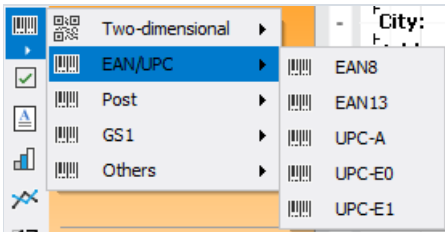
Her karakter, iki çubuk ve iki boşlukla kodlanır. Bir çubuk veya boşluk 1, 2, 3 ya da 4 modül genişliğinde olabilir (bir modül 0,33 mm'dir).

UPC-A barkodu, ABD ve Kanada'da oldukça yaygındır. Süpermarketlerde ürünleri etiketlemek için kullanılır.

Bu barkod, kompakt boyutu, taranmasının kolaylığı ve okuma hatalarına karşı koruma sağlayan kontrol basamağına sahip olması nedeniyle büyük bir popülerlik kazanmıştır.

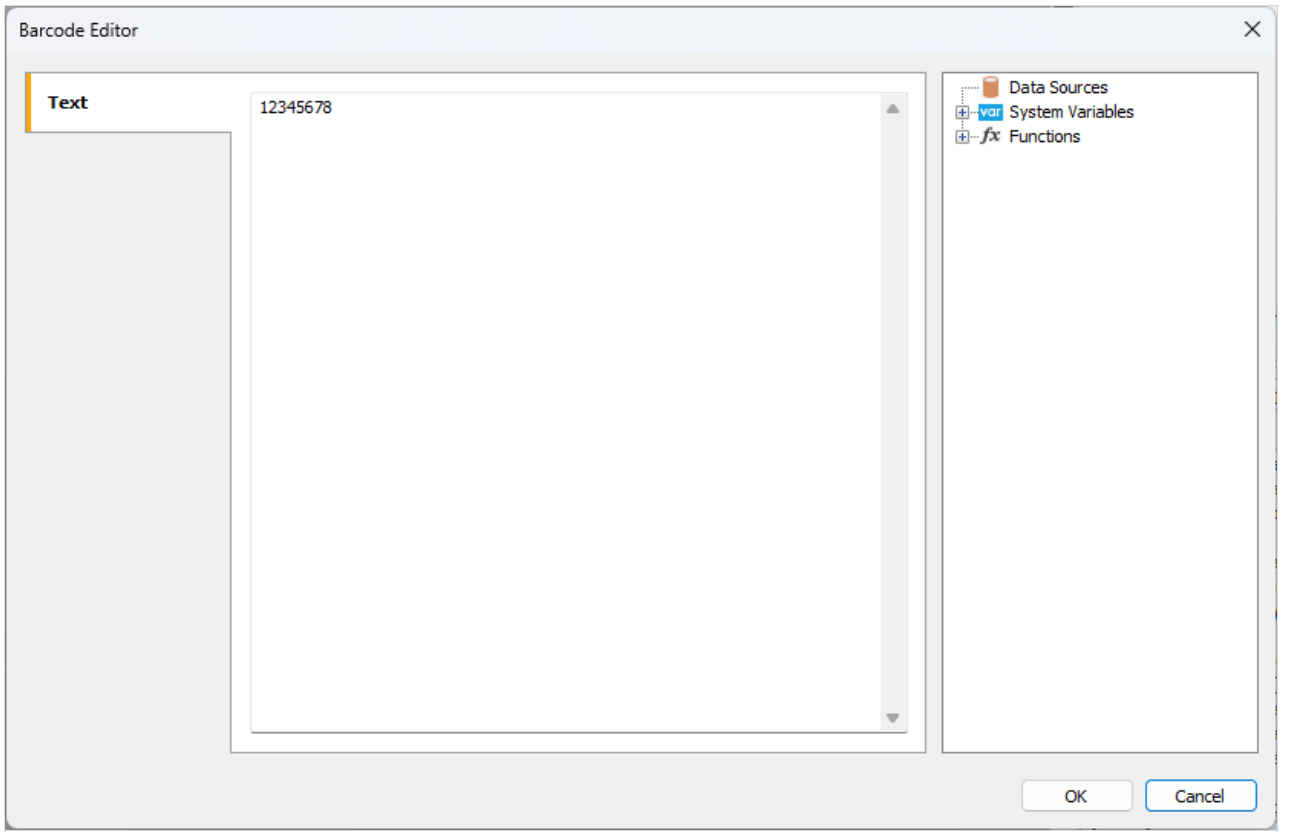
Kodun dezavantajları arasında yalnızca rakamları kodlayabilme yeteneği ve kodun küçük boyutu, bu da kullanım alanını sınırlamaktadır.

FastReport .NET'te bir UPC-A barkodu oluşturmak için, Rapor Tasarımcısı'ndaki Bileşenler Panelinden Barkod nesnesini seçin. Açılır listeden "EAN/UPC" kategorisine gidin ve ardından UPC-A seçeneğini seçin.



Barkodu seçtikten sonra, onu Rapor Sayfasına yerleştirin.

Eklenen barkodu düzenleyiciyi açmak için çift tıklayın. Ayrıca, eklenen nesneye sağ tıklayarak erişilen bağlam menüsündeki düğmeye tıklayarak da barkod düzenleyicisini açabilirsiniz:



Tüm Barkod (Barcode) nesneleri bir dizi özelliğe sahiptir. En sık kullanılanlar şunlardır:

(Buradan sonra istersen, özelliklerin çevirisini de ekleyebilirim.)

Özellik	Açıklama
Döndürme (Angle)	Nesnenin dönüş açısını sabit açılardan biri olarak ayarlamaya olanak tanır – 0, 90, 180, 270 derece.
Yakınlaştırma (Zoom)	Barkodun ölçeklendirilmesini ayarlar. Bu özellik yalnızca "Otomatik Boyut" özelliği ile birlikte kullanılır.
Otomatik Boyut (AutoSize)	Bu özellik etkinleştirilirse, nesne barkodu tamamen gösterecek şekilde genişletilir. Özellik devre dışı bırakılırsa, barkod nesnenin boyutlarına sığacak şekilde genişletilir.
Metni Göster (ShowText)	Barkodun alt kısmında metnin gösterilip gösterilmeyeceğini belirler.
Veri Alanı (DataColumn)	Nesnenin metninin yükleneceği veri alanını belirtir.
İfade (Expression)	Nesnenin metnini döndüren ifadeyi belirtir.
Metin (Text)	Nesnenin metni.
Dolgu Boşluğu (Padding)	Nesnenin kenarlarından olan boşlukları piksel cinsinden ayarlamaya olanak tanır.
Geniş Çubuk Oranı (WideBarRatio)	Bu özellik tüm doğrusal barkodlarda bulunur. Barkoddaki geniş çubukların göreceli boyutunu belirler.
Kontrol Toplamı (CalcChecksum)	Bu özellik birçok doğrusal barkodda bulunur. Kontrol toplamının otomatik olarak hesaplanıp hesaplanmayacağını belirler. Bu özellik devre dışı bırakılırsa, kontrol toplamı nesnenin metninde yer almalıdır.

**Dikey Taşıyıcı Çubukları
Göster
(DrawVerticalBearerBars)**

Bu özellik birçok doğrusal barkodda bulunur. Kontrol toplamının otomatik olarak hesaplanıp hesaplanmayacağını belirler. Bu özellik devre dışı bırakılırsa, kontrol toplamı nesnenin metninde yer almalıdır.

ShowText özelliği devre dışı bırakılırsa, barkod şu şekilde görünecektir:



UPC-E



UPC-E, perakende satıştaki ürün bilgilerini kodlamak için oluşturulmuş doğrusal, sayısal bir barkod türüdür. Bildiğiniz üzere UPC standardı EAN üzerine kuruludur. Amerikan UPC-A barkodu, Avrupa'nın EAN-13 barkodunun bir benzeridir. Temel fark yalnızca barkod uzunluğundadır: UPC-A 12 karakter, EAN-13 ise 13 karakter uzunluğundadır. EAN-13'ün, küçük ambalajlarda daha kompakt yerleştirme amacıyla oluşturulan EAN-8 adı verilen kısaltılmış versiyonu bulunmaktadır. Benzer şekilde, UPC-A'nın da daha hafif bir sürümü olan ve 6 karakterden oluşan UPC-E versiyonu bulunmaktadır. Genellikle UPC-A'da kullanılan ürün ve imalatçı kodlarında çok sayıda sıfır bulunur. Bu sıfırların kaldırılmasıyla, bilgi kaybı olmaksızın barkod uzunluğu 12 karakterden 6 karaktere düşürülmüştür.

Kodun yapısı şu şekildedir:

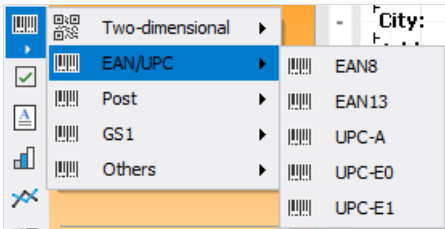
- İlk karakter, sayı sistemini belirten ön ek olup, 0 veya 1 değerini alabilir. Bu, kodun nasıl yorumlanacağını etkiler.
- Daha sonra, imalatçı ve ürün kodunu içeren 6 karakter gelir (UPC-A'da bunlar ayrıdır).
- Son karakter ise, kodun doğruluğunu kontrol etmek için kullanılan kontrol basamağıdır.

UPC-E, aslında 6 ila 8 karakter arasında kodlama yapabilir. Minimum 6 karakter seti kodlanırken sayı sistemini belirten ilk karakter ve kontrol rakamı hariç tutulur. 7 karakter kodlanırken ise yalnızca kontrol rakamı hariç tutulur. 8 karakterlik kod, yukarıda açıklanan tüm yapı unsurlarını içerir. UPC'nin kısaltılmış sürümünde, kompaktlık için başlangıç ve bitiş sembolleri bulunmaz.

Seçilen sayı sistemine bağlı olarak kontrol rakamı otomatik olarak hesaplanır. Böylece UPC-E standardı, UPC-E0 ve UPC-E1 olarak ikiye ayrılır. UPC-E'nin yalnızca ilk rakamı 0 olan ve kodunda sıfır dizisi bulunan GTIN-12 yapılarını kodlamaya izin verdiğine dikkat edilmelidir. Bu sıfır dizisi, UPC-E kodunda kaldırılır.

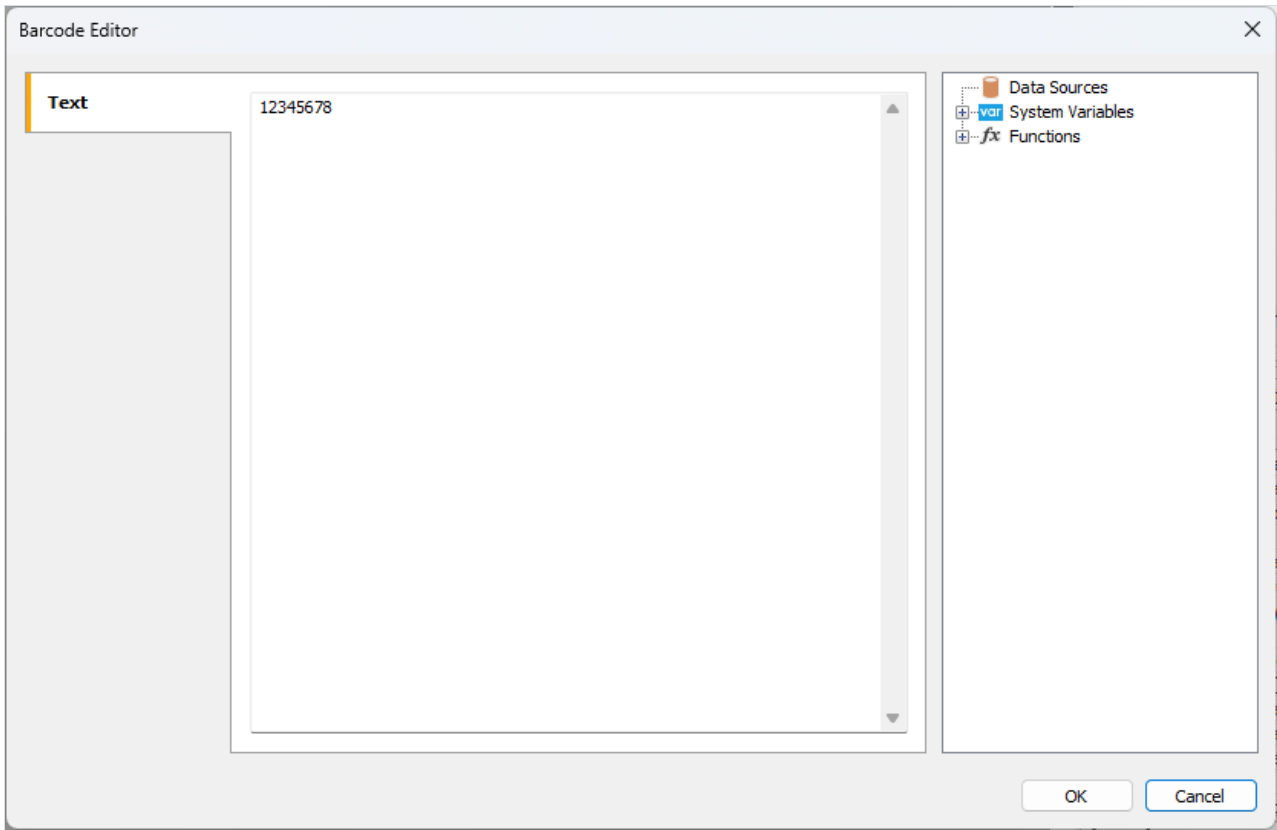
UPC-A'da olduğu gibi, UPC-E'deki her veri karakteri iki çubuk ve iki boşlukla kodlanır. Çubuk veya boşluk 1, 2, 3 veya 4 modül genişliğinde olabilir (bir modül 0,33 mm'ye eşittir).

FastReport .NET'te UPC-E0 veya UPC-E1 barkodlarını oluşturmak için rapor tasarımcısındaki bileşen panelinden Barkod (Barcode) nesnesini seçin. Açılır listede "EAN/UPC" kategorisine, ardından da UPC-E0 veya UPC-E1 seçeneklerine gidin.



Barkodu seçtikten sonra rapor sayfasına yerleştirin.

Eklenen barkoda çift tıklayarak düzenleyici açılır. Ayrıca eklenen nesnenin sağ tık bağlam menüsündeki butona basarak da düzenleyiciyi açabilirsiniz:



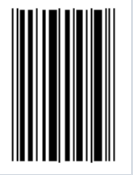
Barkod (Barcode) türündeki tüm nesneler belirli özelliklere sahiptir. En sık kullanılan özellikler şunlardır:

Özellik	Açıklama
Döndürme (Açı)	Nesnenin 0, 90, 180 veya 270 derece gibi sabit açılardan birine döndürülmesini sağlar.
Ölçek (Yakınlaştırma)	Barkodun ölçeklendirilmesini belirler. Bu özellik yalnızca "Otomatik Boyutlandırma" özelliğiyle birlikte kullanılır.
Otomatik Boyutlandırma (AutoSize)	Bu özellik etkinleştirilirse, nesne barkodun tamamını gösterecek şekilde genişletilir. Özellik devre dışı bırakıldığında ise barkod, nesnenin boyutlarına göre ölçeklendirilir.
Metni Göster (ShowText)	Barkodun alt kısmında metnin gösterilip gösterilmeyeceğini belirler.
Veri Alanı (DataColumn)	Nesnenin metninin yükleneceği veri alanını belirtir.
İfade (Expression)	Nesnenin metnini döndüren ifadeyi belirtir.
Metin (Text)	Nesnenin metni.
Dolgu (Padding)	Nesnenin kenarlarından olan boşlukları piksel cinsinden ayarlamayı sağlar.
Geniş Çubuk Oranı (WideBarRatio)	Bu özellik tüm doğrusal barkodlarda bulunur. Barkoddaki geniş çizgilerin göreceli boyutunu belirler.
Kontrol Toplamı (CalcChecksum)	Bu özellik birçok doğrusal barkodda bulunur. Kontrol toplamının otomatik olarak hesaplanıp hesaplanmayacağını belirler. Bu özellik devre dışı bırakıldığında, kontrol toplamı nesnenin metninde yer almalıdır.

**Dikey Taşıyıcı Çubukları
Göster
(DrawVerticalBearerBars)**

Bu özellik etkinleştirilirse, nesnede yan çizgiler görüntülenir.

ShowText özelliği devre dışı bırakılırsa, kod aşağıdaki gibi görünecektir:



Deutsche Post Identcode



Deutsche Post Identcode doğrusal barkodu, yaygın olarak kullanılan 2 of 5 Interleaved barkod türü temel alınarak oluşturulmuştur ve onun sembolojisini kullanmaktadır. Farklar, barkodun kullanım amacındadır. Identcode, posta göndericisini tanımlama amacıyla özellikle Deutsche Post için tasarlanmıştır. Gönderici verilerinin gerekli boyutundan dolayı, barkod uzunluğu Interleaved'in 14 hanesine kıyasla 12 haneye düşürülmüştür. Ancak gerçek kodlanan verinin uzunluğu 13 karakterdir, çünkü 13. karakter, kodun doğru okunmasını doğrulamak için gerekli olan kontrol hanesidir.

Identcode yapısı aşağıdaki gibidir:

- Başlangıç sembolü;
- Birincil dağıtım merkezi tanımlayıcısı için 2 rakam;
- Müşteri tanımlayıcısı için 3 rakam;
- Posta numarası için 6 rakam;
- Kontrol rakamı;
- Bitiş sembolü.

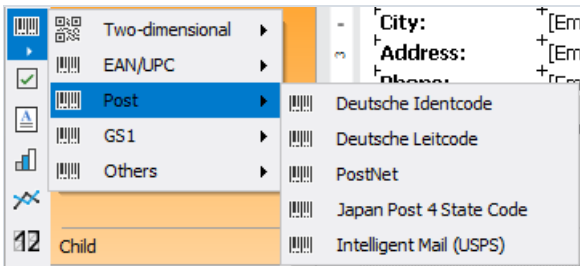
Çizgi ve boşluklar ikili semboller olarak gösterildiğinde başlangıç sembolü 1010 şeklindedir. Bu sembol, okuma yönünü belirlemeye yardımcı olur.

Verilerin rakamlarını kodlamak için 2 of 5 Interleaved kod tablosu kullanılır. Her veri sembolü, 5 çizgi veya 5 boşluktan oluşur; bunlardan iki tanesi geniş, kalanlar ise dar olmalıdır. Tüm 2 of 5 kod ailesi barkodları hem çizgileri hem de boşlukları kodlama için kullandığından, barkodun uzunluğu her zaman çift sayıdadır. Çünkü 5 çizgi ve aradaki 5 boşluktan oluşan bir set, aynı anda iki rakamı kodlar. Bu da kodun yüksek yoğunlukta ve kompakt olmasını sağlar.

11 rakamlık veri dizisinden sonra Modulo 10 algoritmasıyla hesaplanan bir kontrol rakamı gelir.

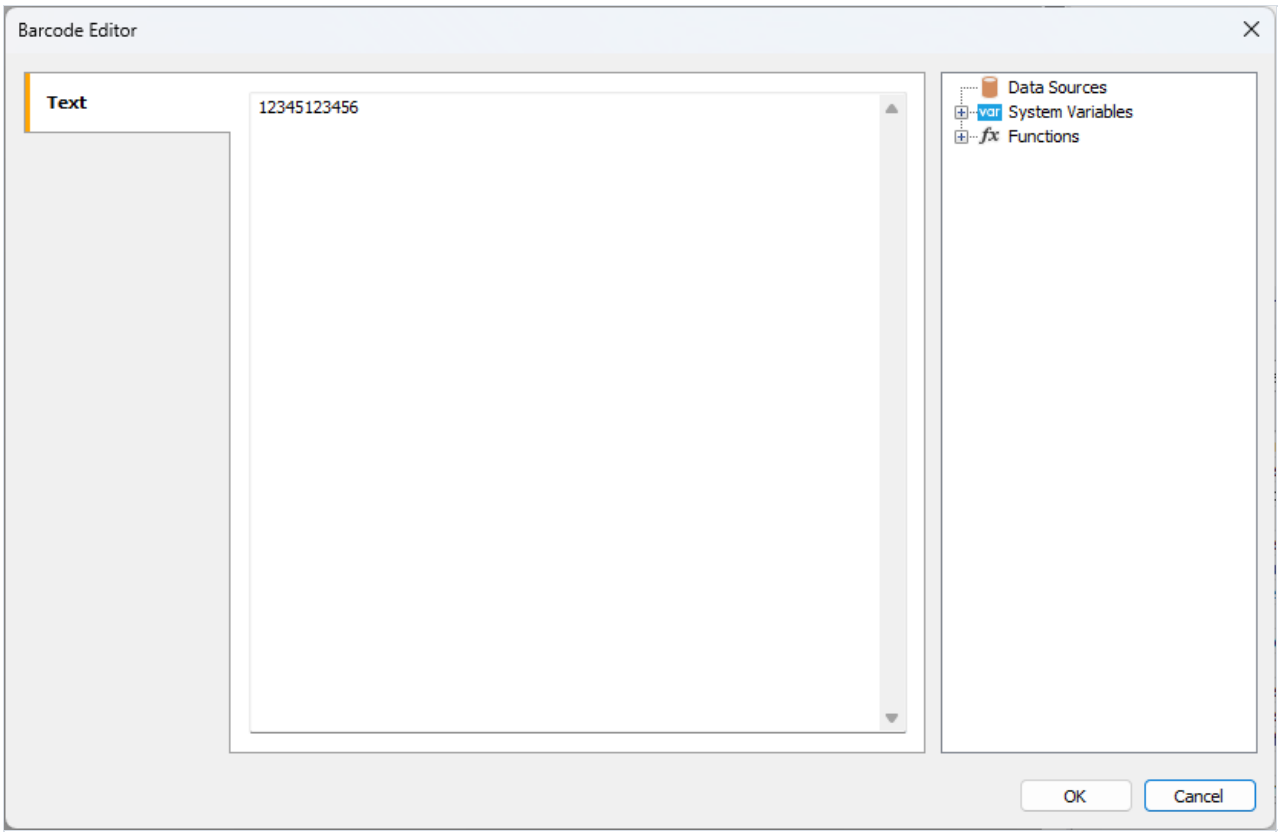
Kod, 101 dizisiyle kodlanan bitiş sembolüyle sonlanır.

FastReport .NET ortamında Deutsche Post Identcode barkodunu oluşturmak için rapor tasarımcısındaki bileşenler panelinden Barkod (Barcode) nesnesini seçin. Açılır menüden "Posta" kategorisine gidin ve buradan Deutsche Identcode seçeneğini seçin:

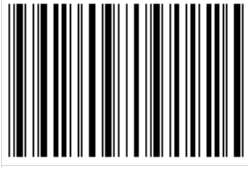


Barkod seçildikten sonra rapor sayfasına yerleştirilir.

Eklenen barkoda çift tıklayarak düzenleyici açılır. Ayrıca barkod nesnesinin bağlam menüsünde bulunan butona basarak veya sağ fare düğmesine tıklayarak açılan bağlam menüsü aracılığıyla da düzenleyiciye erişilebilir:



Eğer barkodun altındaki metnin gizlenmesi gerekiyorsa, ilgili barkodun özellikler panelinde bulunan ShowText özelliği bulunmalı ve değeri False olarak ayarlanmalıdır:



Deutsche Post Leitcode



Deutsche Post Leitcode linear barcode, or simply LeitCode, is based on the widely known 2 of 5 Interleaved code. It allows you to encode 13 digital characters (0-9) and is used in the German postal service (Deutsche Post) and DHL to automate the sorting of mail items. A distinctive feature of Leitcode is its well-defined data structure, which distinguishes it from 2 of 5 Interleaved barcodes.

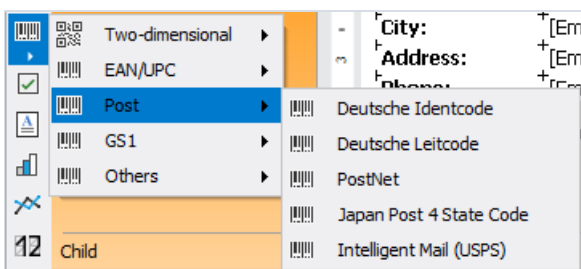
Code structure:

- start character;
- 1-5 – ZIP code;
- 6-8 – street number;
- 9-11 – house number;
- 12-13 – product code;
- 14 – check digit;
- stop character.

Another difference between Leitcode and 2 of 5 Interleaved is the use of the Modulo 10 algorithm to calculate the mandatory checksum.

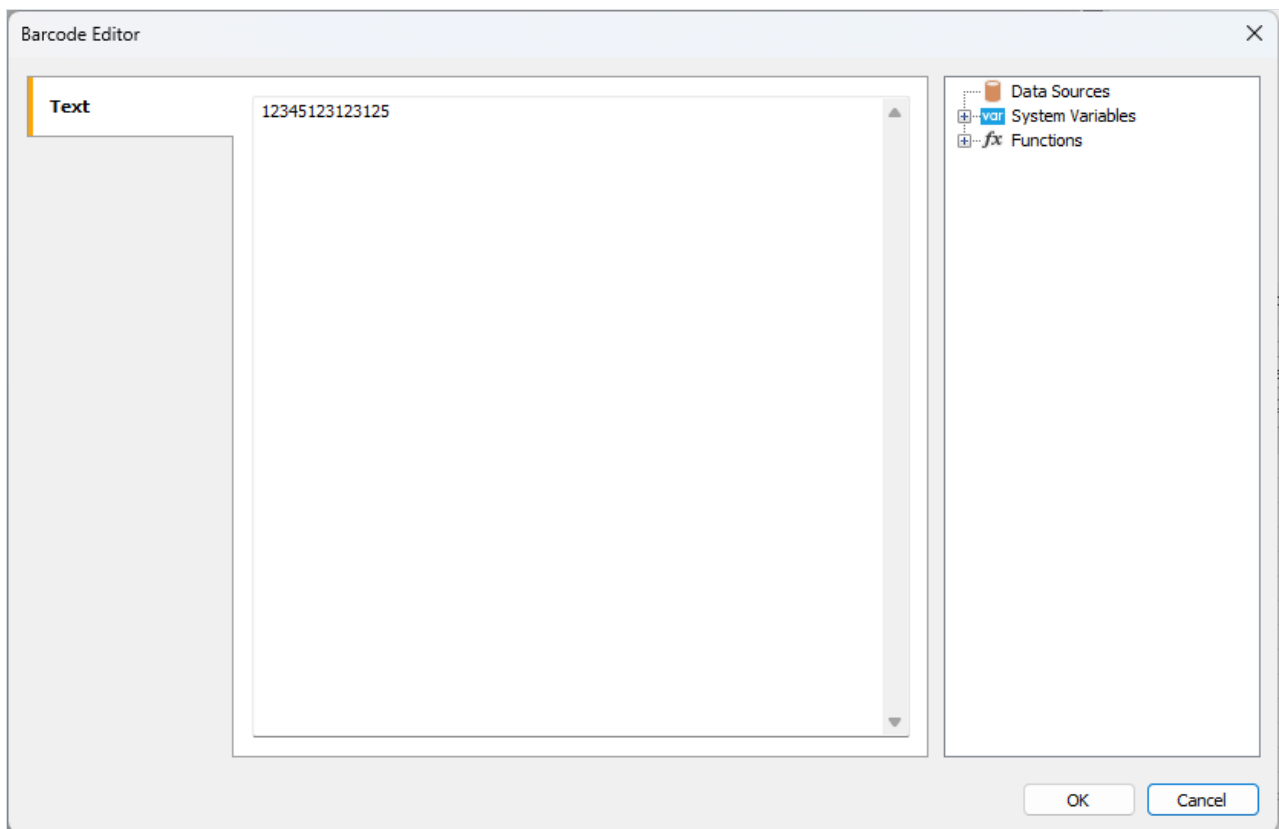
As in the case with 2 of 5 barcodes, Leitcode uses both bars and spaces to encode data. This approach has helped reduce the size of the code. The data character in Leitcode is encoded with five modules (bars/spaces), where two bars are wide and three are narrow. The same applies to spaces. It is thanks to this combination that the barcode got its name 2 of 5.

To generate a Deutsche Post Leitcode barcode in FastReport .NET, select the Barcode object  at the Components Panel in the Report Designer. In the drop-down list, navigate to the "Post" category, and then choose Deutsche Leitcode:

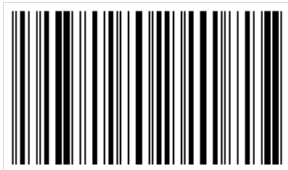


After selecting the barcode, place it on the Report Page.

Double-click on the added barcode to open the editor. You can also open the barcode editor by clicking the button  in the context menu of the added object, accessed by right-clicking:



If you want to hide the text under the barcode, locate the `ShowText` property in the property inspector of the corresponding barcode and set its value to `False` :



PostNet



Bu sayısal doğrusal barkod, özellikle Amerika Birleşik Devletleri posta hizmetinde kullanılmak üzere oluşturulmuştur ve postaların makineli olarak ayrıştırılması için tasarlanmıştır. Bu barkodun dış görünümü, perakende ticarete ürünlerin etiketlenmesinde kullanılan UPC ve EAN kodlarından oldukça farklıdır. Çubuklar farklı yüksekliklere sahiptir ve genel olarak barkodun yüksekliği zarf üzerine yerleştirmeyi kolaylaştıracak şekilde küçüktür.

ABD’de ZIP kodunun uzunluğu başlangıçta 5 karakterdi, ancak kapasitenin yetersiz kalması nedeniyle 1983 yılında 9 karaktere çıkarıldı. Daha sonra bu koda 2 karakter daha eklendi. Sonuç olarak, PostNet barkodunun üç farklı uzunluğu bulunmaktadır:

- 5 karakter;
- 9 karakter;
- 11 karakter.

Barkodun yapısı şu şekildedir:

- başlangıç karakteri;
- veri karakterleri;
- kontrol karakteri;
- bitiş karakteri.

Her karakter 5 çubukla kodlanır. Başlangıç ve bitiş karakterleri tek bir uzun çubukla temsil edilir.

Aşağıda, sayıların karşılık geldiği çubuk kodlarının listesi verilmektedir:

Bir karakterin kodlanması için gereken çubuk sayısının oldukça fazla olması nedeniyle, PostNet barkodunun boyutu oldukça büyüktür. Ayrıca bu barkod yalnızca sayıları kodlamaya olanak tanır. Bu dezavantajlar, barkodun popülaritesinin azalmasına yol açmış ve günümüzde yerini daha modern bir çözüm olan Intelligent Mail barkoduna bırakmaya başlamıştır.

FastReport .NET'te PostNet barkodu için karakter sayısına ilişkin herhangi bir sınırlama bulunmamaktadır. Ancak, daha önce de belirtildiği gibi, standart üç farklı boyut seçeneğini öngörmektedir.

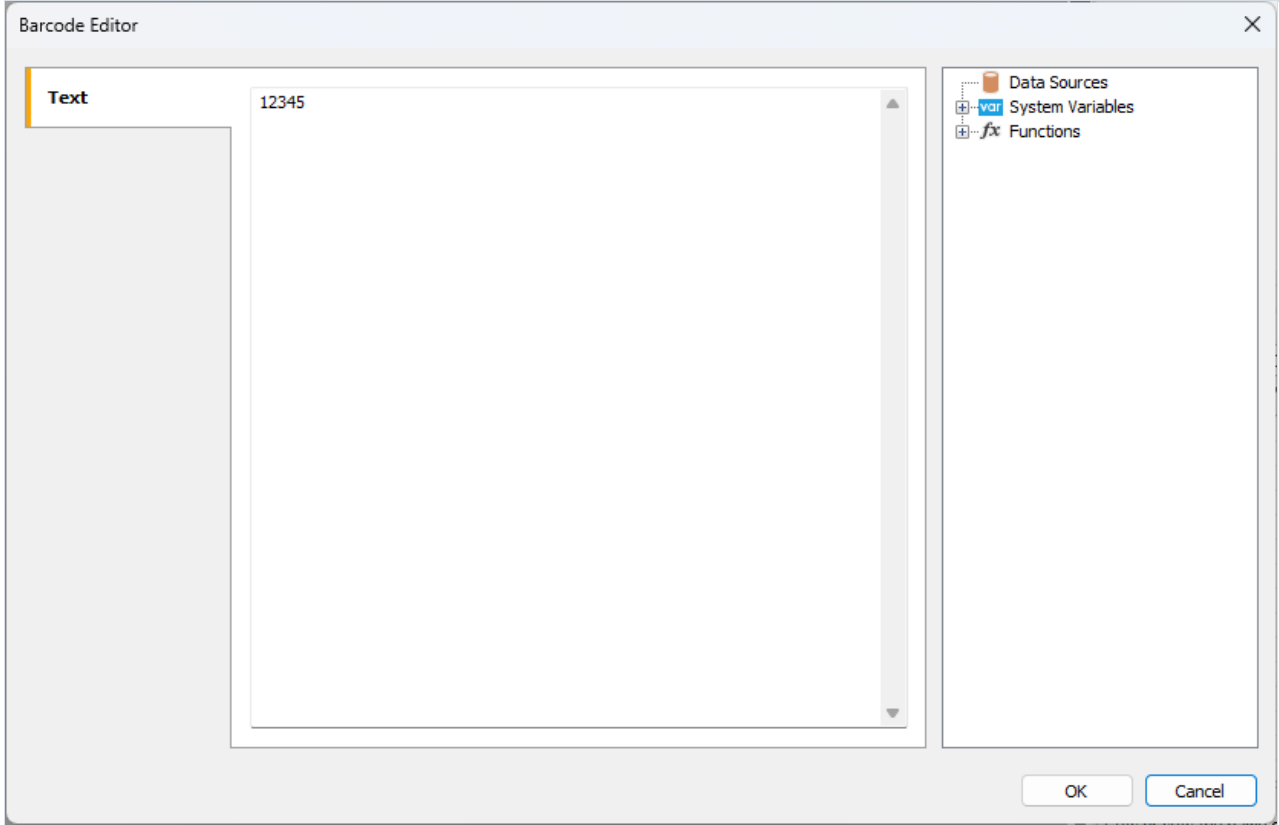
FastReport .NET'te PostNet barkodunu oluşturmak için, rapor tasarımcısında bileşenler panelinden **Barkod (Barcode)** nesnesini seçin. Açılır listeden **"Posta" (Postal)** kategorisine gidin ve ardından **PostNet** seçeneğini seçin:

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.

8. 
9. 
10. 

Barkodu seçtikten sonra, onu rapor sayfasına yerleştirin.

Eklenecek barkoda çift tıklayarak düzenleyici penceresini açabilirsiniz. Ayrıca, eklenen nesneye sağ tıklayarak açılan bağlam menüsünden **düzenleyici butonuna** basarak da düzenleyiciyi açabilirsiniz:



FastReport .NET'teki tüm barkodlarda olduğu gibi, PostNet barkodunun da nesne özellikleri penceresinde düzenlenebilen çeşitli özellikleri bulunmaktadır:

özellik	açıklama
Döndürme (Angle)	Nesnenin 0, 90, 180 veya 270 derece gibi sabit açılardan biriyle döndürülmesini sağlar.
Ölçek (Zoom)	Barkodun ölçeklendirilmesini ayarlar. Bu özellik yalnızca "Otomatik Boyut" özelliği ile birlikte kullanılır.
Otomatik Boyut (AutoSize)	Bu özellik etkinleştirildiğinde, nesne barkodu tamamen gösterecek şekilde genişletilir. Özellik devre dışı bırakıldığında ise barkod, nesnenin boyutlarına göre genişletilir.
Metni Göster (ShowText)	Barkodun alt kısmında metnin gösterilip gösterilmeyeceğini belirler.
Veri Sütunu (DataColumn)	Nesnenin metninin yükleneceği veri alanını belirtir.
İfade (Expression)	Nesnenin metnini döndüren ifadeyi belirtir.
Metin (Text)	Nesnenin metni.

Dolgu (Padding)	Nesnenin kenarlarından olan boşlukları piksel cinsinden ayarlamaya olanak tanır.
Geniş Çubuk Oranı (WideBarRatio)	Bu özellik tüm doğrusal barkodlarda bulunur. Barkoddaki geniş çubukların görelî boyutunu belirler.
Kontrol Toplamı (CalcChecksum)	Bu özellik birçok doğrusal barkodda bulunur. Kontrol toplamının otomatik olarak hesaplanıp hesaplanmayacağını belirler. Bu özellik devre dışı bırakıldığında, kontrol toplamı nesnenin metninde yer almalıdır.
Dikey Taşıyıcı Çubukları Göster (DrawVerticalBearerBars)	Bu özellik etkinleştirildiğinde, nesnede yan çizgiler görüntülenir.

Intelligent Mail

IMb (Intelligent Mail barkodu), ayrıca USPS OneCode olarak da bilinir; yüksekliğe modüle edilmiş bir barkoddur. Bu barkod, dört durumlu semboloji kullanarak 65 dikey çubukta posta gönderisi verilerinden en fazla 31 haneyi kodlayabilir. Dört farklı “çubuk” durumunun kullanılması, tek bir barkotta daha fazla bilginin kodlanmasına imkân tanır.



Intelligent Mail standardı, ABD’de daha önce posta gönderilerinde kullanılan POSTNET ve PLANET standartları temel alınarak oluşturulmuştur. Bu standartlar, posta kodlarını kodlamayı mümkün kılar ve esasen postanın sınıflandırılması ile takibi amacıyla hizmet eder.

POSTNET beş haneli posta kodunu, dört haneli ZIP+4 kodunu ve iki haneli teslim noktası kodunu kodlayabilir. POSTNET barkodları 32 ila 62 çubuk arasında değişken uzunluğa sahipken, PLANET barkodları 62 veya 72 çubuktan oluşur. Bu barkodlar yüksekliğe modüle edilmiş (dikey sütunlar farklı uzunluktadır) olup iki duruma sahiptir (kısa ve uzun çubuklar). Kodlanan verideki her rakam, beş çubukluk bir grup ile temsil edilir. POSTNET barkodunda her grupta tam olarak iki uzun çubuk bulunurken, PLANET barkodunda üç uzun çubuk bulunur.

IMb teknolojisi, PLANET ve POSTNET’in işlevselliğini etkili şekilde tek bir barkotta birleştirir. Bu sayede posta kullanıcıları, birden çok posta hizmetine katılım için tek bir barkod kullanabilir; bu da tekil gönderilerin izlenme olanağını artırır ve posta akışının daha geniş bir görünürlüğünü sağlar. Bu barkodun kullanımı, Posta Servisi’ne gönderi endüstrisine yönelik birden çok hizmet sunmanın yanı sıra posta izleme ve performans takibi fonksiyonları ekleyerek maliyetlerin düşürülmesine katkı sağlar. POSTNET’te olduğu gibi, IMb de kod bütünlüğünü doğrulamak ve zarar görmüş kodu onarmak için bir denetleme toplamına (kontrol toplamı) sahiptir.

POSTNET ile karşılaştırıldığında, IMb çok daha yüksek veri kapasitesine sahiptir (31 karaktere karşı 11 karakter). Ayrıca, yönlendirme kodunun yanı sıra Intelligent Mail barkodunda dört ek alan bulunur: barkod tanımlayıcısı, hizmet türü tanımlayıcısı (STID), posta programı tanımlayıcısı (MID) ve seri numarası. Bu ek alanlar, posta kullanıcılarının posta sınıfını belirlemelerine, almak istedikleri hizmetleri (örneğin izleme ve adres düzeltme) tanımlamalarına ve posta gönderilerini tekil olarak tanımlamalarına olanak tanır.

Dezavantajlar arasında kod uzunluğunun oldukça fazla olması ve yalnızca sayıları kodlayabilme yeteneği sayılabilir.

Intelligent Mail aşağıdaki bilgileri kodlamaya imkân tanır:

Bilgi	Açıklama
Barkod tanımlayıcısı (2 karakter)	Amerika Birleşik Devletleri Posta Servisi tarafından atanır.
Hizmet türü tanımlayıcısı (3 karakter)	Posta sınıfı veya diğer hizmet.
Gönderici tanımlayıcısı (6 veya 9 karakter)	Amerika Birleşik Devletleri Posta Servisi tarafından atanmış belirli şirkete ait tanımlayıcıdır.
Seri numarası (6 veya 9 karakter)	Posta göndericisine, belirli bir alıcı veya haneyi tanımlamak amacıyla bir numara atanır.

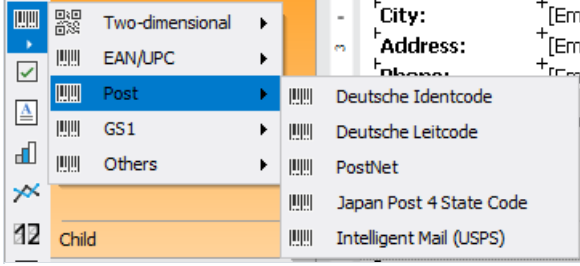
Teslimat noktası posta kodu (11 karakter)

İsteğe bağlı alan.

Çubuklar farklı yükseklik, yön (yukarı, aşağı) ve kalınlığa sahiptir.

Bu barkod sadece zarf üzerine değil, doğrudan belgenin üzerine de yazdırılabilir ve zarf üzerindeki özel şeffaf pencereden okunabilir.

FastReport .NET içerisinde Intelligent Mail barkodunu oluşturmak için rapor tasarımcısının bileşenler panelinden Barkod (Barcode) nesnesini seçin. Açılan listede "Posta" kategorisini bulun ve ardından Intelligent Mail (USPS)'i seçin:



Barkodu seçtikten sonra rapor sayfasına yerleştirin.

Eklenen barkoda çift tıklayarak düzenleyiciyi açabilirsiniz. Ayrıca eklenen nesneye sağ tıklayarak açılan bağlam menüsündeki düğmeye basarak da barkod düzenleyicisini açabilirsiniz.

FastReport'ta barkodun minimum uzunluğu 20 karakterdir. Bunun sebebi, 21. karakterden itibaren, isteğe bağlı olan teslimat noktası posta kodunun başlamasıdır.

Tek bir rakamın değiştirilmesi barkodun tamamen değişmesine neden olur:



GS1-128

Bu barkod, EAN-128 ve UCC-128 olarak da bilinir. Tek boyutlu (1D) bir çizgi barkoddur ve hem sayıları hem de harfleri kodlama imkânı sunar. En çok lojistik alanında, ambalajların etiketlenmesinde yaygın olarak kullanılmış olsa da, başka sektörlerde de uygulanır. GS1 Logistic Label standardı, GS1-128 kodunun kullanıldığı lojistik yük etiketini tanımlar.

Kodun oluşturulmasında, Application Identifiers (AI, Uygulama Tanımlayıcıları) olarak adlandırılan Code-128 tanımlayıcı seti kullanılır. Her AI, belirli bir veri türünü temsil eder ve örneğin (37) şeklinde parantez içinde belirtilir. Parantez içindeki veri türünün bilinmesi, sonraki verinin doğru yorumlanmasını sağlar.

Önce parantez içinde tanımlayıcı kod, ardından onun değeri yazılır. Daha sonra boşluksuz biçimde bir sonraki tanımlayıcı kod ve değeri eklenir. Bu sayede kod şu yapıda oluşur:

(01)12646846874672(10)ABC11(15)100420

Kodun uzunluğu, parantezler hariç olmak üzere 48 karakterle sınırlıdır.

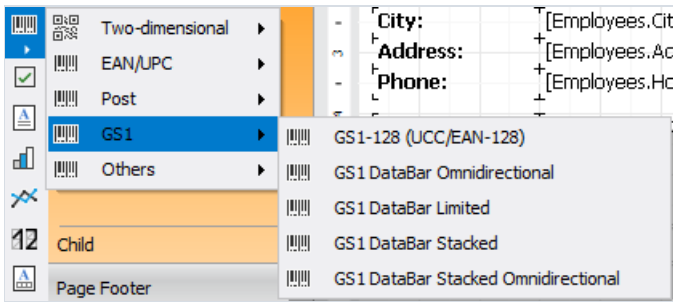
Desteklenen veri türü tanımlayıcılarının listesi oldukça geniştir; bazı örnekler şunlardır:

kod	açıklama	format
00	Küresel benzersiz seri sevkiyat konteyner kodu (Serial Shipping Container Code (SSCC))	N2+N18
01	Küresel benzersiz ticari ürün numarası (Global Trade Item Number (GTIN))	N2+N14
02	Yük içindeki ticari ürünlerin GTIN'i (GTIN of Contained Trade Items)	N2+N14
10	Parti/Lot Numarası (Batch/Lot Number)	N2+X..20
11	Üretim Tarihi (Production Date (YYMMDD))	N2+N6
12	Faturanın ödeneceği son tarih (Due Date (YYMMDD))	N2+N6
13	Paketleme Tarihi (Packaging date (YYMMDD))	N2+N6
15	Tavsiye edilen son tüketim tarihi (Best Before Date (YYMMDD))	N2+N6
37	İçerilen ticari birim sayısı (Number of Units Contained)	N2+N..8
240	Ek Ürün Tanımlayıcısı (Additional Product Identification)	N3+X..30
8002	Hücreli mobil telefon tanımlayıcısı (Cellular mobile telephone identifier)	N4+X..20

Eğer bu barkodu FastReport .NET'te oluştururken yukarıdaki örneği kullanırsak, şu sonucu elde ederiz:

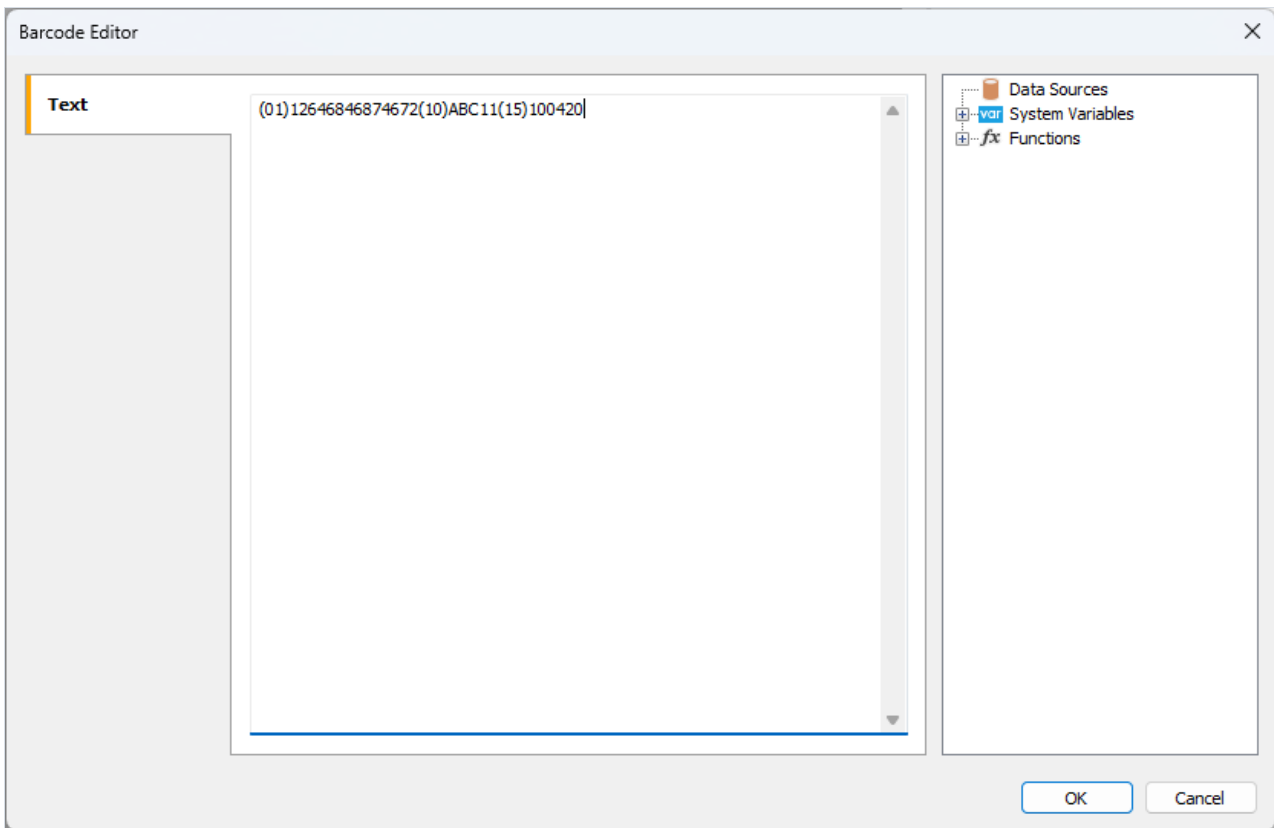


FastReport .NET'te bir GS1-128 barkodu oluşturmak için, Rapor Tasarımcısı'ndaki Bileşenler Panelinden Barkod nesnesini seçin. Açılır listeden "GS1" kategorisine gidin ve ardından GS1-128 (UCC/EAN-128) seçeneğini seçin.



Barkodu seçtikten sonra, onu Rapor Sayfasına yerleştirin.

Eklenecek barkodu düzenleyiciyi açmak için çift tıklayın. Ayrıca, eklenen nesneye sağ tıklayarak erişilen bağlam menüsündeki düğmeye tıklayarak da barkod düzenleyicisini açabilirsiniz:



Interleaved 2 of 5



Interleaved 2 of 5 (ITF 2/5) barkodu, herhangi bir uzunluktaki sayısal bilgileri kodlamak için kullanılan klasik bir doğrusal barkod türüdür. 1972 yılında David Allais tarafından geliştirilmiştir.

Bu barkodun temel özelliği, kodlanacak rakamların sayısının mutlaka çift olması gerektiğidir. Bunun nedeni, karakterlerin çiftler halinde kodlanmasıdır. Eğer kodlanacak rakamların sayısı tek ise, barkodun başına bir sıfır eklenir.

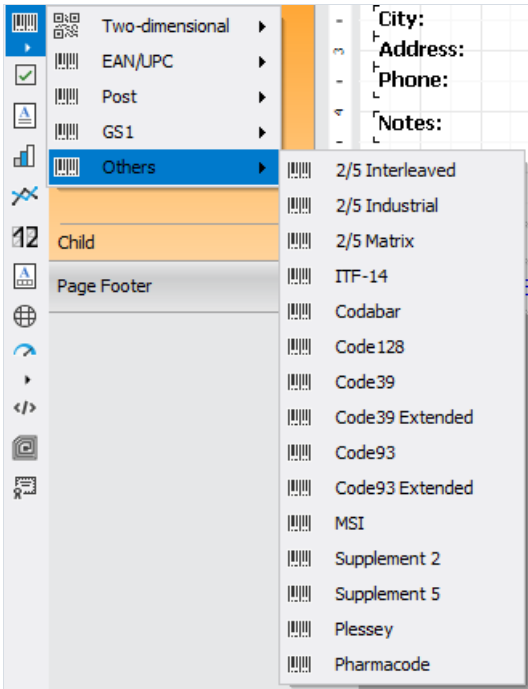
Çiftler halinde kodlamada, ilk rakam çubuklarla, ikinci rakam ise boşluklarla temsil edilir. Çubuk ve boşluklar dar veya geniş olabilir. Her rakam, toplam beş çubuk veya boşluk ile kodlanır; bunların ikisi geniş, üçü ise dardır. Aynı durum boşluklar için de geçerlidir.

Bu özel kodlama yöntemi sayesinde barkod yüksek yoğunluğa sahiptir ve Code 39 gibi diğer türlere kıyasla daha az alan kaplar. Interleaved 2 of 5 barkodu, depolarda ürün etiketleme ve nakliyecilikte sıkça kullanılır.

Kodun yapısı şöyledir:

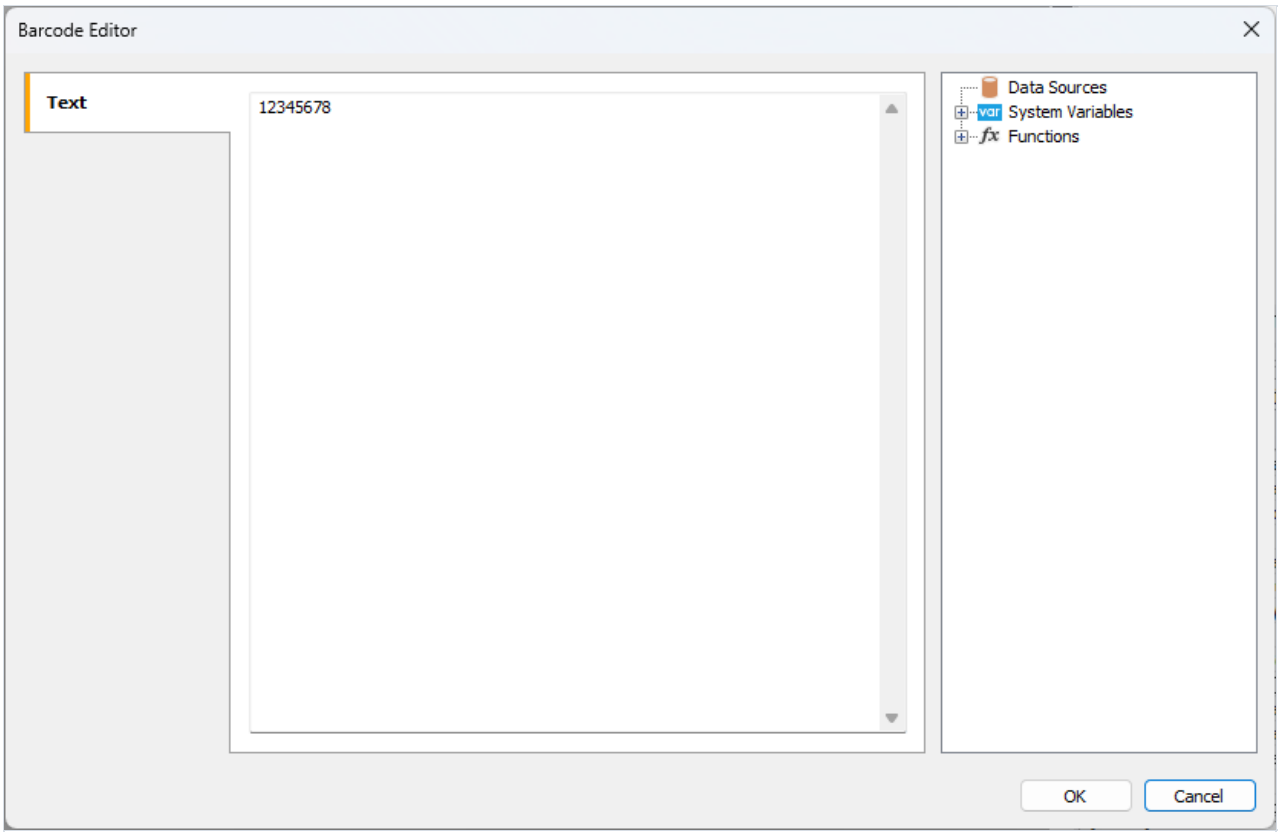
- Başlangıç sembolü – barkodun başlangıcını belirtir;
- Kodlanmış veriler;
- İsteğe bağlı kontrol rakamı;
- Bitiş sembolü.

Interleaved 2 of 5 barkodu, dahili kontrol özelliğine sahip olduğu için kontrol rakamının kullanımı zorunlu değildir:



Barkodu seçtikten sonra, bunu Rapor Sayfasına yerleştirin.

Eklenecek barkoda çift tıklayarak düzenleyiciyi açın. Ayrıca, eklenen nesnenin bağlam menüsünde, sağ tıklayarak erişebileceğiniz düğmeye tıklayarak barkod düzenleyicisini açabilirsiniz:



Barkodun altındaki metni gizlemek istiyorsanız, ilgili barkodun özellik denetleyicisinde **ShowText** (MetniGöster) özelliğini bulun ve değerini False (Yanlış) olarak ayarlayın:



Industrial 2 of 5



Industrial 2 of 5 (Standard 2 of 5 olarak da bilinir), isteğe bağlı uzunlukta sayıların kodlanması için tasarlanmış düşük yoğunluklu doğrusal bir barkoddur. Kodlamada yalnızca farklı kalınlıktaki çizgiler kullanılır ve boşluklar sadece bu çizgileri ayırmak için yer alır. Bu özellik nedeniyle barkodun yoğunluğu düşük ve boyutları büyüktür.

Her karakter, ikisi kalın olmak üzere toplam beş çizgiyle kodlanır. Adını da bu özellikten almıştır.

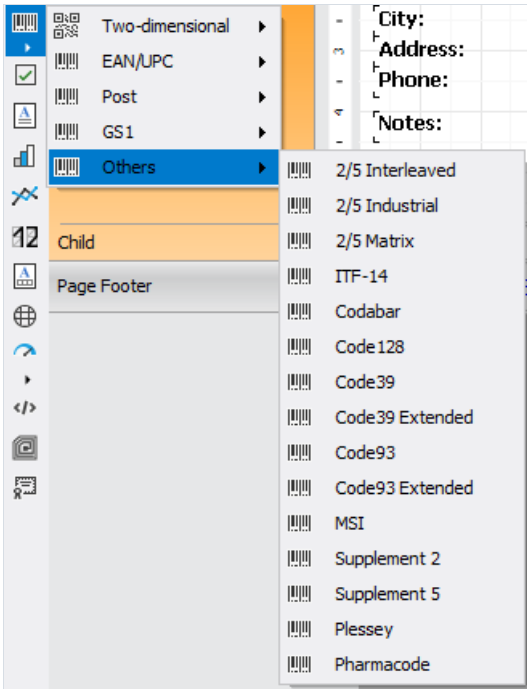
Günümüzde bu barkod türü eskimiş olarak kabul edilir. Geçmişte depo yönetiminde, fotoğraf laboratuvarlarında ve uçak biletlerinin numaralandırılmasında kullanılmıştır.

Industrial 2 of 5, modül 10 kontrol rakamını destekler, ancak bu kontrol rakamının kullanımı zorunlu değildir.

Industrial 2 of 5 barkodunun fiziksel yapısı şöyledir:

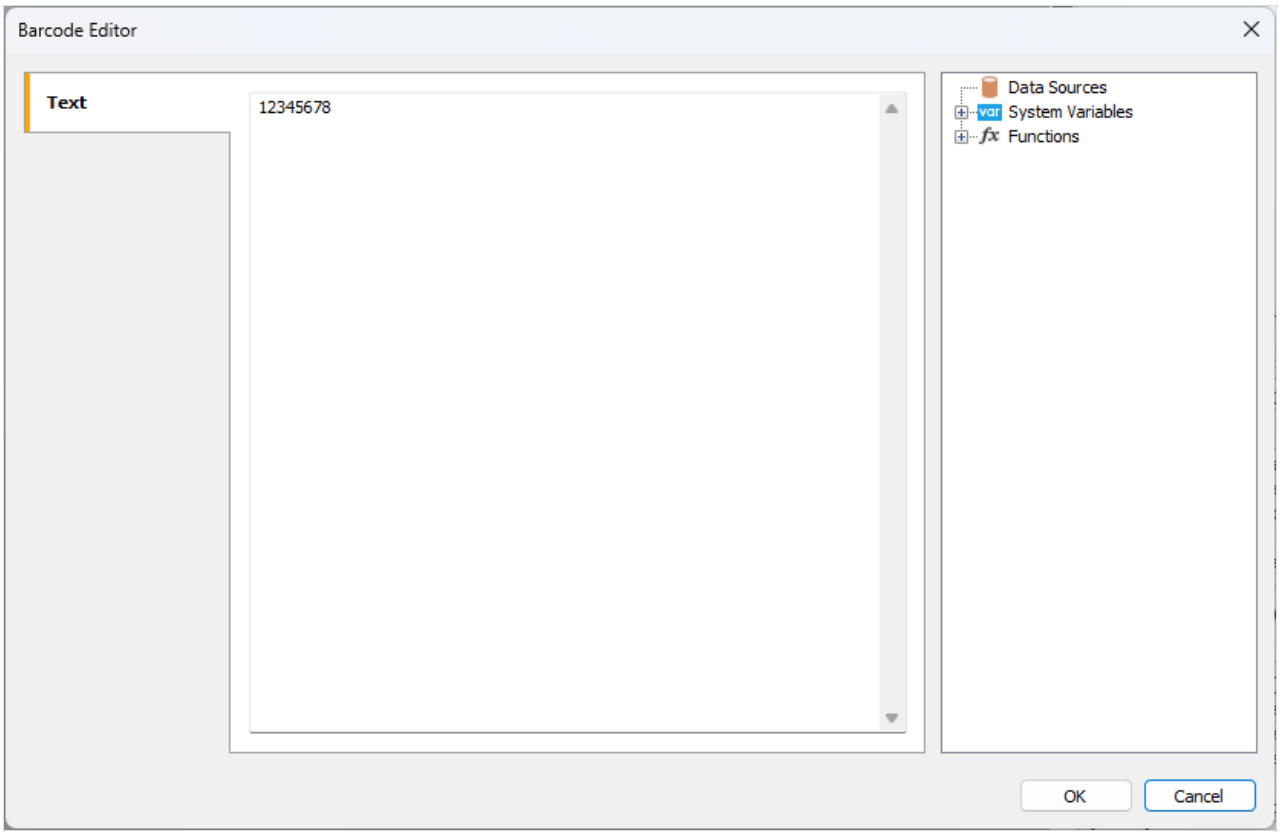
- Başlangıç sembolü;
- Veri sembolleri;
- İsteğe bağlı kontrol sembolü;
- Bitiş sembolü.

FastReport.NET ile Industrial 2 of 5 barkodu oluşturmak için, rapor tasarım aracındaki bileşenler panelinde bulunan Barkod (Barcode) öğesini seçin. Açılan listeden "Diğerleri" kategorisine giderek, "2/5 Industrial" seçeneğini seçin:



Barkodu seçtikten sonra, bunu Rapor Sayfasına yerleştirin.

Eklenen barkoda çift tıklayarak düzenleyiciyi açın. Ayrıca, eklenen nesnenin bağlam menüsünde, sağ tıklayarak erişebileceğiniz düğmeye tıklayarak barkod düzenleyicisini açabilirsiniz:



Barkodun altındaki metni gizlemek istiyorsanız, ilgili barkodun özellik denetleyicisinde **ShowText** (MetniGöster) özelliğini bulun ve değerini False (Yanlış) olarak ayarlayın:



2 of 5 Matrix



2 of 5 Matrix, sayısal verileri (0-9) herhangi bir uzunlukta kodlamak için geliştirilmiş, yüksek yoğunluklu doğrusal bir barkod türüdür. 1968 yılında Identicon şirketi tarafından geliştirilmiştir. Bu barkod türü, çizgiler arasındaki boşlukları da bilgi kodlamak için kullandığı için yüksek veri yoğunluğu sağlar, dolayısıyla hem çizgiler hem de boşluklar kodlama sürecine katılır.

Kod, dahili hata denetimi içerir ve iki yönlü okumayı destekler, yani hem soldan sağa hem de sağdan sola okunabilir.

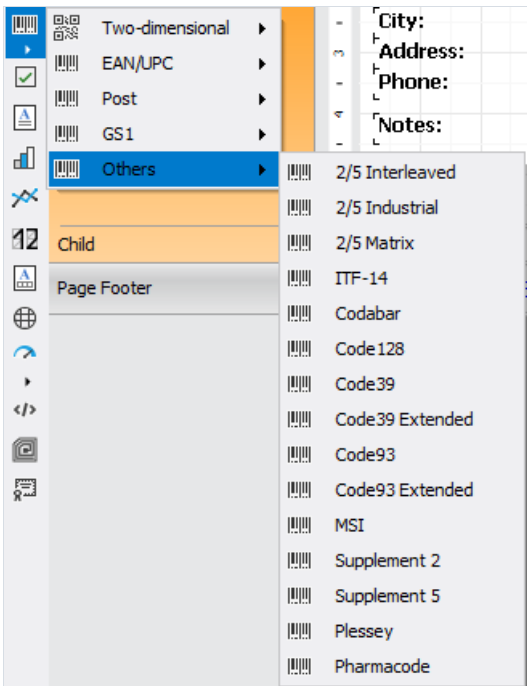
Kodun yapısı:

- Başlangıç karakteri;
- Veri;
- Opsiyonel kontrol karakteri (`mod 10` formülü ile hesaplanır);
- Durdurma (bitiş) karakteri.

Her bir sembol, ikisi kalın olmak üzere toplam beş çizgiden oluşur. Bu iki genişlikli sembol yapısı, ince çizginin genişliğini belirler ve bundan daha kalın olan tüm çizgiler kalın çizgi olarak kabul edilir. Böylece kaliteli olmayan baskı ekipmanlarında bile yüksek toleransla barkod basımı mümkün hale gelir.

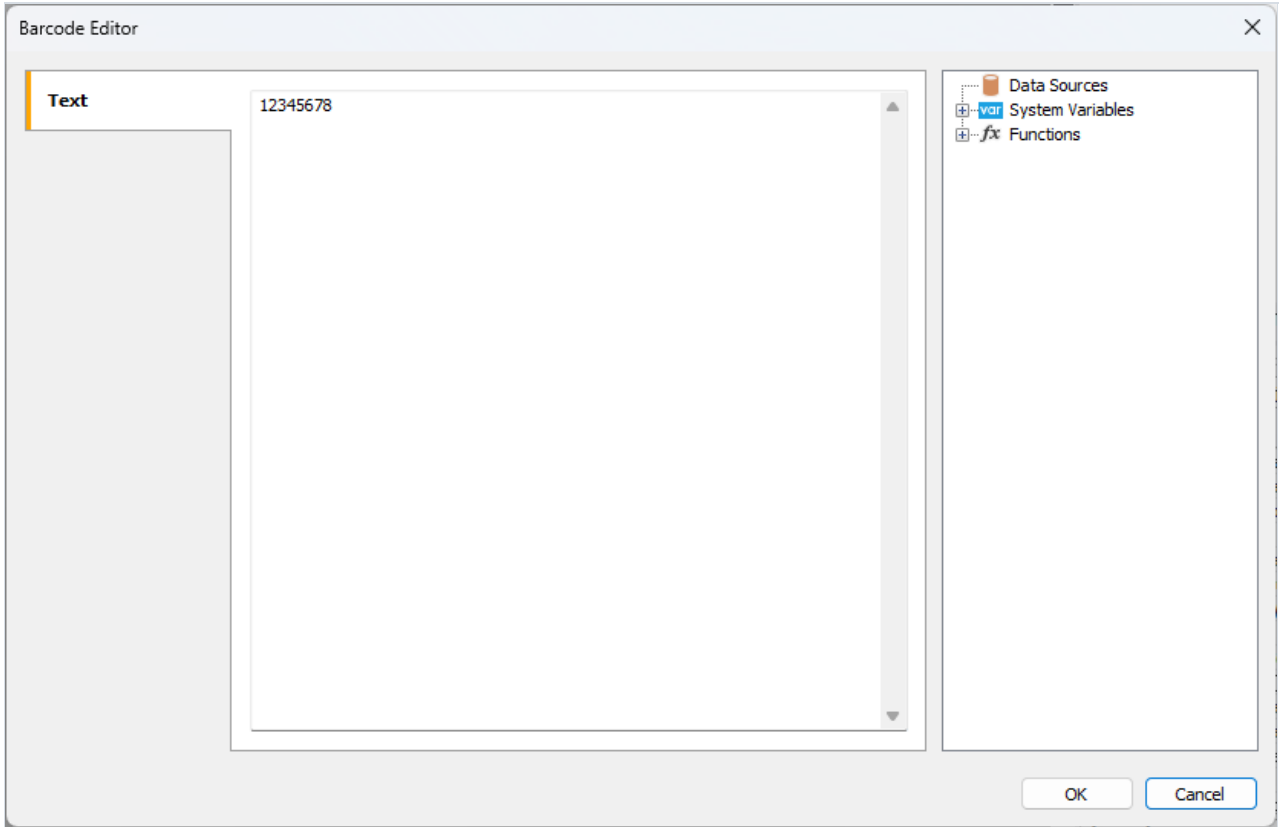
2 of 5 Matrix barkodları, depolarda ürün etiketleme, uçak biletleri ve fotoğraf laboratuvarları gibi alanlarda kullanılır.

FastReport .NET uygulamasında 2 of 5 Matrix barkodu oluşturmak için rapor tasarımcısının bileşen panelinde Barkod (Barcode) nesnesini seçin. Açılan listede "Diğer" kategorisine ve ardından 2/5 Matrix seçeneğine geçin:



Barkodu seçtikten sonra rapor sayfasına yerleştirin.

Eklenen barkoda çift tıklayarak düzenleyici açılır. Ayrıca eklenmiş nesneye sağ tıklayarak çağrılan bağlam menüsündeki butona tıklayarak da düzenleyici açılabilir:



Barkod altındaki metni gizlemek gerekirse, ilgili barkodun özellikler denetleyicisinde **ShowText** özelliğini bulun ve değerini False olarak ayarlayın.



ITF-14



ITF-14 – çift şeritli sayısal kod, yüksek yoğunluklu barkod olarak da bilinir. Sadece çift sayıda rakam kodlayabilir. Bu barkod türünde her tek rakam koyu bir çizgiyle, her çift rakam ise bu çizgiler arasındaki boşlukla temsil edilir. Tek sayıda rakamı kodlamak gerektiğinde en soldaki (en yüksek değerdeki) rakam 0 ile tamamlanır.

Barkodun yapısı şu şekildedir:

- **İlk gösterge karakteri:** Belirli bir karton kutu için ambalaj seviyesini belirtir. Bu tek basamaklı önek 0'dan 8'e kadar değişebilir. (Örneğin, 1 – kutu, 2 – kasa vb.)
- **Ardından 2–3 rakam:** İlgili numaranın kayıtlı olduğu ülkenin bölgesel kodu (ön ek).
- **Sonraki 4–5 rakam:** Ulusal organizasyon içindeki işletme kayıt numarasını temsil eder.
- **Takip eden rakam grubu:** İşletme içindeki ürünün sıra numarasını gösterir.
- **Son 13. rakam:** Önceki on iki rakamdan Modül 10 algoritması ile hesaplanan kontrol rakamıdır.

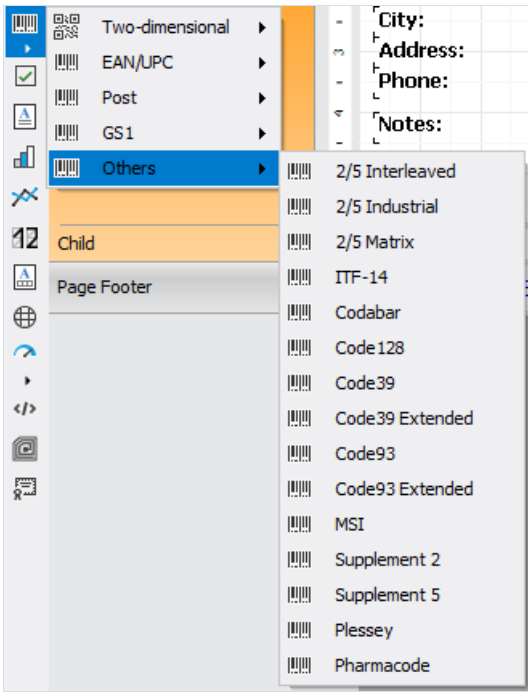
ITF-14 barkodu, bireysel veya grup taşıma ambalajlarına yerleştirilen ürünlerin depo muhasebesini otomatikleştirmek için kullanılır. Bu barkod, bilgisayarlı envanter sisteminin ambalaj içindeki ürünün sadece türünü değil, aynı zamanda miktarını da belirlemesini sağlar.

ITF-14 genellikle oluklu mukavva üzerine baskı yapmak ve karton kutuların, kasaların veya paletlerin işaretlenmesi için kullanılır. Bu barkodlar, perakendeciler, üreticiler ve dağıtıcılar tarafından hassas lojistik kontrolü ve envanter yönetimi için yaygın olarak kullanılır. Ayrıca, havaalanlarında bagaj tanımlaması, uçak biletlerinin numaralandırılması ve posta gönderilerinin tanımlanması amacıyla da kullanılabilir.

ITF-14 barkodu, taşıma ambalajlarındaki ürünlerin işaretlenmesi amacıyla kullanıldığından, kasa terminallerinde işlem görmesi öngörülmemektedir.

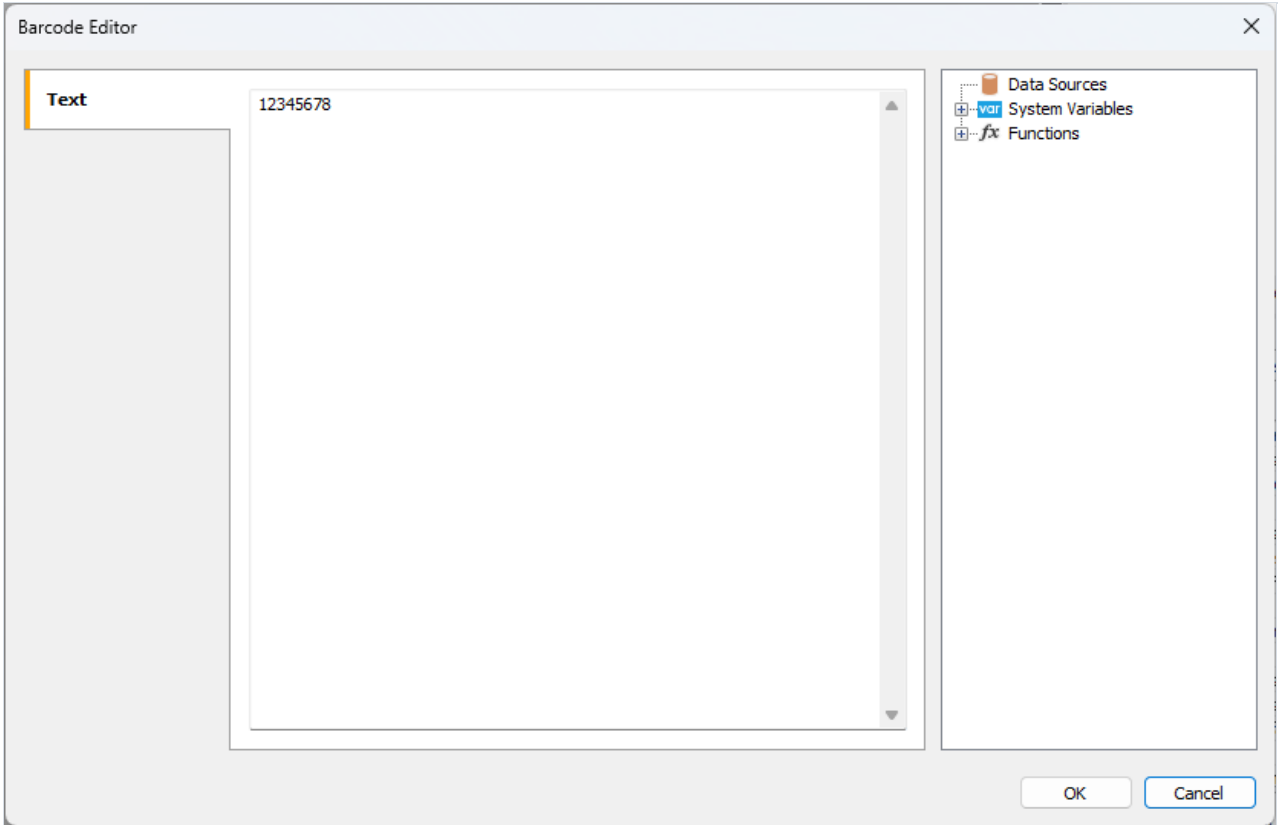
Barkodun etrafındaki kalın siyah çerçeve, Bearer Bar (Taşıyıcı Çubuğu) olarak adlandırılır. Bu çubuk, baskı plakasının barkodun tüm yüzeyi üzerinde oluşturduğu baskıyı dengeler ve okunabilirliği artırarak eksik karakter tarama olasılığını azaltır. ITF-14, görülebilir veya gizli dikey taşıyıcı çubuklarla olabilir.

FastReport .NET'te bir ITF-14 barkodu oluşturmak için, Rapor Tasarımcısı'ndaki Bileşenler Panelinden Barkod nesnesini seçin. Açılır listeden "Diğerleri" kategorisine gidin ve ardından ITF-14 seçeneğini seçin.



Barkodu seçtikten sonra, onu Rapor Sayfasına yerleştirin.

Eklenen barkodu düzenleyiciyi açmak için çift tıklayın. Ayrıca, eklenen nesneye sağ tıklayarak erişilen bağlam menüsündeki düğmeye tıklayarak da barkod düzenleyicisini açabilirsiniz.



Barkod nesnesinin özelliklerine bir göz atalım:

özellik	Açıklama
Açı (Angle)	Nesnenin dönüşünü 0, 90, 180 veya 270 derece olmak üzere sabit açılardan birine ayarlamaya olanak tanır.

Yakınlaştırma (Zoom)	Barkodun ölçeklendirmesini ayarlar. Bu özellik yalnızca "Otomatik Boyutlandırma" özelliğiyle birlikte kullanılır.
Otomatik Boyutlandırma (AutoSize)	Bu özellik etkinleştirildiğinde, barkodu tamamen göstermek için nesne uzatılır. Özellik devre dışı bırakıldığında ise barkod nesnenin boyutlarına göre uzatılır.
Metni Göster (ShowText)	Barkodun alt kısmında metin gösterilip gösterilmeyeceğini belirler.
Veri Sütunu (DataColumn)	Nesnenin metnini yüklemek için kullanılacak veri alanı.
İfade (Expression)	Nesnenin metnini döndüren ifade.
Metin (Text)	Nesnenin metni.
Dolgu (Padding)	Nesnenin kenarlarından piksel cinsinden boşlukları ayarlamaya olanak tanır.
Geniş Çubuk Oranı (WideBarRatio)	Bu özellik tüm lineer barkodlarda bulunur. Barkodun geniş çubuklarının göreceli boyutunu belirler.
Kontrol Toplamını Hesapla (CalcChecksum)	Bu özellik birçok lineer barkodda bulunur. Kontrol toplamının otomatik olarak hesaplanıp hesaplanmayacağını belirler. Özellik devre dışı bırakıldığında, kontrol toplamının nesne metninde bulunması gerekir.
Dikey Taşıyıcı Şeritleri Çiz (DrawVerticalBearerBars)	Bu özellik etkinleştirildiğinde, nesnede yan çizgiler görüntülenir.

DrawVerticalBearerBars özelliği devre dışı bırakıldığında, barkod aşağıdaki şekilde görünür:



Codabar



Codabar, en sık seri numaralarını kodlamak için kullanılan, değişken uzunlukta doğrusal bir barkod türüdür. İlk olarak 1972 yılında Pitney Bowes şirketi tarafından oluşturulmuştur. Çizgileri, barkodun matris yazıcılarla basılması durumunda bile doğru okunmasını sağlar.

Kodlanabilen Karakterler:

- 0-9 rakamları;
- A, B, C, D harfleri;
- -, \$, /, ., + özel karakterleri.

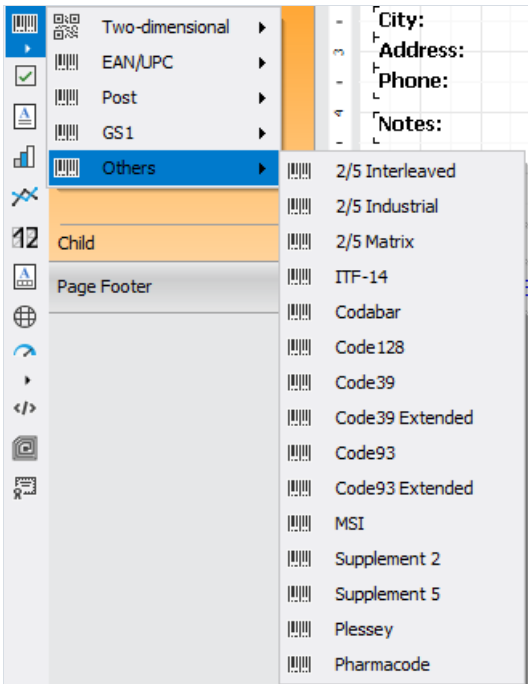
Harfler yalnızca barkodun başlangıç ve bitiş karakterleri olarak kullanılabilir ve ek bilgi kodlama amacıyla tercih edilebilir.

Barkod Yapısı:

- Başlangıç sembolü: veri okuma işleminin başlangıcını belirtir;
- Kodlanacak veriler;
- Bitiş sembolü: kodun bittiğini gösterir.

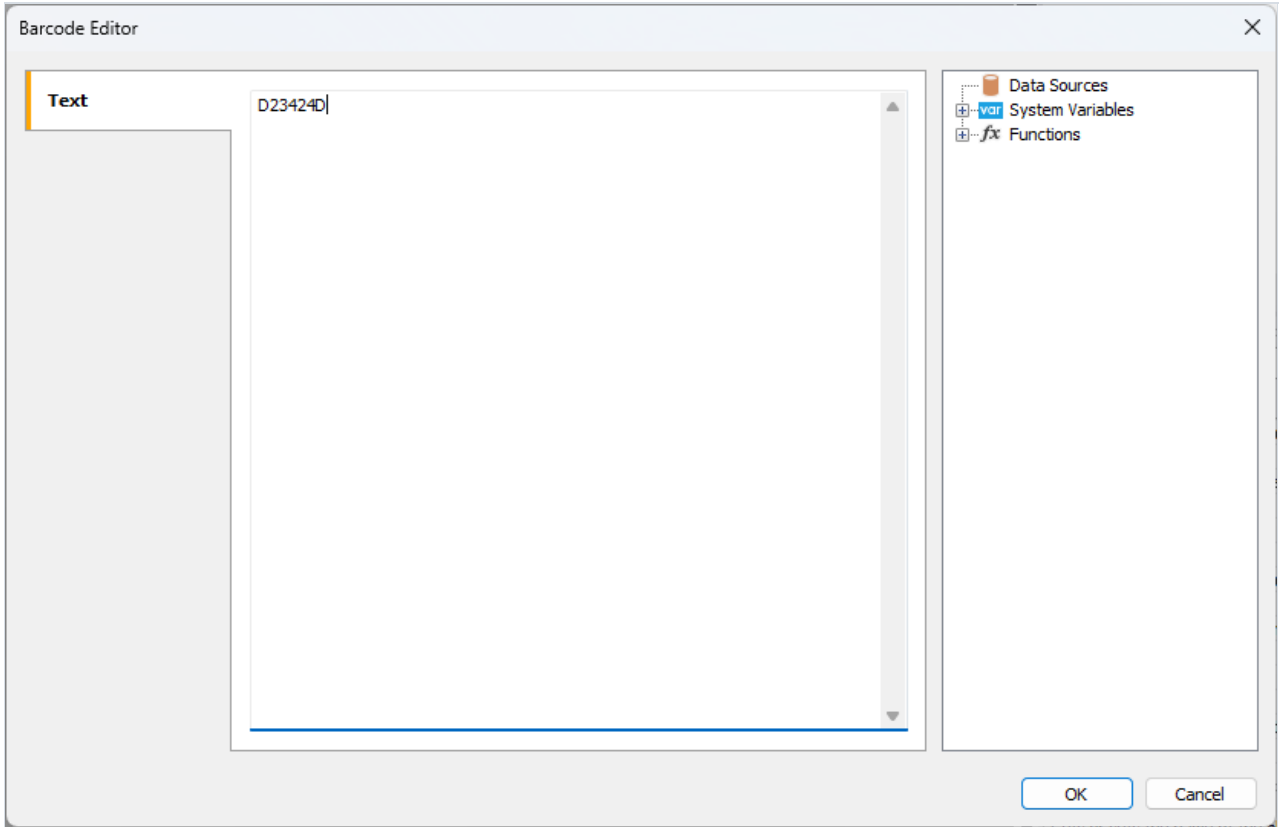
Codabar'da kontrol karakteri (checksum) kullanılmaz, çünkü kodun kendi içinde doğrulama sistemi bulunmaktadır. Codabar'daki her karakter, 4 çizgi ve 3 boşlukla kodlanır.

FastReport .NET'te Codabar barkodunu oluşturmak için rapor tasarımcısındaki bileşenler panelinden Barkod (Barcode) öğesini seçin. Açılan listede "Diğer" kategorisine gidin ve ardından Codabar seçeneğini belirleyin:



Barkod seçildikten sonra rapor sayfasına yerleştirilir.

Eklenen barkoda çift tıklayarak düzenleyici açılır. Ayrıca, sağ tıklayarak açılan bağlam menüsündeki butona basarak da düzenleyiciye erişilebilir:



Barkodun altındaki metni gizlemek isterseniz, özellikler denetleyicisinde ilgili barkodun **ShowText** özelliğini bulun ve değerini False olarak ayarlayın.



Code 128



Code 128 linear barkod birçok alanda geniş kullanım bulmuştur. Adından da anlaşılacağı gibi 128 karakteri kodlamaya izin verir. Ancak bu yalnızca rakamlarla sınırlı değildir, harfleri ve bazı özel karakterleri de kodlayabilmektedir. Bu özellik, onu EAN standart kodlarından ayıran temel avantajıdır.

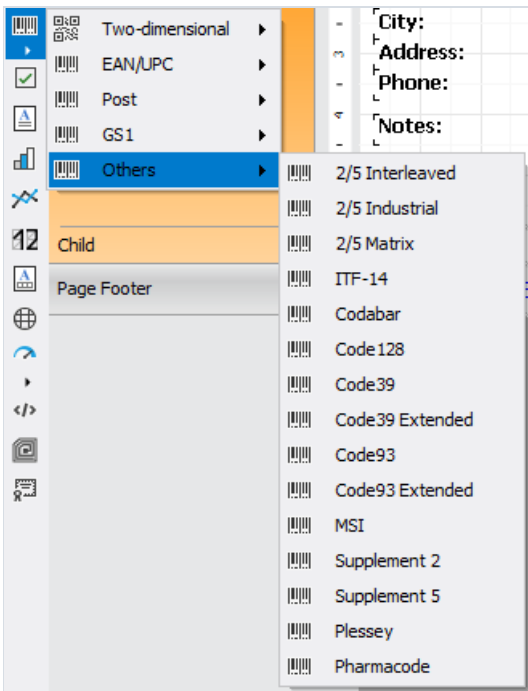
Kodun yapısı çoğu linear barkod gibi basittir. Kodun her iki yanında da başlangıcı net bir şekilde belirleyen boş alanlar bulunur. Sonra, başlangıç sembolü olarak özel çizgi ve boşluk dizisi gelir. Başlangıç sembolünden sonra kodlanacak veriler yer alır. Ardından verilerin bütünlüğünü kontrol etmek için bir kontrol karakteri bulunur. Kodlama işlemini tamamlayan son sembol ise, verilerin bittiğini belirten durdurma sembolüdür.

Farklı bilgilerin kodlanması, harfler, rakamlar ve özel karakterler için farklı karakter setleri kullanılarak gerçekleştirilir. Her sembol üç çizgi ve üç boşluk ile kodlanır. Çizgiler ve boşluklar farklı genişliklerde olabilir. Genişlik, modül sayısına göre belirlenir – 1'den 4'e kadar olabilir.

Böylece Code 128 aşağıdaki avantajlara sahiptir:

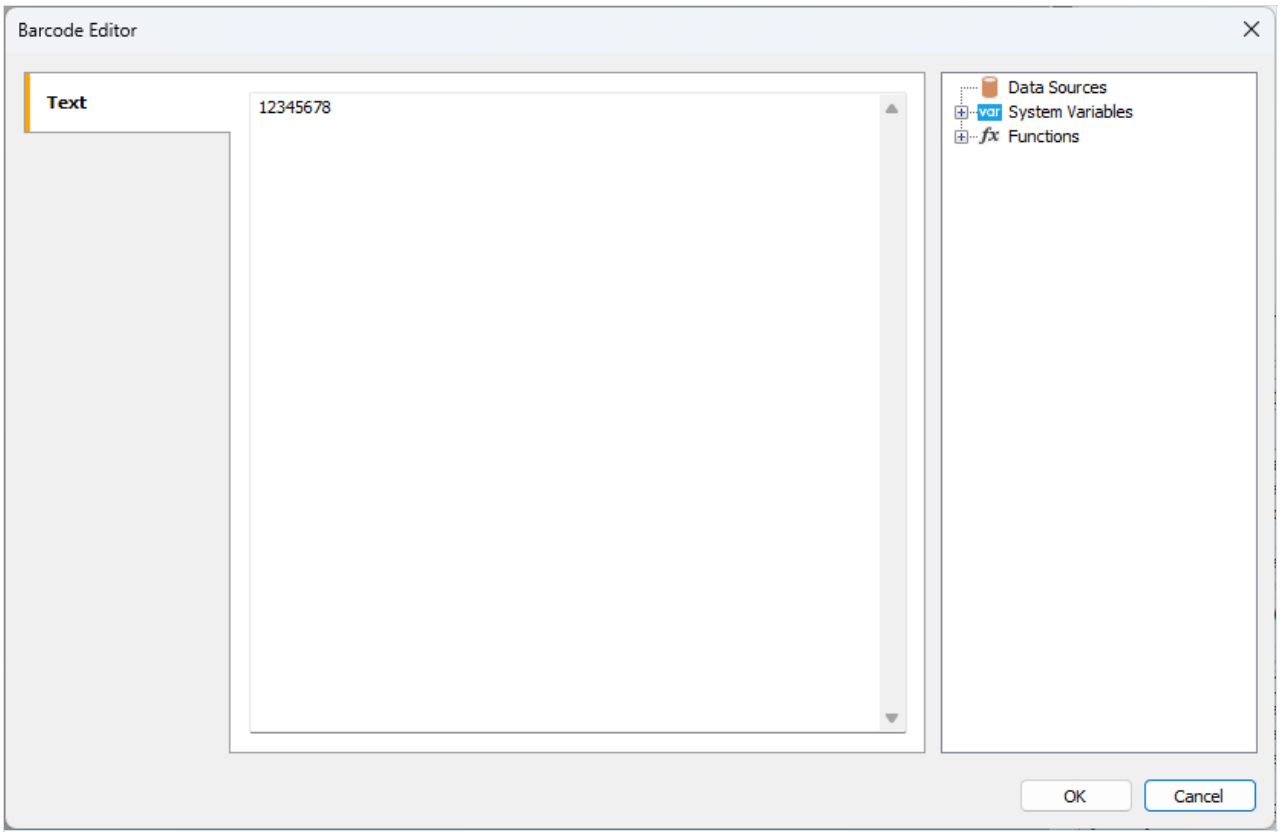
- Hem rakamlar hem de büyük harfler kodlanabilir;
- Yalnızca sayısal veriler için çift paketleme kullanılması sayesinde oldukça kompakt kodlar elde edilir.

FastReport .NET içinde Code 128 barkod oluşturmak için rapor tasarımcısındaki bileşenler panelinden Barkod (Barcode) nesnesini seçin. Açılır menüden "Diğer" kategorisine geçin ve ardından Code128 seçimini yapın:



Barkodu seçtikten sonra rapor sayfasına yerleştirin.

Eklenmiş barkoda çift tıklayarak düzenleyiciyi açabilirsiniz. Ayrıca eklenmiş nesnenin sağ tıklamayla açılan bağlam menüsündeki düğmeye basarak da düzenleyiciyi açmak mümkündür:



Barkodun altındaki metni gizlemek isterseniz, ilgili barkodun özellikler denetleyicisindeki ShowText özelliğini bulun ve değerini False olarak ayarlayın:



Code 39



Code 39, 1975 yılında Intermec şirketi tarafından geliştirilen doğrusal bir barkod türüdür. Bu barkod, alfanümerik verileri kodlayabilme özelliği sayesinde büyük bir popülerlik kazanmıştır ve halen aktif olarak kullanılmaktadır.

Code 39 barkodunun önemli bir özelliği, istenilen uzunlukta verileri kodlayabilmesidir. Barkodun genişliği, yalnızca barkod okuyucusunun okuma kapasitesiyle sınırlıdır.

Bu barkodda her karakter 9 elementten oluşmaktadır: 5 çizgi ve 4 boşluk. Bu kadar fazla öğenin kullanılmasının sebebi, dahili kendini kontrol etme (self-checking) özelliğidir. Bu özellik sayesinde her karakter kontrol edilir ve kontrol hanesine ihtiyaç duyulmaz. Ancak barkodun bütünlüğünü artırmak amacıyla isteğe bağlı olarak kontrol hanesi eklenebilir.

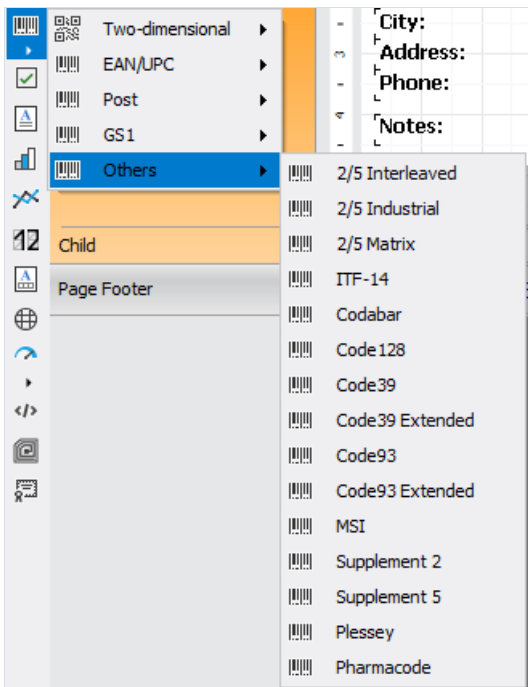
Daha önce belirtildiği gibi Code 39, alfanümerik bilgileri kodlamaya olanak sağlar. Kullanılabilir karakterler:

- Büyük harf Latin alfabe harfleri (A-Z);
- Rakamlar (0-9);
- Özel semboller (-, , , . , \$, / , + , % , :).

Code 39 barkodunun yapısı aşağıdaki gibidir:

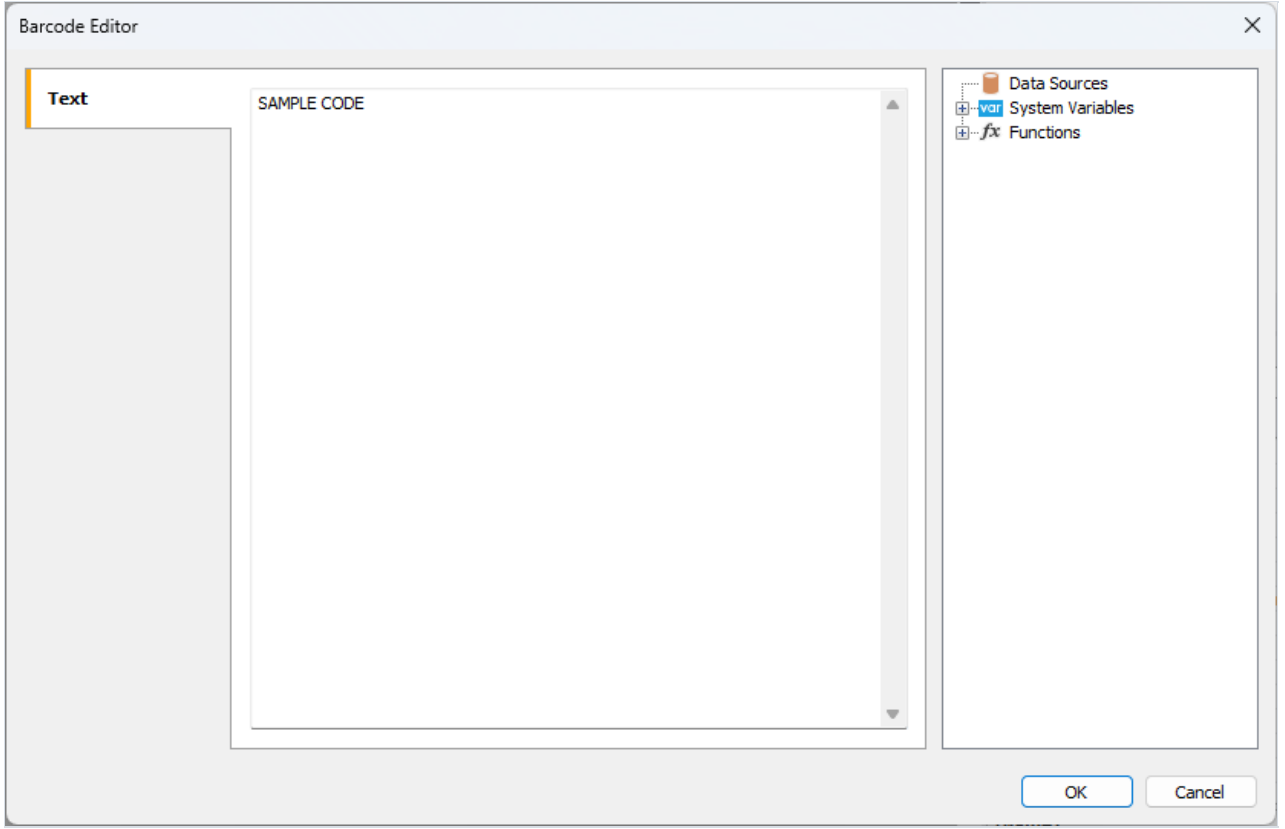
- Başlangıç sembolü (* ile kodlanır). Barkodun başlangıcını gösterir;
- Kodlanmış veriler;
- İsteğe bağlı kontrol hanesi;
- Bitiş sembolü (yine * sembolü ile kodlanır).

FastReport .NET uygulamasında Code 39 barkodunu oluşturmak için rapor tasarımcısının bileşenler panelinden Barkod (Barcode) öğesini seçin. Açılan listede "Diğer" (Diğerleri) kategorisine gidin ve ardından Code39'u seçin:



Barkodu seçtikten sonra, onu Rapor Sayfasına yerleştirin.

Eklenen barkodu düzenleyiciyi açmak için çift tıklayın. Ayrıca, sağ tıklayarak erişilen eklenmiş nesnenin bağlam menüsündeki düğmeye tıklayarak da barkod düzenleyicisini açabilirsiniz:



Barkodun altındaki metni gizlemek isterseniz, ilgili barkodun özellikler denetçisinde (Property Inspector) "ShowText" özelliğini bulun ve değerini "False" olarak ayarlayın:



Code 39 Extended



Başlıktan da anlaşıldığı gibi, bu, Code 39 barkodunun genişletilmiş sürümüdür. Büyük harfler (A-Z), küçük harfler (a-z) ve çeşitli özel karakterler dahil olmak üzere ASCII karakter setinin tamamını destekler. Standart Code 39 gibi, genişletilmiş sürüm de modül 43 formülü kullanılarak hesaplanan bir kontrol karakteri içerebilir.

Barkod, alfanümerik bilgilerin kodlanmasına izin verir. İzin verilen karakterler şunlardır:

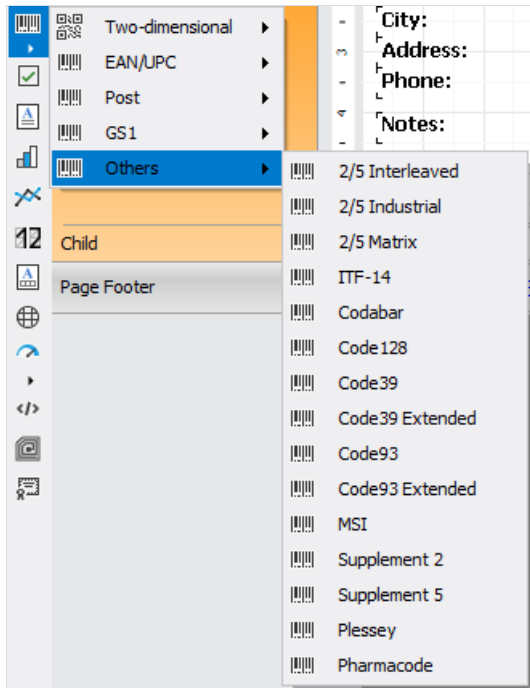
- Büyük (A-Z) ve küçük harfler (a-z);
- Rakamlar (0-9);
- Özel karakterler.

Code 39 Extended barkodunun yapısı şu şekildedir:

- Başlangıç sembolü (* karakteri ile kodlanır);
- Kodlanmış veriler;
- İsteğe bağlı kontrol karakteri;
- Bitiş sembolü (yine * karakteri ile kodlanır).

Küçük harfleri oluşturmak için iki karakter kullanılır: + işareti ve ilgili büyük harf. Bazı özel karakterleri kodlamak için de iki karakterin birleşimi kullanılır: % ve büyük harf veya / ve büyük harf.

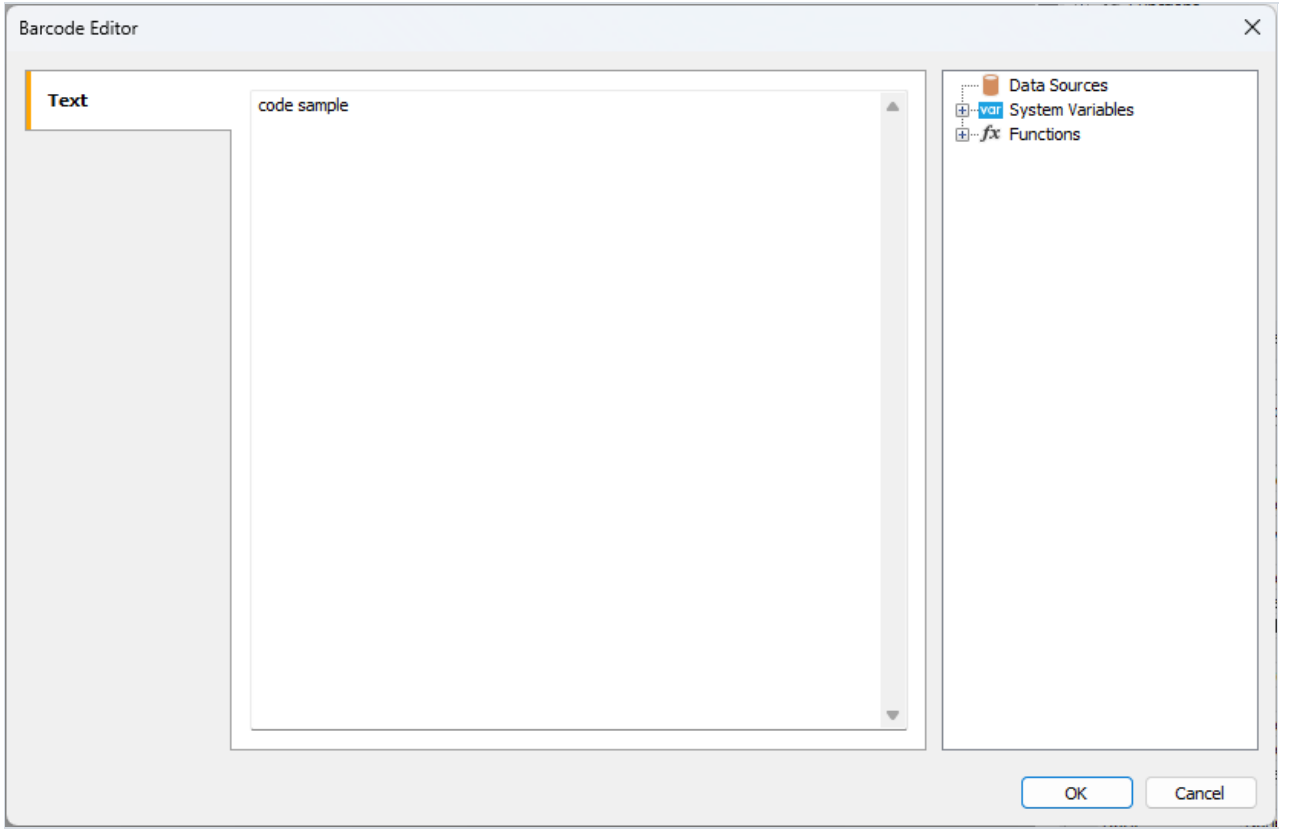
FastReport .NET ortamında Code 39 Extended barkodunu oluşturmak için rapor tasarımcısındaki bileşen panelinden Barkod (Barcode) ögesini seçin. Açılır listeden "Diğer" kategorisine gidin ve ardından "Code39 Extended" seçeneğini işaretleyin:



Barkod seçildikten sonra rapor sayfasına yerleştirilir.

Eklene barkoda çift tıklayarak düzenleyiciyi açabilirsiniz. Ayrıca, eklenen nesneye sağ tıklayarak açılan bağlam

menüsündeki düğmeye basarak da düzenleyici açılabilir:



Barkodun altındaki metni gizlemek gerekirse, ilgili barkodun özellikler denetleyicisinde bulunan **ShowText** özelliğini bulun ve değerini False olarak ayarlayın:



Code 93



Code 93 (AIM-BC5-2000 spesifikasyonu), hem rakamların hem de harflerin kodlanmasına izin veren doğrusal bir barkod türüdür. 1982 yılında Intermec firması tarafından geliştirilmiş olup, Code 39 temel alınarak oluşturulmuştur. Önceki versiyonuna göre temel avantajı, daha kompakt boyutudur.

Code 93, Kanada posta hizmetlerinde gönderi bilgilerini kodlamak için ve otomotiv endüstrisinde kullanılmaktadır.

Kodun uzunluğu 48 karakterle sınırlandırılmıştır ve şu sembollerden oluşabilir: 26 büyük harf, 10 rakam, 7 özel karakter ve 5 yardımcı sembol. Yardımcı semboller, kodun başlangıç ve bitişi belirtmek ile semboloji genişletme amacıyla kullanılır. Bu kod, iki yardımcı sembolün kombinasyonu kullanılarak tüm ASCII karakter setini temsil edebilir.

Kodun yapısı şöyledir:

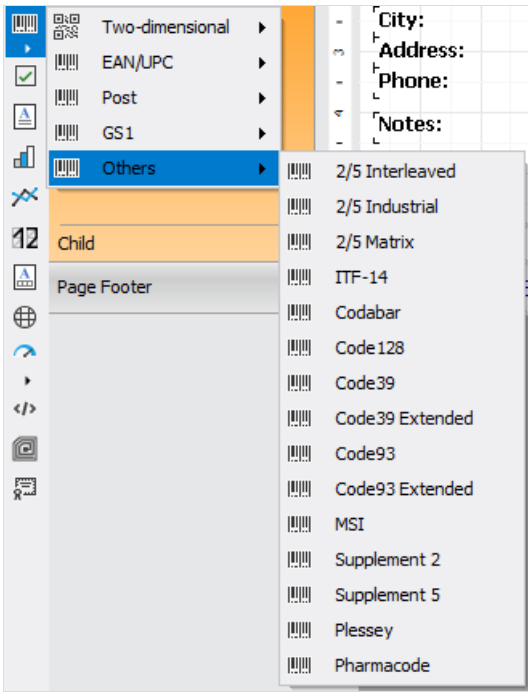
- Başlangıç sembolü
- Kodlanan bilgi
- Modül "C" ile hesaplanan kontrol hanesi
- Modül "K" ile hesaplanan kontrol hanesi
- Bitiş sembolü

Başlangıç sembolü * karakteri ile gösterilir.

Her veri sembolü üç çizgi ve üç boşluktan oluşur. Çizgi veya boşlukların genişliği 1 ile 4 modül arasında değişebilir.

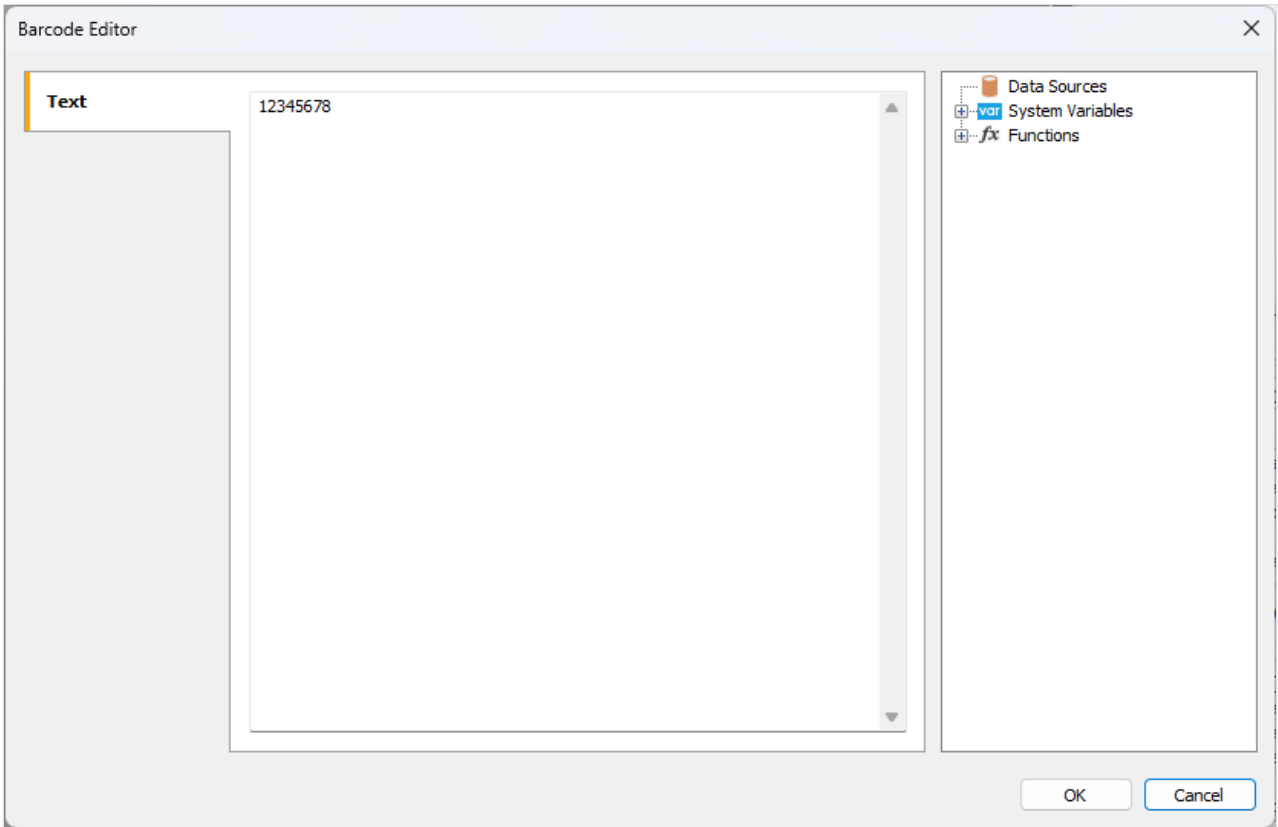
Kontrol haneleri barkod altındaki metinde görüntülenmez. Kontrol toplamı, farklı yöntemlerle hesaplanan iki haneden oluşur: Modül "C" ve Modül "K". Hesaplama formülü, veri değerlerinin ağırlıklı toplamının modül 47'ye göre kalanıdır. "C" ve "K" arasındaki fark, veri değerlerinin ağırlıklandırma yöntemleridir.

FastReport .NET ortamında Code 93 barkodu oluşturmak için rapor tasarımcısındaki bileşen panelinden "Barkod" (Barcode) öğesini seçin. Açılan listeden "Diğer" kategorisine giderek "Code93" seçeneğini seçin:

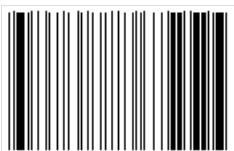


Barkod seçildikten sonra, rapor sayfasına yerleştirin.

Eklenecek barkod üzerine çift tıklayarak düzenleyici açılır. Düzenleyici, barkod üzerine sağ tıklandığında çıkan bağlam menüsünden butonuna basılarak da açılabilir:



Barkodun altındaki metni gizlemek için, ilgili barkodun özellikler denetçisindeki ShowText özelliğini bulun ve değerini False olarak ayarlayın:



Code 93 Extended



Code 93 Extended, Code 93 barkodunun genişletilmiş versiyonudur. Temel versiyondan farklı olarak 48 yerine 128 ASCII karakterini kodlayabilir.

Karakter Kümesi:

- Rakamlar: 0-9;
- Büyük harfler: A-Z;
- Özel karakterler: \$, % , / , + .

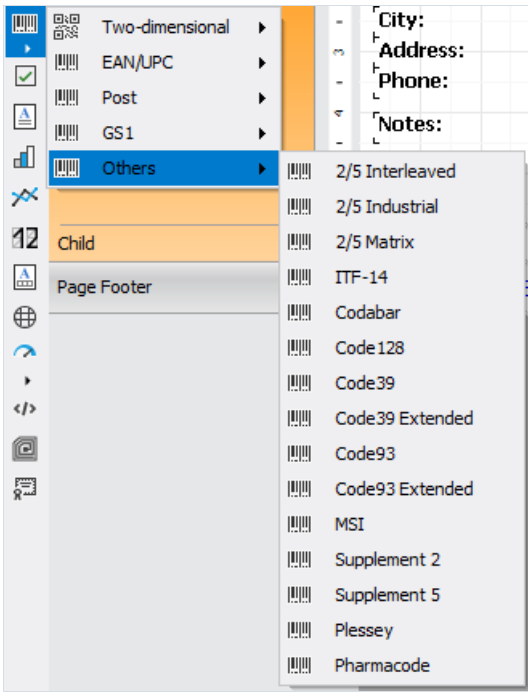
Kod Yapısı:

- Başlangıç sembolü: * ;
- Kodlanacak bilgiler: Veri sembolü üç çizgi ve üç boşluktan oluşur. Çizgi veya boşluk genişliği 1 ile 4 modül arasında değişebilir;
- Modül "C"ye göre hesaplanan kontrol hanesi;
- Modül "K"ye göre hesaplanan kontrol hanesi;
- Bitiş sembolü: * .

Kontrol haneleri barkod altındaki metinde gösterilmez. Kontrol toplamı iki haneden oluşur ve farklı yöntemlerle hesaplanır: "C" modülüne ve "K" modülüne göre. Hesaplama formülü, verilerin ağırlıklarının toplamının 47 modülüne göre kalanıdır. "C" ve "K" arasındaki fark, değerlerin ağırlıklandırma yöntemindedir.

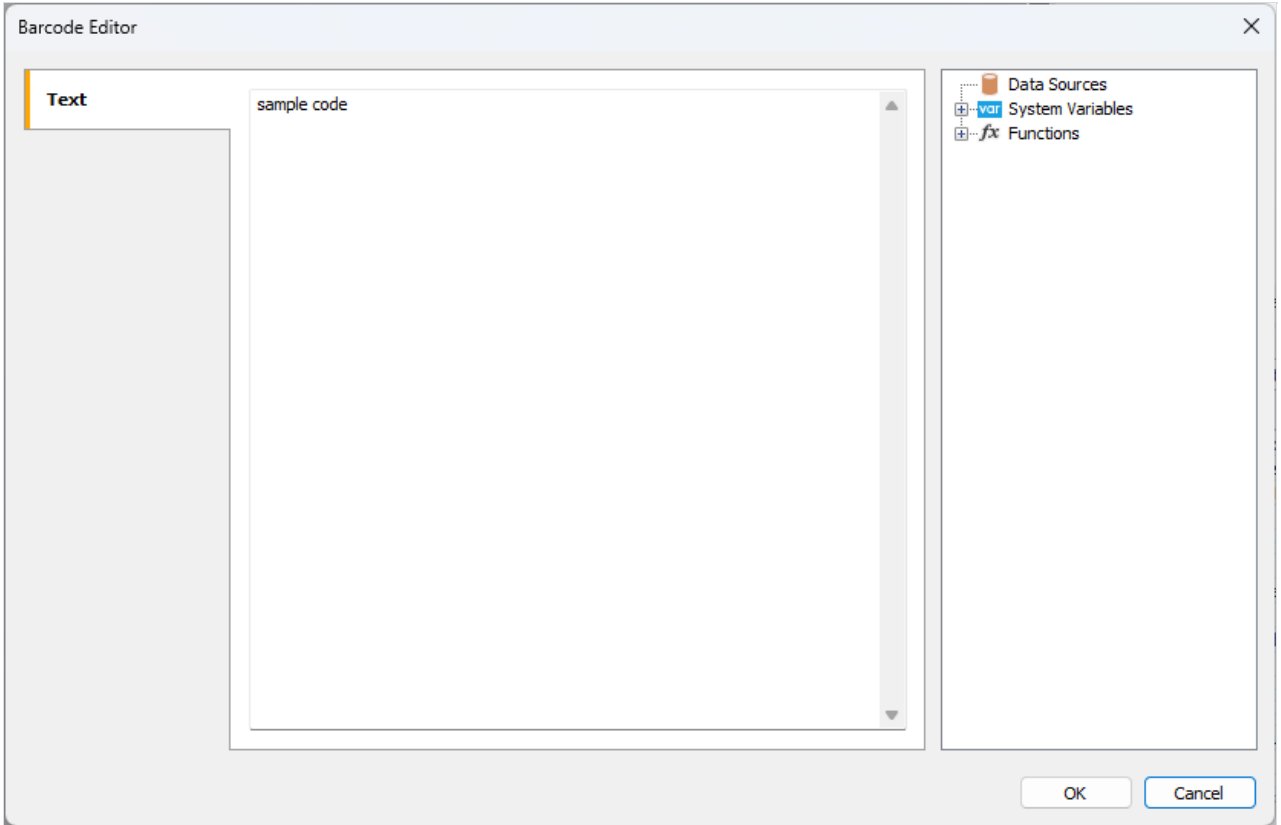
Daha önce belirtildiği gibi, Code 93 Extended, 128 karakterin kodlanmasına izin verir. Ancak standart Code 93 karakter tablosunu genişletmek için ek semboller (\$, /, %, +) kullanılır. Bu nedenle küçük harfleri kodlamak için + sembolü ve ilgili büyük harf kombinasyonu kullanılır. Diğer modifikatör semboller, özel karakter kümesini genişletmek amacıyla kullanılır.

FastReport .NET üzerinde Code 93 Extended barkodunu oluşturmak için, rapor tasarımcısında bileşen panelinden Barkod (Barcode) nesnesini seçin. Açılır listeden "Diğer" kategorisine, ardından da Code93 Extended seçeneğine gidin:



Barkodu seçtikten sonra rapor sayfasına yerleştirin.

Eklenen barkoda çift tıklayarak barkod düzenleyicisini açabilirsiniz. Barkod düzenleyicisi ayrıca eklenen nesnenin sağ tuşla açılan bağlam menüsündeki düğme ile de açılabilir:



Barkod altındaki metni gizlemek isterseniz, ilgili barkodun özellikler denetçisinde (Property Inspector) bulunan ShowText özelliğini False olarak ayarlayın:



MSI



MSI doğrusal barkodu, 1971 yılında Plessey barkoduna dayanılarak oluşturulmuştur ve esas olarak onun geliştirilmiş bir versiyonudur. Yalnızca 0'dan 9'a kadar rakamları kodlamaya izin verir. Kod yapısı şöyledir:

- Veri okumanın başlangıcını belirten başlangıç karakteri (110);
- Veriler;
- İsteğe bağlı kontrol karakteri;
- Veri okumanın bittiğini belirten sonlandırma karakteri (1001).

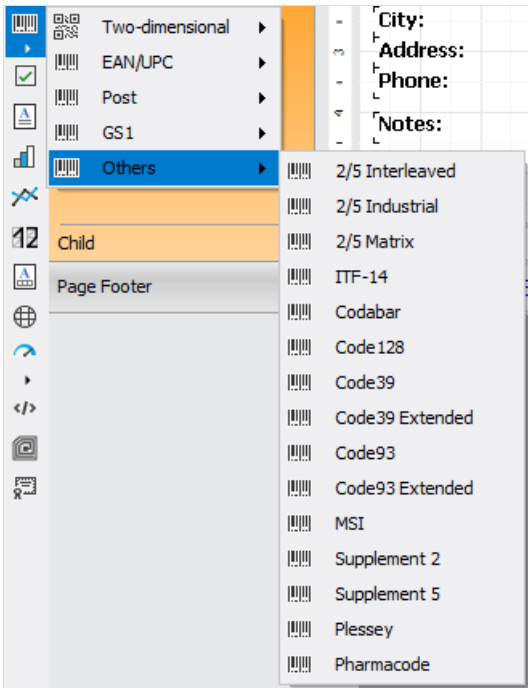
Koddaki her karakter, çizgi ve boşluklar kullanılarak ikili (binary) olarak gösterilir. Çizgi 1, boşluk ise 0 olarak temsil edilir.

Kontrol karakteri, dört farklı yöntemden biriyle hesaplanabilir: Modulo 10, Modulo 11, Modulo 1010 ve Modulo 1110. En yaygın kullanılan yöntem Modulo 10'dur. Kontrol karakteri belirli bir algoritma kullanılarak hesaplanır. Tarayıcı tarafından okunan veriler bu algoritma ile toplanır ve sonuç kontrol karakteri ile karşılaştırılır. Sonuç doğruysa barkodun doğru okunduğu kabul edilir.

MSI barkodunun uzunluğu keyfidir. Uzunluğu yalnızca tarayıcının yetenekleri ile sınırlanır.

Günümüzde bu barkod tipi eski kabul edilir ve neredeyse hiç kullanılmaz. Eskiden depo ve süpermarketlerdeki ürünlerin etiketlenmesi için kullanılırdı.

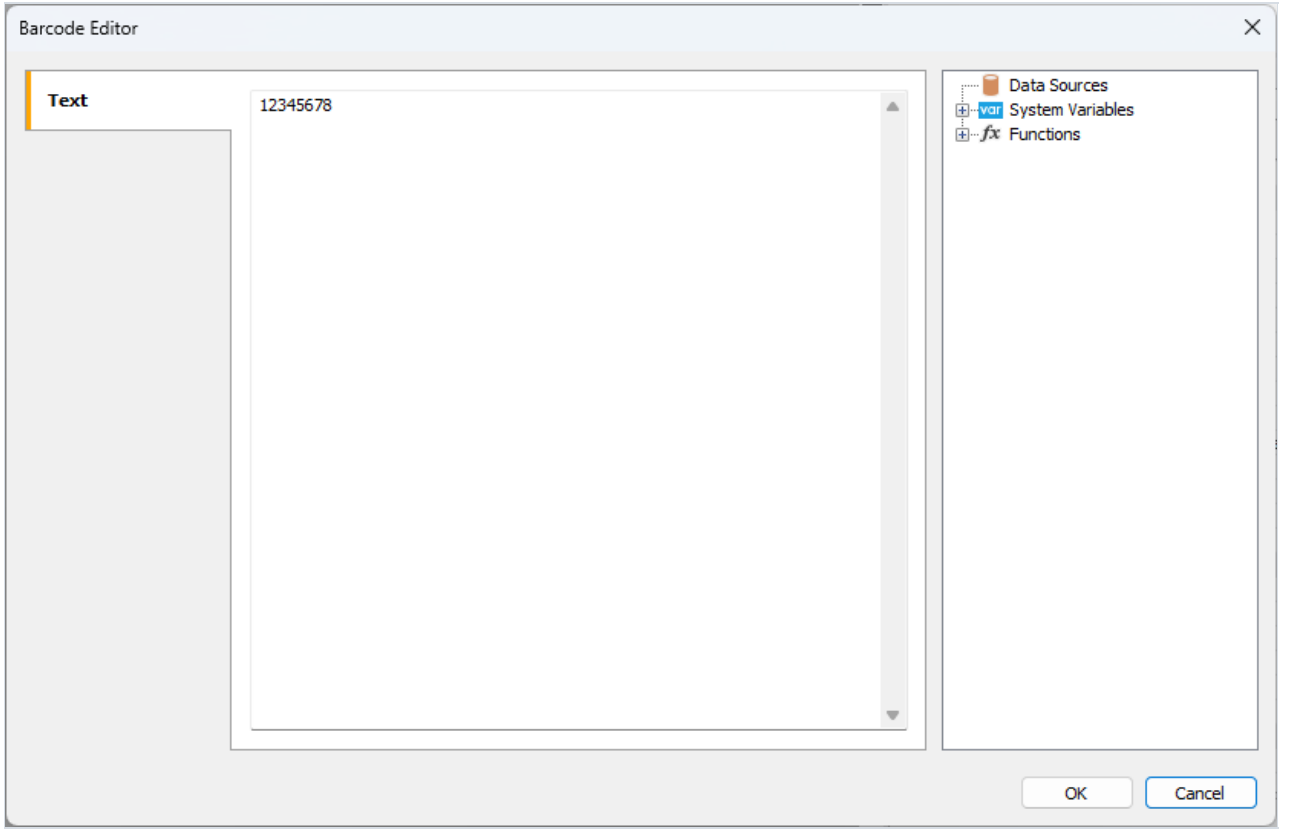
FastReport .NET ortamında MSI barkodunu oluşturmak için, rapor tasarımcısındaki bileşenler panelinden Barkod (Barcode) nesnesini seçin. Açılır listeden "Diğer" kategorisine gidin ve ardından MSI seçeneğini tıklayın:



Barkodu seçtikten sonra, rapor sayfasında uygun yere yerleştirin.

Eklenen barkoda çift tıklayarak düzenleyici açılır. Barkod düzenleyicisini ayrıca eklenen nesnenin sağ tuş

menüsünden çağırabilirsiniz:



Barkodun altındaki metni gizlemek istiyorsanız, ilgili barkodun özellikler denetleyicisinde ShowText özelliğini bulun ve değerini False olarak ayarlayın:



Supplement

Avrupa'da yaygın olarak kullanılan EAN ve ABD'de kullanılan UPC gibi yaygın tek boyutlu barkodlar, ana barkodun sağına yerleştirilen ek bir kodla genişletilebilir.

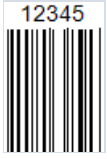
UPC-A, UPC-E, EAN-13 ve EAN-8, ana barkodun sağına ek bir barkod içerebilir. Genellikle ana barkod kadar yüksek olmayan bu ikinci barkod, gazeteler, kitaplar ve diğer periyodik yayınlar için ek bilgi kodlamak amacıyla kullanılır. Ek barkod 2 veya 5 haneli bilgi kodlayabilir.

İki haneli ek kod genellikle periyodik yayınlarda (dergiler, gazeteler) sayı numarasını kodlamak için kullanılır.



Ek kod, ana barkoddan yayın numarasını atlamayı sağlar ve bu numara tüm yayınlar için aynı kalır. Böylece, sadece yayın bilgisi gerektiğinde büyük barkodu taramaya gerek kalmaz.

Beş haneli ek kod, genellikle kitaplarda kullanılır ve ürünün önerilen perakende fiyatı hakkında bilgi içerir. Beş karakterin ilki para birimi kodunu, geri kalan 4 karakter ise fiyatı belirtir.



Fiyatsız ürünler için özel kodlar mevcuttur:

- 90000 kodu – kitabın perakende satış fiyatı belirlenmemiştir.
- 99991 kodu – kitap ücretsiz dağıtılmaktadır.

Bununla birlikte, ek kod aynı zamanda yayınevi içindeki dahili kullanım için başka bilgiler de saklayabilir.

Her rakam 7 modül (çubuklar ve boşluklar) ile kodlanır.

Kod yapısı:

- başlangıç karakteri (çubuklar ve rakamlar arasında bir benzerlik kurarsak 1011);
- ilk veri karakteri;
- ayırıcı (01);
- ikinci veri karakteri veya 5 haneli kod durumunda 4 karakter.

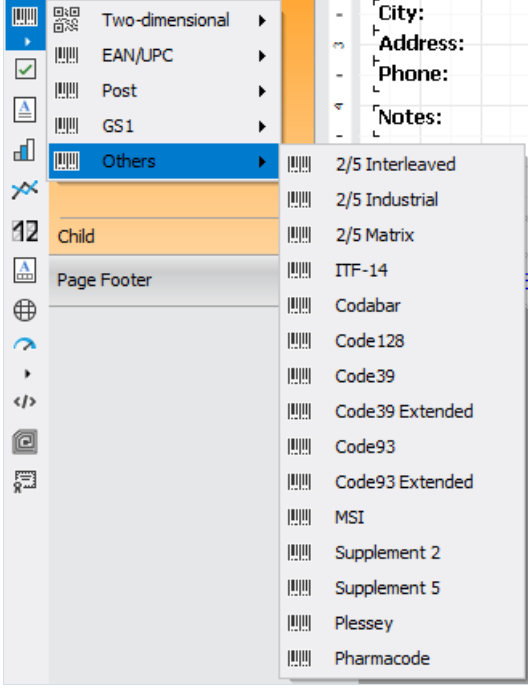
Açıkça bir kontrol karakteri ve durdurma karakteri yoktur. Ayırıcının ardından, kabul edilebilir sayıda karakter okunur.

Veri, EAN'da kullanılan "sol çift" ve "sol tek" kümelerine göre kodlanır. Ek barkodun boyutuna bağlı olarak, farklı karakter parite desenleri kullanılır. Karakterin paritesi, doğrulama kodunu (checksum) belirler. Eğer ek barkod iki haneli ise, birinci ve ikinci rakamların eklenmesiyle oluşan iki haneli sayı 4'e bölünmelidir. Ardından, bölme işleminden kalan sayı çift bir rakam ise, birinci karakter çift pariteyle, ikinci karakter ise tek pariteyle kodlanır. Bu, birinci rakamın kodlanmasında çift değerler kümesinin, ikinci rakamın kodlanmasında ise tek değerler kümesinin kullanılacağı anlamına gelir.

Şimdi, tarama yapıldığında, okunan değer paritesi belirlenecektir. Eğer bu, hesaplanan pariteyle uyuşmazsa, yani beklenen pariteyle örtüşmezse, bu barkodun yanlış okunduğu anlamına gelir.

Beş haneli kod durumunda, doğrulama kodu hesaplaması daha karmaşıktır. Kodun son rakamının tek bir pozisyonda olduğu kabul edilir. Son rakamdan başlayarak, sırayla tek ve çift pozisyonlar atanır. Ardından, tüm tek rakamların toplamı alınır ve 3 ile çarpılır. Tüm çift rakamların toplamı ise 9 ile çarpılır. Son olarak, önceki iki hesaplamadan toplamından ölçü birimi alınır, yani en sağdaki rakam. Bu doğrulama rakamı, özel bir tabloda parite desenini belirlemek için kullanılır.

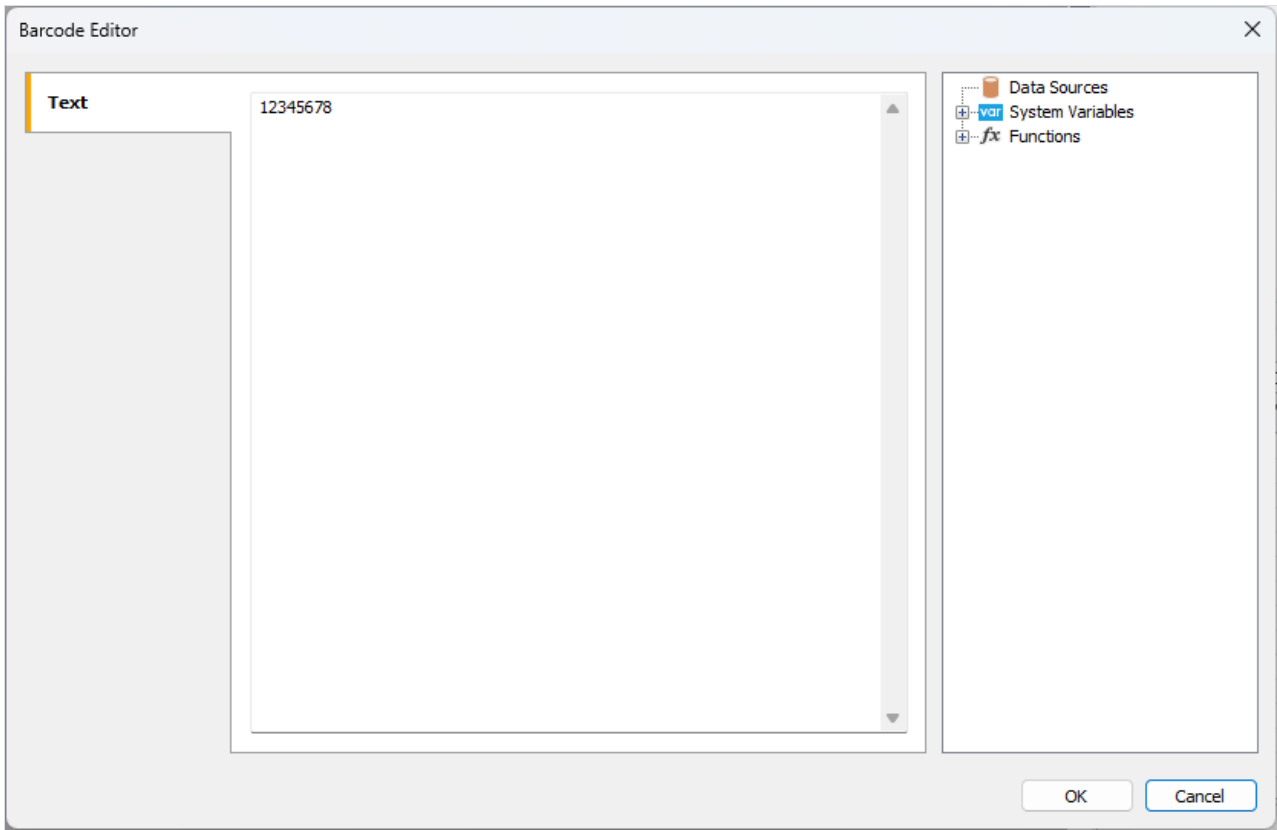
FastReport .NET'te bir Supplement 2 (iki haneli kod için) veya Supplement 5 (beş haneli kod için) barkodu oluşturmak için, Rapor Tasarımcısı'ndaki Bileşenler Panelinden Barkod nesnesini seçin. Açılır listeden "Diğerleri" kategorisine gidin ve ardından Supplement 2 veya Supplement 5 seçeneğini seçin:



Barkodu seçtikten sonra, onu Rapor Sayfasına yerleştirin. Ek kodu ana kodun yanına yerleştirin:



Eklenen barkodu düzenleyiciyi açmak için çift tıklayın. Ayrıca, eklenen nesneye sağ tıklayarak erişilen bağlam menüsündeki düğmeye tıklayarak da barkod düzenleyicisini açabilirsiniz:



Tüm barkodlarda olduğu gibi, **Supplement** barkodunun da FastReport .NET içinde nesne özellikleri denetleyicisinde (Property Inspector) düzenlenebilecek çeşitli özellikleri vardır:

özellik	açıklama
Döndürme (Angle)	Nesnenin 0, 90, 180 veya 270 derece gibi sabit açılardan biriyle döndürülmesini sağlar.
Yakınlaştırma (Zoom)	Barkodun ölçeklendirilmesini belirler. Bu özellik yalnızca "Otomatik Boyutlandırma" (AutoSize) özelliği ile birlikte kullanılır.
Otomatik Boyutlandırma (AutoSize)	Bu özellik etkinleştirildiğinde, nesne barkodun tamamını gösterecek şekilde genişletilir. Özellik devre dışı bırakıldığında, barkod nesnenin boyutlarına sığacak şekilde ölçeklendirilir.
Metni Göster (ShowText)	Barkodun alt kısmında metnin gösterilip gösterilmeyeceğini belirler.
Veri Sütunu (DataColumn)	Nesnenin metnin yükleneceği veri alanını belirtir.
İfade (Expression)	Nesnenin metnini döndüren ifadeyi belirtir.
Metin (Text)	Nesnenin metni.
Dolgu (Padding)	Nesnenin kenarlarından olan boşlukları piksel cinsinden belirlemenizi sağlar.
Geniş Çubuk Oranı (WideBarRatio)	Bu özellik tüm çizgisel (lineer) barkodlarda mevcuttur. Barkodun geniş çizgilerinin göreceli boyutunu belirler.
Doğrulama Toplamı (CalcChecksum)	Bu özellik birçok çizgisel barkodda mevcuttur. Denetim toplamının otomatik olarak hesaplanıp hesaplanmayacağını belirler. Bu özellik devre dışı bırakıldığında, denetim toplamı nesnenin metninde yer almalıdır.

**Dikey Taşıyıcı Çubukları
Göster
(DrawVerticalBearerBars)**

Bu özellik etkinleştirildiğinde, nesnede yan çizgiler görüntülenir.

Plessey

Plessey barkodu, 1971 yılında aynı adı taşıyan şirket tarafından geliştirilmiştir. Bu, esas olarak mağaza raflarında ürünlerin etiketlenmesinde ve depo takibinde kullanılan klasik bir boyutlu (1D) çizgisel barkoddur.

Oluşturulduğu dönemdeki temel avantajı, nokta vuruşlu yazıcılarda kolayca yazdırılabilmesiydi. Günümüzde ise eski kabul edilmekte olup neredeyse hiç kullanılmamaktadır.

Plessey, onaltılı (hexadecimal) rakamları (0-F) kodlayabilir. Her rakam dört bitten – çizgilerden – oluşur. Sıfır ince çizgi, bir kalın çizgi ile temsil edilir. Rakamların yanı sıra A'dan F'ye kadar olan harfler de kodlanabilir. Barkodun yapısı; başlangıç kodu, kodlanmış veriler, kontrol toplamı kodu, bitiş işareti ve ters yönde okuma için bir kod içerir. Bu sayede barkod her iki yönden de okunabilir.



Plessey'in bir türevi olan **MSI** barkodu ise yalnızca sayıları kodlayabilir ve ters yönde okuma kodu içermez. MSI, Mod-10, Mod-11, Mod-1010, Mod-1110 gibi çeşitli kontrol toplamı kodlarını destekler.

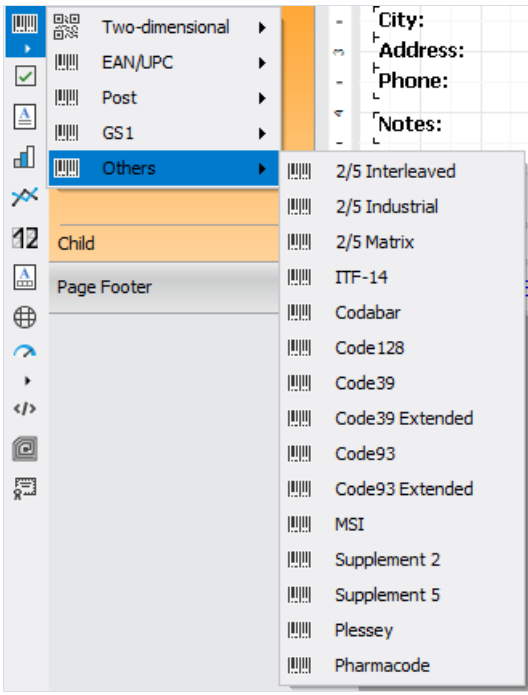


Hem MSI hem de standart Plessey barkodları uzunluk açısından sınırlı değildir; ancak kod çok uzun olursa ambalaj üzerine sığmayabilir ve barkod okuyucu bu uzunlukta bir kodu okuyamayabilir.

Aşağıda böyle uzun bir kodun nasıl görüneceği belirtilmiştir:



FastReport .NET'te Plessey barkodu oluşturmak için rapor tasarımcısındaki bileşen panelinden Barkod (Barcode) nesnesini seçin. Açılır listeden "Diğer" (Other) kategorisine, ardından Plessey seçeneğine gidin:



Barkod seçildikten sonra rapor sayfasına yerleştirin. Eklenecek barkoda çift tıklayarak düzenleyiciyi açabilirsiniz. Ayrıca, eklenen nesneye sağ tıklayarak açılan bağlam menüsünden düzenleme simgesine tıklayarak da barkod düzenleyicisini açabilirsiniz.

Pharmacode

Pharmacode – Alman şirketi **LAETUS GMBH** tarafından özellikle **ilaç ambalajları** için geliştirilmiş ikili (binary) bir koddur. **CODE39** standardının bir alt kümesidir. Bu kod, **ilaç endüstrisinde** ürün ambalaj kontrol sisteminin bir parçası olarak yaygın şekilde kullanılır.

Otomatik ambalaj sistemleri çerçevesinde, Pharmacode evrensel tanımlayıcılar kullanılarak ilaç sevkiyatlarının kolayca taranmasını ve kaydedilmesini sağlar. Aynı zamanda tarayıcılar yardımıyla bir ilaç partisinin başka bir partiyle karışıp karışmadığı kolayca tespit edilebilir.

Pharmacode, baskı hatalarına rağmen kodun okunmasını garanti eder. Ayrıca, kod dışında ambalajın geri kalan kısmının da doğru şekilde basıldığından emin olmak için Pharmacode, diğer lazerle ya da lazer öykünmeli okuyucular için tasarlanmış barkodların aksine, **farklı renklerde** (kod ve arka plan) basılabilir. Bu, Pharmacode'un **LAETUS'un özel beyaz ışıklı tarayıcılarıyla** okunabilmesi sayesinde mümkündür. Bu özellik, Pharmacode'u siyah mürekkep içermeyen ambalaj veya belgelerde baskı için oldukça **kullanışlı** hale getirir.

Yukarıda belirtildiği gibi, **Pharmacode farklı renklerde basılabilir**. Kodun ve arka planın rengi siyah ve beyaz dışında da olabilir. Okuma işlemi için kullanılan tarayıcı tipine bağlı olarak uygulanan özel **renk kombinasyonları** standartları mevcuttur. Örneğin, standart siyah-beyaz tarayıcılar yalnızca yüksek kontrastlı kod ve arka planı algılayabilirken, renk algılayabilen özel tarayıcılar bu konuda katı sınırlamalara sahip değildir.

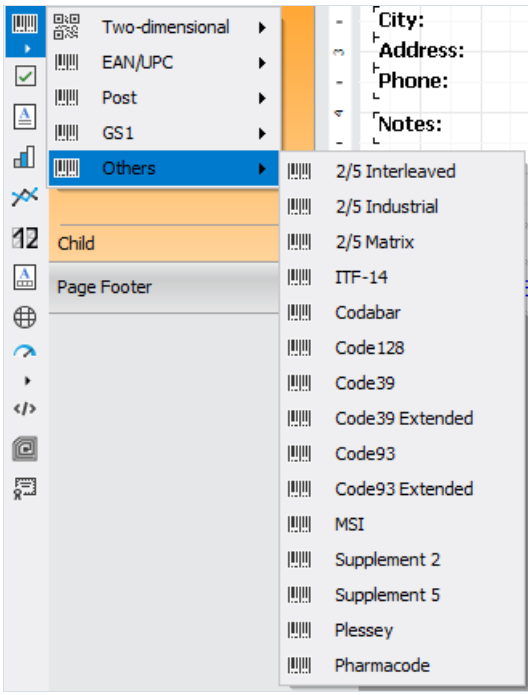
Diğer 1D barkodlardan farklı olarak, **Pharmacode verileri onluk değil ikilik sistemde** saklar. Ayrıca yalnızca **3 ile 131070 arasında** tekil tamsayıları temsil edebilir. Minimum çizgi sayısı, 3 sayısı için **2 çizgi**, maksimum değer olan 131070 için ise **16 çizgidir**. Pharmacode'un bir diğer benzersiz özelliği, diğer doğrusal barkodların aksine, **sağdan sola doğru okunmasıdır**. Soldan sağa okunursa tamamen farklı bir sayı dizisi elde edilir. Pharmacode standardı, geliştirici **LAETUS** tarafından belirlenmiş olup **PharmaCode Guide** adlı belgede tanımlanmıştır.

Aşağıda bir Pharmacode örneği verilmiştir:

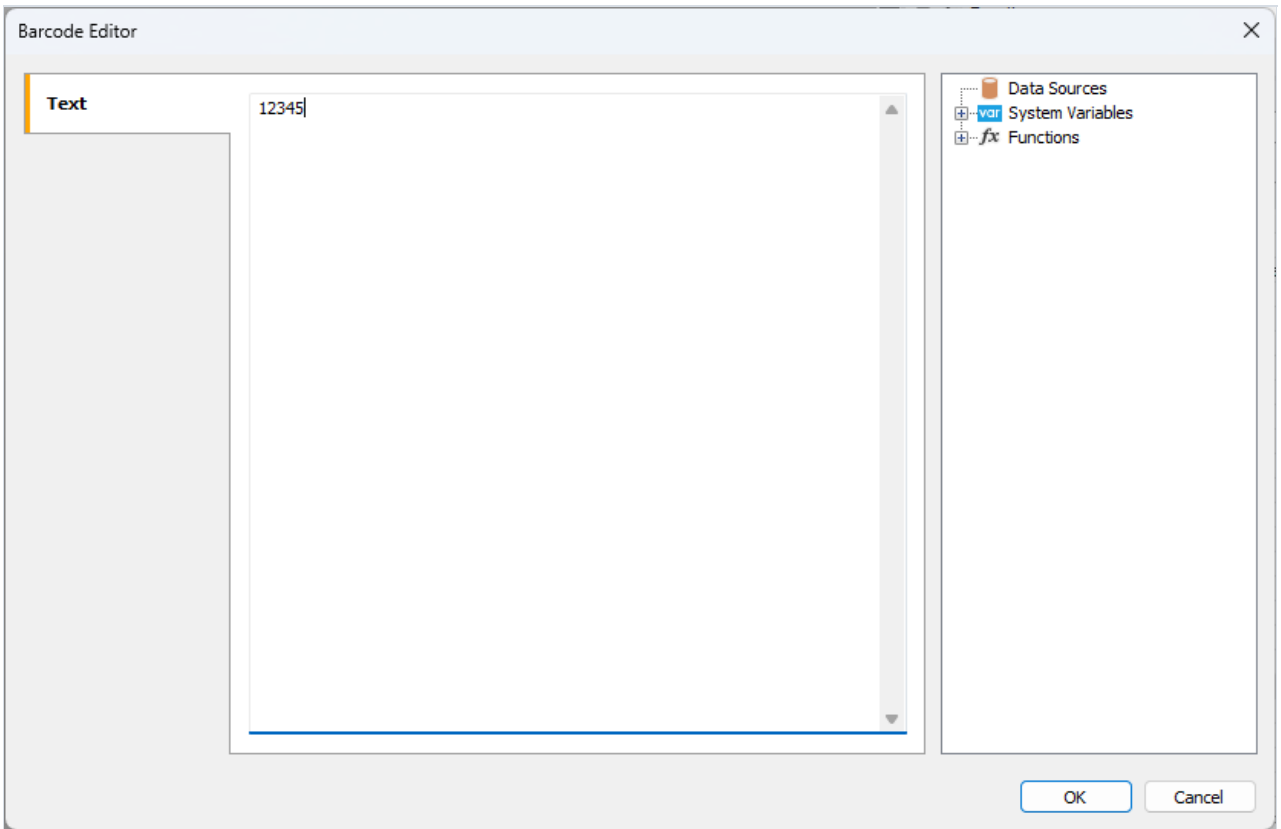


FastReport .NET, raporlarınızda bu tür barkodları oluşturmanıza olanak tanır. Barkod içeren ambalaj tasarımınızı doğrudan rapor tasarımcısında geliştirebilirsiniz.

FastReport .NET'te **Pharmacode barkodunu oluşturmak** için rapor tasarımcısındaki bileşenler panelinden **Barkod (Barcode)** nesnesini seçin. Açılır listede "**Diğer**" kategorisine geçin ve **Pharmacode**'u seçin:



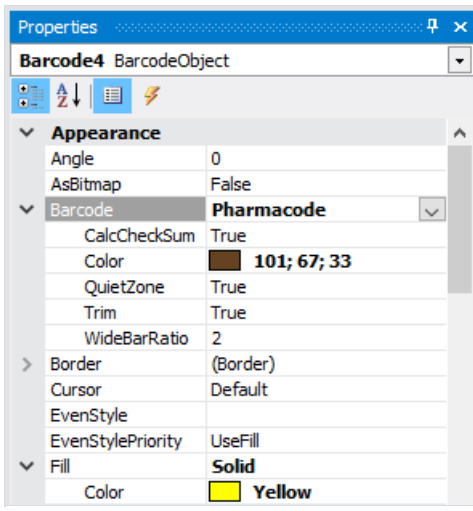
Barkodu seçtikten sonra rapor sayfasına yerleştirin. Eklenen barkod üzerine çift tıklayarak düzenleyiciyi açabilirsiniz. Ayrıca sağ tıklamayla açılan bağlam menüsündeki simgeye tıklayarak da düzenleyiciye ulaşabilirsiniz:



Kod değeri olarak bir sayı dizisi girebilir, bir fonksiyon, rapor değişkeni ya da veritabanından gelen bir değer atayabilirsiniz.

Barkodun özelliklerinde, çizgiler arasındaki boşluk oranını (**WideBarRatio**), barkodun yüksekliğini (**Height**) ve altına sayısal değerini yazılıp yazılmayacağını (**ShowText**) ayarlayabilirsiniz.

Varsayılan olarak barkod, **beyaz arka plan üzerinde siyah renktedir**. Kodun rengini, **Barcode -> Color** özelliğinden; arka plan rengini ise **Fill -> Color** özelliğinden değiştirebilirsiniz:

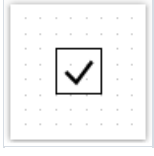


Renk ayarlarından sonra Pharmacode şu şekilde görünebilir:



Nesne "Onay Kutusu"

Nesne, raporda bir onay kutusu görüntülenmesini sağlar. Görünümü aşağıdaki gibidir:



Nesne, iki durumu gösterebilir: "açık" ve "kapalı". Nesnenin durumunu aşağıdaki yöntemlerle belirleyebilirsiniz:

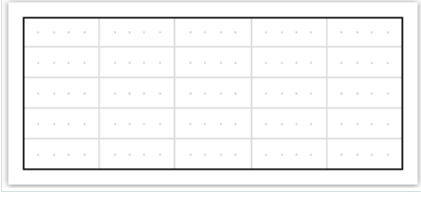
- **Checked özelliği:** Durumu mantıksal olarak belirtin;
- **DataColumn özelliği:** Nesneyi veri alanına bağlayın;
- **Expression özelliği:** "Doğru" veya "Yanlış" döndüren bir ifade belirtin.

Nesnenin sahip olduğu özellikler şunlardır:

Özellik	Açıklama
İşaretili Sembol (CheckedSymbol, UncheckedSymbol)	Bu özellikler, işaretili ve işaretsiz durumlarda görüntülenen sembol tipini belirler.
İşaret Rengi (CheckColor)	Bu özellikte, işaret sembolünün rengini belirtebilirsiniz.
İşaret Genişliği Oranı (CheckWidthRatio)	İşaret genişliği, nesnenin boyutlarına bağlıdır. 0,2 ile 2 arasında herhangi bir değeri ayarlayabilirsiniz. Varsayılan olarak, özellik 1 değerindedir.
Gizle, eğer kapalı (HideIfUnchecked)	Bu özellik, nesnenin "Kapalı" durumunda olması halinde gizlenmesini sağlar.
İşaretili (Checked)	Bu özellik, nesnenin durumunu doğrudan kontrol etmenizi sağlar.
Veri Sütunu (DataColumn)	Veri sütunu, nesnenin durumunun yükleneceği alandır. Bu alanın Boolean türünde olması gerekmektedir.
İfade (Expression)	Nesnenin durumunu döndüren ifade. İfade true veya false değeri döndürmelidir.

"Tablo" Nesnesi

"Tablo" nesnesi, satırlardan, sütunlardan ve hücrelerden oluşur ve Microsoft Excel tablosunun basitleştirilmiş bir benzerini temsil eder. Şu şekilde görünür:



Bu nesnenin kullanımını daha ayrıntılı olarak ["Rapor Oluşturma"](#) bölümünde okuyabilirsiniz.

Nesnenin aşağıdaki özellikleri vardır:

Özellik	Açıklama
"Sütun Sayısı" (ColumnCount)	Gerekli sütun sayısını hızlıca oluşturmanıza olanak tanır. Tabloyu daha az sütun içeriyorsa, sütunlar eklenir; fazla ise, sütunlar silinir.
"Satır Sayısı" (RowCount)	Gerekli satır sayısını hızlıca oluşturmanıza olanak tanır. Tabloyu daha az satır içeriyorsa, satırlar eklenir; fazla ise, satırlar silinir.
"Sabit Sütunlar" (FixedColumns)	Tablonun başındaki sabit sütun sayısını belirler. Sabit sütunlar, tablonun başlığını oluşturur. Başlıkların yazdırılması, "Başlıkları Tekrarlama" özelliği ile kontrol edilir. Bu özellik yalnızca dinamik olarak oluşturulan tablolarda çalışır.
"Sabit Satırlar" (FixedRows)	Tablonun başındaki sabit satır sayısını belirler. Sabit satırlar, tablonun başlığını oluşturur. Başlıkların yazdırılması, "Başlıkları Tekrarlama" özelliği ile kontrol edilir. Bu özellik yalnızca dinamik olarak oluşturulan tablolarda çalışır.
"Başlıkları Tekrarlamak" (RepeatHeaders)	Tablonun başlıklarını her yeni sayfada yazdırmanıza olanak tanır. Bu özellik yalnızca dinamik olarak oluşturulan tablolarda çalışır.

"Matris" Nesnesi

"Matris" nesnesi bir tablo türüdür ve "Tablo" nesnesi gibi satırlar, sütunlar ve hücrelerden oluşur. Ancak, matrisin kaç satır ve sütundan oluşacağı önceden bilinmez – bu, bağlandığı verilere bağlıdır.

Nesne şu şekilde görünür:

Employee	[Year]	Total
[Name]	[Revenue]	
Total		

Bu nesnenin kullanımı hakkında daha fazla bilgiyi ["Rapor Oluşturma"](#) bölümünde bulabilirsiniz.

Nesnenin aşağıdaki özellikleri vardır:

özellik	açıklama
"Başlıkları Tekrarla" (RepeatHeaders)	Eğer matris yazdırıldığında birden fazla sayfaya bölünürse, bu özellik matrisin başlıklarının her yeni sayfada yazdırılmasını sağlar.
"Hücreleri Yan Yana" (CellsSideBySide)	Özellik, matrisin veri hücrelerinde birden fazla değer bulunduğunda hücrelerin nasıl yerleştirileceğini belirler. Olası seçenekler şunlardır: - Hücreler yan yana (tek satır halinde) yerleştirilir. - Hücreler alt alta yerleştirilir.
"Stil" (Style)	Bu özellik kullanılarak tüm matris için bir stil belirlenebilir. Önceden tanımlanmış stillerden birini seçebilirsiniz.
"Otomatik Boyut" (AutoSize)	Bu özellik etkinleştirildiğinde, matrisin boyutları otomatik olarak ayarlanır. Nesnenin boyutunu manuel olarak kontrol etmek istiyorsanız, bu özelliği devre dışı bırakın.
"Veri Kaynağı" (DataSource)	Bu özellik, matrisi bir veri kaynağına bağlamayı sağlar. Veri alanı matrise taşındığında bu özellik otomatik olarak doldurulur. Ancak, hücrelerde ifadeler kullanıyorsanız, bu özelliğin doğru şekilde ayarlandığını kontrol edin.
"Filtre" (Filter)	Bu özellik, matrisin veri kaynağına (bkz. DataSource özelliği) uygulanacak veri filtreleme ifadesini içerir.

"Geliştirilmiş Matris" Nesnesi

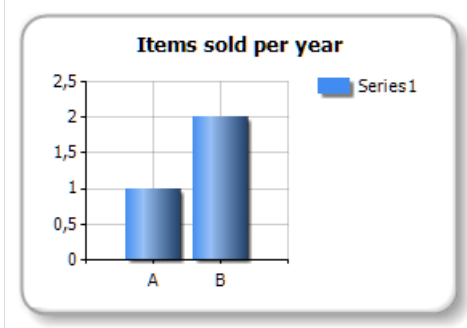
Bu nesne, tıpkı "Matris" nesnesi gibi, özet raporlar oluşturmayı mümkün kılar.

	Top 5 customers		Total
	Total	Others (84)	
► Beverages (12)	\$144 007,88	\$192 684,30	\$336 692,18
► Condiments (12)	\$123 407,20	\$105 665,89	\$229 073,09
► Confections (13)	\$54 823,13	\$112 615,09	\$167 438,23
► Dairy Products (10)	\$103 074,28	\$166 433,01	\$269 507,29
► Grains/Cereals (7)	\$31 368,01	\$64 376,58	\$95 744,59
▼ Meat/Poultry (6)	\$269 495,38	\$113 526,98	\$383 022,36
1. Alice Mutton	\$13 428,48	\$19 269,90	\$32 698,38
2. Mishi Kobe Niku	\$1 319,20	\$5 907,30	\$7 226,50
3. Pâté chinois	\$7 137,60	\$10 288,80	\$17 426,40
4. Perth Pasties	\$6 916,56	\$13 657,61	\$20 574,17
5. Thüringer Rostbratwurst	\$239 795,40	\$60 573,28	\$300 368,67
6. Tourtière	\$898,14	\$3 830,10	\$4 728,24
► Produce (5)	\$67 154,22	\$71 955,36	\$139 109,58
► Seafood (12)	\$31 732,55	\$99 591,19	\$131 323,74
Total	\$825 062,63	\$926 848,41	\$1 751 911,04

Bu nesnenin kullanımı hakkında daha fazla bilgiyi ["Geliştirilmiş Matris Nesnesi"](#) başlıklı bölümde bulabilirsiniz.

"Grafik" nesnesi

"MS Chart Grafik" nesnesi, grafik oluşturmaya sağlar. Toplamda 30'dan fazla farklı grafik türü mevcuttur – histogramlar, alan grafikler, çizgi grafikler, baloncuk grafikler, dairesel grafikler, yaprak grafikler, finansal grafikler, piramit grafikler ve aralık grafikler. Nesne şu şekilde görünür:



Bu nesnenin kullanımı hakkında daha detaylı bilgiyi "[Rapor Oluşturma](#)" bölümünde bulabilirsiniz.

Nesnenin aşağıdaki özellikleri vardır:

Özellik	Açıklama
Grafik (Chart)	Microsoft Chart nesnesine referans.
Değerleri Hizala (AlignXValues)	Bu özellik, birden fazla serideki değerlerin hizalanmasını (eksik değerlerin eklenmesini) sağlar. İki veya daha fazla serinin tek bir grafikte gösterilmesinde kullanılır.
"Otomatik Seriler" (AutoSeriesColumn, AutoSeriesColor, AutoSeriesSortOrder)	Bu özellikler, otomatik olarak oluşturulan serilerin ayarlanmasını sağlar. Detaylar için "Rapor Oluşturma" bölümüne bakınız.
"Veri Kaynağı" (DataSource)	Bu özellik, grafiğin veri kaynağına bağlanmasını sağlar.
"Filtre" (Filter)	Bu özellik, veri kaynağına uygulanacak olan filtreleme ifadesini içerir (bkz. Veri Kaynağı özelliği, DataSource).

"Zip Code" nesnesi

"Zip Code" nesnesi, zarf üzerinde posta kodunu yazdırmanıza olanak tanır. Sayısal karakterleri (0-9) gösterebilir. Nesne aşağıdaki gibidir:



Bir nesneyi verilere bağlamak için aşağıdaki yöntemlerden birini kullanabilirsiniz:

- Zip kodu verisini **Text** özelliğine ayarlayarak;
- Nesneyi, **DataColumn** özelliğini kullanarak veri sütununa bağlayarak;
- Zip kodu verisini döndüren bir ifade belirleyerek **Expression** özelliğini ayarlayarak.

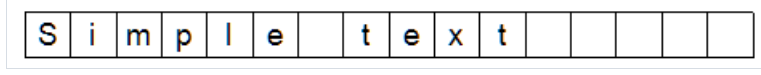
"Zip Code" nesnesinin aşağıdaki özellikleri vardır:

Özellik	Açıklama
"Segment Sayısı" (SegmentCount)	Posta kodundaki segment sayısını belirler. Varsayılan olarak 6'dır.
"Segment Boyutu" (SegmentWidth, SegmentHeight)	Bir segmentin boyutu. Standartlara göre, segmentin boyutu 0,5x1 cm'dir.
"Segmentler Arasındaki Mesafe" (Spacing)	İki komşu segmentin sol kenarları arasındaki mesafe. Standartlara göre, mesafe 0,9 cm'dir.
"Izgarayı Göster" (ShowGrid).	Özellik, nokta izgarasının gösterilip gösterilmeyeceğini belirler.
"İşaretçileri Göster" (ShowMarkers).	Özellik, işaretçilerin (posta kodunun üzerinde bulunan kalın yatay çizgiler) gösterilip gösterilmeyeceğini belirler.
Veri Sütunu" (DataColumn)	Nesnenin metninin yükleneceği veri sütunu.
"İfade" (Expression)	Nesnenin metnini döndüren ifade.
"Metin" (Text).	Nesnenin metni.

"Hücreli Metin" nesnesi

"Hücrelerde Metin" nesnesi, her karakterin ayrı bir hücrede görüntülediği metnin basımına yönelik tasarlanmıştır. Bu nesne, örneğin vergi raporlaması gibi formların basımında sıklıkla kullanılır.

Görünümü şu şekildedir:



Özünde, bu nesne "Metin" nesnesinin bir türevidir. Benzer şekilde veriye bağlanabilir. Bunun için, nesnenin metnine veri alanına bir referans yerleştirin, örneğin:

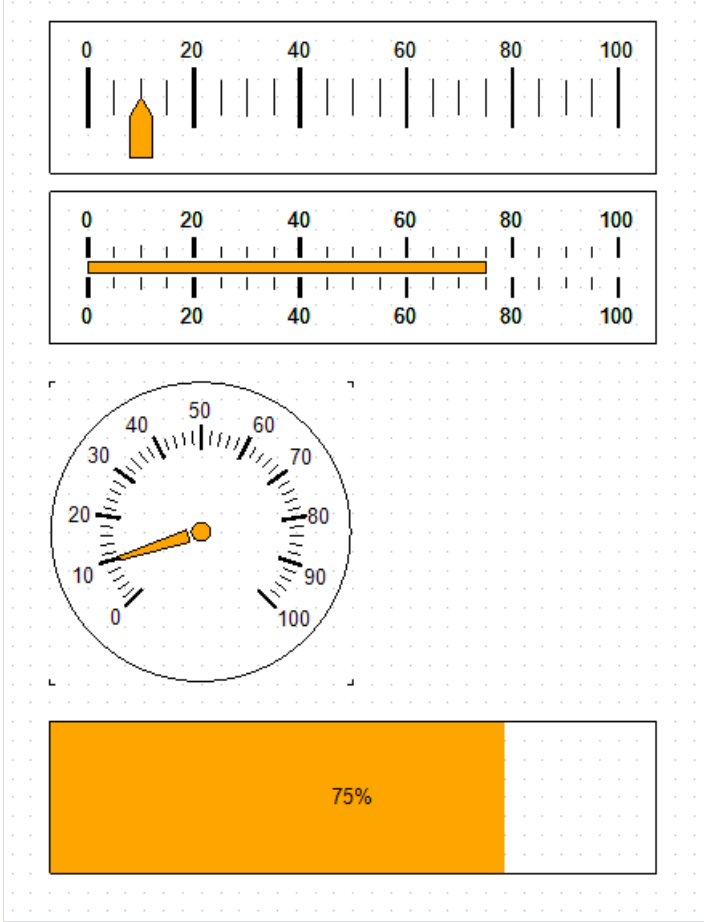
[Employees.FirstName]

Nesnenin aşağıdaki özellikleri vardır:

Özellik	Açıklama
"Hücre Boyutları" (CellWidth, CellHeight)	Özellikler, hücre boyutlarını belirler. Eğer her iki özellik de 0 (varsayılan) ise, hücre boyutları yazı tipi boyutuna bağlı olarak otomatik hesaplanır.
"Hücreler Arası Boşluk" (HorzSpacing, VertSpacing)	Özellikler, bitişik hücreler arasındaki yatay ve dikey boşlukları belirler.

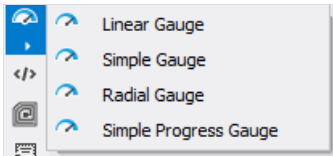
Sensör Nesneleri Grubu

Bu nesneler, belirli bir değerin görsel olarak görüntülenmesi için tasarlanmıştır. Şu anda desteklenen dört farklı sensör türü şunlardır:

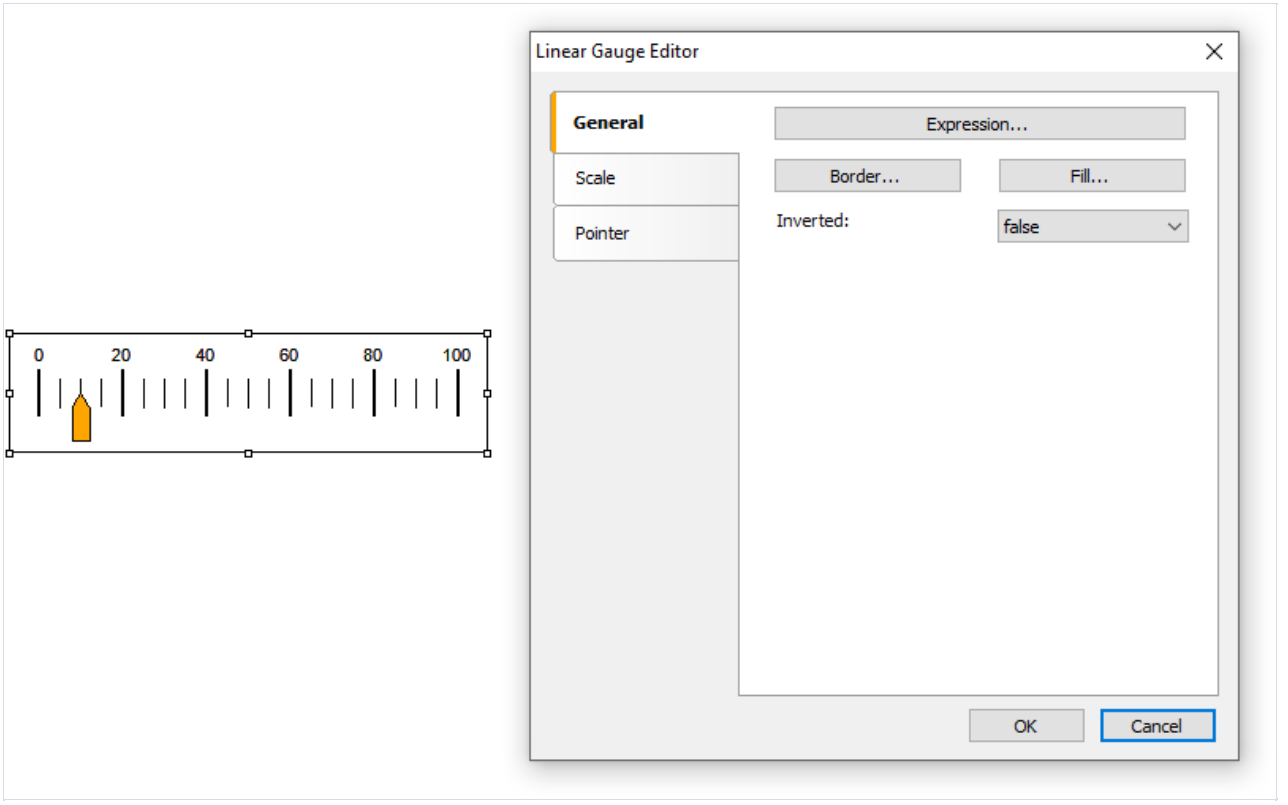


- Doğrusal sensör
- Basit sensör
- Dairesel sensör
- Doğrusal ilerleme sensörü

Raporunuza bir **Sensör** nesnesi eklemek için, düğmeye tıkladığınızda açılan alt menüden bir seçenek belirleyin .



Sensörün bazı görsel özellikleri, örneğin gösterge rengi, değiştirilebilir. Sensör parametrelerini düzenlemek için sensörün üzerine **çift tıklayın** veya bağlam menüsündeki düğmeye tıklayın .



"İfade" düğmesi, bir değeri manuel olarak girebileceğiniz veya sensörü verilere bağlayarak bir ifade oluşturabileceğiniz bir metin düzenleyicisini açar.

"Dijital İmza" nesnesi

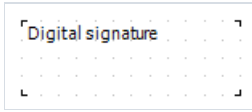
Dijital imza – benzersizliği ve özgünlüğü garanti eden, yazarlığın kesin olarak belirlenmesini ve belgedeki değişikliklere karşı korunmasını sağlayan bir şifredir. Güvenilir şifreleme algoritmaları sayesinde bu tür imzalar el yazısıyla atılmış imzalardan geri kalmaz; hatta daha iyi ve daha güvenilirdir.

Mevcut sürümde iki tür imza mevcuttur:

1. İmza alanı (signature field) – belge içinde, kullanıcıya kendi sertifikasını ekleme imkânı tanıyan özel bir alanın varlığını ifade eder. Bu alana tıklanarak imza işlemi gerçekleştirilir.

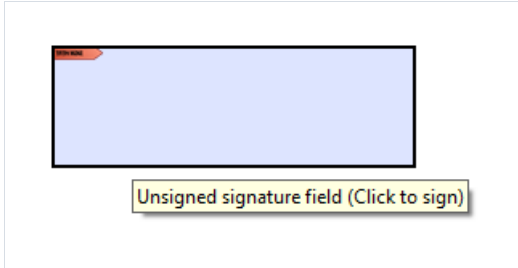
Bu özelliği uygulayan nesne, bu tuşa basılarak eklenir: 

Buna Dijital İmza denir. Bu denetim rapor sayfasına yerleştirildiğinde şu şekilde görünür:



Rapor görünümünde görünmez. Fonksiyonu yalnızca PDF dışı aktarma ile sınırlıdır. Yani, bu alanı bir PDF dosyasını Acrobat Reader'da görüntülerken göreceksiniz.

Dışa aktarıldıktan sonra, alan şu şekilde görünecektir:



İmza alanına tıklayın ve belgeyi imzalamak için sertifika seçme penceresini görün:



2. Görünmeyen imza – ardından sertifika ile imzalanır. Görsel olarak görünmez, ancak belge özelliklerinde imzalayan kişi, imzanın geçerliliği, imzalama anındaki belge sürümü ve diğer bilgiler hakkında bilgi edinilebilir.

Görünmeyen imza ile dışa aktarılan bir PDF dosyasını imzalamak için, rapor sayfasına "Dijital İmza" nesnesi eklemenize gerek yoktur. Dışa aktarma ayarlarında imzalama seçeneğini etkinleştirmeniz ve orada bir sertifika seçmeniz yeterlidir:

Export to PDF

Export

Options

Vector Graphics

Information

Security

Digital signature

Viewer

Sign Information

☒ Sign document

Location

Reason

Contact info

Certificate Information

Certificate path

C:\Certificate.p12

Certificate password

☐ Save password

☐ Open after export

OK

Cancel

FastReport'da İlk Rapor

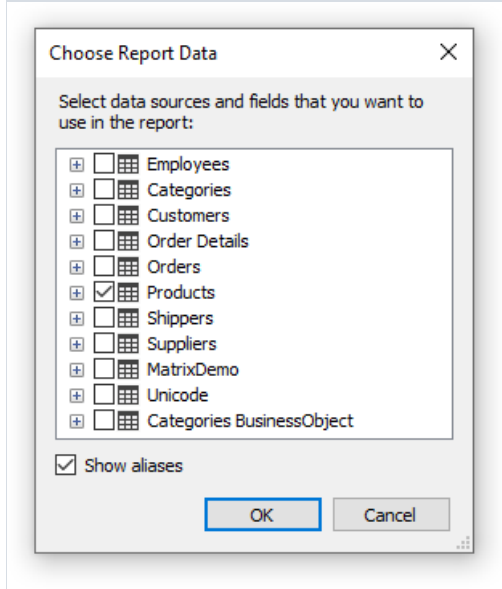
Basit bir rapor oluřturalım; bu rapor ürün listesini yazdıracaktır. Veri kaynağı olarak, demo veritabanında yer alan Products tablosu kullanılacaktır.

Ařağıda açıklanan işlemleri, rapor tasarımcısını çağırabileceğiniz Demo.exe adlı demo program üzerinden gerçekleřtireceksiniz.

Örnek 1. Raporun Elle Oluşturulması

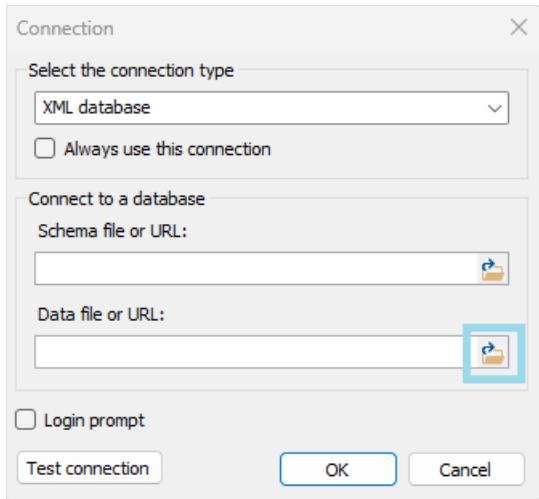
Bu örnekte bir raporu manuel olarak oluşturacağız. Bunun için aşağıdaki adımları uygulayın:

- Araç çubuğundaki  düğmesine tıklayın ve açılan "Oluştur" penceresinde "Boş Rapor" seçeneğini seçin;
- Menüden "Veri" > "Rapor için verileri seç..." seçeneğine tıklayın ve "Products" veri kaynağını işaretleyin:

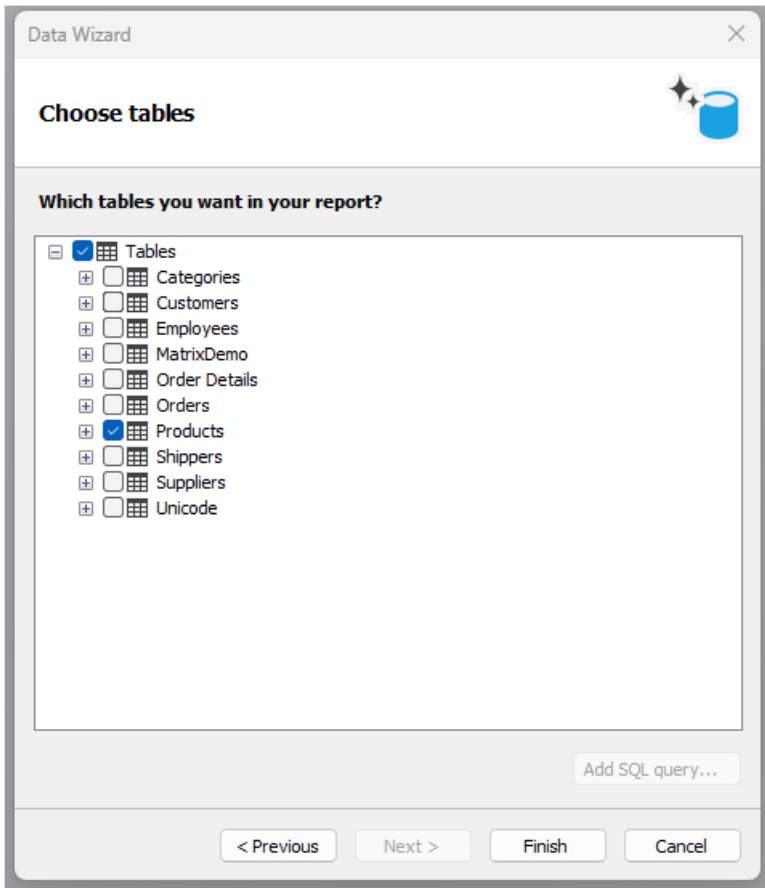


Alternatif olarak XML dosyasından veri kaynağı eklemek için:

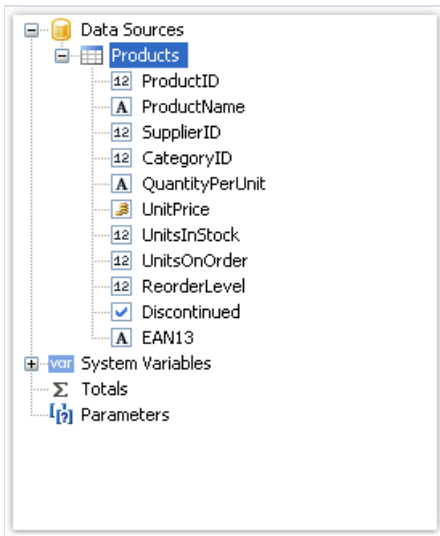
- Menüden "Veri" > "Yeni veri kaynağı" seçeneğine tıklayın. "Veri Bağlantı Sihirbazı" penceresi açılır;
- "Yeni bağlantı" düğmesine tıklayın;
- Bağlantı türü olarak XML veritabanını seçin. Veri dosyasını seçmek için dosya simgesine veya URL adresine tıklayın;



- Açılan "Dosya Aç" iletişim kutusunda örneğin `C:\Program Files (x86)\Fast Reports\2024.2.12\FastReport .NET Trial\Demos\Reports` konumuna gidin ve `nwind.xml` dosyasını seçin;
- Veritabanı bağlantısını test etmek için "Test" düğmesine tıklayın. Test başarılı olursa, bağlantı penceresini kapatmak için "Tamam" düğmesine tıklayın;
- "Veri Bağlantı Sihirbazı"nda "İleri" düğmesine tıklayın;
- Kök düğüm olan "Tablolar" bölümünü genişletin, "Products" tablosunu seçin ve "Son" düğmesine tıklayın;



- “Veri” hizmet penceresine geçin (ekranda yoksa “Veri | Veri penceresini göster” menü seçeneğiyle görüntülenebilir) ve “Veri Kaynakları” > “Products” ögesini açın;

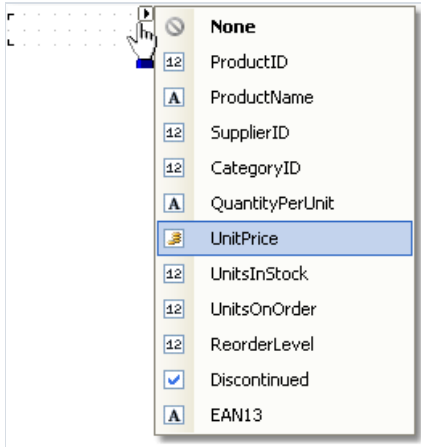


- **ProductName** alanını tutup “Veri” bandına sürükleyin. FastReport bu alanla bağlı bir “Metin” nesnesi ve başlık oluşturur;
- **UnitPrice** alanını farklı bir şekilde ekleyelim. Araç çubuğundaki “Nesneler” bölümünde “Metin” düğmesine tıklayın;

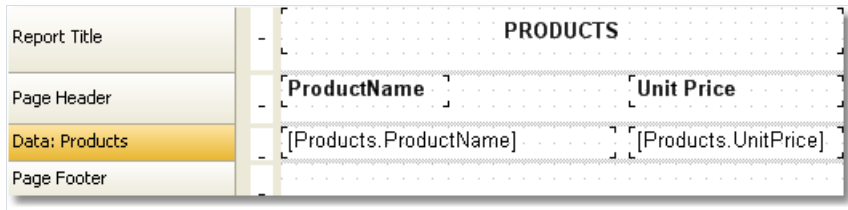


- Farenin düğmesini bırakın ve imleci tasarım alanına getirin – FastReport bir nesne eklemeyi önerir. Uygun bir konum seçip farenin düğmesine tıklayarak nesneyi yerleştirin;

- İmleci nesnenin üzerine getirin ve sağ üst köşedeki küçük oka tıklayın. Açılan alanlar listesinden UnitPrice alanını seçin;



- UnitPrice alanı için bir başlık oluşturun: Aynı yöntemle "Sayfa Başlığı" bandına yeni bir "Metin" nesnesi ekleyin. Nesneye çift tıklayın ve içine "Unit Price" yazın;
- "Rapor Başlığı" bandına bir "Metin" nesnesi yerleştirin ve içine "PRODUCTS" yazın;
- Rapor ve sayfa başlıklarında bulunan nesneler için yazı tipi stilini kalın yapın. Bunun için üç nesneyi Shift tuşuna basılı tutarak seçin ve araç çubuğundaki "Metin" stil düğmesine tıklayın. Sonuç şu şekilde görünecektir:



Raporu çalıştırma yöntemleri:

- Kontrol panelindeki  düğmesiyle;
- Klavye kısayolu olan Ctrl+P ile.

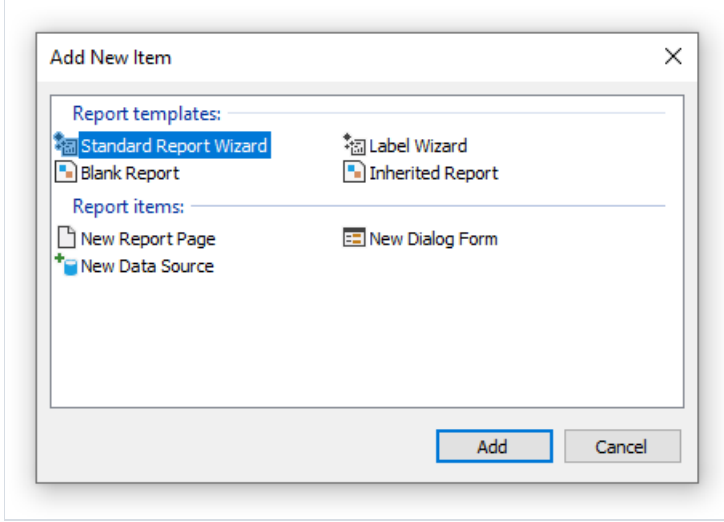
Rapor oluşturulacak ve önizleme penceresinde görüntülenecektir:

PRODUCTS	
Product Name	Unit Price
Chai	18
Chang	19
Aniseed Syrup	10
Chef Anton's Cajun Seasoning	22
Chef Anton's Gumbo Mix	21,35
Grandma's Boysenberry Spread	25
Uncle Bob's Organic Dried	30
Northwoods Cranberry Sauce	40
Mishi Kobe Niku	97
Ikura	31

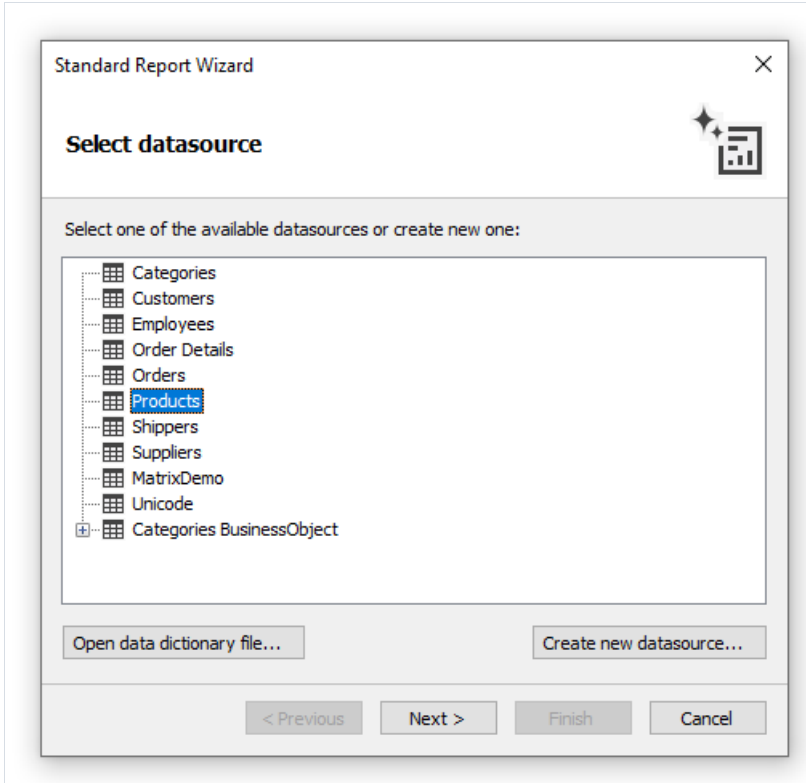
2. Örnek. Sihirbaz kullanarak rapor oluşturma

Bu örnekte, "Standart Rapor Sihirbazı" yardımıyla bir rapor oluşturacağız. Bunun için aşağıdaki adımları izleyin:

- Araç çubuğundaki  düğmesine tıklayın ve açılan "Oluştur" penceresinden Standart Rapor Sihirbazı seçeneğini seçin:



- Sihirbazın ilk adımında "Products" (Ürünler) tablosunu seçin ve "İleri" düğmesine tıklayın:



- Sihirbazın ikinci adımında **ProductName** (Ürün Adı) ve **UnitPrice** (Birim Fiyatı) alanlarını seçin ve bunları seçilen alanlar listesine aktarın:

Standard Report Wizard

Select data columns

Select data columns that you want to show in a report.

Available columns:

- ProductID
- SupplierID
- CategoryID
- QuantityPerUnit
- UnitsInStock
- UnitsOnOrder
- ReorderLevel
- Discontinued
- EAN13

Selected columns:

- ProductName
- UnitPrice

< Previous Next > Finish Cancel

- Sihirbazın geri kalan adımlarını "İleri" düğmesine tıklayarak atlayabilirsiniz;
- Sihirbazın son adımında Blue (Mavi) stilini seçin ve "Bitir" düğmesine tıklayın:

Standard Report Wizard

Select style

Select the report style.

Standard
Gray
Green
Sand
Blue

CUSTOMERS

ID	Company Name
A	
ALFIO	Alfreda Fullerton
ANATR	Ana Trujillo Emparedados y helados
ANTON	Antonio Moreno Taquería
AROUT	Around the Horn
B	
BERRG	Berglund's smörgåsar
BIALS	Bialas Szwedzkie
BOLID	Bonadelli's pizza & ice
BOLID	Bolid's Comidas preparadas
BONAP	Bon app'
BOTTU	Bottou-Collier Markets
BUSEV	B's Beverages
C	
CACTU	Cactus Comidas para llevar
CEMTC	Centro comercial Modocums
CHOPS	Chop-suey Chinese
COMUS	Comerio Uniano

Page | Page

< Previous Next > Finish Cancel

FastReport, aşağıdaki görünüme sahip bir rapor oluşturacaktır:

Report Title	Products
Page Header	ProductName UnitPrice
Data: Products	[Products.ProductName] [Products.UnitPrice]
Page Footer	[PageN]

Raporu çalıştırmak için kontrol panelindeki  düğmesine tıklayın. Rapor oluşturulacak ve önizleme penceresinde

gösterilecektir:

Products	
ProductName	UnitPrice
Chai	18
Chang	19
Aniseed Syrup	10
Chef Anton's Cajun Seasoning	22
Chef Anton's Gumbo Mix	21,35
Grandma's Boysenberry Spread	25
Uncle Bob's Organic Dried Pears	30
Northwoods Cranberry Sauce	40
Mishi Kobe Niku	97
Ikura	31
Que Pasa	21

Report creation

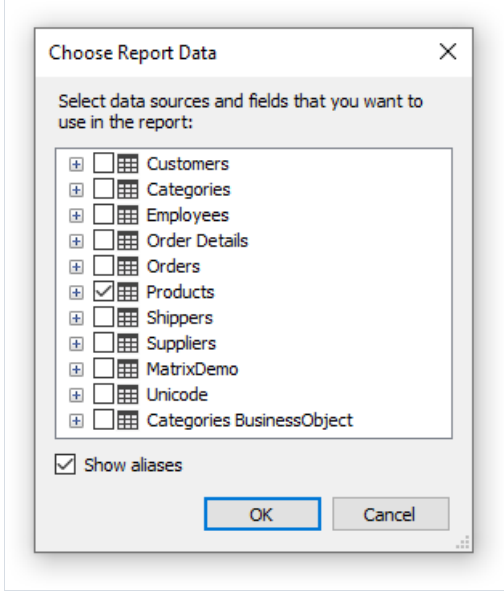
In this chapter we will look at the methods of creating common types of reports. In order to create any report, as a rule, you need to do the following:

1. Choose or create data, which will be used in the report.
2. Create the report structure, by adding the needed bands into the report.
3. Connect the band to a data source.
4. Place the "Text" objects on the bands to print data.
5. Setup the appearance, formatting.

Rapor için Veri Seçimi

Rapor oluşturma işlemine başlamadan önce, raporda basılacak verileri seçmeniz gerekmektedir. Bunu iki farklı şekilde yapabilirsiniz:

Raporla programlama yöntemiyle kaydedilmiş veri kaynaklarından birini seçebilirsiniz. Bunu, "Data|Choose Report Data..." menüsünde, gerekli veri kaynağını işaretleyerek yapabilirsiniz.



Yeni bir veri kaynağı oluşturmak için "Data|New Data Source..." menüsünü kullanın.

Veri kaynakları hakkında daha fazla bilgi için ["Data sources"](#) bölümüne bakın.

Veri kaynağını seçer seçmez, bu kaynak "Data" penceresinde görünür. Artık bu kaynağı raporda kullanabilirsiniz. Birçok rapor yalnızca bir veri kaynağı kullanır. "master-detail" türündeki raporlar için, birbirine bağlı iki veri kaynağını seçmeniz gerekir (ilişkiler hakkında daha fazla bilgi için "Data sources" bölümüne bakabilirsiniz). İlişkili kaynaklardan veri yazdıran bir raporda da birden fazla veri kaynağına ihtiyaç duyulabilir.

Nesnelerin Yüksekliklerinin Otomatik Ayarlanması

Rapor oluşturma aşamasında metnin boyutu genellikle bilinmediğinden, metin basılması gerekebilmektedir. Örneğin, bu durum bir ürün açıklaması içeren metinler için geçerli olabilir. Bu durumda şu görevlerin çözülmesi gerekmektedir:

- Nesnenin, tüm metni kapsayacak şekilde yüksekliğinin ayarlanması;
- Değişken miktarda metin içeren nesneleri alacak biçimde bant yüksekliğinin belirlenmesi;
- Raporun genel düzenini bozmadan, bant üzerindeki diğer nesnelerin konumlarının kaydırılması veya yüksekliklerinin değiştirilmesi.

Bu görevler, nesnelerin ve bantların bazı özellikleri kullanılarak çözülebilir:

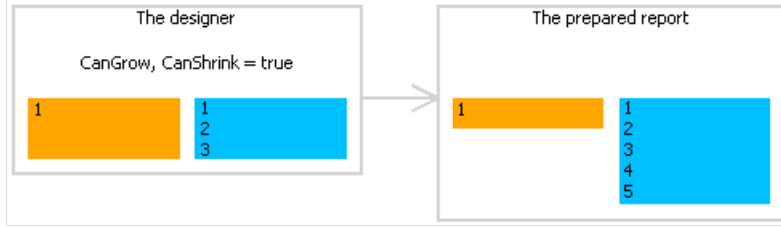
- **Büyüyebilir** ve **Küçülebilir** özellikleri, nesnenin yüksekliğinin otomatik olarak ayarlanmasına olanak tanır;
- **Kaydır** özelliği, genişleyebilen nesnelerin altındaki nesnelerin konumlarının kaydırılmasını sağlar;
- **Aşağıya Doğru Büyü** özelliği, nesnelerin bantın alt kenarına kadar uzamasını mümkün kılar;
- **Çapa** ve **Yapışma** özellikleri, nesnelerin boyutunun bantın boyutuna bağlı olarak kontrol edilmesine imkân tanır.

Bu özelliklerin tümü aşağıda detaylı olarak ele alınacaktır.

CanGrow, CanShrink özellikleri

Her bant ve rapor nesnesinin bu özellikleri vardır. Bu özellikler, bir nesnenin içeriğinin boyutuna bağlı olarak büyüyüp küçülüp küçülmeyeceğini belirler. Her iki özellik devre dışı bırakılmışsa, nesne her zaman tasarımcıda belirtilen boyuta sahip olur.

Bu özellikler, tasarım aşamasında boyutu bilinmeyen bir metni yazdırmak gerektiğinde çok kullanışlıdır. Bir nesnenin tüm metni alabilmesi için CanGrow ve CanShrink özelliklerinin etkinleştirilmesi gerekir:

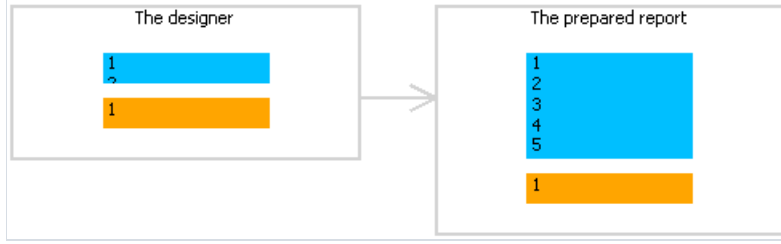


Aşağıdaki nesneler, bir bandın yüksekliğini etkileyebilir:

- "Metin";
- "Rich Text";
- "Resim" (`AutoSize` özelliği etkinleştirilmişse);
- "Tablo".

ShiftMode özelliği

Her rapor nesnesinin bu özelliği vardır. Bu özellik yalnızca "Özellikler" penceresinde erişilebilir. ShiftMode özelliği etkinleştirilen bir nesne, üstündeki nesne büyüyüp küçüldükçe yukarı ve aşağı hareket eder.



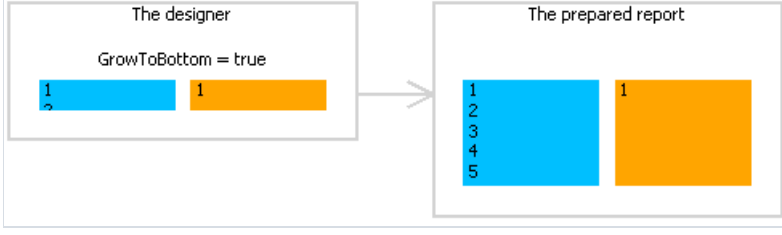
ShiftMode özelliği aşağıdaki değerlerden birine sahip olabilir:

- Her Zaman (varsayılan olarak) - nesnenin her zaman hareket etmesi gerektiğini belirtir;
- Asla - nesnenin hareket etmesine gerek olmadığını belirtir;
- Çakıştığında - nesnenin yalnızca, genişleyen nesne tam olarak üzerine yerleştiğinde (yani, her iki nesne yatay olarak çakıştığında) hareket etmesi gerektiğini belirtir.

Bu özellik, bilgilerin tablo şeklinde yazdırılması sırasında kullanışlıdır; özellikle tablonun birkaç hücresinin üst üste yerleştiği ve değişken miktarda metin içerebileceği durumlarda.

GrowToBottom özelliği

Her rapor nesnesinin bu özelliği vardır. Bu özelliğe sahip bir nesne yazdırıldığında, bir bandın alt kenarına kadar uzar:



Bu, bilgilerin tablo şeklinde yazdırılması gerektiğinde kullanılır. Bir tablo satırında, uzayabilen birkaç nesne olabilir. Bu özellik, tüm nesnelerin yüksekliğini bandın maksimum yüksekliğine ayarlamayı mümkün kılar.

"Çapa" Özelliği (Anchor)

Bu özellik, rapor nesnelerine ait olup, nesnenin konumu ve/veya boyutunun, içinde bulunduğu kapsayıcının boyutları değiştikçe nasıl değişeceğini belirler. Çapa kullanılarak, nesnenin kapsayıcıyla senkronize şekilde genişlemesi veya yer değiştirmesi sağlanabilir.

Kapsayıcı genellikle bir bant (band) olsa da, içinde başka nesneler barındırabilen "Tablo" veya "Matris" gibi nesneler de olabilir.

Özellik, aşağıdaki değerlerden birini veya bu değerlerin herhangi bir kombinasyonunu alabilir.

Değer	Açıklama
Left	Nesnenin sol kenarı sabitlendi. Konteynerin genişliği değiştirildiğinde, nesne sola veya sağa kaymayacaktır.
Top	Nesnenin üst kenarı sabitlendi. Konteyner yüksekliği değiştirildiğinde, nesne yukarı veya aşağı kaymayacaktır.
Right	Nesnenin sağ kenarı sabitlendi. Konteynerin genişliği değiştirildiğinde, nesnenin sağ kenarı ile konteynerin sağ kenarı arasındaki mesafe sabit kalacaktır. Eğer aynı zamanda nesnenin sol kenarı da sabitlenirse, nesne konteynerle senkronize olarak büyüyecek ve küçülecektir.
Bottom	Nesnenin alt kenarı sabitlendi. Konteynerin yüksekliği değiştirildiğinde, nesnenin alt kenarı ile konteynerin alt kenarı arasındaki mesafe sabit kalacaktır. Eğer aynı zamanda nesnenin üst kenarı da sabitlenmişse, nesne konteynerle senkronize olarak büyüyüp küçülecektir.

Varsayılan olarak, bu özelliğin değeri Left, Top olarak ayarlanmıştır. Bu, konteyner boyutları değiştirildiğinde nesnenin konumunun sabit kalacağı anlamına gelir. Aşağıdaki tabloda, sık kullanılan bazı değer kombinasyonları yer almaktadır.

Değer	Açıklama
Left, Top	Varsayılan değer. Konteyner boyutları değiştirildiğinde nesne değişmez.
Left, Bottom	Konteynerin yüksekliği değiştirildiğinde nesne yukarı veya aşağı kayar. Ancak, nesnenin konteynerin alt kenarıyla olan pozisyonu sabit kalır.
Left, Top, Bottom	Konteynerin yüksekliği değiştirildiğinde, nesnenin yüksekliği onunla birlikte senkronize olarak değişir.
Left, Top, Right, Bottom	Konteynerin genişliği ve yüksekliği değiştirildiğinde, nesne onunla senkronize olarak büyür/küçülür.

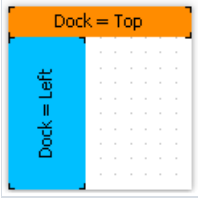
Dock özelliği

Her rapor nesnesinin bu özelliği vardır. Bu özellik, nesnenin kapsayıcısının hangi tarafına yerleştirileceğini belirler.

Dock özelliği aşağıdaki değerlerden birine sahip olabilir:

değer	açıklama
None	Varsayılan değer. Nesne uyumuyor.
Left	Nesne kapsayıcının sol kenarına yapışık. Aynı zamanda, nesnenin yüksekliği kapsayıcının yüksekliğine eşit olacaktır.*
Top	Nesne kapsayıcının üst kenarına yapışık. Aynı zamanda, nesnenin genişliği kapsayıcının genişliğine eşit olacaktır.*
Right	Nesne kapsayıcının sağ kenarına yapışık. Aynı zamanda, nesnenin yüksekliği kapsayıcının yüksekliğine eşit olacaktır.*
Bottom	Nesne kapsayıcının alt kenarına yapışık. Aynı zamanda, nesnenin genişliği kapsayıcının genişliğine eşit olacaktır.*
Fill	Nesne kapsayıcının tüm boş alanını kaplar.

* Bu tam olarak böyle değildir, eğer birkaç nesne aynı anda yerleştirilmişse. Aşağıdaki şekil, iki nesneyi göstermektedir; ilki, kapsayıcının üst kenarına yerleştirilmiş ve ikincisi ise sola:



Görüldüğü gibi, ikinci nesnenin yüksekliği, ilk nesnenin yerleştirilmesinden sonra kalan boş alanın yüksekliğine eşittir.

Yerleştirme davranışı, nesnenin oluşturulma sırasına bağlıdır. Sıralamayı, bir nesnenin bağlam menüsünden değiştirebilirsiniz. Bunu yapmak için, "Öne getir" veya "Sona gönder" menü öğelerinden birini seçin.

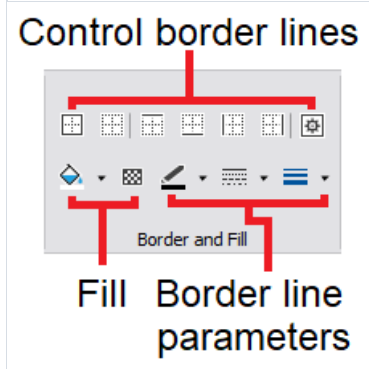
Biçimlendirme

Bu bölümde aşağıdaki konular ele alınmaktadır:

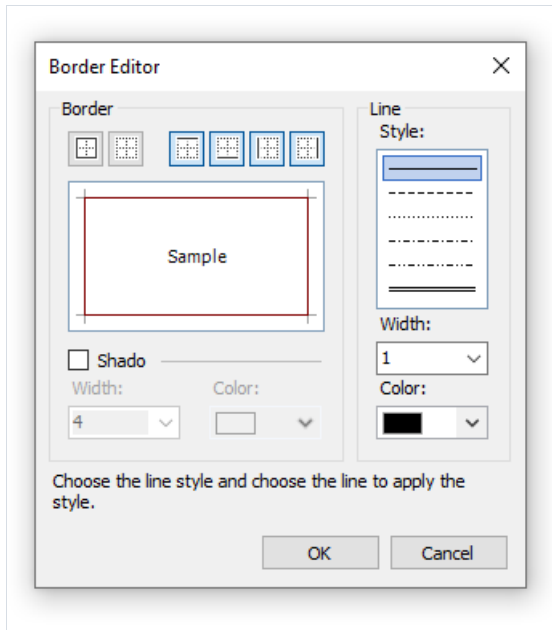
- Nesnenin görünüşünün değiştirilmesi;
- Yazdırılan değerlerin formatının değiştirilmesi;
- Belirli bir koşul sağlandığında nesnenin görünüşünün otomatik olarak değiştirilmesi;
- Gereksiz değerlerin gizlenmesi;
- Çift satırların farklı bir renkle vurgulanması.

Kenarlık ve dolgu

Neredeyse tüm rapor nesneleri kenarlık ve dolguya sahiptir. Bu özelliklerle çalışmak için "Kenarlık ve Dolgu" araç çubuğunu kullanın:

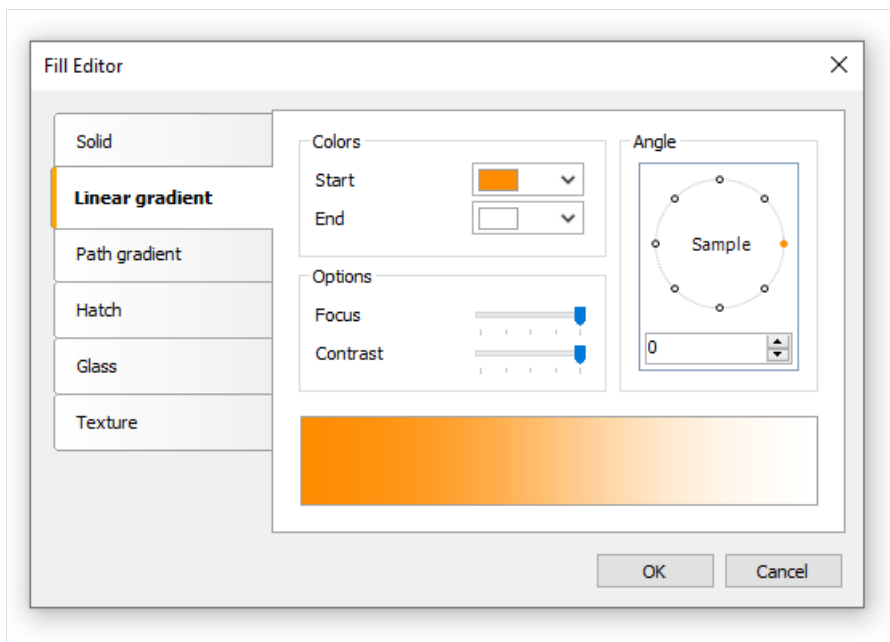


Nesnenin kenarlığı dört çizgiden oluşur. Her çizgi farklı genişlik, renk ve stile sahip olabilir. Araç çubuğundaki düğmeler, çerçevenin tüm çizgilerini etkiler. Bu düğme, her bir çizgiyi ayrı ayrı ayarlamaya olanak tanıyan bir iletişim penceresini görüntüler:



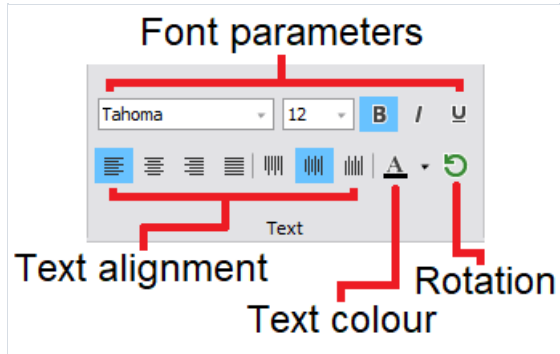
Dolgu ile çalışmak için araç çubuğunda iki düğme bulunur. **Düz dolgu türü** için bir renk seçmenizi sağlayan düğme:

Farklı dolgu türleri arasından seçim yapmanızı sağlayan bir iletişim penceresi açan düğme:



Metin formatı

'Metin' nesnesinin görünümünü değiştirmek için 'Metin' araç çubuğunu kullanın:

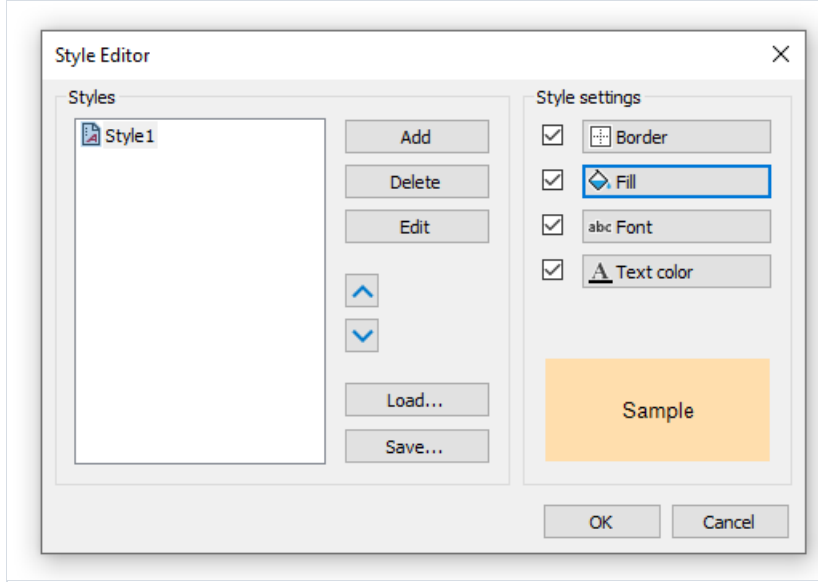


Stiller

Nesnelerin biçimlendirilmesinde stiller kullanılabilir. Stil, aşağıdaki özelliklerin bir kümesidir:

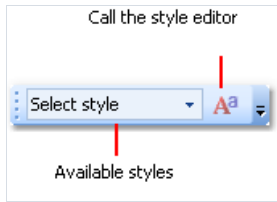
- Çerçeve;
- Dolgu;
- Yazı tipi;
- Metin rengi.

Stil listesi raporda yer alır. Rapor menüsünden "Rapor | Stiller..." seçeneği veya araç çubuğundaki "Stil" düğmesi ile yönetilebilir:



Nesneler için stil seçimi iki şekilde yapılabilir:

1. "Özellikler" penceresinde Style özelliğine değer atayarak;
2. "Stiller" araç çubuğunu kullanarak.



Eğer araç çubuğu ekranda görünmüyorsa, "Görünüm | Araç Çubukları | Stil" menüsünden gösterilebilir.

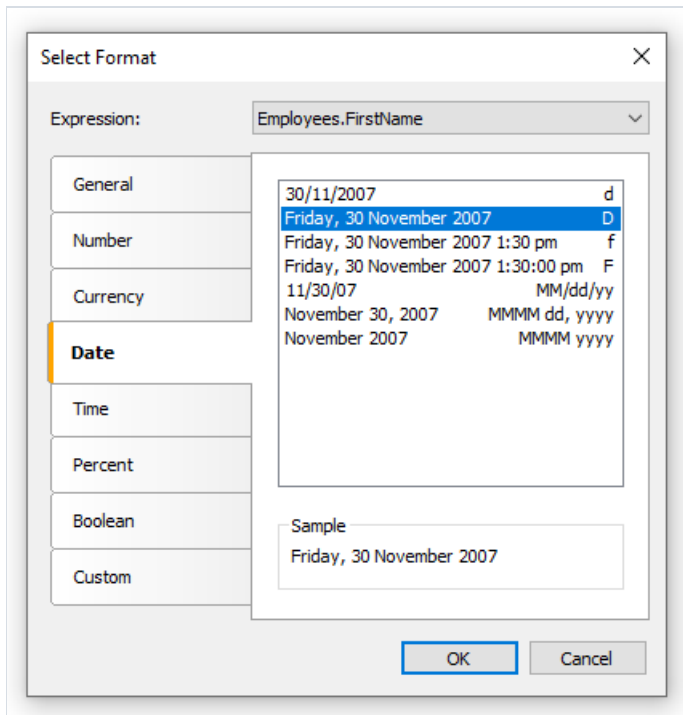
Bu durumda nesne, stilde belirtilen şekilde görünür. Stil parametreleri değiştirildiğinde, nesnenin görünümü otomatik olarak güncellenir.

Veri Formatı

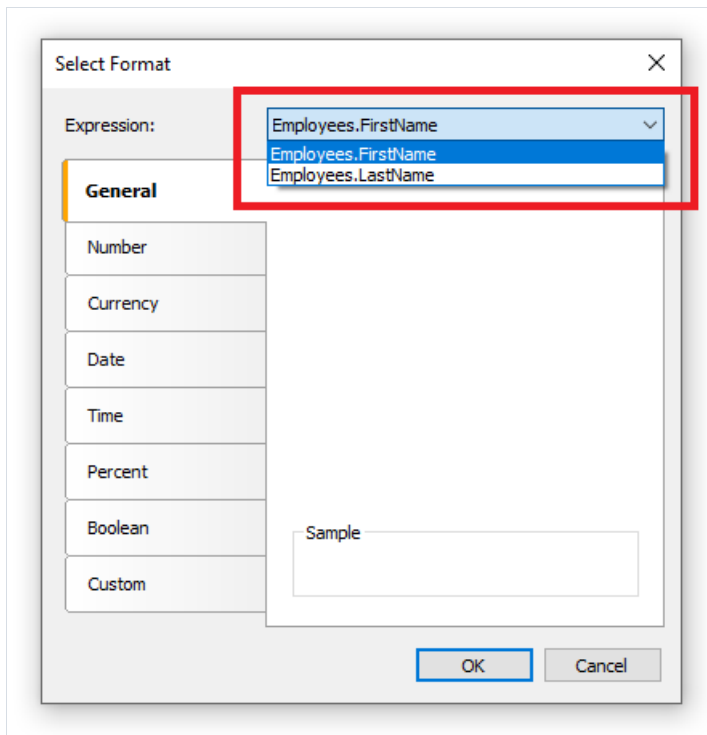
Raporunuzda metin bilgilerini yazdırmak için "Metin" nesnesi kullanılır. Varsayılan olarak, kaynak veride sunulduğu gibi yazdırılır. Örneğin, DateTime türündeki bir veri alanı; tarih bilgisine ek olarak saat bilgisini de içerir (bu, .NET'te DateTime türünün bir özelliğidir). Böyle bir alan raporda yazdırıldığında, aşağıdakine benzer bir çıktı elde edersiniz:

4/2/2024 6:04:52 PM

Eğer yalnızca tarihi yazdırmanız gerekiyorsa, veri formatlaması kullanabilirsiniz. Bunun için "Metin" nesnesini seçip sağ tıklayarak bağlam menüsünü açın. Menüden "Veri Biçimlendir..." seçeneğini seçtiğinizde aşağıdaki iletişim kutusunu göreceksiniz:



Diyalogda mevcut biçimlerden birini seçebilir veya "Özel" biçimi seçerek kendi biçim dizisini belirleyebilirsiniz. Eğer nesnenin metninde birden fazla ifade bulunuyorsa, her ifade için ayrı ayrı biçimlendirme ayarlayabilirsiniz. Bunun için pencerenin üst kısmında ilgili ifadeyi seçip biçimi belirleyin:

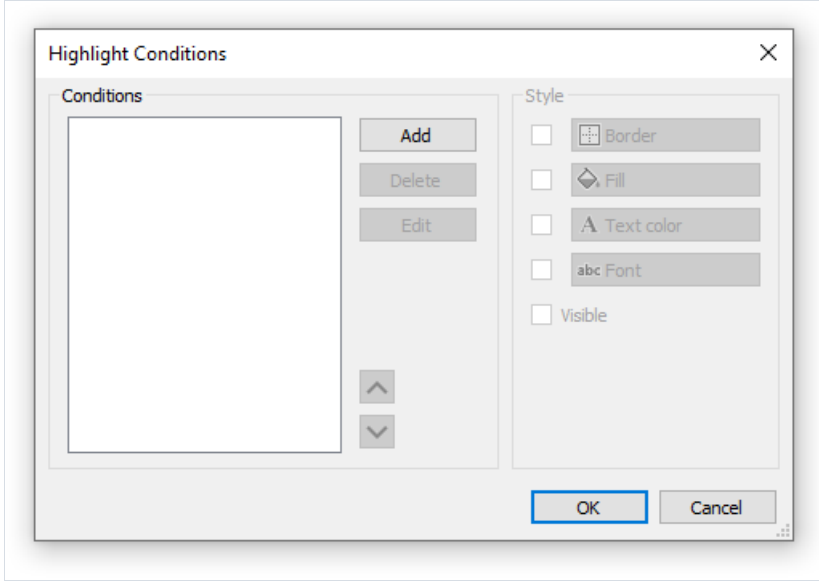


Ayrıca, veri formatlaması için sistem fonksiyonu `String.Format`'i de kullanabilirsiniz. Bu fonksiyonla ilgili ayrıntılı bilgiyi MSDN'in yardım sisteminde bulabilirsiniz.

```
Today is [String.Format("{0:d}", [Date])]
```

Koşullu Vurgulama

"Metin" nesnesinde, nesnenin görünümünü belirlenen koşullara bağlı olarak değiştirme imkânı sağlanmıştır. Örneğin, nesne negatif bir değer içeriyorsa kırmızı renkle vurgulanabilir. Bu imkân "koşullu vurgulama" olarak adlandırılır. Ayarlamak için nesneyi seçin ve "Metin" araç çubuğunda yer alan ilgili düğmeye basın. Aşağıdaki iletişim penceresi açılacaktır:

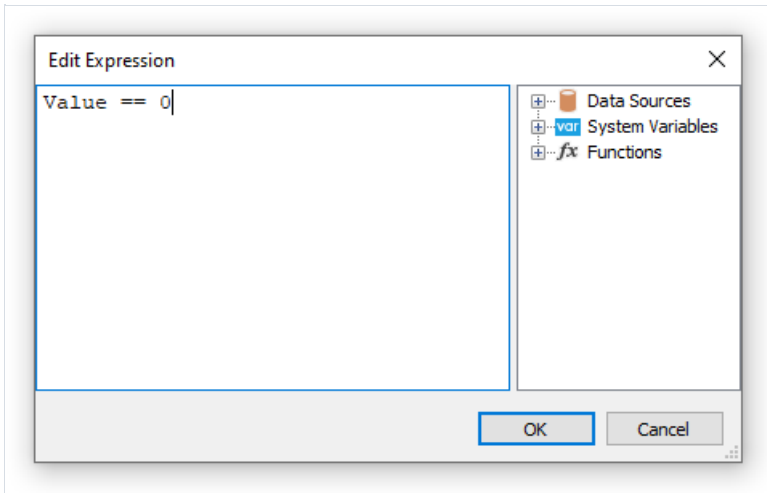


Burada bir veya daha fazla koşul tanımlayabilir ve her koşul için bir stil belirleyebilirsiniz. Stil, şu özelliklerden bir veya birkaçını içerebilir:

- Dolgu;
- Metin rengi;
- Yazı tipi ayarları;
- Nesnenin görünürlüğü.

Koşul tetiklendiğinde hangi özelliklerin değiştirileceğini belirtebilirsiniz. Bunun için pencerenin sağ kısmında yer alan onay kutularını kullanın. Varsayılan olarak, yeni stil yalnızca metin rengini ayarlar.

Yeni bir koşul oluşturmak için "Ekle" düğmesine basın:



Burada, doğru/yanlış (evet/hayır) sonucu döndüren herhangi bir ifadeyi yazabilirsiniz. Çoğu durumda, ifadede mevcut basılan değeri içeren ve Value değişkeni aracılığıyla erişilebilen bir ifade kullanılır.

Aşağıdaki örneği ele alalım: Stoktaki ürün miktarını gösteren bir "Metin" nesnesi var:

```
[Products.UnitsInStock]
```

Ürün adedi 0 olduğunda nesnenin kırmızı renkle vurgulanması gerekmektedir. Bunun için aşağıdaki koşulu oluşturun:

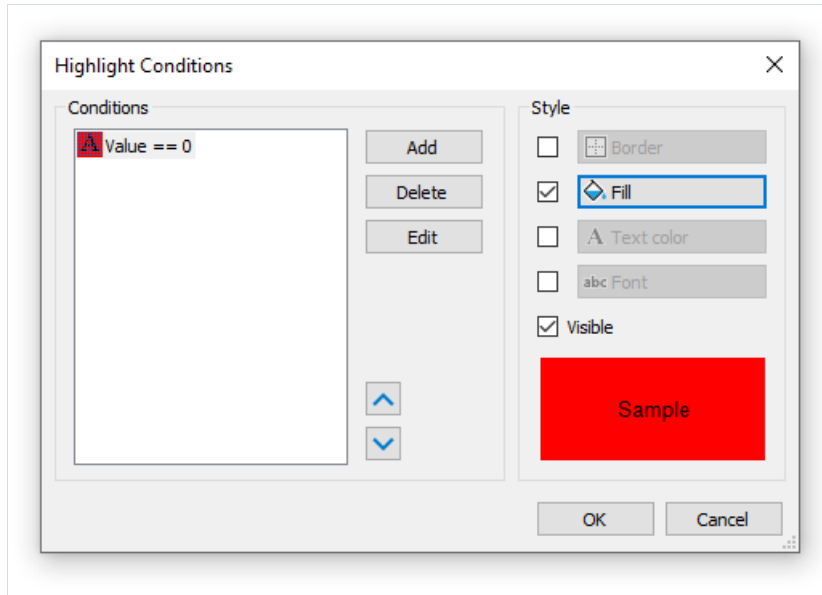
```
Value == 0
```

Bu örnekte, basılan değeri içeren `Value` değişkenini kullandık. Nesnede birden fazla ifade varsa, bu değişken en son ifadenin değerini tutacaktır. Value yerine, veri kaynağından gelen bir alan da kullanılabilir, örneğin:

```
[Products.UnitsInStock] == 0
```

Karşılaştırma işlemi C# stilinde yazılmıştır. Bu, rapor dili olarak C# seçildiyse doğrudur. VisualBasic.NET için karşılaştırma işlemi tek eşittir (=) ile yapılır. Rapor dilini "Rapor|Özellikler..." menüsünden değiştirebilirsiniz.

Bu koşul için yalnızca dolgu kullanılacak şekilde stili ayarlayalım ve dolgu rengi olarak kırmızıyı belirleyelim:



Baskıda, sıfır değeri içeren nesneler kırmızı olarak görünecektir. Şimdi örneğimizi biraz daha karmaşık hale getirip ek bir koşul ekleyelim. Ürün adedi 10'dan azsa, nesne sarı renkle basılmalıdır. Bunun için koşul düzenleyicisini açıp "Ekle" düğmesine basın. İkinci koşul şu şekilde olacaktır:

```
Value < 10
```

Birden fazla koşul belirtildiğinde, FastReport koşulları ilk koşuldan başlayarak kontrol eder. Bir koşul sağlandığında, FastReport nesneye ilgili stili uygular ve işlem sona erer. Bu yüzden koşulları doğru sırayla yerleştirmek önemlidir. İncelemekte olduğumuz örnekte doğru sıra şöyledir:

1. Value == 0
2. Value < 10

Eğer koşullar yer değiştirilirse, vurgulama yanlış çalışacaktır:

1. Value < 10
2. Value == 0

Bu durumda, sıfır değeri için ilk koşul sağlanacağından `Value == 0` koşulu uygulanmayacaktır. Koşulların sırasını değiştirmek için koşul düzenleyicisindeki yukarı/aşağı ok düğmelerini kullanın.

Sıfır Değerlerin Gizlenmesi

"Text" nesnesinin **HideZeros** özelliği, sıfır değerleri gizlemeyi sağlar. Aşağıdaki içerikle bir nesneye bakalım:

```
Total elements: [CountOfElements]
```

Eğer **CountOfElements** değişkeninin değeri 0 ise ve **HideZeros** özelliği true olarak ayarlanmışsa, nesne şu şekilde yazdırılır:

```
Total elements:
```

"Text" nesnesinin ayrıca, belirli bir değere eşit olan ifade değerlerini gizlemeye yarayan string tipinde **HideValue** özelliği bulunmaktadır. Örneğin, eğer bu özelliğin değeri 0 ise, tüm sıfır alanlar gizlenecektir. Bu özellik, "sıfır" tarihleri gizlemek için de kullanılabilir. Genellikle, bu tarihler 1.1.0001 veya 1.1.1900'dir. Bu durumda **HideValue** özelliğinin değeri şu şekilde olmalıdır:

```
1/1/1900 0:00:00 AM
```

Görüldüğü gibi, tarih ile birlikte saat de yazılmalıdır. Bunun nedeni, .NET'te tarih değerinin aynı zamanda zamanı da içermesidir.

Önemli not: Bu mekanizma, Denetim Masası'nda ayarlanan bölgesel ayarlara bağlıdır. Çünkü FastReport, ifadelerin değerlerini karşılaştırmak için ToString() metodunu kullanır. Bu metod, ifadelerin değerini mevcut bölgesel ayarları kullanarak dizeye dönüştürür. Bu nedenle, farklı bölgesel ayarlara sahip bilgisayarlarda çalıştırılacak raporlar geliştirirken dikkatli olunmalıdır.

Son olarak, "Text" nesnesinin **NullValue** özelliği, boş (null) değerler yerine belirli bir metin görüntülemeyi sağlar. Bu, genellikle boş değer yerine kısa çizgi yazdırmak için kullanılır. Aşağıdaki içerikle bir nesneye bakalım:

```
Total elements: [CountOfElements]
```

Eğer **CountOfElements** değişkeninin değeri null ise ve **NullValue** özelliği -- olarak ayarlanmışsa, nesne şu şekilde yazdırılır:

```
Total elements: --
```

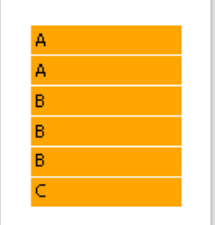
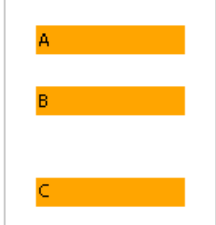
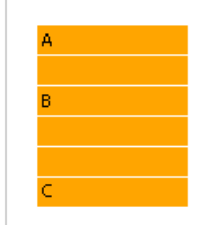
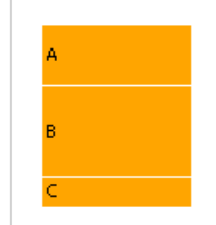
Tekrarlayan Değerleri Gizleme

"Metin" nesnesinin, tekrarlayan değerleri gizlemeye yarayan **Duplicates** adlı bir özelliği vardır. Bu özellik yalnızca "Veriler" bandında bulunan "Metin" nesneleri için geçerlidir. Tekrarlayan değerler, ardışık veri satırlarında yazılmış aynı değerler olarak kabul edilir.

Özellik aşağıdaki değerlerden birini alabilir:

- **göstermek (Show)** – Tekrarlayan değerleri gösterir (varsayılan değer);
- **gizlemek (Hide)** – Tekrarlayan değere sahip nesneyi gizler;
- **temizlemek (Clear)** – Nesnedeki metni temizler, ancak nesnenin kendisini gösterir;
- **birleştirmek (Merge)** – Aynı değerlere sahip nesneleri birleştirir.

Şekilde, modlar arasındaki farklar gösterilmiştir:

Duplicates = Show	Duplicates = Hide	Duplicates = Clear	Duplicates = Merge
			

Veri Satırlarının Aralıklı Vurgulanması

Raporun görünümünü iyileştirmek için veri satırlarının çift numaralı olanlarını farklı bir renkle vurgulayabilirsiniz. Bunu, bantın kendisinde veya bant üzerinde yer alan nesnelerde bulunan **EvenStyle** özelliğini kullanarak gerçekleştirebilirsiniz. Bu özelliğe, çift numaralı satırlar için kullanılacak stilin adını atarsınız.

Bant üzerindeki nesnelerde **EvenStyle** özelliğini kullanmak tercih edilir; böylece rapor dışı aktarılırken olası sorunların önüne geçilmiş olur.

Satır vurgulamasını ayarlamak için şu adımları izleyin:

1. Stili Tanımlayın: Vurgulamada kullanılacak stili belirleyin. Bu işlemi "Rapor | Stiller..." menüsü üzerinden yapabilirsiniz. Genellikle vurgulama için soluk gri dolgu rengi tercih edilir.
2. Stil Adını Atayın: Tanımladığınız stilin adını, bant veya bant üzerindeki nesnelerin **EvenStyle** özelliğine girin.

Varsayılan olarak nesneler, yalnızca **EvenStyle** ile belirlenen stilin dolgu rengini kullanır. Bu davranış, **EvenStylePriority** özelliği ile kontrol edilir; varsayılan değeri **UseFill** 'dir. Eğer stilin diğer parametrelerine de ihtiyaç duyuyorsanız, **EvenStylePriority** özelliğini **UseAll** olarak ayarlayın.

Bu teknik kullanılarak oluşturulmuş nihai rapor aşağıdaki gibi görünebilir:

Product name	Unit price
Chai	18,00
Chang	19,00
Chartreuse verte	18,00
Côte de Blaye	263,50
Guaraná Fantástica	4,50
Iphoh Coffee	46,00

Tek bantlı "Veri" raporu

Bu tip rapor en çok talep edilen rapor türüdür. Rapor, veri kaynağındaki satır listesini yazdırmaya olanak tanır. Örneğin, bu bir müşteri listesi olabilir.

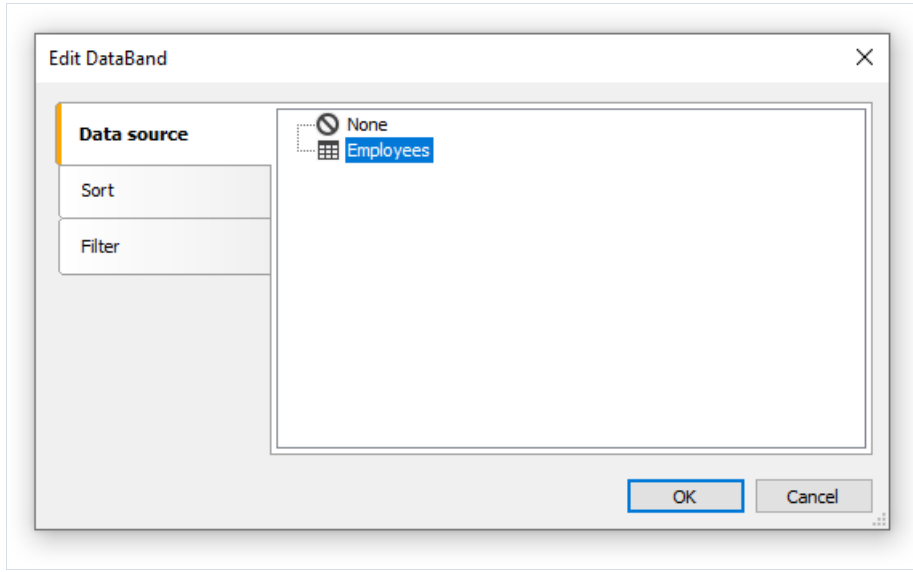
Bantın Veriye Bağlanması

Raporunuzda verileri liste şeklinde yazdırmak için, veri kaynağına bağlanması gereken **"Veriler"** bantına ihtiyacınız olacaktır. Bant, veri kaynağındaki her satır için bir kez basılır.

Eğer **"Veriler"** bandı veri kaynağına bağlanmamışsa, yalnızca bir kez basılır.

Yeni bir rapor oluşturduğunuzda, rapor zaten **"Veriler"** bandı dahil olmak üzere birkaç boş bant içerir. Bu bant, ayrıca **"Rapor | Bant Ayarları..."** menü seçeneğini seçerek de eklenebilir.

Bantı veriye bağlamak için, bant üzerinde çift tıklayın. Açılan düzenleyici penceresinde veri kaynağını seçin ve **"Tamam"** butonuna tıklayın.



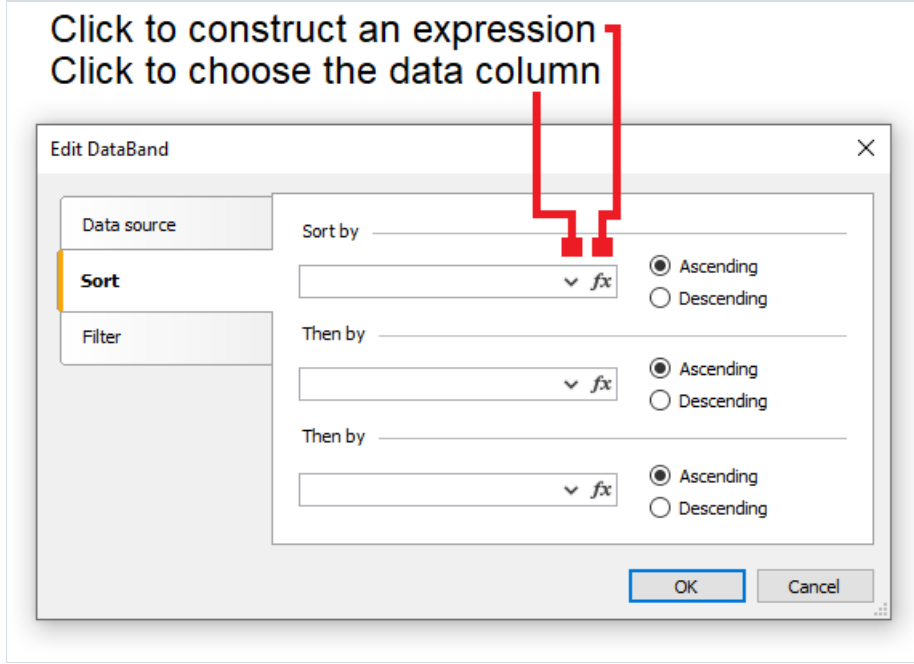
Metin Verilerini Yazdırma

Bant veriye bağlandıktan sonra, üzerinde veri alanlarındaki bilgileri gösterecek "Metin" nesnelerini yerleştirebilirsiniz. Bunu yapmanın en hızlı yolu, "Veriler" penceresindeki alanları banta sürükleyip bırakmaktır. "Metin" nesnesinin tüm olanakları hakkında daha detaylı bilgi için "Metin Nesnesi" başlıklı bölümü inceleyin.

Veri Sıralaması

Varsayılan olarak, "Veri" bandı verileri doğal sıralamayla yazdırır. Ancak, verilerin yazdırılmadan önce sıralanması gerekebilir. Örneğin, müşteri listesi alfabetik sıraya göre sıralanarak daha düzenli bir şekilde sunulabilir.

Sıralamayı "Veri" bandı düzenleyicisinden kontrol edebilirsiniz. Düzenleyiciyi açmak için, band üzerindeki boş bir alana çift tıklayın:

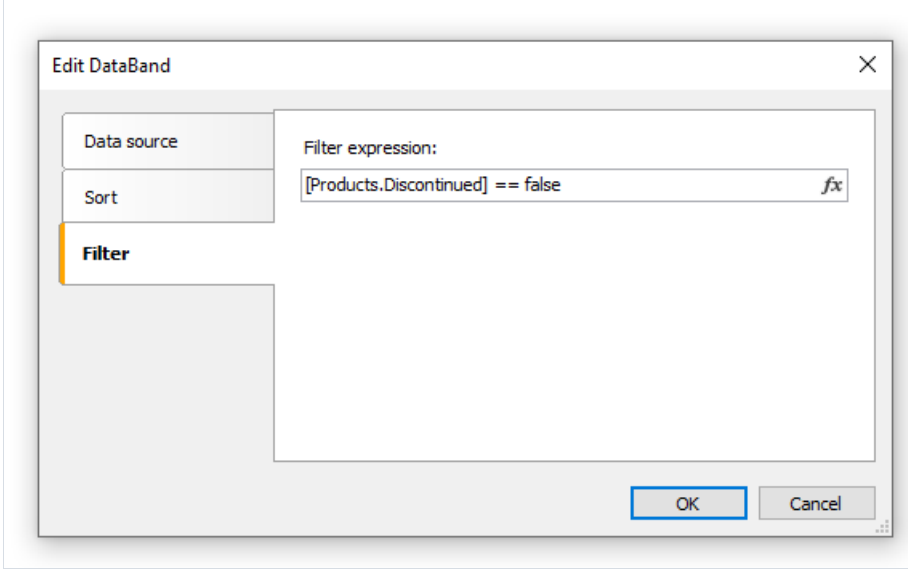


Sıralama kriteri olarak, bir veri sütunu veya bir ifade kullanabilirsiniz. Birden fazla (üçten fazla olmamak kaydıyla) sıralama koşulu belirtebilirsiniz. Örneğin, müşterilerin listesini şehirlerine göre sıralamak ve ardından müşterinin adına göre sıralamak isteyebilirsiniz. Her bir koşul için sıralama düzenini - artan veya azalan - seçebilirsiniz.

Veri sıralamanın bir diğer yöntemi, SQL sorgusunu veri kaynağı olarak kullanmaktır. Sorgu, veri sunucusunda çalıştırılacak ve sıralanmış satırları döndürecektir.

Veri Filtreleme

Veri bandında yazdırılan satırları filtrelemek için, düzenleyicisini açın ve "Filtre" sekmesine geçin:



Filtre ifadesi olarak, FastReport açısından geçerli herhangi bir ifadeyi belirtebilirsiniz. İfadeler hakkında daha ayrıntılı bilgi için "İfadeler" bölümüne bakınız.

Yukarıdaki örnekte şu filtre kullanılmıştır:

```
[Products.Discontinued] == false
```

Bu, **Discontinued** bayrağı **false** olan tüm veri satırlarının seçileceği anlamına gelir.

Başka bir karmaşık ifade içeren filtre örneği:

```
[Products.Discontinued] == false && [Products.UnitPrice] < 10
```

Bu, **Discontinued** bayrağı **false** olan ve fiyatı 10'dan küçük olan tüm veri satırlarının seçileceği anlamına gelir. \

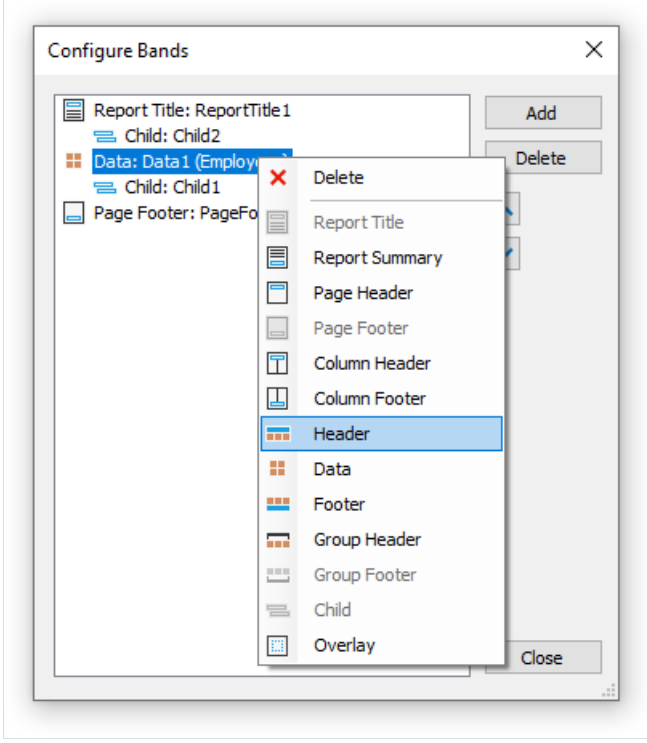
Bu filtreleme yöntemi, veri kaynağının tüm satırları içerdiğini ve bunlardan bir kısmının filtreleneceğini varsayar. Eğer veri kaynağı çok sayıda satır içeriyorsa, bu raporun performansını ciddi şekilde yavaşlatabilir. Bu durumda, gerekli filtreleme koşulunu içeren bir SQL sorgusunu veri kaynağı olarak kullanabilirsiniz. Sorgu, veri sunucusunda çalıştırılacak ve yalnızca raporda ihtiyaç duyulan satırları döndürecektir.

Veri filtrelemesi için raporda diyalog formları da kullanılabilir. Daha fazla bilgi için "Diyalog Formları" bölümüne bakınız.

Veri Başlığı ve Altbilgisi

Bir "Veri" bandı, bir başlık ve bir altbilgi içerebilir. Başlık, veriden önce bir kez basılacak, altbilgi ise tüm verilerin çıktıdan sonra basılacaktır.

Bir "Veri" bandına başlık ve altbilgi eklemek için "Rapor|Bantları Yapılandır..." menü öğesini seçin. Açılan pencerede "Veri" bandını seçin ve sağ tıklayın. Açılan bağlam menüsünde "Başlık" ve/veya "Altbilgi" öğelerini seçin:



Bu bantlar aşağıdaki durumlarda faydalı olabilir:

Tek bir sayfada birden fazla liste yazdırılırken ("[ana-ana](#)" raporları). Bu durumda her veri bandının kendi başlığı ve altbilgisi olabilir.

Data Header 1	Data 2
Data 1	Data 2
Data 1	Data 2
Data 1	Data Footer 2
Data 1	
Data Footer 1	
Data Header 2	
Data 2	
Data 2	

Hazırlanan raporun tek bir sayfasına sığmayan bir liste yazdırılırken. Veri başlığı/altbilgisinin "Her Sayfada Tekrarla" özelliğini kullanarak, bu bantları raporun her sayfasında yazdırabilirsiniz.

Data Header	Data Header
Data	Data
Data	Data
Data	Data Footer
Data	
Data	
Data	
Data	
Data Footer	

Veri Kesme ve Bir Arada Tutma

Bu bölümde, FastReport'ta bantların yazdırılması için iki mod ele alınacaktır: "**Kesme**" ve "**Bir Arada Tutma**".

Normal bant yazdırma modunda, FastReport mevcut sayfada yeterli alan olup olmadığını kontrol eder. Eğer yeterli alan yoksa, bant tamamen sonraki sayfada yazdırılır. Bantın "Bölünebilir" (Может разрываться) özelliği etkinleştirilmişse, FastReport mevcut alanda bantın bir kısmını yazdırmayı dener; yani bant "kesilir".

Bantın kesilebilme denemesi başarılı olabileceği gibi başarısız da olabilir. Bu durum, bant üzerinde yer alan nesneler ve onların ayarlarına bağlıdır. Aşağıdaki nesneler kesilebilir:

- **Metin**
- **Biçimlendirilmiş Metin**
- **Tablo**

Bu nesnelerin de "Bölünebilir" özelliği bulunmaktadır. Eğer bu özellik etkinse, nesne kesilebilir. Kesilemeyen bir nesne ise, kendisine yeterli alan bulunduğu durumda her zaman tamamen yazdırılır.

Aşağıdaki çizimde, bir bantın nasıl kesilebileceğine dair örnek gösterilmiştir.



Kesme algoritması her zaman düzgün çalışmayabilir. Özellikle aynı anda, farklı yazı tipi boyutlarına veya karmaşık çerçeveselendirmeye sahip birden fazla nesnenin kesilmesi durumunda, bazı görsel hatalar (artefaktlar) ortaya çıkabilir.

Veri Kesmenin Amacı Veri kesmenin temel amacı, sayfadaki mevcut alanı maksimum derecede kullanarak rapor yazdırılırken kağıt tasarrufu sağlamaktır.

Bir Arada Tutmanın Amacı Bir arada tutmanın hedefi ise, belirli bir bant grubunun tamamının tek bir sayfada yer almasını sağlamaktır. Bu durumda rapor sayfalarında kullanılmayan boş alanlar fazla olsa da, veriler daha okunabilir bir biçimde yazdırılır.

Bir Arada Tutma Mekanizması (KeepTogether) Bu mekanizma, belirli bir bant grubunun (veya sütun bazında, eğer rapor çok sütunluyorsa) aynı sayfada tutulmasını sağlar. Eğer bir arada tutulan veri yazdırılırken sayfanın sonuna ulaşırsa, FastReport, daha önce yazdırılan tüm verileri yeni sayfaya taşır.

Bir Arada Tutmanın Kullanım Durumları:

- "Veri" bantındaki tüm satırların birlikte yazdırılması;
- Grup elemanlarının (başlık, veri, altbilgi) birlikte yazdırılması;
- Master-detail tipindeki raporlarda, ana veri kaynağına ait satırın alt veri kaynağına ait tüm satırlarla birlikte yazdırılması;
- Rapor başlığının veya veri başlığının en az bir veri satırı ile birlikte yazdırılması;
- Rapor altbilgisinin veya veri altbilgisinin en az bir veri satırı ile birlikte yazdırılması;
- Ana bant ile alt bantın birlikte yazdırılması.

"Birlikte tut" mekanizmasının kullanımına bakalım.

Tüm veri satırlarını veya grup öğelerini (başlık, veri, altbilgi) birlikte tutmak için `KeepTogether` (BirlikteTut) özelliğini etkinleştirin. Bu özellik "Veri" ve "Grup Başlığı" bantlarında kullanılır. Aşağıdaki görsel, verilerin birlikte tutularak ve tutulmadan nasıl yazdırıldığını göstermektedir.



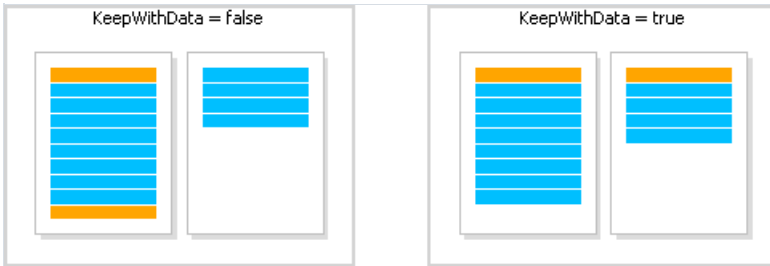
Ana veri satırını detay veri satırlarıyla birlikte tutmak için "Veri" bandının `KeepDetail` (DetaylaBirlikteTut) özelliğini etkinleştirin. Bu özellik, "ana-detay" türündeki raporlarda kullanılır.



"Bağımsız kalan" başlık ve altbilgileri önlemek için `KeepWithData` (VeriyleBirlikteTut) özelliğini kullanın. Aşağıdaki bantlar bu özelliğe sahiptir:

- report header;
- report footer;
- data header;
- data footer;
- group header;
- group footer.

Bu özellik, başlık/altbilginin en az bir veri satırıyla birlikte tutulmasına olanak tanır.

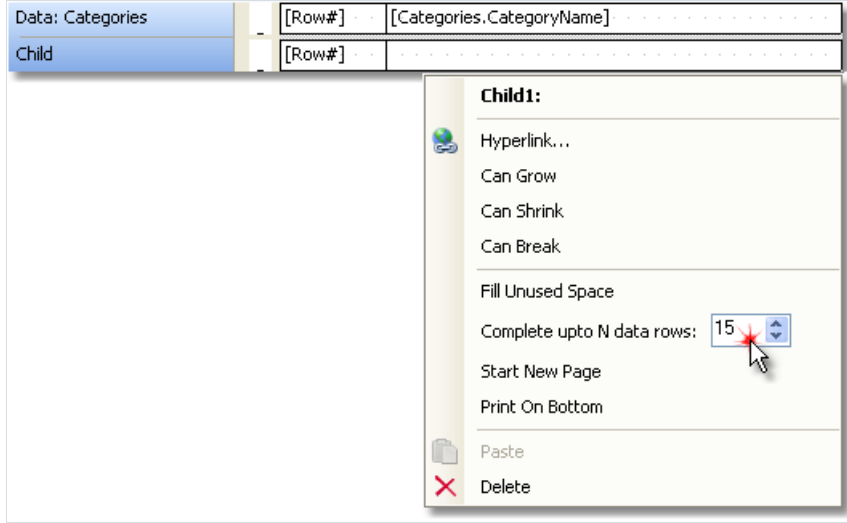


Bir bandı ve ona bağlı alt bandı birlikte tutmak için `KeepChild` (AltBandıBirlikteTut) özelliğini etkinleştirin.

Veri Satırlarının Boş Yazdırılması

Baskı formlarında belirli sayıda veri satırının yazdırılması sıklıkla gerekmektedir. Eğer gerçek veri sayısı istenen satır sayısından az ise, boş satırlar yazdırılmalıdır. Bu durum, "Alt bant"ın "Verileri N Satıra Tamamla" (CompleteToNRows) özelliği kullanılarak ve bu bantın "Veri bantı"na eklenmesiyle sağlanır.

"Alt bant"ın "Verileri N Satıra Tamamla" özelliğine 0'dan büyük bir değer girilirse, bant belirtilen satır sayısına ulaşmak için eksik satırları tamamlar. Örneğin, 15 veri satırı yazdırılması gerekiyorsa fakat veri kaynağında yalnızca 8 satır mevcutsa, bu durumda "Alt bant" 7 kere yazdırılarak eksik satırlar tamamlanır.



Hazır rapor aşağıdaki gibi görünecektir:

1	Beverages
2	Condiments
3	Confections
4	Dairy Products
5	Grains/Cereals
6	Meat/Poultry
7	Produce
8	Seafood
9	
10	
11	
12	
13	
14	
15	

Veri kaynağında, **CompleteToNRows** (NSatıraTamamla) özelliğinde belirtilenden daha fazla satır varsa, boş bir satır yazdırılmayacaktır.

Boş bir satır yazdırmanın bir diğer yolu, sayfadaki boş alanı doldurmaktır. Bu durumda, "Child" (AltBand) bantı, "Veri Altbilgisi" veya "Grup Altbilgisi" türündeki bantlara bağlanır ve sayfadaki boş alanı doldurur. Altbilgi bandı, sayfanın alt kısmında yazdırılacaktır.

Boş bir satır bu şekilde yazdırmak için "Child" (AltBand) bantını altbilgi bandına bağlayın ve **FillUnusedSpace** (KullanılmayanAlanıDoldur) özelliğini etkinleştirin. Şimdi, altbandın bağlı olduğu bandın üzerinde görüntülendiğini göreceksiniz. Aşağıdaki görselde, "Child" bandı "Rapor Özeti" bandına bağlanmıştır.

Data: Categories	[Categories.CategoryName]
Child	
Report Summary	report summary

Bu şekilde bir raporu çalıştırdığımızda, aşağıdaki gibi görüntü elde edeceğiz:

Beverages
Condiments
Confections
Dairy Products
Grains/Cereals
Meat/Poultry
Produce
Seafood
report summary

'No data' Metnini Yazdırma

Eğer "Veriler" bandı boş bir veri kaynağına bağlıysa, yazdırılmaz. Böyle bir durumda bazen "Veri Yok" ifadesinin yazdırılması gerekebilir. Bunun için şu adımları izleyin:

- "Veriler" bandına bir alt bant ekleyin.
- Alt bant için `PrintIfDataBandEmpty` özelliğini etkinleştirin (bu, "Özellikler" penceresinden yapılabilir).
- Alt banda, "Metin" nesnesini ekleyin ve içine "Gösterilecek veri yok" metnini yerleştirin.

Page Header	-	CompanyName	Country
Data: Customers	-	[Customers.CompanyName]	[Customers.Country]
Child	-	No data to display	

Rapor aşağıdaki şekilde oluşturulacaktır:

- Eğer veri kaynağında veri varsa, "Veriler" bandı tüm ilgili bantlarla (veri başlığı, alt bilgi) birlikte yazdırılacaktır.
- Eğer veri kaynağı boşsa, yalnızca "Gösterilecek veri yok" metnini içeren alt bant yazdırılacaktır.

Hiyerarşi Yazdırma

"Veri" bandı, hiyerarşik bir liste yazdırmaya olanak tanır. Bunun için tek bir bant ve tek bir veri kaynağı kullanılır. Hiyerarşi, veri kaynağında iki alan yardımıyla tanımlanmalıdır:

1. **Anahtar Alan:** Veri satırının kimlik değerini içerir.
2. **Ebeveyn Anahtar Alanı:** Veri satırının bağlı olduğu (ebeveyn) satırın anahtarını içerir.

Böyle bir veri kaynağını hiyerarşik olarak yazdırmak için "Veri" bandının aşağıdaki özelliklerini yapılandırmanız gerekmektedir. Bu ayarları "Özellikler" penceresinde yapabilirsiniz:

☒ Hierarchy	
IdColumn	
Indent	1 cm
ParentIdColumn	

- **IdColumn:** Anahtar alan burada belirtilir.
- **ParentIdColumn:** Ebeveyn değerini içeren alan burada belirtilir.
- **Indent:** Girinti değeri burada tanımlanır.

Örneğin, Employees tablosundan çalışanların hiyerarşisini yazdırmayı ele alalım. Bu tabloda ihtiyacımız olan iki alan bulunmaktadır:

- **EmployeeID:** Anahtar alandır ve çalışanın numarasını içerir.
- **ReportsTo:** Çalışanın bağlı olduğu çalışanın numarasını içerir.

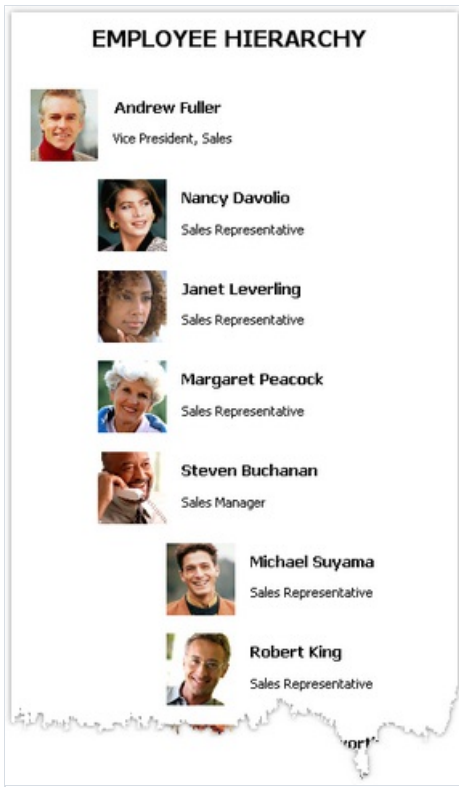
Aşağıdaki gibi bir rapor oluşturabiliriz:

Report Title	EMPLOYEE HIERARCHY
Data: Employees	[Employees.FirstName] [Employees.LastName] [Employees.Title]

Aşağıdaki şekilde, hiyerarşiden sorumlu olan "Veri" bandı özelliklerini ayarlayın:

☒ Hierarchy	
IdColumn	Employees.EmployeeID
Indent	1,5 cm
ParentIdColumn	Employees.ReportsTo

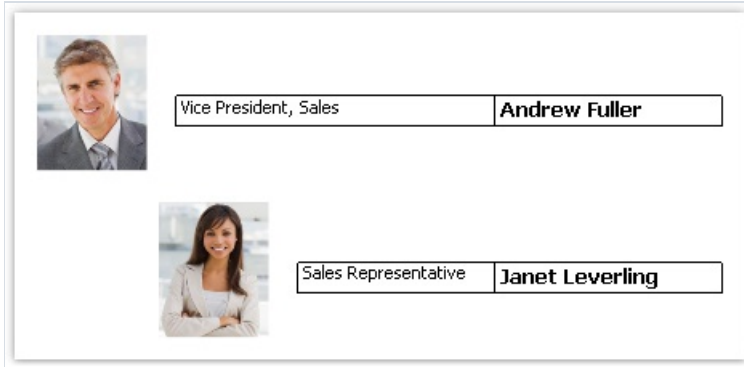
Bir raporu çalıştırdığımızda, aşağıdaki gibi bir görüntü elde edeceğiz:



Hiyerarşi yazdırıldığında, FastReport bandı sağa kaydırır (Indent (Girinti) özelliğinde belirtilen bir değere göre) ve aynı değeri kullanarak bandın genişliğini de küçültür. Bu, bandın nesnelerinin Anchor (Çapa) özelliğini kullanmanıza olanak tanır. Bu durumda kullanılacak bu özelliğin olası değerleri şunlardır:

- Sol, Üst (varsayılan) - nesne, bandla birlikte hareket eder;
- Sağ, Üst - nesne, orijinal pozisyonunda kalır;
- Sol, Sağ, Üst - nesnenin sağ tarafı orijinal pozisyonunda kalır, sol tarafı ise bandla birlikte hareket eder.

Bu, bazı faydalı efektler elde etmenizi sağlar:



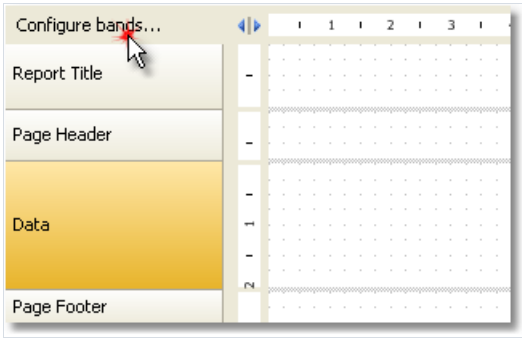
Bu örnekte, resim nesnesinin **Anchor** (Çapa) özelliği Sol, Üst olarak ayarlanmıştır; iş unvanını içeren nesne Sol, Sağ, Üst olarak çapa yapılmıştır; isim içeren nesne ise Sağ, Üst olarak çapa yapılmıştır.

Ana-Detay Raporu (Master-detail)

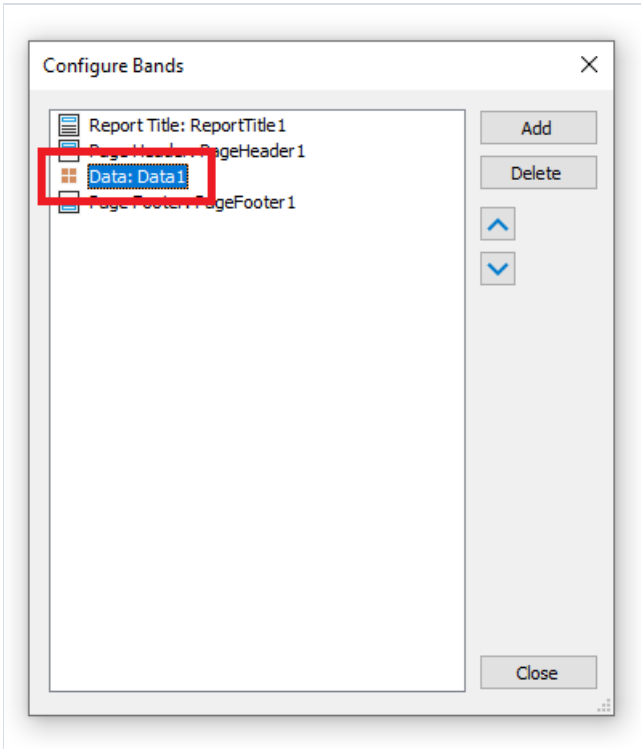
İki "Veri" bandı kullanarak, ana-detay (master-detail) tipinde bir rapor oluşturmak oldukça kolaydır. Bu raporda, birbirine bağlı olan iki veri kaynağı kullanılmaktadır; ana veri kaynağındaki bir satıra, detay veri kaynağından birden fazla satır karşılık gelebilir. Bağlantılar hakkında daha fazla bilgi için "Veri" bölümüne bakınız.

Bantların raporda, ana bandın içinde detay bandını barındıracak şekilde yerleştirilmesi gerekir. Bu işlem, "Bant Ayarları" penceresinde yapılır; ilgili pencereye "Rapor|Bant Ayarları..." menüsünden ulaşabilirsiniz.

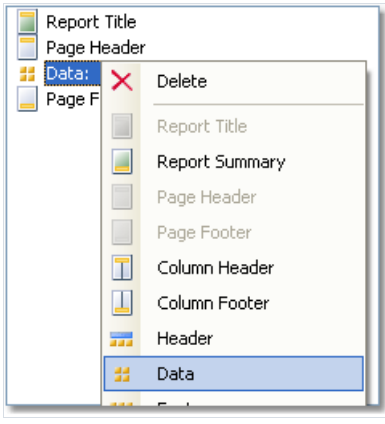
Sıfırdan bir master-detail raporu oluşturalım. Bunun için rapor tasarımcısını çalıştırıp yeni, boş bir rapor oluşturun. Oluşturulan rapor zaten bir "Veri" bandı içermektedir:



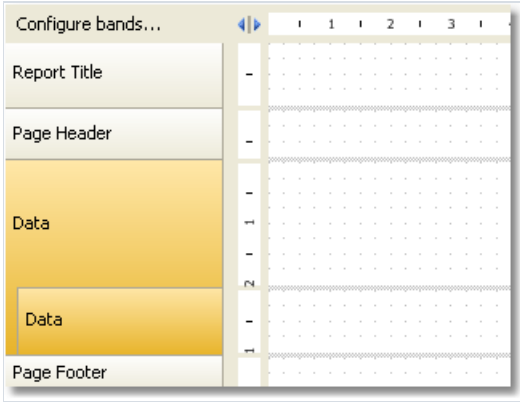
Detay "Veri" bandını eklemek için "Bant Ayarları" penceresini açın. Bunu, resimde gösterilen "Bant Ayarları..." butonuna tıklayarak veya "Rapor|Bant Ayarları..." menü seçeneğini seçerek yapabilirsiniz. Ayarlar penceresinde mevcut bant yapısı görüntülenecektir:



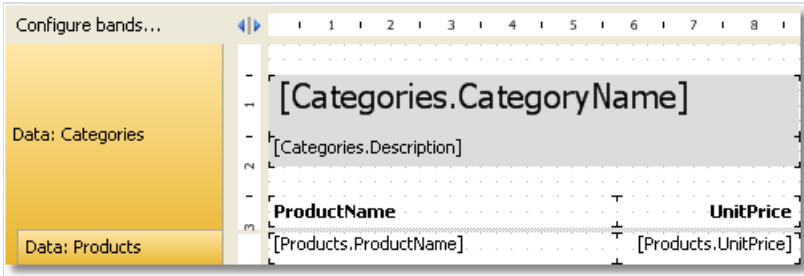
Resimde gösterildiği gibi "Veri" bandını seçin ve sağ tıklayarak bağlam menüsünü açın (veya pencerenin alt kısmındaki "Ekle" butonuna basın). Açılan menüden "Veri" bandını seçin:



Bunun sonucunda, seçilen banda iç içe geçmiş bir "Veri" bandı eklenecektir. Pencereyi "Kapat" butonuna tıklayarak kapatın. Rapor şablonunun aşağıdaki şekilde değiştiğini göreceksiniz:



Veri bantlarının iç içe yapısı, pencerenin sol tarafındaki bant yapısında net bir şekilde görülmektedir. Bundan sonra, bantları ilgili veri kaynaklarına bağlamalı ve veri alanlarını bantlara yerleştirmelisiniz. FastReport ile birlikte gelen demo veritabanındaki Categories (Kategoriler) ve Products (Ürünler) veri kaynaklarını kullanacağız:



Raporu çalıştırdığınızda aşağıdakini göreceksiniz:

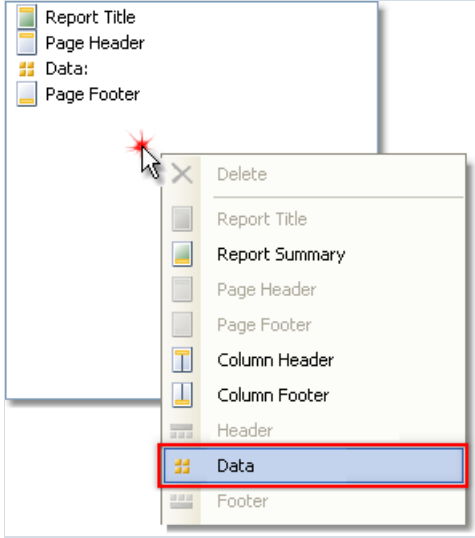
Beverages	
Soft drinks, coffees, teas, beers, and ales	
ProductName	UnitPrice
Chai	18,00p.
Chang	19,00p.
Guaraná Fantástica	4,50p.
Sasquatch Ale	14,00p.
Steeleye Stout	18,00p.
Côte de Blaye	263,50p.
Chartreuse verte	18,00p.
Ippoh Coffee	46,00p.
Laughing Lumberjack Lager	14,00p.
Outback Lager	15,00p.
Rhönbräu Klosterbier	7,75p.
Lakkaikööri	18,00p.
Condiments	
Sweet and savory sauces, relishes, spreads, and seasonings	
ProductName	UnitPrice
Aniseed Syrup	10,00p.
Chef Anton's Cajun Seasoning	22,00p.
Chef Anton's Gumbo Mix	21,35p.

Bu şekilde, örneğin master-detail-subdetail gibi, sınırsız veri iç içe geçirme ile master-detail raporları oluşturabilirsiniz. Master-detail raporlarının oluşturulmasında kullanılan bir diğer yöntem ise iç içe raporların kullanılmasıdır. İç içe raporlar daha sonra ele alınacaktır.

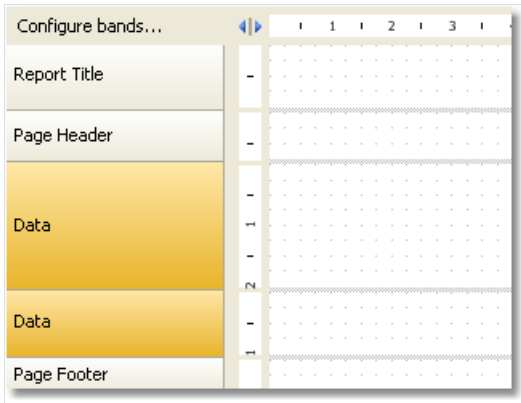
Master-master tipi rapor

Rapor sayfasında birden fazla basit liste yazdırabilirsiniz. Bunu, sayfaya iki veya daha fazla "Veri" bandı yerleştirerek yapabilirsiniz. Master-detail raporda bantlar birbirinin içine yerleştirilmiş olup ilişkili veri kaynaklarından veri yazdırırken, bu tip raporda (buna master-master diyelim) hem bantlar hem de veri kaynakları birbirinden bağımsızdır.

Örneğin, sayfada iki liste yazdıran bir rapor oluşturmayı göstereceğiz – kategoriler listesi (Categories tablosu) ve müşteriler listesi (Customers tablosu). Yeni bir rapor oluşturun ve gerekli veri kaynaklarını ekleyin. İkinci "Veri" bandını eklemek için "Bant Ayarları" penceresini açın.



Listenin boş bir yerine sağ tıklayın (resimde gösterildiği gibi) ve açılan bağlam menüsünden "Veri" bandını seçin. Bu, ikinci bağımsız "Veri" bandını oluşturacaktır. Rapor şablonu şu şekilde görünecektir:



Şimdi bantları veri kaynaklarına bağlayıp üzerine birkaç veri alanı yerleştirelim:



Raporu çalıştırırsanız, aşağıdaki görüntüyü göreceksiniz:

Beverages

Condiments

Confections

Dairy Products

Grains/Cereals

Meat/Poultry

Produce

Seafood

Alfreds Futterkiste

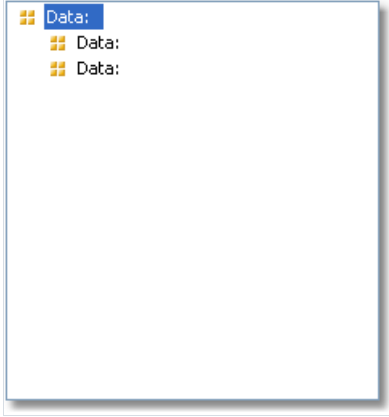
Ana Trujillo Emparedados y helados

Antonio Moreno Taquería

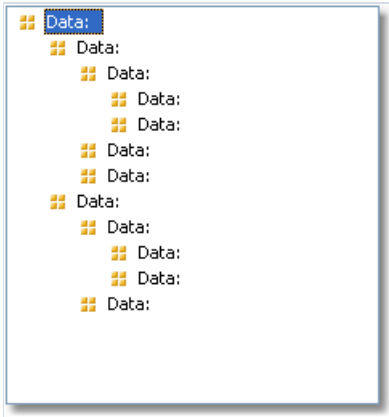
Around the Horn

Master-detail-detail tipi rapor

"Veri" bandı, bir veya daha fazla iç içe "Veri" bandı içerebilir. Bu, master-detail-detail tipi raporların oluşturulmasına olanak tanır. Böyle bir rapor için bant yapısını "Bant Ayarları" penceresinde yapılandırabilirsiniz. Bunun için ana "Veri" bandına sağ tıklayın ve ona bağlı bir "Veri" bandı ekleyin. İkinci bağlı bandı eklemek için bu işlemi tekrarlayın:



Bu şekilde, ana "Veri" bandına sınırsız sayıda bağlı bant ekleyebilirsiniz. Örneğin, rapor yapısı şu şekilde görünebilir (bu, FastReport'un yetenekleri hakkında yalnızca bir fikir veren uydurma bir yapıdır):



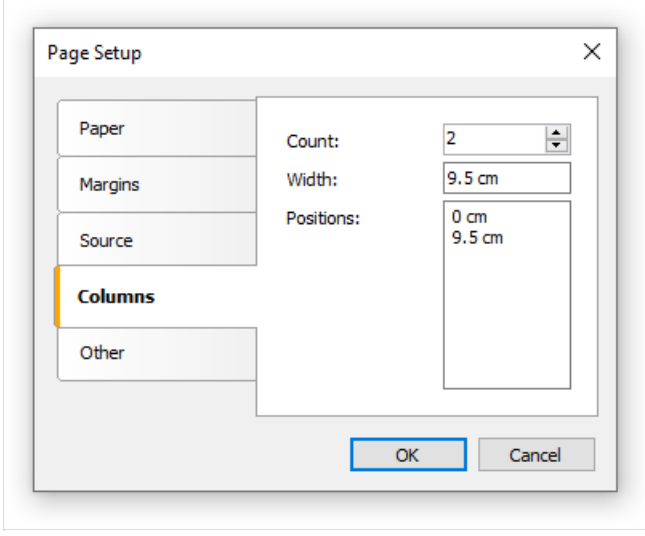
Çok sütunlu rapor

Normal rapor, sayfa sonuna ulaşınca kadar verileri yazdırır. Bunun ardından yeni bir sayfa oluşturulur ve yazdırma bu sayfada devam eder. Sütunlu rapor, verileri birden fazla sütun halinde yazdırır. Sayfa sonuna ulaşıldığında, yazdırma aynı sayfa üzerinde yeni bir sütundan devam eder. Bu anlamda, normal rapor tek sütunlu bir rapor olarak değerlendirilebilir.

FastReport'ta sütun yazdırmanın iki yöntemi vardır.

Sayfa Sütunları

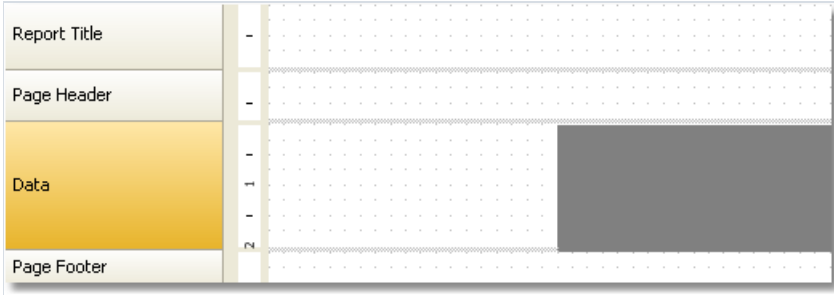
İlk yöntem, rapor sayfasındaki sütun sayısının belirtilmesidir. Bu, sayfa ayarlarında "Sütunlar" sekmesinde yapılır:



Görüldüğü gibi, sütunlar için aşağıdaki parametreleri belirtebilirsiniz:

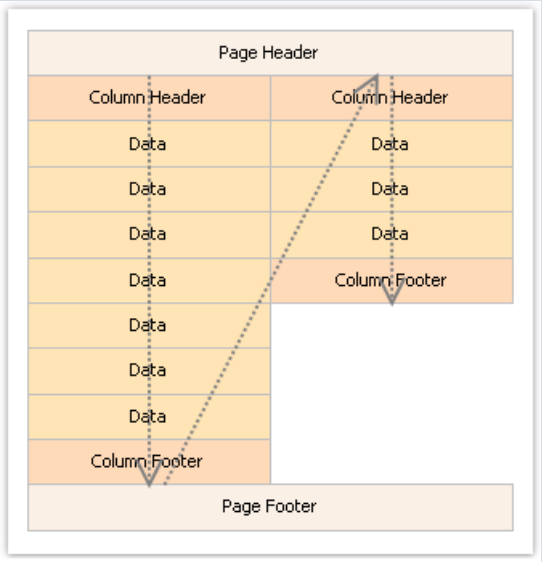
- sayı;
- sütun genişliği;
- her sütunun pozisyonu.

Normal bir raporu sütunlu bir rapora çevirmek için, sayfadaki sütun sayısını belirtmeniz yeterlidir. Diğer tüm parametreleri FastReport otomatik olarak seçecektir. Sütunları etkinleştirdiğinizde, tasarımcıdaki bant görünümü değişecektir:



"Veriler", "Grup Başlığı" ve veri çıktısıyla ilişkili diğer bantlar, yazdırma için yalnızca sütun genişliğine karşılık gelen alanı kullanır. Bu alan her zamanki gibi beyaz renkle gösterilir. Gri renkle gösterilen alan ise veri yerleştirme için kullanılamaz; çünkü bu alanda komşu sütunlardaki veriler yazdırılır.

Sütunlarla çalışmak için "Sütun Başlığı" ve "Sütun Altbilgisi" bantları kullanılır. Adlarından da anlaşılacağı üzere, bunlar sırasıyla her sütunun üstünde ve altında yazdırılır. Sütunlu rapor yazdırma şeması resimde gösterilmiştir.



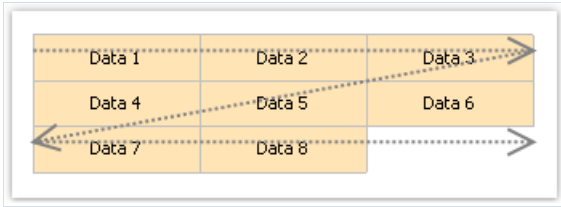
Veri Bandı Sütunları

Çok sütunlu baskının ikinci yöntemi, "Veri" bandının sütunlarının kullanılmasıdır. Bu durumda, diğer bantlar tek sütun halinde basılmaya devam eder.

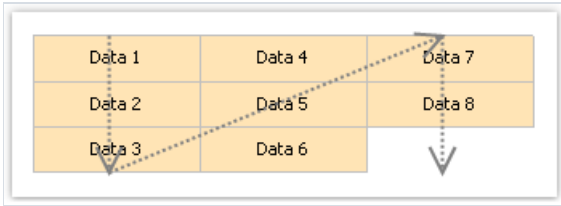
Sütun ayarları, "Özellikler" penceresinde değiştirilebilen `Columns` özelliği aracılığıyla yapılandırılır. Aşağıdaki parametreleri belirleyebilirsiniz:

- Sütun sayısı;
- Sütun genişliği;
- Sütun baskı modu. İki moddan birini seçebilirsiniz – "soldan sağa, yukarıdan aşağı" ve "yukarıdan aşağı, soldan sağa";
- "Yukarıdan aşağı, soldan sağa" baskı modu seçildiğinde, bir sütundaki minimum satır sayısı.

Sütun bandı iki farklı modda yazdırılabilir. `AcrossThenDown` modunda (varsayılan mod), sütunlar şu şekilde yazdırılır:



`DownThenAcross` modunda ise, sütunlar aşağıdaki şekilde yazdırılır:



Bu modda, FastReport sütunlardaki veri satırlarının sayısını, sütunların eşit şekilde doldurulacağı şekilde hesaplar. Ayrıca, sütundaki minimum satır sayısını `Columns.MinRowCount` özelliği ile belirleyebilirsiniz.

Buklet tarzında çok sayfalı rapor

Buklet şeklinde baskıya alınan raporda genellikle aşağıdaki gereksinimlerle karşılaşılır:

- Raporun ayrı sayfaları: kapak sayfası (cover), içindekiler (table of contents), rapor içeriği ve arka kapak (back cover);
- Çift ve tek sayfalar için farklı sayfa kenar boşlukları;
- Çift ve tek sayfalarda başlık ve altbilgilerin farklı biçimlendirilmesi.

Rapora Sayfa Ekleme

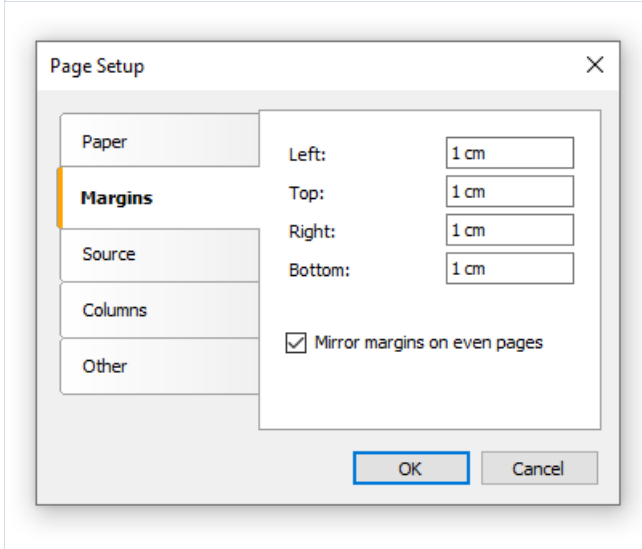
Rapor şablonuna istediğiniz sayıda sayfa ekleyebilirsiniz. Her sayfada ayrı bir rapor yerleştirilebilir.

Yeni bir sayfa eklemek için araç çubuğundaki ilgili düğmeye tıklayın. Alternatif olarak, sayfayı eklemek amacıyla "Oluştur..." düğmesine tıklayıp açılan pencereden "Rapor Sayfası" seçeneğini seçebilirsiniz.

"İçindekiler" bölümünün yazdırılması için "[Etkileşimli Raporlar](#)" bölümünde açıklanan yöntemi kullanabilirsiniz.

Sayfa Ayarları

Sayfa ayarlarında, **Kenar Boşlukları** sekmesinde, çift sayfalar için sol ve sağ kenar boşluklarının yer değiştireceğini belirtebilirsiniz:



Eğer sayfanın tek numara ile başlaması gerekiyorsa, sayfa özelliği **StartOnOddPage**'i true olarak ayarlayın. Gerekirse, FastReport belirtilen sayfanın baskısına başlamadan önce boş bir sayfa ekleyecektir.

Çift ve Tek Sayfalarda Baskı

Rapor nesnelerinin "BaskıYap" (**PrintOn**) özelliğini kullanarak, farklı nesneleri çift ve tek sayfalarda yazdırabilirsiniz.

"PrintOn" özelliğini değiştirmek için, öncelikle ilgili nesneyi seçip "Özellikler" penceresini açın.

Bu özellik, nesnenin hangi sayfalarda basılabileceğini belirler. Aşağıdaki değerler ve bunların herhangi bir kombinasyonu kullanılabilir:

- ilk sayfa (FirstPage);
- son sayfa (LastPage) – raporda çift geçişli mod etkinleştirilmiş olmalıdır;
- tek sayfalar (OddPages);
- çift sayfalar (EvenPages);
- bantın tekrarlı baskısı (RepeatedBand). Bu, bantın "Her sayfada tekrarla" bayrağı ayarlanmışsa ortaya çıkar;
- tek sayfa (SinglePage) – raporda çift geçişli mod etkinleştirilmiş olmalıdır.

Varsayılan olarak, bu özelliğin değeri FirstPage, LastPage, OddPages, EvenPages, RepeatedBand, SinglePage olarak ayarlanmıştır. Bu, nesnenin raporun tüm sayfalarında basılacağı anlamına gelir. Eğer rapor tek sayfadan oluşuyorsa, nesnenin basılması yalnızca SinglePage bayrağının varlığına bağlıdır.

Bu özelliğin kullanımına ilişkin birkaç tipik örneği aşağıda verelim.

Özellik değeri	Nesnenin yazdırılacağı yer
FirstPage	Sadece ilk sayfada.
LastPage, OddPages, EvenPages, RepeatedBand	İlk sayfa hariç tüm sayfalarda.
FirstPage, OddPages, EvenPages, RepeatedBand	Son sayfa hariç tüm sayfalarda.
RepeatedBand	Sadece her sayfada tekrarlanan bantlarda.
FirstPage, LastPage, OddPages, EvenPages	Her sayfada tekrarlanan bantlar hariç, tüm bantlarda.
FirstPage, LastPage, OddPages, RepeatedBand	Sadece tek numaralı sayfalarda.
FirstPage, LastPage, EvenPages, RepeatedBand	Sadece çift numaralı sayfalarda.

Not: Sayfaların tek veya çift olması, hazırlanan rapordaki sayfa numarasına göre belirlenir; sayfa numaralandırması 0'dan başlar. Örneğin, önizlemede görüntülenen ikinci sayfa, rapordaki sayfa numarası 1 olduğu için tek numaralı sayfadır.

Örneğin, çift ve tek sayfalarda farklı metinler yazdırmak için, bant üzerine iki "Metin" nesnesi yerleştirin ve bunları aşağıdaki şekilde yapılandırın:

- İlk nesne, tek sayfalarda yazdırılacaktır. Özelliğini şu şekilde ayarlayın: **PrintOn = FirstPage, LastPage, OddPages** (yani **EvenPages** hariç tüm değerler).
- İkinci nesne, çift sayfalarda yazdırılacaktır. Özelliğini şu şekilde ayarlayın: **PrintOn = FirstPage, LastPage, EvenPages** (yani **OddPages** hariç tüm değerler).

Nesneler, birbirlerinin üstüne veya farklı şekillerde yerleştirilebilir. Bunlar hiçbir zaman aynı anda yazdırılmaz.

Bantlarda benzer bir özellik bulunmaktadır. Farklı sayfalarda yazdırılan iki bant örneği oluşturmak için "Alt Bant" kullanın. Bu bant, "Bant Ayarları" penceresindeki herhangi bir banta iliştilirilebilir. Ana ve alt bantları aşağıdaki şekilde yapılandırın:

- Ana bant, tek sayfalarda yazdırılacaktır. Özelliğini şu şekilde ayarlayın: **PrintOn = FirstPage, LastPage, OddPages** (yani **EvenPages** hariç tüm değerler).
- Alt bant, çift sayfalarda yazdırılacaktır. Özelliğini şu şekilde ayarlayın: **PrintOn = FirstPage, LastPage, EvenPages** (yani **OddPages** hariç tüm değerler).

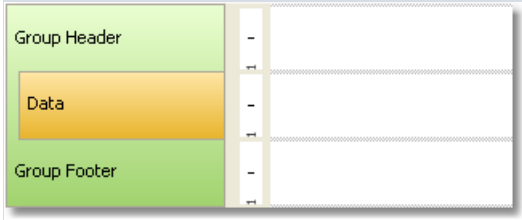
Bantlar, farklı yüksekliklere ve farklı içeriğe sahip olabilir. Farklı sayfa başlıkları yazdıran bir örneğe bakalım:

Page Header	-	PrintOn = FirstPage, LastPage, OddPages	[PageN]
Child	-	[PageN]	PrintOn = FirstPage, LastPage, EvenPages

Gruplama, sonuçlar

Daha önce, iki ilişkili kaynaktan veri yazdıran “Ana-Alt” tipi raporu incelemiştik. FastReport, yalnızca bir veri kaynağı kullanarak benzer görünümlü bir rapor oluşturmanıza olanak tanır. Bunun için gruplar kullanılır.

Bir grup, üç banttıan oluşur: **Grup Başlığı**, **Veri** ve **Grup Altbilgisi**. Tasarımcıda bu, aşağıdaki gibi görünür:



Bir grup her zaman bir başlık ve veri içerir. Grup altbilgisi isteğe bağlıdır; dilerseniz kaldırabilirsiniz.

Grup başlığı için gruplama koşulu belirtilir. Bu herhangi bir ifade olabilir, ancak genellikle veri kaynağındaki alanlardan biridir. Veri kaynağının kendisi “Veri” bandına bağlanır. Grubun yazdırılması şu şekilde gerçekleştirilir:

1. Grup başlığı yazdırılır;
2. Bir veri satırı yazdırılır;
3. Gruplama koşulunun değişip değişmediği kontrol edilir;
4. Eğer koşul değişmediyse, bir sonraki veri satırı yazdırılır (madde 2);
5. Eğer koşul değişmişse, grup altbilgisi yazdırılır ve yeni bir grubun yazdırılması başlatılır (madde 1).

Diyelim ki elimizde şu verilerle **Products** tablosu var:

CategoryName	ProductName
Beverages	Côte de Blaye
Beverages	Chartreuse verte
Beverages	Steeleye Stout
Beverages	Guaraná Fantástica
Beverages	Sasquatch Ale
Beverages	Rhönbräu Klosterbier
Beverages	Lakkaikööri
Beverages	Outback Lager
Beverages	Ipoh Coffee
Beverages	Laughing Lumberjack Lager
Beverages	Chang
Beverages	Chai
Condiments	Original Frankfurter grüne Soße
Condiments	Siroop d'érable
Condiments	Chef Anton's Gumbo Mix
Condiments	Northwoods Cranberry Sauce
Condiments	Grandma's Boysenberry Spread
Condiments	Chef Anton's Cajun Seasoning
Condiments	Aniseed Syrup
Condiments	Louisiana Hot Spiced Okra
Condiments	Vegie-spread
Condiments	Louisiana Fiery Hot Pepper Sauce
Condiments	Gula Malacca
Condiments	Genen Shouyu

Veriler, **CategoryName** alanına göre gruplanabilir. Bu alan, grup başlığında yazdırılacaktır. Verinin kendisi **ProductName** alanı ile temsil edilir. Rapor şu şekilde görünebilir:

Group Header:	-	[[Products.Categories.CategoryName]]
CategoryName	-	
Data: Products	-	[Products.ProductName]
Group Footer	-	

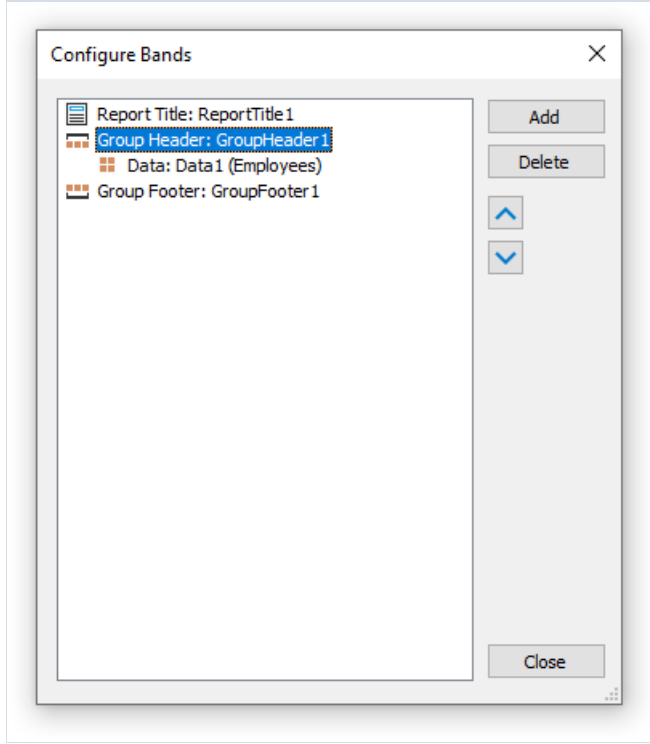
Rapor çalıştırılırsa, aşağıdaki gibi sonuçlanır:

Beverages
Côte de Blaye
Chartreuse verte
Steeleye Stout
Guaraná Fantástica
Sasquatch Ale
Rhönbräu Klosterbier
Lakkaikööri
Outback Lager
Ipoh Coffee
Laughing Lumberjack Lager
Chang
Chai
Condiments
Original Frankfurter grüne Soße
Sirop d'érable
Chef Anton's Gumbo Mix
Northwoods Cranberry Sauce
Grandma's Boysenberry Spread
Chef Anton's Cajun Seasoning
Aniseed Syrup
Louisiana Hot Spiced Okra
Veggie-spread
Louisiana Fiery Hot Pepper Sauce
Gula Malacca
Genen Shouyu

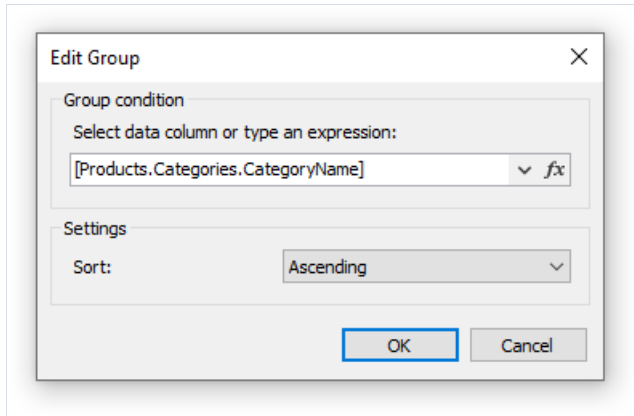
Grup Oluřturma

Bir rapora grup eklemek iki yöntemle yapılabilir.

Birinci yöntem: "Bantları Yapılandır" penceresinden "Grup başlığı" bandı eklemektir. Bunu yapmak için "Ekle" düğmesine basın ve "Grup başlığı" bandını seçin. FastReport mevcut bir "Veri" bandına grubu ekler veya raporda böyle bir bant yoksa tüm grubu oluşturur:

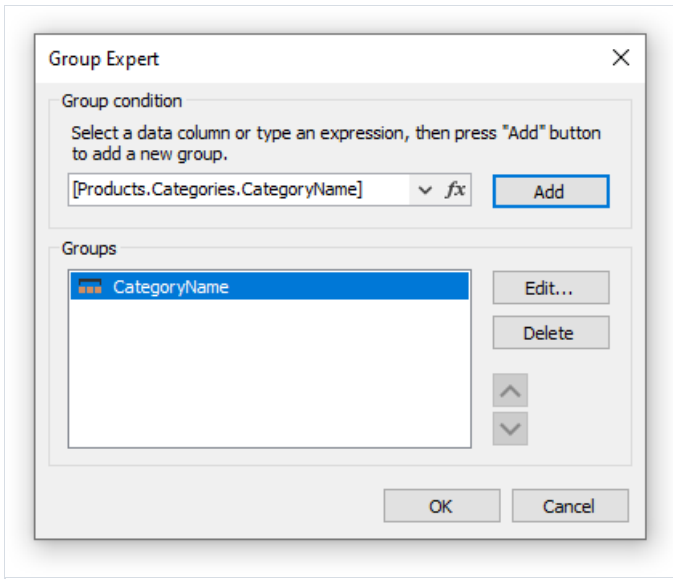


Bir grubu yapılandırmak için "Grup başlığı" bandına çift tıklayın. Grup başlığı düzenleyici penceresi görünecektir:



Grup koşulunu belirtmeniz gerekir. Bu, herhangi bir ifade veya veri kaynağı sütunu olabilir. Ayrıca sıralama türünü de seçin. Varsayılan olarak veriler artan düzende sıralanır.

İkinci yöntem: "Rapor | Grup Sihirbazı..." menüsünden çağrılabilen sihirbazı kullanmaktır. Bir grup oluşturmak için, grup koşulunu girin ve "Ekle" düğmesine basın:



Wizard, grubun tüm öğelerini rapora ekleyecektir. Ayrıca, grup koşulunun yazdırıldığı grup başlığında bir "Metin" nesnesi oluşturur.



Veri Sıralaması

Grubun doğru çalışması için aşağıdaki koşulun yerine getirilmesi gerekmektedir: Veri kaynağı, grupta koşulunda kullanılan alana göre sıralanmalıdır. Bu koşul sağlanmadığında, 1-2 satırlık veriler içeren birçok aynı grup görürsünüz:

Beverages
Sasquatch Ale
Steeleye Stout
Seafood
Inlagd Sill
Gravad lax
Beverages
Côte de Blaye
Chartreuse verte
Seafood
Boston Crab Meat
Jack's New England Clam Chowder

Neyse ki, FastReport'ta veri kaynağını iki farklı yöntemle sıralama imkanı bulunmaktadır:

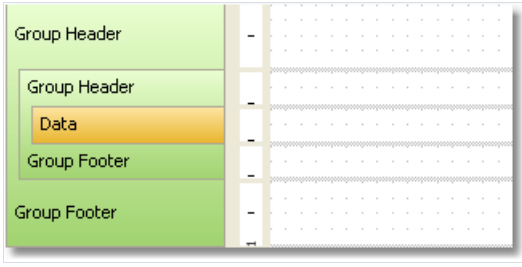
- Grup editöründe veri sıralama düzenini belirtirsiniz. Veri kaynağı otomatik olarak grupta koşuluna göre sıralanır;
- Gruba dahil olan "Veriler" bandı editöründe sıralama belirtirsiniz.

Her iki yöntem eşdeğerdir, ancak ilk yöntemi kullanmak daha kullanışlıdır – grup oluşturulurken, tek bir diyalogda grupta ve veri sıralamasını belirtirsiniz.

Bazı durumlarda ilk yöntemi kullanmak mümkün değildir. Örneğin, ürün adının ilk harfine göre grupta yapıyoruz. Bu durumda ürünler sadece ilk harfe göre sıralanır, bu istenmeyen bir durumdur. Bu durumda ikinci yöntemi kullanarak ürünün tam adına göre sıralama yapılması gerekmektedir.

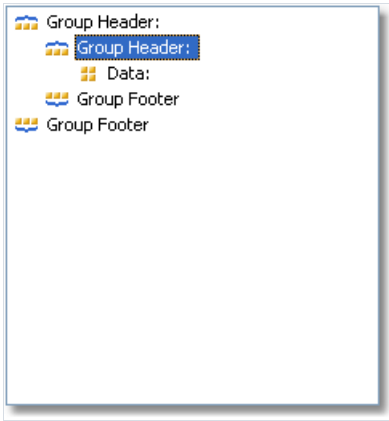
İç içe Gruplar

İç içe grup, birden fazla "Grup Başlığı" bandı içerir. Son band, "Veriler" bandını içerir:



Her grup başlığının kendi grup koşulu vardır.

İç içe grup oluşturmak, normal bir grup oluşturmakla aynı şekilde yapılabilir. İlk durumda, basit bir grup oluşturur ve iç içe grubu "Bantları Yapılandır" penceresinde eklersiniz. Bunun için mevcut "Grup başlığı" bandını seçin, "Ekle" düğmesine basın ve başka bir "Grup başlığı" bandı ekleyin:



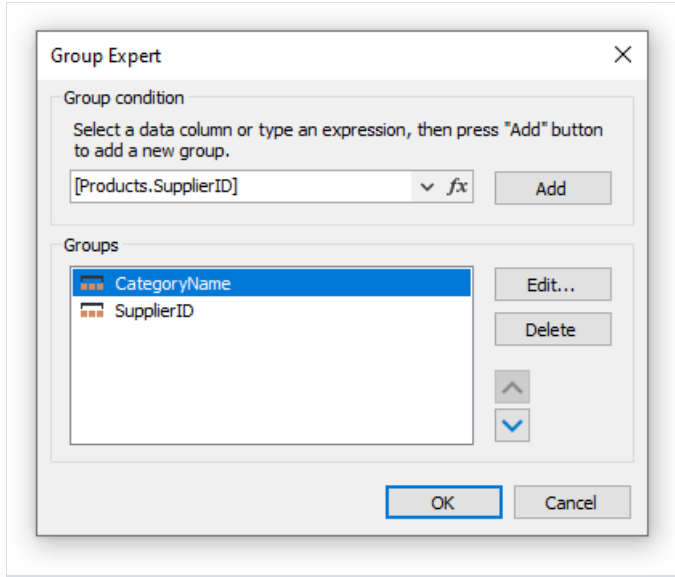
Bundan sonra, eklenen grubun düzenleyicisini çağırın ve grup koşulunu ayarlayın.

İkinci durumda, daha önce incelediğimiz grup sihirbazını kullanırsınız. Grup koşulunu ayarlayın ve "Ekle" düğmesine basın. Sihirbaz, yeni grubu mevcut gruba ekleyecektir.

İç içe grubun yazdırılması, normal bir grubun yazdırılmasından çok farklı değildir. Veriler yazdırılırken, FastReport tüm grupların grup koşullarını kontrol eder. Eğer koşul değişirse, ilgili grup tamamlanır ve yeni bir grup yazdırılmaya başlanır.

Grup Yönetimi

Grupları yönetmek için, "Rapor | Gruplama Sihirbazı..." menüsünden çağrılabilen gruptaştırma sihirbazını kullanın:



Sihirbaz sayesinde, grubu ekleyebilir veya kaldırabilir, ayrıca gruptaştırma sırasını deęiřtirmek için ilgili butonları kullanabilirsiniz. Seili grubun gruptaştırma kořulunu deęiřtirmek için "Deęiřtir..." butonunu kullanın.

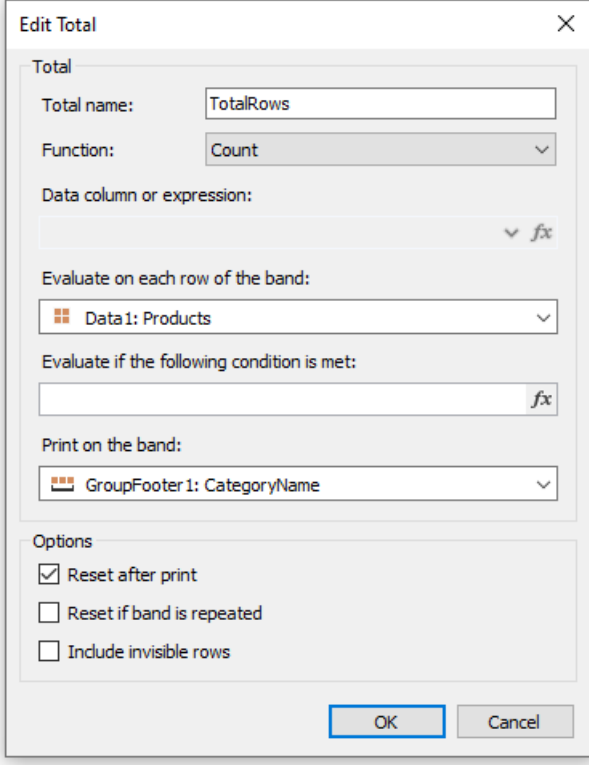
Toplam Değerlerin Yazdırılması

Gruplama, her grupta bazı özet değerleri yazdırmak için sıklıkla kullanılır. Örneğin, bu bir gruptaki satır sayısı veya veri alanlarından birinin toplamı olabilir. Bu tür değerleri yazdırmak için özetler kullanılır. Özetlerin kullanımı, "Veriler" başlığında ayrıntılı olarak açıklanmıştır.

Bir grupta özet değeri yazdırmak için şu adımları izleyin:

1. Veri penceresinde **Eylemler | Yeni Özet** seçeneğini seçerek bir özet oluşturun.
2. Veri satırı olarak, gruba dahil olan **Veriler** bandını belirtin.
3. **Bantta Yazdır** alanında, grup altbilgisini (footer) seçin.
4. Özet değerini yazdıran **Metin** nesnesini, grup altbilgisine yerleştirin.

Örneğin, her gruptaki satır sayısını yazdırmak için özet ayarı şu şekilde görünecektir:



Özet değerini görüntülemek için, onu grup altbilgisine sürükleyip bırakın:

Group Header:		[Products.Categories.CategoryName]
CategoryName		
Data: Products		[Products.ProductName]
Group Footer		[TotalRows]

Hazır rapor şu şekilde görünecektir:

Meat/Poultry

Alice Mutton

Perth Pasties

Thüringer Rostbratwurst

Pâté chinois

Tourtière

Mishi Kobe Niku

6**Produce**

Rössle Sauerkraut

Uncle Bob's Organic Dried Pears

Manjimup Dried Apples

Longlife Tofu

Tofu

5

Başlıkların ve Altlıkların Tekrarı

Grup başlığı ve altlığı için "Her Sayfada Tekrarla" (**RepeatOnEveryPage**) özelliği bulunmaktadır. Bu özellik, grupta çok sayıda satır bulunup raporun tek bir sayfasına sığmadığı durumlarda yararlı olabilir. Bu özellik kullanılarak, grubun kapladığı her sayfada grup başlığı/altlığı yazdırılabilir.

"Tekrarla..." özelliği etkinleştirildiğinde, "Tekrarlanan" (**Repeated**) bayrağı ayarlanmış başlık yazdırılır. Bu durum, normal grup başlığında ve tekrarlanan başlıkta farklı nesneler yazdırmak için kullanılabilir; örneğin, yeni sayfada "Devamı" yazısını yazdırmak gibi. Bunun için "Metin" nesnesinin **PrintOn** özelliği kullanılır (ayrıntıları "Broşür Tipi Çok Sayfalı Rapor" bölümünde incelenmiştir).

Farklı metin yazdırmak için, grup başlığı üzerine iki nesne yerleştirin, biri diğerinin üstünde:

- İlk nesne normal başlıklarda yazdırılacaktır. Bu nesnenin PrintOn özelliği, FirstPage, LastPage, OddPages, EvenPages değerlerine (yani RepeatedBand dışında kalan tüm mevcut değerlere) ayarlanmalıdır.
- İkinci nesne ise yalnızca tekrarlanan başlıklarda yazdırılacaktır. Bu nesnenin PrintOn özelliği RepeatedBand olarak ayarlanmalı ve üzerine "devamı" metni eklenmelidir.

Sonuç olarak, rapor aşağıdaki gibi yazdırılacaktır:

Group header	Group - continued
Data	Data
Data	Data
Data	Data
Group footer	Data
Group header	Data
Data	Data
Data	
Data	Group footer

Grup altlığı da her sayfada tekrar edilebilir. Bu, örneğin, düzenlemenin dikey çizgilerle yapıldığı durumlarda grubun görünümünü korumak için kullanılabilir:

R	
Raclette Courdavault	55,00
Ravioli Angelo	19,50
Rhönbräu Klosterbier	7,75
Röd Kaviar	15,00
Røgede sild	9,50
R	
Rössle Sauerkraut	45,60
Total products: 6	

Grup altlığı, birbirinin üstünde yer alan iki nesne içerir:



Grup Özellikleri

"Grup Başlığı" bandında bulunan grup özelliklerini inceleyelim.

Özellik "Yeni Sayfa Başlat" (`StartNewPage`), grubun yazdırılmasından önce yeni bir sayfa oluşturulmasını sağlar. Böylece, her grup yeni bir sayfada yer alır.

İlk grup için yeni sayfa oluşturulmayacaktır. Bu durum, boş bir ilk sayfa oluşmasını önlemek amacıyla yapılmıştır.

Özellik "Sayfa Numarasını Sıfırla" (`ResetPageNumber`), grubun yazdırılması sırasında sayfa numarasının sıfırlanmasını sağlar. Genellikle bu özellik, "Yeni Sayfa Başlat" özelliğiyle birlikte kullanılır. Dolayısıyla, her iki özellik de etkinleştirildiğinde, her grup yeni bir sayfada yazdırılır ve kendi sayfa numaralandırmasına sahip olur.

Gömülü Raporlar

Bazen ana raporun belirli bir bölümünde, ayrı ve oldukça karmaşık yapıya sahip bir raporu andıran ek verilerin sunulması gerekebilir. Bu sorunu FastReport'un sunduğu zengin bant seti kullanılarak çözmeye çalışabilirsiniz. Ancak bazı durumlarda "Gömülü Rapor" nesnesini kullanmak daha tercih edilir.

"Gömülü Rapor" nesnesi, raporun bir bandına yerleştirilebilen sıradan bir rapor nesnesidir. Bu durumda FastReport, rapora ek bir sayfa ekler ve bu sayfayı gömülü rapor nesnesiyle ilişkilendirir. Ek sayfada, herhangi bir yapıya sahip ek bir rapor oluşturabilirsiniz.

Gömülü rapor nesnesi bulunan bir rapor yazdırılırken şu adımlar izlenir:

- Ana rapor, gömülü rapor nesnesi ile karşılaşılana kadar yazdırılır;
- Gömülü raporun bantları yazdırılır;
- Yazdırma işlemi ana rapor için devam ettirilir.

Gömülü rapor, ana rapor sayfası üzerinde oluşturulduğu için aşağıdaki bantları içeremez:

- Rapor Başlığı/Alt bilgi
- Sayfa Başlığı/Alt bilgi/Arka Plan
- Kolon Başlığı/Alt bilgi

Baskı Modları

Gömülü rapor iki modda yazdırılabilir.

Birinci Yazdırma Modu (Ana Mod): Bu modda, bantlar ve gömülü raporun nesneleri ana rapor sayfasında yazdırılır. Gömülü raporu içeren bant üzerindeki nesnelerin yerleştirilmesiyle ilgili şu kısıtlamalar vardır:

- **"Gömülü rapor" nesnesi**, bantın alt sınırında yer almalıdır;
- **"Gömülü rapor" nesnesinin altına** başka nesneler yerleştirilemez. Rapor çalışırken bu nesneler, gömülü raporun nesneleri tarafından örtülecektir:



Gömülü rapordan aşağıya diğer nesneleri yerleştirmek için *****Child** bant ****kullanın**. Bu bant, ana banda eklenerek nesnelerin istenilen şekilde yerleştirilmesine olanak tanır:



İkinci Yazdırma Modu: Bu modda, gömülü rapor, **"Gömülü rapor" nesnesini** içeren bant üzerinde yazdırılır. Bu modu etkinleştirmek için, gömülü rapor nesnesinin bağlam menüsünden **"Ebeveyn üzerinde yazdır"** seçeneğini seçin.

Bu modun avantajları:

- Nesnelerin yerleştirilmesi konusunda herhangi bir kısıtlama yoktur.
- Ebeveyn bant, gömülü raporda yazdırılan veri miktarına bağlı olarak genişleyip daralabilir.

Ancak bu modun tek sorunu, gömülü raporda çok fazla veri bulunması durumunda baskı sırasında ebeveyn bantın yüksekliğinin artmasıdır. Böyle bir bantı düzgün şekilde yazdırabilmek için, içeriğinin kesintiye uğratılması (özellik: **"Kesilebilir"**) uygulanmalıdır. Kesinti algoritması %100 kalite sağlamaz ve bazı durumlarda nesnelerin yer değiştirmesine neden olabilir.

Yan Yana Birkaç Raporun Yazdırılması (side-by-side)

İki "İç Rapor" nesnesini yan yana yerleştirerek, yan yana konumlanacak iki bağımsız veri listesinin yazdırılması sağlanır. Böyle bir rapor yazdırılırken FastReport şu şekilde çalışır:

- Temel rapor, "İç Rapor" nesnesine rastlanıncaya kadar yazdırılır;
- İlk iç rapor yazdırılır;
- İç raporun yazdırılmaya başlandığı sayfaya geri dönülerek sonraki iç rapor yazdırılır;
- Tüm iç raporlar yazdırıldıktan sonra, temel rapor, en uzun iç raporun yazdırılmasının sona erdiği yerden devam eder.

Birkaç iç içe geçme seviyesi

İç içe rapor sayfasına, "İç içe rapor" nesnesi yerleştirilerek iç içe geçme seviyesi artırılabilir. İç içe geçme seviyelerinin sayısı resmi olarak hiçbir şekilde sınırlandırılmamaktadır, ancak buna fazla kapılmamalısınız. Çok katmanlı iç içe raporlar anlaşılması oldukça güçtür. İmkanınız varsa, iç içe verilerin yazdırılması için FastReport bantlarını kullanın. Hatırlatmak isteriz ki, "Veri" bandı bir veya daha fazla iç içe "Veri" bandına sahip olabilir ve bu şekilde devam edebilir. Böylece, master-detail veya master-detail-subdetail tipindeki raporların yazdırılması için iç içe raporlar kullanmaya gerek yoktur.

Tablo Raporları

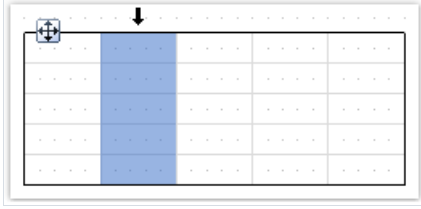
"Tablo" nesnesi, satırlardan, sütunlardan ve hücrelerden oluşur ve Microsoft Excel tablosunun basitleştirilmiş bir versiyonudur. Bu şekilde görünür:

...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...

Kolon ayarları

Kolonları sağ tıklama menüsü aracılığıyla silebilir veya ekleyebilirsiniz. Bunu yapmak için:

- Tabloyu veya herhangi bir ögesini seçin ve farenizi gerekli kolon üzerine getirin. İmleç küçük siyah bir ok şeklini alacaktır:



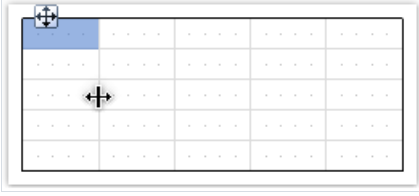
- Kolonu seçmek için sol fare düğmesine basın;
- Kolonun sağ tıklama menüsünü açmak için sağ fare düğmesine basın;
- Bitişik birkaç kolonu seçmeniz gerekiyorsa, sol düğmeye basılı tutun ve farenizi sola veya sağa hareket ettirerek bitişik kolonları seçin.

Kolonun sağ tıklama menüsünü "Rapor Ağacı" penceresinde de açabilirsiniz. Pencereyi açın, gerekli kolonu seçin ve sağ fare düğmesine basın.

Kolon Boyutunu Yönetme

Kolon genişliğini aşağıdaki yöntemlerden biriyle belirleyebilirsiniz:

- Tabloyu veya herhangi bir ögesini seçin ve imleci iki sütun arasındaki kenarlığa getirin. İmlecin şekli yatay bir ayırıcıya dönüşecektir.



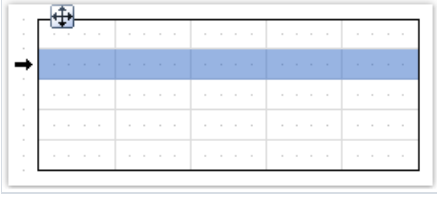
- Bir sütunu seçin ve Genişlik (**width**) özelliğinde gerekli genişliği belirtin. Bu özellik "Özellikler" (Properties) penceresinde erişilebilirdir.

Sütunun genişliği hiçbir zaman sayfanın genişliğinden büyük olmamalıdır.

Satır Ayarları

Satırlar benzer şekilde ayarlanır. Bir satırı seçmek için aşağıdaki adımları izleyin:

- Tabloyu veya herhangi bir ögesini seçin ve imleci gerekli satırın sol tarafına yerleştirin. İmlecın şekli küçük siyah bir oka dönüşecektir:



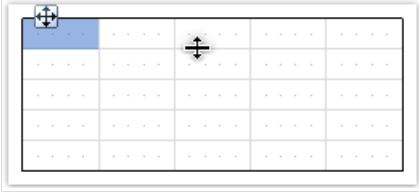
- Satırı seçmek için fareyle sol tıklayın.
- Satırın bağlam menüsünü göstermek için fareyle sağ tıklayın.

Birden fazla bitişik satırı seçmeniz gerekiyorsa, fareyle sol tıklayın ve bırakmadan fareyi sağa veya sola doğru hareket ettirerek bitişik satırları seçin.

Satır Boyutunu Yönetme

Satır yüksekliğini aşağıdaki yollardan biriyle belirtebilirsiniz:

- Tabloyu veya herhangi bir ögesini seçin ve imleci iki satır arasındaki kenarlığa getirin. İmlecin şekli dikey bir ayırıcıya dönüşecektir:



Satırın boyutunu değiştirmek için sol tıklayın ve fareyi hareket ettirin.

- Satırı seçin ve Genişlik özelliğinde gerekli yüksekliği belirtin. Bu özellik "Özellikler" penceresinden erişilebilirdir.

Ayrıca satırın **AutoSize** (OtomatikBoyut) özelliğini de etkinleştirebilirsiniz. Rapor çalıştırıldığında, satırın yüksekliği otomatik olarak hesaplanacaktır. Satırın yüksekliğini sınırlamak için **MinHeight** (MinimumYükseklik) ve **MaxHeight** (MaksimumYükseklik) özelliklerini kullanabilirsiniz.

Satırın yüksekliği hiçbir zaman sayfanın yüksekliğinden büyük olmamalıdır.

Hücre Ayarları

Hücre, bir metin nesnesidir. Temelde, hücre sınıfı "Metin" nesnesinden miras alır. Yukarıda "Metin" nesnesi hakkında söylenenlerin tamamı, tablo hücresine de uygulanır.

Hücre metnini "Metin" nesnesi gibi düzenleyebilirsiniz. Ayrıca, "Veri" penceresinden bir ögeyi hücre içine sürükleyip bırakabilirsiniz.

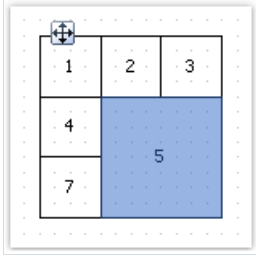
Hücrelerin kenarlık ve dolgu ayarları, "Kenarlık ve Dolgu" araç çubuğu ile yapılandırılabilir.

Hücrenin bağlam menüsünü açmak için üzerine sağ tıklayın. Bağlam menüsünde aşağıdaki işlemleri gerçekleştirebilirsiniz:

Hücre Birleştirme

Tablodaki bitişik hücreleri birleştirebilirsiniz. Sonuç olarak, tek büyük bir hücre oluşur. Bunu yapmak için:

- İlk hücreyi fare yardımıyla seçin.
- Sol tıklayın ve tıklamayı bırakmadan fareyi hareket ettirerek hücre grubunu seçin.
- Seçili alanda hücrelerin bağlam menüsünü açmak için fareyle sağ tıklayın.
- Seçilen bölgede hücrelerin bağlam menüsünü göstermek için fareye sağ tıklayın.



Bir hücreyi bölmek için, hücrenin bağlam menüsünü açın ve "Hücreyi Böl" seçeneğini seçin.

Tablonun Yazdırılması

Bir tablo iki modda yazdırılabilir:

İlk modda, tablo ait olduğu bandın içinde yazdırılır ve tasarımcıda görüldüğü gibi aynı şekilde görünür. Bu modda, tablonun genişliği rapor sayfasının genişliğinden büyükse, tablo sayfalar arasında bölünmez. Bu, varsayılan yazdırma modudur.

İkinci mod dinamik moddur. Bu modda, tablo script yardımıyla oluşturulur. Bu esnada, oluşan tablo, başlangıçtaki tablodan farklı olabilir, tıpkı FastReport'un hazırlanan raporunun rapor şablonundan farklı olması gibi. Dinamik modda, tablo rapor sayfasına sığmazsa sayfalar arasında bölünebilir.

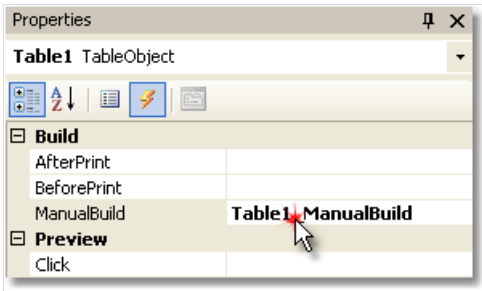
Dinamik modda, tablo bulunduğu bandın üzerine yazdırılmaz. Bunun yerine, tablo kendisi, oluşan tablonun parçalarını içeren bir dizi band oluşturur. Bu çalışma modu, aşağıdaki sınırlamaları getirmektedir:

- Tablonun altına veya yanına başka nesneler yerleştirmeyin. Bunun yerine, "Child" (AltBand) bandını kullanın.
- Bir bandın üzerine asla iki "Tablo" nesnesi yerleştirmeyin.

Dinamik modu detaylı olarak inceleyelim.

Bu mod, programlama ile bağlantılıdır ve rapor geliştiricisinden daha yüksek nitelikler gerektirir.

Tablonun oluşumu, script yardımıyla yapılır. Bir script oluşturmak için, "Tablo" nesnesini seçin, "Özellikler" penceresinde "Olaylar" (Events) düğmesine tıklayın ve ManualBuild olayına çift tıklayın.



yöntem	parametreler	açıklama
PrintRow	int index	Belirtilen numaralı tablo satırını yazdırır. Satır numaralandırması 0'dan başlar.
PrintColumn	int index	Belirtilen numaralı tablo sütununu yazdırır. Sütun numaralandırması 0'dan başlar.
PrintRows	int[] indices	Tablonun birkaç satırını yazdırır.
PrintRows	-	Tablonun tüm satırlarını yazdırır.
PrintColumns	int[] indices	Tablonun birkaç sütununu yazdırır.
PrintColumns	-	Tablonun tüm sütunlarını yazdırır.
PageBreak	-	Her yeni satır veya sütun yazdırılmadan önce sayfa sonu ekler.

Tablonun yazdırılması iki yöntemden biriyle gerçekleştirilmelidir.

Yöntem 1: Yukarıdan Aşağıya, Sonra Soldan Sağa Yazdırma Bu yöntem, değişken sayıda satıra sahip tabloların yazdırılması için daha uygundur. Yazdırma metodlarını aşağıdaki sırayla çağırmanız gerekir:

- PrintRow (satır numarası);
- Belirtilen sütunları yazdırmak için bir veya birden fazla PrintColumn (sütun numarası) veya PrintColumns (sütun numaraları) metod çağırısı;
- Ya da tüm sütunları yazdırmak için bir kez PrintColumns() metod çağırısı;
- Gerekli tüm tablo satırlarını yazdırmak için bu sıralamayı tekrarlayın.

Her tablo satırı aynı sayıda sütun içermelidir. PrintColumn(int index) ve PrintColumns(int[] indices) metodlarını kullanırken bunu göz önünde bulundurun.

Yöntem 2: Soldan Sağa, Sonra Yukarıdan Aşağıya Yazdırma Bu yöntem, değişken sayıda sütuna sahip tabloların yazdırılması için daha uygundur. Yazdırma metodlarını aşağıdaki sırayla çağırmanız gerekir:

- PrintColumn (sütun numarası);
- Belirtilen satırları yazdırmak için bir veya birden fazla PrintRow (satır numarası) veya PrintRows (satır numaraları) metod çağırısı;
- Ya da tüm satırları yazdırmak için bir kez PrintRows() metod çağırısı;
- Gerekli tüm tablo sütunlarını yazdırmak için bu sıralamayı tekrarlayın.

Her tablo sütunu aynı sayıda satır içermelidir. PrintRow(int index) ve PrintRows(int[] indices) metodlarını kullanırken bunu göz önünde bulundurun.

Önemli Not: Yazdırma metodlarının çağrı sırasına uyulmaması, raporun çalışması sırasında hatalara yol açacaktır. Örneğin, aşağıdaki kod hatalıdır:

```
Table1.PrintRows();  
Table1.PrintColumns();
```

Bu metod sıralaması yanlıştır. Tablonun yazdırılmasına PrintRow veya PrintColumn metoduyla başlamalısınız.

Karmaşık Başlıkların Yazdırılması

Burada kastedilen, birleşik hücreler içeren başlıklardır. Birleşik hücre içeren bir tablo satırı veya sütunu yazdırıldığında, hücre otomatik olarak boyutunu genişletir. Bu durumu aşağıdaki örnekle göstereceğiz:

Header	
1	2

Örnekte, ilk sütunu 3 kere ve ikinci sütunu 1 kere yazdıracak bir `ManualBuild` olay işleyicisi oluşturuyoruz:

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // Satır 0'ı yazdırıyoruz ve sütunları sırasıyla 0, 0, 0, 1 yazdırıyoruz.
    Table1.PrintRow(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(1);
    // Satır 1'i yazdırıyoruz ve sütunları sırasıyla 0, 0, 0, 1 yazdırıyoruz.
    Table1.PrintRow(1);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(1);
}
```

Her satırda aynı sayıda sütun yazdırdığımıza dikkat edin. Bu kurala uyulmazsa, sonuç öngörülemez hale gelir.

Bu kodun çalıştırılması sonucunda aşağıdaki çıktıyı elde ederiz:

Header			
1	1	1	2

Görüldüğü üzere, başlık otomatik olarak genişlemiştir. Şimdi kodu biraz daha karmaşık hale getirerek iki sütun grubunu yazdırmayı görelim:

```

private void Table1_ManualBuild(object sender, EventArgs e)
{
    // Satır 0'ı yazdırıyoruz ve iki sütun grubu (0, 0, 0, 1) yazdırıyoruz.
    Table1.PrintRow(0);
    // Grup 1
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(1);
    // Grup 2
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(1);

    // Satır 1'i yazdırıyoruz ve iki sütun grubu (0, 0, 0, 1) yazdırıyoruz.
    Table1.PrintRow(1);
    // Grup 1
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(1);
    // Grup 2
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(1);
}

```

Raporu çalıştırdığınızda aşağıdaki sonucu görürsünüz:

Header				Header			
1	1	1	2	1	1	1	2

İkinci sütunun yazdırılması sırasında:

```
Table1.PrintColumn(1);
```

başlık kapanır ve takip eden ilk sütun yazdırması yeni bir başlık açar:

```

// Grup 2
Table1.PrintColumn(0);

```

Toplamların Yazdırılması

Dinamik modda, "Tablo" nesnesi aşağıdaki özet (toplam) fonksiyonlarını destekler:

Fonksiyon	Parametreler	Açıklama
Sum	TableCell cell4	Hücredeki değerlerin toplamını döndürür.
Min	TableCell cell	Hücrede bulunan değerler arasından en küçük olanı döndürür.
Max	TableCell cell	Hücrede bulunan değerler arasından en büyük olanı döndürür.
Avg	TableCell cell	Hücrede bulunan değerlerin ortalamasını döndürür.
Count	TableCell cell	Hücreyi içeren satırların sayısını döndürür.

Dinamik mod dışında yapılan baskıda toplam fonksiyonları çalışmaz.

Toplam fonksiyonunu kullanmak için, fonksiyonu bir tablo hücresine yerleştirin. Örneğin, aşağıdaki fonksiyon "Cell1" adlı hücredeki değerlerin toplamını hesaplar:

```
[Sum(Cell1)]
```

Bu durumda, mevcut hücreden yukarıda ve solda bulunan tüm hücreler analiz edilir.

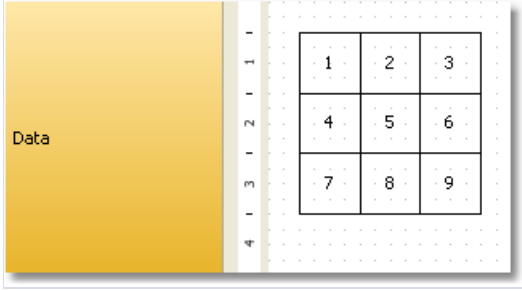
Tablo Düzeni

Dinamik modda oluşturulan tablo, hem genişlik hem de yükseklik açısından otomatik olarak sayfalara bölünebilir. Tablonun Layout özelliği, üç farklı bölme algoritmasından birini seçmenizi sağlar:

Değer	Açıklama
AcrossThenDown	Önce yana, sonra aşağıya doğru açılım.
DownThenAcross	Önce aşağıya, sonra yana doğru açılım.
Wrapped	Geniş tablo tek sayfada görüntülenir.

Tablo

Örneklerle tablo yazdırmayı inceleyelim. Şablon olarak aşağıdaki raporu kullanacağız:



1	2	3
4	5	6
7	8	9

Tabloyu seçin ve **ManualBuild** olay işleyicisini oluşturun.

Örnek 1. Tablonun yukarıdan aşağıya doğru yazdırılması

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // 0. satırı ve tüm sütunlarını yazdır
    Table1.PrintRow(0);
    Table1.PrintColumns();
    // 1. satırı ve tüm sütunlarını yazdır
    Table1.PrintRow(1);
    Table1.PrintColumns();
    // 2. satırı ve tüm sütunlarını yazdır
    Table1.PrintRow(2);
    Table1.PrintColumns();
}
```

Sonuç olarak, şablondan farklı olmayan aşağıdaki tablo yazdırılacaktır:

1	2	3
4	5	6
7	8	9

Örnek 2. Satırların tekrarıyla tablonun yukarıdan aşağıya doğru yazdırılması

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // 0. satırı ve tüm sütunlarını yazdırıyoruz
    Table1.PrintRow(0);
    Table1.PrintColumns();
    // 1. satırın 3 kopyasını ve tüm sütunlarını yazdırıyoruz
    for (int i = 0; i < 3; i++)
    {
        Table1.PrintRow(1);
        Table1.PrintColumns();
    }
    // 2. satırı ve tüm sütunlarını yazdırıyoruz
    Table1.PrintRow(2);
    Table1.PrintColumns();
}
```

Bu örnekte, tablonun orta satırı 3 kez yazdırılmaktadır. Sonuç olarak elde edilen çıktı şöyledir:

1	2	3
4	5	6
4	5	6
4	5	6
7	8	9

Örnek 3. Tabloyu soldan sağa doğru yazdırma

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // 0. sütunu ve tüm satırlarını yazdır
    Table1.PrintColumn(0);
    Table1.PrintRows();
    // 1. sütunu ve tüm satırlarını yazdır
    Table1.PrintColumn(1);
    Table1.PrintRows();
    // 2. sütunu ve tüm satırlarını yazdır
    Table1.PrintColumn(2);
    Table1.PrintRows();
}
```

Sonuç olarak şablondan farklı olmayan aşağıdaki tablo yazdırılacaktır:

1	2	3
4	5	6
7	8	9

Örnek 4. Sütunların tekrarı ile tablonun soldan sağa yazdırılması

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // sütun 0'ı ve tüm satırlarını yazdırır
    Table1.PrintColumn(0);
    Table1.PrintRows();
    // sütun 1'in 3 kopyasını ve tüm satırlarını yazdırır
    for (int i = 0; i < 3; i++)
    {
        Table1.PrintColumn(1);
        Table1.PrintRows();
    }
    // sütun 2'yi ve tüm satırlarını yazdırır
    Table1.PrintColumn(2);
    Table1.PrintRows();
}
```

Bu örnekte, tablonun orta sütunu 3 kez yazdırılır. Sonuç olarak aşağıdakiler elde edilir:

1	2	2	2	3
4	5	5	5	6
7	8	8	8	9

Örnek 5. Satır ve Sütunların Tekrarlı Yazdırılması

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // ----- 0. satırını yazdırıyoruz
    Table1.PrintRow(0);
    // 0. sütunu yazdırıyoruz
    Table1.PrintColumn(0);
    // 1. sütunun 3 örneğini yazdırıyoruz
    for (int i = 0; i < 3; i++)
        Table1.PrintColumn(1);
    // 2. sütunu yazdırıyoruz
    Table1.PrintColumn(2);

    // ----- 1. satırın 3 örneğini yazdırıyoruz
    for (int j = 0; j < 3; j++)
    {
        Table1.PrintRow(1);
        // 0. sütunu yazdırıyoruz
        Table1.PrintColumn(0);
        // 1. sütunun 3 örneğini yazdırıyoruz
        for (int i = 0; i < 3; i++)
            Table1.PrintColumn(1);
        // 2. sütunu yazdırıyoruz
        Table1.PrintColumn(2);
    }
    // ----- 2. satırını yazdırıyoruz
    Table1.PrintRow(2);
    // 0. sütunu yazdırıyoruz
    Table1.PrintColumn(0);
    // 1. sütunun 3 örneğini yazdırıyoruz
    for (int i = 0; i < 3; i++)
        Table1.PrintColumn(1);
    // 2. sütunu yazdırıyoruz
    Table1.PrintColumn(2);
}
```

Her satırda aynı sayıda sütun yazdırdığımıza dikkat edin. Bu kurala uyulmazsa, öngörülemeyen sonuçlar elde edilir.

Bu örnekte, tablonun orta satırı ve orta sütunu 3 kez yazdırılmaktadır. Sonuç olarak aşağıdakiler elde edilir:

1	2	2	2	3
4	5	5	5	6
4	5	5	5	6
4	5	5	5	6
7	8	8	8	9

Örnek 6. Veri Kaynaklarına Bağlama

Tüm incelenen örneklerde normal metin içeren bir tablo bastırmıştık. Bu örnekte, bir veri kaynağı kullanarak nasıl tablo oluşturulacağını göstereceğiz. Bunun için aşağıdaki gibi bir tablo oluşturacağız:

Product Name	Unit Price	Units In Stock
[Products.ProductName]	[Products.UnitPrice]	[Products.UnitsInStock]

ManualBuild olay işleyicisini aşağıdaki gibi oluşturacağız; bu işleyici şunları yapacak:

- Rapor içinde tanımlı olan veri kaynağına referans alacak;
- Onu başlatacak (verilerle dolduracak);
- Veri kaynağındaki kayıt sayısı kadar tablo satırlarını bastırarak.

İşte olay işleyicisinin kodu:

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // Adıyla veri kaynağını alıyoruz
    DataSourceBase rowData = Report.GetDataSource("Products");
    // Onu başlatıyoruz
    rowData.Init();

    // Tablonun başlığını bastırıyoruz
    Table1.PrintRow(0);
    Table1.PrintColumns();

    // Veri kaynağında kayıtlar olduğu sürece döngü gerçekleştiriyoruz
    while (rowData.HasMoreRows)
    {
        // Tablonun satırını bastırıyoruz
        Table1.PrintRow(1);
        Table1.PrintColumns();

        // Veri kaynağındaki bir sonraki kayda geçiyoruz
        rowData.Next();
    }

    // Tablonun alt bilgisini bastırıyoruz
    Table1.PrintRow(2);
    Table1.PrintColumns();
}
```

FastReport, "Veriler" bandını yazdırırken benzer işlemleri gerçekleştirir. Raporu çalıştırırsak, aşağıdakini elde ederiz:

Product Name	Unit Price	Units In Stock
Chai	18,00	39
Chang	19,00	17
Aniseed Syrup	10,00	13
Chef Anton's Cajun Seasoning	22,00	53
Chef Anton's Gumbo Mix	21,35	0
Grandma's Boysenberry Spread	25,00	120
Uncle Bob's Organic Dried Pears	30,00	15
Northwoods Cranberry Sauce	40,00	6

Örnek 7. Sayfa Sonu Ekleme

Sayfa sonu eklemek için "Tablo" nesnesinin `PageBreak` metodu kullanılır. Bu metodu, tablonun bir sonraki satırını veya sütununu basmadan önce çağırın.

`PageBreak` metodunun kullanımını Örnek 1 üzerinden inceleyelim. Üçüncü satırın yeni bir sayfada basılmasını sağlayacağız.

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // 0. satırı ve tüm sütunlarını basıyoruz
    Table1.PrintRow(0);
    Table1.PrintColumns();
    // 1. satırı ve tüm sütunlarını basıyoruz
    Table1.PrintRow(1);
    Table1.PrintColumns();
    // sayfa sonu ekliyoruz
    Table1.PageBreak();
    // 2. satırı ve tüm sütunlarını basıyoruz
    Table1.PrintRow(2);
    Table1.PrintColumns();
}
```

Sonuç olarak, aşağıdaki gibi bir tablo basılacaktır:

1	2	3
4	5	6

7	8	9
---	---	---

Örnek 8. Toplamların Yazdırılması

Örnek 6’da toplam fonksiyonlarının kullanımını ele alalım. Bunu aşağıdaki şekilde modifiye edelim:

Product Name	Unit Price	Units In Stock
[Products.ProductName]	[Products.UnitPrice]	[Products.UnitsInStock]
	Total:	[Sum(Cell8)]

The cell Cell8, which we will summarize

Summary value

Raporu çalıştırsak, aşağıdaki çıktıyı elde ederiz:

Outback Lager	15,00	15
Fløtemysost	21,50	26
Mozzarella di Giovanni	34,80	14
Röd Kaviar	15,00	101
Longlife Tofu	10,00	4
Rhönbräu Klosterbier	7,75	125
Lakkaikkööri	18,00	57
Original Frankfurter grüne Soße	13,00	32
Total:	3119	

Matris (özet) raporları

"Matris" nesnesi, "Tablo" nesnesinin bir çeşididir ve tıpkı "Tablo" nesnesi gibi satırlar, sütunlar ve hücrelerden oluşur. Ayrıca, matriste kaç satır ve sütun olacağı önceden bilinmez – bu, bağlı olduğu verilere bağlıdır. \

Nesne aşağıdaki gibi görünür:

Employee	[Year]	Total
[Name]	[Revenue]	
Total		

Yazdırıldığında, matris değerlerle doldurulur ve hem aşağıya hem yana doğru büyür. Sonuç aşağıdaki gibi görünebilir:

Employee	1999	2000	2001	2002	Total
Andrew Fuller	\$3,900.00	\$2,100.00		\$1,800.00	\$7,800.00
Janet Leverling	\$6,100.00	\$3,200.00			\$9,300.00
Nancy Davolio	\$3,300.00	\$2,700.00	\$3,100.00	\$1,700.00	\$10,800.00
Steven Buchanan		\$3,999.00	\$8,100.00		\$12,099.00
Total	\$13,300.00	\$11,999.00	\$11,200.00	\$3,500.00	\$39,999.00

Biraz teori

Matrisin öğelerini inceleyelim:

	10		20	
	1	2	1	2
a	a10.1	a10.2	a20.1	a20.2
b	b10.1	b10.2	b20.1	b20.2

Bu örnekte bir sütun bileşiktir, yani iki değere sahiptir. Bu rapor için aşağıdaki verilere ihtiyaç vardır:

```
a 10 1 a10.1
a 10 2 a10.2
a 20 1 a20.1
a 20 2 a20.2
b 10 1 b10.1
b 10 2 b10.2
b 20 1 b20.1
b 20 2 b20.2
```

Burada, ilk sütun satırı temsil eder, ikinci ve üçüncü sütunlar matris sütununu temsil eder. Son veri sütunu ise hücrenin değerini içerir.

Sonraki matris öğesi — ara toplam ve genel toplamdır, bunu aşağıdaki şekil göstermektedir:

	10			20			Total
	1	2	Total	1	2	Total	
a	a10.1	a10.2	a10.1+a10.2	a20.1	a20.2	a20.1+a20.2	sum(a)
b	b10.1	b10.2	b10.1+b10.2	b20.1	b20.2	b20.1+b20.2	sum(b)
Total	a10.1+b10.1	a10.2+b10.2	a10.1+b10.1+a10.2+b10.2	a20.1+b20.1	a20.2+b20.2	a20.1+b20.1+a20.2+b20.2	sum(a)+sum(b)

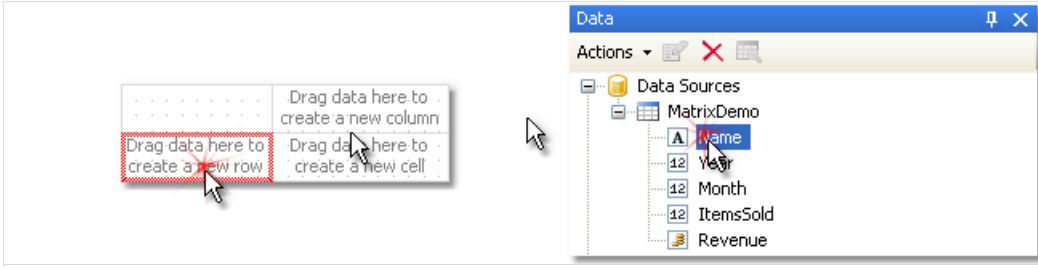
Bu rapor, önceki örnekte kullanılan aynı verilere dayanarak oluşturulmuştur. Şekilde gri renkle gösterilen satırlar otomatik olarak hesaplanmaktadır.

Matris Yapısının Ayarlanması

Rapor sayfasına yeni bir **"Matris"** nesnesi ekledikten sonra, aşağıdaki şekilde görünür:



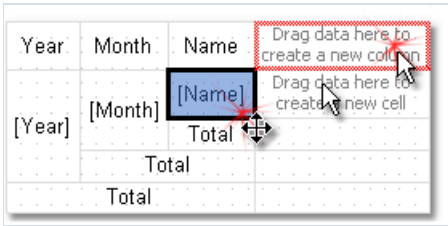
Matrisin yapısını ayarlamak için fareyi kullanabilirsiniz. Bunun için **"Veriler"** penceresinden veri kaynak alanlarını sürükleyip (drag&drop) matrise bırakın. Böylece satırlar, sütunlar ve



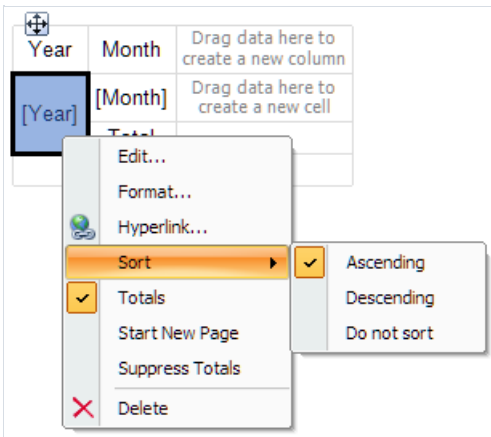
hücreler oluşturulur. Matris, yeni verilerin ekleneceği yeri kırmızı çerçeve ile vurgular:



Eğer matris içerisinde zaten öğeler (başlıklar, hücreler) varsa, yeni bir öğe eklenirken bir gösterge belirir. Bu durumda yeni alan, **Year** ve **Name** alanları arasında eklenecektir:



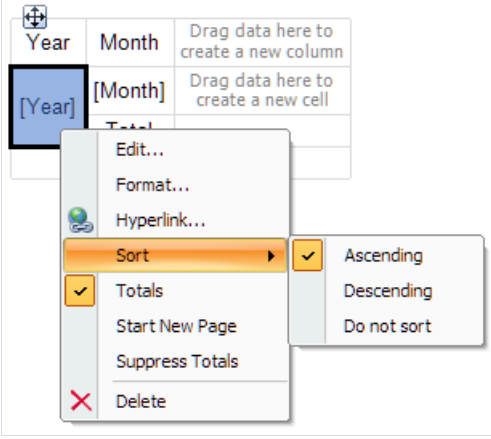
Mevcut öğelerin sırasını da değiştirebilirsiniz. Bunun için öğeye tıklayın ve çerçevesinden tutarak yeni konuma sürükleyin:



Bir öğeyi silmek için öğeyi fare ile seçin ve **Delete** tuşuna basın.

Başlık Ayarları

Bir başlık öğesini ayarlamak için onu seçin ve bağlam menüsünü görüntülemek için sağ tıklayın:



Varsayılan olarak, matris başlığındaki veriler artan sıraya göre sıralanır. Sıralama düzenini değiştirmek için menüden "**Sıralama**" seçeneğini seçebilirsiniz.

Matrise eklediğiniz her alana FastReport bir toplam (üzerinde "**Toplam**" yazan hücre) ekler. Toplamı kaldırmak için onu seçin ve **Delete** tuşuna basın. Toplamı yeniden etkinleştirmek için ilgili öğeyi seçin ve bağlam menüsünden "**Toplamlar**" seçeneğini seçin.

Menüdeki "**Yeni sayfa oluştur**" seçeneği, matris yazdırılırken bir sayfa sonu eklemenizi sağlar. Yukarıdaki şekilde gösterildiği gibi **Year** hücresinde sayfa sonu etkinleştirilirse, matris yazdırıldığında her yıl ayrı bir sayfada yer alacaktır.

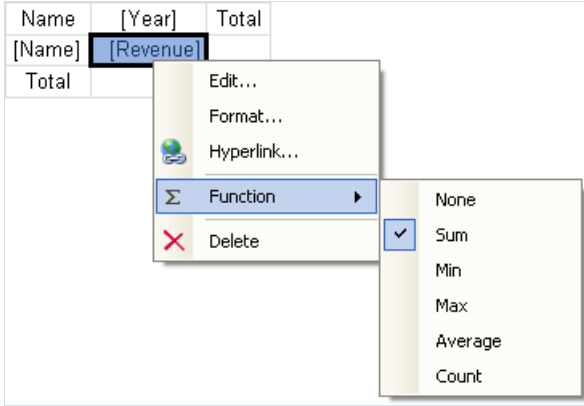
Menüdeki "**Toplamları bastır**" seçeneği, toplamın hesaplandığı grupta yalnızca bir değer olması durumunda toplam satırını gizlemenizi sağlar.

Matris Hücrelerinin Ayarlanması

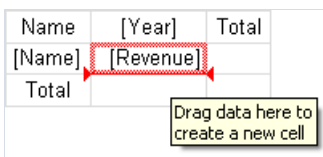
Matris hücreleri için, toplamların hesaplanmasında kullanılacak bir fonksiyon seçilebilir. Bu fonksiyonun değeri, toplam satırlarında ve sütunlarında görüntülenir. Aşağıda kullanılacak fonksiyonların listesi verilmiştir:

Fonksiyon	Açıklama
Yok	Hücrelerin değerleri işlenmiyor.
Toplam	Matristeki hücrelerin değerlerinin toplamını döndürür.
Minimum	Değerlerin en küçüğünü döndürür.
Maksimum	Değerlerin en büyüğünü döndürür.
Ortalama	Ortalama değeri döndürür.
Miktar	Boş olmayan değerlerin sayısını döndürür.
Farklı değerlerin sayısını döndürür.	Farklı (benzersiz) değerlerin sayısını döndürür.

Matris'e eklenen hücreler varsayılan olarak toplama fonksiyonunu kullanır. Başka bir fonksiyon seçmek için hücreyi seçip bağlam menüsünden **"Fonksiyon"** seçeneğini kullanabilirsiniz:



- **"Yok"** fonksiyonunu seçerek, bu hücre için herhangi bir toplam hesaplaması yapmaktan kaçınabilirsiniz.
- Matris içinde bir veya birden fazla veri hücresi bulunabilir. Birden fazla hücre varsa, bunlar yan yana veya alt alta yerleştirilebilir. Bu düzeni, bağlam menüsünden **"Hücreler tek satırda"** özelliğini değiştirerek ayarlayabilirsiniz.
- İkinci bir hücre eklerken, yerleşim sırasını seçebilirsiniz. Bunu belirlerken, kırmızı göstergeleri takip edin. Bu göstergeler, yeni değerın yukarı-aşağı mı yoksa sağa-sola mı ekleneceğini gösterir.



İkinci hücreyi ekledikten sonra, sonraki tüm değerler seçilen yönde eklenmeye devam edecektir.

Görünüm Ayarları

Bir matris hücresinin görünümünü değiştirmek için, gerekli hücreye tıklayın. Araç çubuğu yardımıyla yazı tipi, kenarlık ve dolgu ayarlarını yapabilirsiniz. Birden fazla hücrenin görünümünü aynı anda değiştirmek için, hücre grubunu seçin. Bunu yapmak için, sol üst hücreyi seçin ve fareyi bırakmadan, grup seçenecek şekilde fareyi hareket ettirin:

Name	[Year]	Total
[Name]	[Month]	Total
[Name]	[Revenue]	
Total		

Tüm matrisin görünümünü değiştirmek için stilleri kullanabilirsiniz. Bunu yapmak için, "Matrix" nesnesinin bağlam menüsünü açın ve stili seçin:

Satır ve Sütun Boyutlarını Yönetme

Matris, "Tablo" nesnesinin bir türü olduğundan satır ve sütun boyutları benzer yöntemlerle ayarlanabilir.

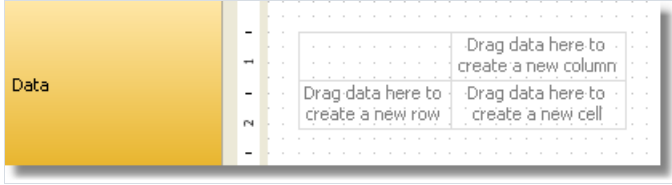
Matris varsayılan olarak "Otomatik Boyut" modundadır. Bu modda, nesne içindeki verilere göre sütun genişlikleri ve satır yükseklikleri otomatik olarak belirlenir. İsterseniz nesnenin boyutlarını elle de yönetebilirsiniz. Bunun için matrisin "Otomatik Boyut" özelliğini devre dışı bırakmanız gerekir. Satırlar ve sütunlar da aynı özelliğe sahiptir; ancak matrisin otomatik boyutlandırması kapalı olduğunda bu özellik kullanılabilir.

Bir satırın minimum ve maksimum yüksekliğini sınırlandırmak için ilgili satırı seçin ve "Özellikler" penceresinde **MinHeight** ve **MaxHeight** özelliklerine değer girin.

Bir sütunun minimum ve maksimum genişliğini sınırlandırmak için ilgili sütunu seçin ve "Özellikler" penceresinde **MinWidth** ve **MaxWidth** özelliklerine değer girin.

Matris

"Matris" nesnesinin kullanım örneklerini inceleyelim. Başlamak için yeni bir rapor oluşturun ve "Matris" nesnesini raporun "Rapor Başlığı" bandına yerleştirin (veya "Veriler" bandına yerleştirin; bu durumda bandı veri kaynağına bağlamanız gerekmez). Bu durumda matrisin belirtilen iki banttı hangisine yerleştirildiği önemli değildir, çünkü rapor çalıştırıldığında her iki bant da yalnızca bir kez yazdırılacaktır. Rapor şu şekilde görünecektir:



Matris nesnesini "Sayfa Başlığı", "Sayfa Alt Bilgisi" gibi her yeni sayfada yazdırılan bantlara yerleştirmeyin. Bu durumda, matris her bant yazdırıldığında tekrar oluşturulur ve bu durum yığın taşmasına ve programın beklenmedik şekilde kapanmasına neden olur.

Örneklerin çoğunda FastReport ile birlikte gelen MatrixDemo tablosu kullanılacaktır. Bu tablo aşağıdaki verileri içermektedir:

Name	Year	Month	ItemsSold	Revenue
Nancy Davolio	1999	2	1	1000
Nancy Davolio	1999	11	1	1100
Nancy Davolio	1999	12	1	1200
Nancy Davolio	2000	1	1	1300
Nancy Davolio	2000	2	2	1400
Nancy Davolio	2001	2	2	1500
Nancy Davolio	2001	3	2	1600
Nancy Davolio	2002	1	2	1700
Andrew Fuller	2002	1	2	1800
Andrew Fuller	1999	10	2	1900
Andrew Fuller	1999	11	2	2000
Andrew Fuller	2000	2	2	2100
Janet Leverling	1999	10	3	3000
Janet Leverling	1999	11	3	3100

Janet Leverling	2000	3	3	3200
Steven Buchanan	2001	1	3	4000
Steven Buchanan	2001	2	4	4100
Steven Buchanan	2000	1	4	3999

Örnek 1. Basit Matris

Matris, satır ve sütunda bir değer ve tek bir veri hücresi içerecektir. Matris oluşturmak için, MatrixDemo veri kaynağı alanlarını aşağıdaki şekilde ekleyin:

- **Year** alanını satır başlığına ekleyin;
- **Name** alanını sütun başlığına ekleyin;
- **Revenue** alanını matris hücresine ekleyin.

Bundan sonra matris aşağıdaki gibi görünecektir:

Year	[Name]	Total
[Year]	[Revenue]	
Total		

Matrisin görünümünü iyileştirelim:

- Matris için "Turuncu" stilini seçin;
- Tüm matris hücreleri için "Tahoma, 8" yazı tipini seçin;
- "Toplam" kelimesini kalın yazı tipiyle vurgulayın;
- Üst satırdaki hücreler için "Cam" tipi doldurma seçeneğini tercih edin;
- Matrisin otomatik boyutlandırmasını devre dışı bırakın ve satır ile sütunların boyutunu artırın.

Bundan sonra matris aşağıdaki görünüme sahip olacaktır:

Year	[Name]	Total
[Year]	[Revenue]	
Total		

Raporu çalıştırın ve aşağıdaki sonucu göreceksiniz:

Year	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	Total
1999	3900	6100	3300		13300
2000	2100	3200	2700	3999	11999
2001			3100	8100	11200
2002	1800		1700		3500
Total	7800	9300	10800	12099	39999

Matrisde aşağıdaki eksiklikler var:

- **Name** alanı için bir başlık yok;
- Toplamlar para biçiminde yazdırılmıyor.

Name alanına başlık eklemek için aşağıdaki yöntemler kullanılabilir:

- Matrisin sol üst köşesine "Yıl/Çalışan" metnini yerleştirin;
- Ayrıca, oraya çapraz bir çizgi ve ikinci bir "Metin" nesnesi ekleyebilirsiniz (aşağıdaki resimde gösterildiği gibi):

Employee	[Name]	Total
Year	[Revenue]	
Total		

Matris başlığını etkinleştirin. Bunu yapmak için, "Matrix" nesnesinin bağlam menüsünden "Başlığı Göster" öğesini seçin. Başlığa herhangi bir metin eklenebilir:

Employee		
Year	[Name]	Total
[Year]	[Revenue]	
Total		

Veri biçimlendirmesini ayarlamak için, aşağıdaki şekilde tüm hücre alanını seçin ve bağlam menüsünden "Biçim..." ögesini seçerek formatı ayarlayın:

Employee		
Year	[Name]	Total
[Year]	[Revenue]	
Total		

Bundan sonra hazırlanmış rapor şu şekilde olacaktır:

Employee					
Year	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	Total
1999	\$3,900.00	\$6,100.00	\$3,300.00		\$13,300.00
2000	\$2,100.00	\$3,200.00	\$2,700.00	\$3,999.00	\$11,999.00
2001			\$3,100.00	\$8,100.00	\$11,200.00
2002	\$1,800.00		\$1,700.00		\$3,500.00
Total	\$7,800.00	\$9,300.00	\$10,800.00	\$12,099.00	\$39,999.00

Örnek 2. Çok seviyeli başlıklar

Matris, satırda bir değer, sütunda iki değer ve bir veri hücresi içerecektir. Önceki örneği esas alarak bir alan ekleyelim:

Year alanının sağındaki satır başlığına **Month** alanını ekleyin.

Yeni alanı ekledikten sonra, matrisin görünümünü düzenleyin. Ayrıca hücreler için biçimlendirmeyi tekrar belirtmek gerekmektedir. Bunun ardından, matris aşağıdaki gibi görünecektir:

		Employee	
Year	Month	[Name]	Total
[Year]	[Month]	[Revenue]	
	Total		
Total			

Raporu çalıştırın ve aşağıdaki sonucu göreceksiniz:

		Employee				
Year	Month	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	Total
1999	2			\$1,000.00		\$1,000.00
	10	\$1,900.00	\$3,000.00			\$4,900.00
	11	\$2,000.00	\$3,100.00	\$1,100.00		\$6,200.00
	12			\$1,200.00		\$1,200.00
	Total	\$3,900.00	\$6,100.00	\$3,300.00	\$0.00	\$13,300.00
2000	1			\$1,300.00	\$3,999.00	\$5,299.00
	2	\$2,100.00		\$1,400.00		\$3,500.00
	3		\$3,200.00			\$3,200.00
	Total	\$2,100.00	\$3,200.00	\$2,700.00	\$3,999.00	\$11,999.00
2001	1				\$4,000.00	\$4,000.00
	2			\$1,500.00	\$4,100.00	\$5,600.00
	3			\$1,600.00		\$1,600.00
	Total	\$0.00	\$0.00	\$3,100.00	\$8,100.00	\$11,200.00
2002	1	\$1,800.00		\$1,700.00		\$3,500.00
	Total	\$1,800.00	\$0.00	\$1,700.00	\$0.00	\$3,500.00
Total		\$7,800.00	\$9,300.00	\$10,800.00	\$12,099.00	\$39,999.00

Örnek 3. Ay numarası yerine ay adının yazdırılması

Önceki örnekte, matris ay numaralarını yazdırıyordu. Bu durum, veri kaynağında Month alanının ay adını değil, ay numarasını içermesinden kaynaklanıyordu. Ay adını yazdırmak için aşağıdakileri yapın:

- Ay numarasının yazdırıldığı hücreyi seçin. Bizim örneğimizde bu, Cell8 adlı hücredir;
- Özellikler penceresinde ilgili butona tıklayın ve BeforePrint olayına çift tıklayın;
- FastReport, raporun betiğine boş bir olay işleyicisi ekleyecektir. Aşağıdaki kodu yazın:

```
private void Cell8_BeforePrint(object sender, EventArgs e)
{
    string[] monthNames = new string[] {
        "Ocak", "Şubat", "Mart", "Nisan", "Mayıs", "Haziran",
        "Temmuz", "Ağustos", "Eylül", "Ekim", "Kasım", "Aralık" };
    // Cell8, ay numarasını yazdıran hücredir.
    // Cell8.Value, hücrede yazdırılan değerdir (ay numarası).
    // Bu değer System.Object türündedir, bu yüzden int'e dönüştürülmelidir.
    Cell8.Text = monthNames[(int)Cell8.Value - 1];
}
```

Raporu çalıştırdığınızda aşağıdaki sonucu görürsünüz:

Year	Month	Andrew Fuller	Janet Leverli
1999	February		
	October	\$1,900.00	\$2,000.00
	November	\$2,000.00	
	December		
	Total	\$3,900.00	
2000	January		
	February		
	March		

Örnek 4. Koşullu Vurgulama

Matris hücreleri için, "Metin" nesnesinde olduğu gibi koşullu vurgulama ayarlamak mümkündür. Bu konu hakkında daha fazla bilgiyi "Koşullu Vurgulama" bölümünde okuyabilirsiniz.

Örnek 2 üzerinden, 3000'den büyük olan toplamların nasıl kırmızı renkle vurgulanacağını inceleyelim. Bunun için "Revenue" metnine sahip hücreyi seçin ve araç çubuğundaki "Metin" düğmesine basın. Koşullar düzenleyicisinde aşağıdaki vurgulama koşulunu belirleyin:

```
Value > 3000
```

ve metin rengini kırmızı olarak ayarlayın. Hazır rapor şu şekilde görünecektir:

		Employee				
Year	Month	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	Total
1999	2			\$1,000.00		\$1,000.00
	10	\$1,900.00	\$3,000.00			\$4,900.00
	11	\$2,000.00	\$3,100.00	\$1,100.00		\$6,200.00
	12			\$1,200.00		\$1,200.00
	Total	\$3,900.00	\$6,100.00	\$3,300.00	\$0.00	\$13,300.00
2000	1			\$1,300.00	\$3,999.00	\$5,299.00
	2	\$2,100.00		\$1,400.00		\$3,500.00
	3		\$3,200.00			\$3,200.00
	Total	\$2,100.00	\$3,200.00	\$2,700.00	\$3,999.00	\$11,999.00
2001	1				\$4,000.00	\$4,000.00
	2			\$1,500.00	\$4,100.00	\$5,600.00
	3			\$1,600.00		\$1,600.00
	Total	\$0.00	\$0.00	\$3,100.00	\$8,100.00	\$11,200.00
2002	1	\$1,800.00		\$1,700.00		\$3,500.00
	Total	\$1,800.00	\$0.00	\$1,700.00	\$0.00	\$3,500.00
Total		\$7,800.00	\$9,300.00	\$10,800.00	\$12,099.00	\$39,999.00

Görüldüğü üzere, toplam değerler vurgulanmamaktadır. Bunun nedeni, yalnızca bir hücre için vurgulama koşulu belirlemiş olmamızdır. Diğer değerleri vurgulamak için, çizimde gösterildiği gibi matrisin tüm hücreleri için koşul belirlenmelidir.

Bu örnekte, hücrenin kendi değerine bağlı koşullu vurgulama kullanılmıştır. Ayrıca, matris başlığındaki değerlere bağlı olarak bir hücreyi vurgulamak da mümkündür. Aşağıdaki örnekte, 2000 yılına ait tüm toplamların nasıl vurgulanacağını göstereceğiz. Bunun için, çizimde gösterildiği gibi matris hücrelerini seçin:

		Employee	
Year	Month	[Name]	Total
[Year]	[Month]	[Revenue]	
	Total		
Total			

Hücreler için aşağıdaki vurgulama koşulunu belirleyin:

```
(int)Matrix1.RowValues[0] == 2000
```

Bu durumda "Matrix1", bizim matrisimizin adıdır. Matrisin `RowValues` özelliği `object[]` türündedir ve mevcut yazdırılan satır için satır başlığındaki değerleri içeren bir dizi içerir. Dizide, başlık seviyeleri kadar değer bulunur.

Örneğimizde dizide 2 değer vardır; birincisi yıl, ikincisi ay.

Son satırı vurgulamayın. Çünkü onun için **RowValues** özelliği tanımsız bir değere sahiptir ve rapor oluşturma sürecinde hata verecektir.

Raporu çalıştırırsanız, aşağıdaki sonucu alırsınız:

		Employee				
Year	Month	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	Total
1999	2			\$1,000.00		\$1,000.00
	10	\$1,900.00	\$3,000.00			\$4,900.00
	11	\$2,000.00	\$3,100.00	\$1,100.00		\$6,200.00
	12			\$1,200.00		\$1,200.00
	Total	\$3,900.00	\$6,100.00	\$3,300.00	\$0.00	\$13,300.00
2000	1			\$1,300.00	\$3,999.00	\$5,299.00
	2	\$2,100.00		\$1,400.00		\$3,500.00
	3		\$3,200.00			\$3,200.00
	Total	\$2,100.00	\$3,200.00	\$2,700.00	\$3,999.00	\$11,999.00
2001	1				\$4,000.00	\$4,000.00
	2			\$1,500.00	\$4,100.00	\$5,600.00
	3			\$1,600.00		\$1,600.00
	Total	\$0.00	\$0.00	\$3,100.00	\$8,100.00	\$11,200.00
2002	1	\$1,800.00		\$1,700.00		\$3,500.00
	Total	\$1,800.00	\$0.00	\$1,700.00	\$0.00	\$3,500.00
Total		\$7,800.00	\$9,300.00	\$10,800.00	\$12,099.00	\$39,999.00

Ayrıca, sütun değerlerine erişmek için matrisin **ColumnValues** özelliğini kullanabilirsiniz.

Örnek 5. Çift Satırların Vurgulanması

Matrisin görünümünü iyileştirmek amacıyla, çift satırları ya da sütunları farklı bir renkle vurgulamak mümkündür. Bunu, Örnek 2 üzerinden nasıl yapacağımızı göstereceğiz.

Matrisin tüm veri alanını, resimde gösterildiği gibi seçin:

		Employee	
Year	Month	[Name]	Total
[Year]	[Month]	[Revenue]	
	Total		
Total			

Koşullu biçimlendirme düzenleyicisini çağırın. Koşul olarak aşağıdakini belirtin:

```
Matrix1.RowIndex % 2 != 0
```

ve mevcut olandan biraz daha koyu bir arka plan rengi belirtin. Bu örnekte, Matrix1 matrisimizin adıdır; RowIndex özelliği, mevcut basılan satırın numarasını döndürür.

Sütunların vurgulanması için, benzer şekilde matrisin `ColumnIndex` özelliğini kullanın.

Raporu çalıştırırsak, aşağıdakileri görürüz:

		Employee				
Year	Month	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	Total
1999	2			\$1,000.00		\$1,000.00
	10	\$1,900.00	\$3,000.00			\$4,900.00
	11	\$2,000.00	\$3,100.00	\$1,100.00		\$6,200.00
	12			\$1,200.00		\$1,200.00
	Total	\$3,900.00	\$6,100.00	\$3,300.00	\$0.00	\$13,300.00
2000	1			\$1,300.00	\$3,999.00	\$5,299.00
	2	\$2,100.00		\$1,400.00		\$3,500.00
	3		\$3,200.00			\$3,200.00
	Total	\$2,100.00	\$3,200.00	\$2,700.00	\$3,999.00	\$11,999.00
2001	1				\$4,000.00	\$4,000.00
	2			\$1,500.00	\$4,100.00	\$5,600.00
	3			\$1,600.00		\$1,600.00
	Total	\$0.00	\$0.00	\$3,100.00	\$8,100.00	\$11,200.00
2002	1	\$1,800.00		\$1,700.00		\$3,500.00
	Total	\$1,800.00	\$0.00	\$1,700.00	\$0.00	\$3,500.00
Total		\$7,800.00	\$9,300.00	\$10,800.00	\$12,099.00	\$39,999.00

Örnek 6. İfade Kullanımı

Önceki örneklerde, “Veriler” penceresindeki alanları matrise sürükleyerek başlıklar ve hücreler oluşturduk. Bu amaçla ifadeleri de kullanabilirsiniz. Bir ifadeyi matrise eklemek için aşağıdakileri yapın:

- Matrise “Veriler” penceresinden herhangi bir öge ekleyin. Bu, sistem değişkeni Date gibi herhangi bir öge olabilir; \
- Ögeye çift tıklayın ve İfade Düzenleyici penceresinde gerekli ifadeyi girin.

Eğer matrisiniz veri alanları yerine ifadeler içeriyorsa, matrisin DataSource özelliğinin doğru şekilde doldurulduğundan emin olun. Veri alanları ile çalışırken, bu özellik bir alan matrise sürüklendiğinde otomatik olarak doldurulur.

İfadelerin kullanımını bir örnek üzerinden inceleyelim. Bunun için veri kaynağı olarak, satışa sunulan ürünlerin çalışanlara göre gruplanmış listesini içeren Order Details tablosunu kullanacağız. Tablo, birkaç ana ilişki içerdiği için çalışan adını, ürün adını ve ürün kategorisini elde etmek mümkündür.

Matrisimiz, her çalışanın satışlarını ürün kategorilerine göre gruplayarak gösterecek. Matrisin oluşturulması için şunları yapın:

- Sütun başlığına `[Order Details.Products.Categories.CategoryName]` alanını ekleyin;
- Satır başlığına, matris ögesi oluşturmak için herhangi bir alan ekleyin. Ardından, başlık ögesi için aşağıdaki ifadeyi belirtin:

```
[Order Details.Orders.Employees.FirstName] + " " + [Order Details.Orders.Employees.LastName]
```

- Veri hücresine, matris ögesi oluşturmak için herhangi bir alan ekleyin. Ardından, hücre için aşağıdaki ifadeyi belirtin:

```
[Order Details.UnitPrice] * [Order Details.Quantity] * (decimal)(1 - [Order Details.Discount])
```

Peki, neden çalışan adı için bu kadar uzun bir veri alanı kullandık, oysa adı doğrudan Employees.FirstName alanından alabilirsiniz? Çünkü matris, Order Details veri kaynağına bağlıdır. Bu veri kaynağının diğer tablolarla olan ilişkilerini kullanarak onların alanlarına erişim sağlayabilirsiniz (ilişkiler hakkında detaylı bilgi için “Veriler” bölümüne bakınız). Eğer örneğimizde doğrudan Employees.FirstName alanına başvurursak, ilk çalışanın adını elde ederiz.

Matrisin düzenlemesini ayarlayın. Sonrasında, matris aşağıdaki şekilde görünecektir:

Employee	[CategoryName]	Total
[Order Details.Orders.Employees.FirstName] + " " + [Order Details.Orders.Employees.LastName]	[[Order Details.UnitPrice] * [Order Details.Quantity] * (decimal)(1 - [Order Details.Discount])]	
Total		

Raporu çalıştırdığınızda, 2 sayfayı kaplayan oldukça büyük bir matris göreceksiniz:

Employee	Beverages	Condiments	Confections	Dairy Products	Grains/Cereals
Andrew Fuller	40 248,25	14 850,67	21 455,69	23 812,55	11 111,11
Anne Dodsworth	19 642,56	10 125,55	8 053,16	21 101,13	1 234,56
Janet Leverling	44 757,41	13 381,64	33 622,40	32 320,84	2 345,67
Laura Callahan	17 897,85	14 637,66	21 699,91	21 269,47	1 345,67
Margaret Peacock	50 308,21	23 314,87	27 768,73	35 980,12	1 234,56
Michael Suyama	9 450,20	4 648,47	6 859,63	12 345,67	1 234,56
Nancy Davolio	46 599,36	13 561,56	28 568,21	32 320,84	1 234,56
Robert King	27 963,83	8 851,38	12 345,67	21 101,13	1 234,56
Steven Buchanan	11 000,53	12 345,67	12 345,67	12 345,67	1 234,56

Örnek 7. Hücrelerde Resimler

Matris hücreleri "Metin" nesnesinden türetilmiştir ve metin verilerini gösterebilir. Eğer bu yeterli gelmiyorsa, hücreye herhangi bir nesne ekleyebilirsiniz. Şimdi, matris içerisinde bir resmi nasıl göstereceğimizi inceleyelim.

Bunun için, örnek 6'yı temel alacağız. Başlık hücrelerine çalışan fotoğrafını (`Employees.Photo` alanı) ve kategori resmini (`Categories.Picture` alanı) ekleyelim. Aşağıdaki adımları izleyin:

1. Çalışan ismi ve kategori adı bulunan hücrelerin boyutunu büyütün.
2. Her bir hücreye birer "Resim" nesnesi ekleyin.
3. Çalışan fotoğrafını göstermek için, "Resim" nesnesi için aşağıdaki veri alanını belirtin (bu ayarı nesne düzenleyicisinde yapabilirsiniz):

```
Order.Details.Orders.Employees.Photo
```

4. Kategori resmini göstermek için, "Resim" nesnesi için aşağıdaki veri alanını belirtin:

```
Order.Details.Products.Categories.Picture
```

Bunun ardından matris aşağıdaki gibi görünecektir:

Employee		[CategoryName]		Total
[Order]		[[Order.Details.UnitPrice] * [Order.Details.Quantity] * (decimal)(1 - [Order Details.Discount])]]		
Total				

Raporu çalıştırdığınızda, aşağıdakini göreceksiniz:

Employee	Beverages	Condiments	Confections
Andrew Fuller 	40 248,25	14 850,67	
Anne Dodsworth 	19 642,56	10 125,6	
Janet Leverling 	44		

Örnek 8. Hücrelerdeki Nesneler

Matris hücrelerine entegre edilen nesneleri kullanarak çeşitli görsel efektler elde edilebilir. Aşağıdaki örnekte, bir çalışanın satış seviyesini gösteren basit bir ölçeğin nasıl çizileceği gösterilmektedir. \

Matris, MatrixDemo veri kaynağını kullanacaktır. Matrisi oluşturmak için, MatrixDemo veri kaynağındaki alanları aşağıdaki şekilde ekleyin:

- **Year** alanını satır başlığına ekleyin;
- **Name** alanını sütun başlığına ekleyin;
- **Revenue** alanını matris hücresine ekleyin.

Matrisin görünümünü aşağıdaki gibi ayarlayın:

Employee	[Year]	Total
[Name]	[Revenue]	
Total		

Şimdi **Revenue** değeri bulunan hücreye 3 adet "Şekil" nesnesi ekleyeceğiz. Bu nesneler gösterge işlevi görecektir. Şekilde ayarlanacaktır:

- Hücredeki değer 100'den küçükse, bir adet kırmızı nesne gösterilir;
- Hücredeki değer 3000'den küçükse, iki adet sarı nesne gösterilir;
- Hücredeki değer 3000 veya daha büyükse, üç adet yeşil nesne gösterilir.

Matris artık şu şekilde görünecektir:

Employee	[Year]	Total
[Name]	■■■ [Revenue]	
Total		

Nesnelerin durumunu kontrol etmek için, matris hücresinin BeforePrint olayına bir işleyici oluşturacağız. Bunun için Revenue hücresini seçin, "Özellikler" penceresinde ilgili butona tıklayın ve BeforePrint olayına çift tıklayın. FastReport boş bir olay işleyicisi oluşturacak ve "Kod" sekmesine geçecektir. İşleyiciye aşağıdaki kodu yazın:

```

private void Cell4_BeforePrint(object sender, EventArgs e)
{
    // Bu örnekte hücre adı Cell4'tür.
    // Cell4.Value özelliğinde bulunan hücre değerini alıyoruz.
    // Matrisimizde bazı hücreler boş olabilir, bunu göz önünde bulundurun (null kontrolü).
    // Değeri decimal türüne dönüştürmemiz gerekiyor, çünkü [MatrixDemo.Revenue] veri alanı
    // System.Decimal türündedir.
    decimal value = Cell4.Value == null ? 0 : (decimal)Cell4.Value;

    // Shape1..3 - bunlar "Şekil" nesneleridir.
    // Değere bağlı olarak nesneleri etkinleştirip devre dışı bırakıyoruz:
    // Değer < 100 ise bir nesne; < 3000 ise iki nesne; >= 3000 ise üç nesne.
    Shape1.Visible = true;
    Shape2.Visible = value >= 100;
    Shape3.Visible = value >= 3000;

    // Renk seçimi: 100'den küçükse kırmızı;
    // 3000'den küçükse sarı;
    // 3000 veya daha büyükse yeşil.
    Color color = Color.Red;
    if (value >= 100)
        color = Color.Yellow;
    if (value >= 3000)
        color = Color.GreenYellow;

    // Tüm üç nesne için rengi ayarlıyoruz.
    Shape1.Fill = new SolidFill(color);
    Shape2.Fill = new SolidFill(color);
    Shape3.Fill = new SolidFill(color);
}

```

Raporu çalıştırırsanız, yukarıdaki gibi bir görünüm elde edeceksiniz:

Employee	1999	2000	2001	2002	Total
Andrew Fuller	3 900,00	2 100,00		1 800,00	7 800,00
Janet Leverling	6 100,00	3 200,00			9 300,00
Nancy Davolio	3 300,00	2 700,00	3 100,00	1 700,00	10 800,00
Steven Buchanan		3 999,00	8 100,00		12 099,00
Total	13 300,00	11 999,00	11 200,00	3 500,00	39 999,00

Örnek 9. Matrisi Elle Doldurma

İncelenen tüm örneklerde, matris verilerle otomatik olarak dolduruluyordu çünkü veri kaynağına bağlıydı. Matris için veri kaynağı, `DataSource` özelliği ile belirtilir. Bu özelliğin değerini manuel olarak ayarlamamış olsak da, veri alanı matrise eklendiğinde bu durum dolaylı olarak gerçekleşmiş olur.

Script kullanarak matrisi elle doldurmak da mümkündür. Bunun için, matrisin `ManualBuild` olay işleyicisini oluşturmanız gerekir. Olay işleyicisi kodunda, değeri eklemek amacıyla `AddValue` metodunu çağırmanız. Aşağıda, diyagonal üzerinde yer alan değerlerle 10×10'luk bir tablo yazdıracak bir matrisi nasıl oluşturacağımız gösterilmektedir:

	1	2	3	...
1	1			
2		2		
3			3	
...				...

Yapılması Gereken İşlemler:

- Raporunuza boş bir matris ekleyin.
- Matrisin satır, sütun ve hücre alanlarına "Veriler" penceresinden herhangi bir öge yerleştirin. Ardından, matris ögesine çift tıklayarak ifade editörünü açın ve ifadenin tamamını temizleyin.
- Matrisin `DataSource` özelliğini temizleyin.

Bu işlemleri tamamladığınızda matris aşağıdaki gibi görünecektir:

		Total
Total		

Şimdi `ManualBuild` olay işleyicisini oluşturun. Bunun için matrisi seçin, "Özellikler" penceresinde ilgili düğmeye tıklayın ve `ManualBuild` olayına çift tıklayın. FastReport, boş bir olay işleyicisi oluşturacaktır. Oluşturulan olay işleyicisinde aşağıdaki kodu yazın:

```

private void Matrix1_ManualBuild(object sender, EventArgs e)
{
    // Matrisimizde, satır, sütun ve hücre verileri için 1 seviye bulunmaktadır.
    // Her ölçü için 1 eleman içeren 3 adet object[] dizisi oluşturalım.
    object[] columnValues = new object[1];
    object[] rowValues = new object[1];
    object[] cellValues = new object[1];

    for (int i = 1; i <= 10; i++)
    {
        // Dizileri dolduruyoruz
        columnValues[0] = i;
        rowValues[0] = i;
        cellValues[0] = i;

        // Verileri matrise ekliyoruz
        Matrix1.AddValue(columnValues, rowValues, cellValues);
    }
}

```

Bir işleyicide, "Matrix" nesnesini veriyle doldurmak için `AddValue` yöntemini kullanmalısınız. Bu yöntemin her biri `System.Object` türünde dizi olan üç parametresi vardır. İlk parametre sütun değeri, ikinci satır değeri, üçüncüsü ise hücre değeridir. Her dizideki değer sayısının nesnenin ayarlarına uygun olması gerektiğini unutmayın! Bizim durumumuzda nesnenin sütun, satır ve hücrelerinde birer seviye bulunmaktadır; dolayısıyla sütunlar için bir, satırlar için bir ve hücreler için bir değer veriyoruz.

Raporu çalıştırdığımızda aşağıdakini göreceğiz:

	1	2	3	4	5	6	7	8	9	10	Total
1	1										1
2		2									2
3			3								3
4				4							4
5					5						5
6						6					6
7							7				7
8								8			8
9									9		9
10										10	10
Total	1	2	3	4	5	6	7	8	9	10	55

Sütun 7 ile satır 3'ün kesişiminde matrise "21" değerini nasıl ekleyeceğimizi gösterelim. Bunun için kodu şu şekilde değiştirin:

```
private void Matrix1_ManualBuild(object sender, EventArgs e)
{
    object[] columnValues = new object[1];
    object[] rowValues = new object[1];
    object[] cellValues = new object[1];

    for (int i = 1; i <= 10; i++)
    {
        columnValues[0] = i;
        rowValues[0] = i;
        cellValues[0] = i;

        Matrix1.AddValue(columnValues, rowValues, cellValues);
    }
    columnValues[0] = 7;
    rowValues[0] = 3;
    cellValues[0] = 21;
    Matrix1.AddValue(columnValues, rowValues, cellValues);
}
```

Sonuç olarak aşağıdakini elde ederiz:

	1	2	3	4	5	6	7	8	9	10	Total
1	1										1
2		2									2
3			3				21				24
4				4							4
5					5						5
6						6					6
7							7				7
8								8			8
9									9		9
10										10	10
Total	1	2	3	4	5	6	28	8	9	10	76

Görüldüğü gibi, matris toplamaları otomatik olarak hesaplar.

Veriye bağlı olan matris için `ManualBuild` (Elle Oluştur) olay işleyicisini kullanabilirsiniz.

Bu durumda, önce olay işleyici çağrılır, ardından matris veri kaynağından gelen verilerle doldurulur.

Geliştirilmiş Matris Nesnesi

Bu nesne, "Matris" nesnesi gibi, özet raporlar oluşturmayı sağlar.

	Top 5 customers		Total
	Total	Others (84)	
► Beverages (12)	\$144 007,88	\$192 684,30	\$336 692,18
► Condiments (12)	\$123 407,20	\$105 665,89	\$229 073,09
► Confections (13)	\$54 823,13	\$112 615,09	\$167 438,23
► Dairy Products (10)	\$103 074,28	\$166 433,01	\$269 507,29
► Grains/Cereals (7)	\$31 368,01	\$64 376,58	\$95 744,59
▼ Meat/Poultry (6)	\$269 495,38	\$113 526,98	\$383 022,36
1. Alice Mutton	\$13 428,48	\$19 269,90	\$32 698,38
2. Mishi Kobe Niku	\$1 319,20	\$5 907,30	\$7 226,50
3. Pâté chinois	\$7 137,60	\$10 288,80	\$17 426,40
4. Perth Pasties	\$6 916,56	\$13 657,61	\$20 574,17
5. Thüringer Rostbratwurst	\$239 795,40	\$60 573,28	\$300 368,67
6. Tourtière	\$898,14	\$3 830,10	\$4 728,24
► Produce (5)	\$67 154,22	\$71 955,36	\$139 109,58
► Seafood (12)	\$31 732,55	\$99 591,19	\$131 323,74
Total	\$825 062,63	\$926 848,41	\$1 751 911,04

İşte temel özelliklerinin listesi:

- Satır ve sütun başlıkları, grupları ve basit öğeleri rastgele sırayla içerebilir. Bu, asimetrik raporlar oluşturmayı mümkün kılar;
- Katlama düğmeleri, belirli öğelerin görünürliğini interaktif olarak yönetmeyi sağlar;
- Sıralama düğmeleri, seçilen değerlere göre, ayrıca toplam değerlere göre de, matrisin interaktif olarak sıralanmasını sağlar;
- Top N gruplaması, başlıkta N değeri gösterme ve diğer değerleri açılabilir ayrı bir öge içinde gruplama olanağı sağlar;
- Matris başlıklarının basamaklı olarak görüntülenmesi;
- Toplam değerlerine göre başlıkların sıralanması;
- Geniş kapsamlı toplulaştırma fonksiyonları seti;
- Kullanıcı tanımlı toplulaştırma fonksiyonlarının desteği;
- Toplam değerler, komşu hücre değerleri elde etmeye olanak tanıyan ve yüzde hesaplaması için fonksiyonlar sunan geniş kapsamlı özel fonksiyon seti;
- Veri hücrelerinde "İskrografik" ve "İllerleme Göstergesi" nesnelerinin desteği.

Matris Yapısı

"Advanced Matrix" (Gelişmiş Matris) nesnesi aşağıdaki öğelerden oluşur:

Corner		Column header	
Employee	[Year]	Total	
		[Month]	
[Name]	[Sum(Revenue)]		[Sum(Revenue)]
Total	[Sum(Revenue)]		[Sum(Revenue)]
Row header		Data area	

Köşe

Matrisin köşesinde bulunan hücreler rastgele bilgiler içerebilir. Bu hücreleri istediğiniz gibi bölebilir veya birleştirebilirsiniz.

Başlık

Matris başlığı iki tür öge içerebilir:

- Basit öge: "Toplam" gibi statik bilgileri görüntüler.
- Grup: Belirli bir kritere göre gruplanmış değerler listesini görüntüler.

Başlık, ağaç yapısına sahiptir. Kök öge görünmezdir ve görünür birinci seviye öğeleri içerir.

Öğelerin herhangi bir düzeni mümkündür; örneğin, başlıkta grup olmayabilir veya birden fazla yan yana grup olabilir. Toplamlar da rastgele bir şekilde düzenlenebilir.

Tasarım modunda, matris başlık alanında görsel ipuçları görüntüler:

Group		Simple item	
Employee	[Year]	Total	
		[Month]	
[Name]	[Sum(Revenue)]		[Sum(Revenue)]
Total	[Sum(Revenue)]		[Sum(Revenue)]
Dashed line separates matrix areas		Select DB field button	

Bu durumda, başlık yapısı şu şekildedir:

```
Row header
- "Name" group
- "Total" element

Column header
- "Year" group
- "Month" group
- "Total" element
```

Daha fazla bilgi için [Başlık Yapılandırması](#) bölümüne bakınız.

Veri Alanı

Veri alanındaki hücreler genellikle bir agregat fonksiyonu içerir. Veri Alanı Yapılandırması bölümüne bakınız.

Matris Yapılandırması

Yapılandırma Ayarları

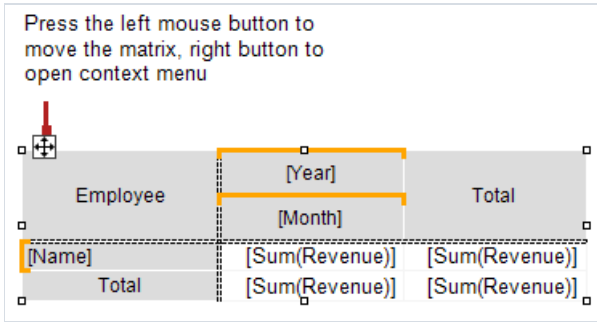
Matris yapılandırmasını yapmak için şu adımları izleyin:

1. Başlıkları yapılandırın ([Başlık Yapılandırması](#) bölümüne bakınız).
2. Veri hücrelerini yapılandırın ([Veri Alanı Yapılandırması](#) bölümüne bakınız).
3. Toplamları ekleyin ([Başlık Yapılandırması](#) bölümüne bakınız). Bu adım, yeni veri hücrelerini yapılandırırken zaman kazandırmak için en son yapılmalıdır.

Matris, veri kaynağına bağlanmış olmalıdır — **DataSource** özelliği bununla sorumludur. Genellikle, bu özellik başlık ve hücre yapılandırması sırasında otomatik olarak ayarlanır.

Bağlam Menüsü

Bağlam menüsünü açmak için, matrisin herhangi bir ögesini seçin, ardından matrisin sol üst köşesindeki alana sağ tıklayın:



Menüde aşağıdaki komutlar mevcuttur:

- **"Stil"** — mevcut stillerden birini seçmenizi sağlar;
- **"Sütunlar ve Satırları Değiştir"** — matris içinde sütunlar ile satırları hızlıca yer değiştirmenizi sağlar;
- **"Başlıkları Tekrarla"** — eğer matris birden fazla sayfaya yayılıyorsa, sütun ve satır başlıkları her sayfada yazdırılır.

"Özellikler" penceresinde kullanılabilen ayarlar

"Advanced Matrix" (Gelişmiş Matris) nesnesine özgü olarak, "Özellikler" penceresinde aşağıdaki özellikler mevcuttur:

Özellik	Değer	Açıklama
DataRowPriority	Rows	Veritabanı alanlarına veri hücreleri aracılığıyla erişimde başlıkların öncelik sırası. Ayrıntılar için "Veri Hücrelerinden Erişilebilen Özellikler" bölümüne bakınız.
DataSource		Veri kaynağı.
EvenStylePriority	Rows	EvenStyle özelliğinin uygulanmasında, numaraların öncelik sırası şu şekildedir: önce satır numarası değerlendirilir, eğer satır numarası belirlenmemişse sütun numarası esas alınır.

Filter		Veri filtreleme ifadesi. Bakınız: "Veri Filtreleme" bölümü.
Layout	AcrossThenDown	Bakınız: "Tablo Açılımı" bölümü.
PrintIfEmpty	True	Matris boşsa yazdır.
RepeatHeaders	True	Yeni sayfada başlıkları tekrar et.
ResetDataOnRun	False	Her rapor çalıştırıldığında verileri sıfırla. Varsayılan olarak, matris interaktif işlemler sırasında yeniden yapılandırılmaz (bkz. "İnteraktif Özellikler" bölümü).
Style		Matris stili.
WrappedGap	0	Layout = Wrapped modunda matris parçaları arasındaki boşluk.

Başlık bağlam menüsünü kullanarak.

Yeni ögeyi eklemek istediğiniz ögeyi seçin ve bağlam menüsünden **"Yeni öge ekle"** komutlarından birini seçin. Boş bir yeni öge eklenecektir;

Ayrıca, bağlam menüsünü kullanarak bir **"Toplam"** ögesi de ekleyebilirsiniz (seçilen ögeden önce veya sonra). Yeni bir öge, **"Toplam"** metniyle (metin, geçerli yerelleştirmeye bağlı olarak değişir) eklenir.

Bir toplam eklemek, boş bir öge eklemek ve metnini düzenlemekle eşdeğerdir.

"Veri" penceresinden bir veritabanı alanı boş bir matrise eklendiğinde, **DataSource** özelliği otomatik olarak yapılandırılır.

Öge Kaldırma

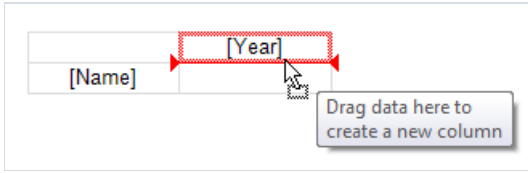
Bir ögeyi, bağlam menüsünde **"Sil"** ögesini seçerek silebilirsiniz. Yalnızca seçilen ögeyi veya öge ağacını (seçilen öge ve tüm alt ögeleri) silebilirsiniz.

Ayrıca, Delete tuşuna basarak bir ögeyi silebilirsiniz. Bu durumda, yalnızca seçilen öge silinir.

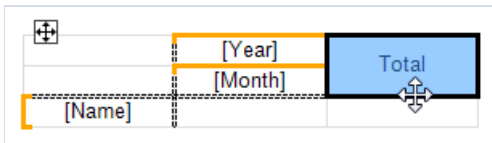
Kilitleme simgesine sahip bir öge, yukarıda açıklanan şekilde silinemez. **TopN Gruplama** bölümüne bakınız.

Bir Ögeyi Taşıma

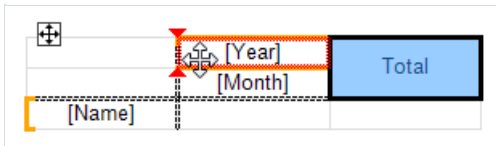
Bir ögeyi yeni bir konuma taşımak için, ögeyi sol fare tuşuyla seçin. Öge, kalın siyah bir çerçeve ile işaretlenecektir:



Ögeyi çerçevesinden tutun ve yeni bir konuma taşıyın. Sürüklerken, yeni bir ögenin başlığın hangi kısmına ekleneceği gösterilecektir:



Fare tuşunu bıraktığınızda, öge yeni bir konuma taşınacaktır:



Eğer ögenin alt ögeleri varsa, ana öge ile birlikte taşınacaklardır, eğer bu mümkünse.

Ögeleri Düzenleme

Öge düzenleyicisini çağırmak için, ögeye sol fare tuşuyla çift tıklayın veya bağlam menüsünden **"Düzenle..."** seçeneğini seçin. Ayrıca, düzenleyiciyi Enter tuşuna basarak da çağırabilirsiniz.

Gruplama

Daha önce açıklandığı gibi, matris başlığı iki tür öge içerebilir: **grup** ve **basit öge**. Ögenin türü, başlık düzenleyicisinde belirlenir:

The screenshot shows a dialog box with a 'General' tab. On the left is a sidebar with buttons: 'Sort', 'Filter', 'Top N', 'Visibility', and 'Other'. The main area has 'Item type:' with radio buttons for 'Simple' and 'Group' (selected). Below it is a text field containing '[MatrixDemo.Year]' with a dropdown arrow and a 'fx' icon. Further down is 'Display text:' with an empty text field and a 'fx' icon. At the bottom are 'OK' and 'Cancel' buttons.

- Basit bir öge bir gruba dönüştürülebilir veya tam tersi yapılabilir.
- Basit öge için düzenleyici penceresinde bazı ayarlar kullanılamaz.

Grup, belirli bir koşula göre gruplandırılmış değerler listesini görüntülemeyi sağlar. Yukarıdaki örnekte koşul olarak `[MatrixDemo.Year]` belirtilmiştir — bu, ögenin yıl listesini getireceği anlamına gelir. Liste tekrar eden değerleri içermez; aynı değerler birleştirilerek (gruplanarak) tek bir değer gibi gösterilir.

Varsayılan olarak grup, gruplama koşulunda belirtilen değerleri görüntüler. Örneğin, koşul `[MatrixDemo.Month]` olarak ayarlanırsa ay numaraları gösterilir. “**Görüntülenen metin**” özelliği sayesinde, isterseniz farklı bir değer gösterebilirsiniz. Örneğin:

```
[MonthName([MatrixDemo.Month])]
```

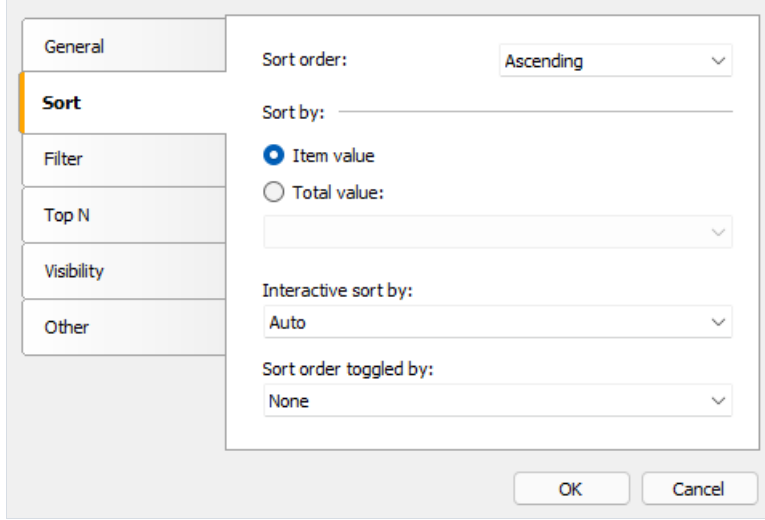
ay numarası yerine ayın adını gösterecektir.

Ayrıca bu özellikte, matrisin özel fonksiyonlarına başvurabilirsiniz (bkz. “Başlık hücrelerinden erişilebilen özellikler” bölümü). Örneğin:

```
[Matrix1.RowNo].[MatrixDemo.Name]
```

Sıralama

Gösterilen değerlerin sıralama ayarları, başlık düzenleyici üzerindeki aynı adlı sekmede sunulmaktadır. Ayarlar, öge tipi "Grup" seçildiğinde aktif hale gelir:



Ayarların Açıklaması:

"Sıralama Düzeni" – Sıralamanın artan, azalan ya da sıralama yapılmadan gerçekleştirileceğini belirler.

"Şuna Göre Sırala" – Gösterilen değerlere göre veya toplam değere göre olmak üzere iki sıralama seçeneğinden birini seçin. Aşağıdaki resimlerde "Yıl" ögesinin sıralanma şekli gösterilmiştir:

Öge değerine göre:

	2011	2012	2013	2014	2015
Andrew Fuller	\$3,900.00	\$2,100.00			\$1,800.00
Janet Leverling	\$6,100.00	\$3,200.00			
Nancy Davolio	\$3,300.00	\$2,700.00	\$3,100.00		\$1,700.00
Steven Buchanan			\$3,999.00	\$8,100.00	
Total	\$13,300.00	\$8,000.00	\$7,099.00	\$8,100.00	\$3,500.00

Toplam değere göre:

	2015	2013	2012	2014	2011
Andrew Fuller	\$1,800.00		\$2,100.00		\$3,900.00
Janet Leverling			\$3,200.00		\$6,100.00
Nancy Davolio	\$1,700.00	\$3,100.00	\$2,700.00		\$3,300.00
Steven Buchanan		\$3,999.00		\$8,100.00	
Total	\$3,500.00	\$7,099.00	\$8,000.00	\$8,100.00	\$13,300.00

"Interactive sort by total" — etkileşimli sıralama etkinse, başlık değerlerinin nasıl sıralanacağını tanımlar (bkz. Interactive sort). Seçenekler: None (Yok), Auto (Otomatik) ve toplamın adı.

"Sort order toggled by button" — burada, matrisin köşesinde veya dışında bulunan MatrixSortButton türündeki düğmenin adı belirtilir. Rapor görüntüleyicisinde bu düğmeye tıkladığınızda sıralama düzeni değiştirilir ve matris güncellenir.

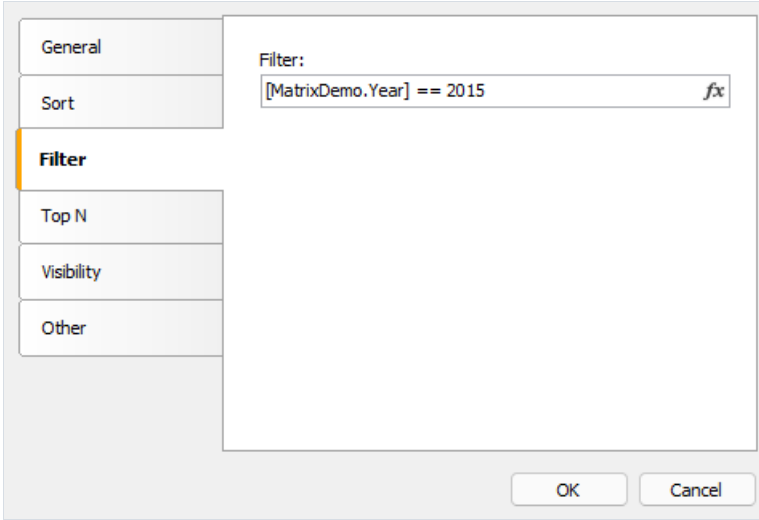
Veri Filtreleme

Matris içinde görüntülenen verileri filtrelemenin iki yolu vardır.

Yöntem 1: Matrisin **Filter** özelliğini kullanarak filtreleme. Bunun için **Matris** nesnesini seçin ve **Özellikler** penceresinde **Filter** özelliğini aşağıdaki gibi doldurun:

```
[MatrixDemo.Year] == 2015
```

Yöntem 2: Başlık düzenleyicisinde, benzer bir ifade kullanarak filtreleme:



Her iki durumda da filtreleme ifadesi **bool** (mantıksal) türünde bir değer döndürür.

Bu iki yöntem arasındaki fark, matrise hangi miktarda veri aktarıldığıyla ilgilidir. İlk yöntemde, yılın 2015'e eşit olduğu kayıtlar dışındaki tüm değerler elenir. Sonuç, aşağıdakine benzer şekilde görünebilir:

Çalıştırdığımızda, aşağıdakini göreceğiz:

	2015
Andrew Fuller	\$1,800.00
Nancy Davolio	\$1,700.00

İkinci yöntemde ise başlıktaki belirli bir öğenin değerleri filtrelenir. Önceki seçeneğin aksine, sonuç boş değerler içerebilir:

	2015
Andrew Fuller	\$1,800.00
Janet Leverling	
Nancy Davolio	\$1,700.00
Steven Buchanan	

TopN Gruplaması

Başlık grubundaki değer sayısı fazlaysa, bu durum aşırı sayıda rapor sayfası oluşturabilir. TopN gruplama, ilk N değeri görüntülenize ve kalan değerleri daraltılmış biçimde göstermenize olanak tanır:

	Top 5 customers						Total
	Lonesome Pine Restaurant	Rattlesnake Canyon Grocery	Save-a-lot Markets	QUICK-Stop	Ernst Handel	Others (84)	
► Beverages (12) ⇅	\$2 708,00	\$26 876,15	\$65 498,00	\$36 216,43	\$12 709,30	\$192 684,30	\$336 692,18
► Condiments (12) ⇅	\$82 480,00	\$6 075,20	\$11 567,00	\$9 214,94	\$14 070,06	\$105 665,89	\$229 073,09
► Confections (13) ⇅	\$549,00	\$10 947,21	\$11 981,07	\$18 530,09	\$12 815,76	\$112 615,09	\$167 438,23
► Dairy Products (10) ⇅	\$815,00	\$42 854,87	\$21 107,10	\$13 800,85	\$24 496,46	\$166 433,01	\$269 507,29
► Grains/Cereals (7) ⇅	\$190,00	\$4 831,31	\$8 298,10	\$5 310,90	\$12 737,70	\$64 376,58	\$95 744,59
► Meat/Poultry (6) ⇅	\$140 098,40	\$83 657,28	\$27 659,18	\$9 754,96	\$8 325,56	\$113 526,98	\$383 022,36
► Produce (5) ⇅	\$16 379,20	\$13 836,05	\$16 387,90	\$8 081,40	\$12 469,67	\$71 955,36	\$139 109,58
► Seafood (12) ⇅	\$625,00	\$884,73	\$13 604,60	\$9 367,74	\$7 250,48	\$99 591,19	\$131 323,74
Total	\$243 844,60	\$189 962,80	\$176 102,95	\$110 277,31	\$104 874,98	\$926 848,41	\$1 751 911,04

Nasıl çalışır

TopN fonksiyonu, verileri görüntülemek için dört öge kullanır:

1. TopN grubu, çok sayıda değeri içeren kaynak gruptur.
2. TopN toplamı, TopN değerlerinin toplamını gösterir.
3. "Diğerleri" grubu, TopN'a dahil olmayan değerleri gösterir.
4. "Diğerleri" grubunun sonucu, bu grubun toplam verisidir.

Eğer kaynak grubundaki değer sayısı **TopN.Count** özelliğinde belirtilenden azsa, grup normal şekilde görüntülenir, TopN gruplama uygulanmaz. Aksi takdirde aşağıdakiler gerçekleşir:

- Ana grupta N değer bırakılır,
- Kalan değerler "Diğerleri" grubuna aktarılır,
- Ana grup ve "Diğerleri" grubundaki veriler ayrı ayrı toplanır,
- Elde edilen değerler, TopN toplamı ve "Diğerleri" grubu toplamı olarak görüntülenir.

Kurulum

TopN, ana grup için yapılandırılır. Bunu yapmak için ögeye çift tıklayın veya sağ tıklama menüsünden "Düzenle..." seçeneğini seçin.

General
Sort
Filter
Top N
Visibility
Other

Display Top N values: 3
Additional items:
☒ Top N total Total
☐ Others
☒ Others total Others
☐ Create additional items in the matrix template for better customization
OK Cancel

Ek öğelerle çalışmak için iki seçenek vardır:

- "TopN toplamı", "Diğerleri" ve "Diğerleri toplamı" öğeleri, matris oluşturulurken otomatik olarak eklenir. Görsel tasarımları, ana öğeden kopyalanır. Bu öğelerin görünürlüğü yönetebilir ve toplam öğeler için metin belirtebilirsiniz. Görünümü özelleştirmek için başka bir seçenek yoktur.
- Yukarıdaki öğeler, matris şablonuna manuel olarak eklenebilir. Bu, görünümü tamamen özelleştirmenize ve öğelerin sırasını değiştirmenize olanak tanır. Ayrıca, her bir öğenin görünürlüğü etkileşimli olarak yönetmek için daraltma düğmeleri ekleyebilirsiniz.

Aşağıda, ek öğelerin şablona eklendiğinde matris şablonunun nasıl görüldüğü gösterilmektedir:

	Top 5 customers				Total
	Total	[CompanyName]	Others		
[CategoryName]	[Sum(Price)]	[Sum(Price)]	[Sum(Price)]	[Sum(Price)]	[Sum(Price)]
[ProductName]	[Sum(Price)]	[Sum(Price)]	[Sum(Price)]	[Sum(Price)]	[Sum(Price)]
Total	[Sum(Price)]	[Sum(Price)]	[Sum(Price)]	[Sum(Price)]	[Sum(Price)]

Bu durumda, ana grup kırmızı renkte vurgulanmış ve ek öğeler kilit simgesiyle işaretlenmiştir. Bu öğeleri, "Ek öğeler oluştur" onay kutusunu işaretini kaldırarak ana öğe düzenleyici penceresinde silebilirsiniz.

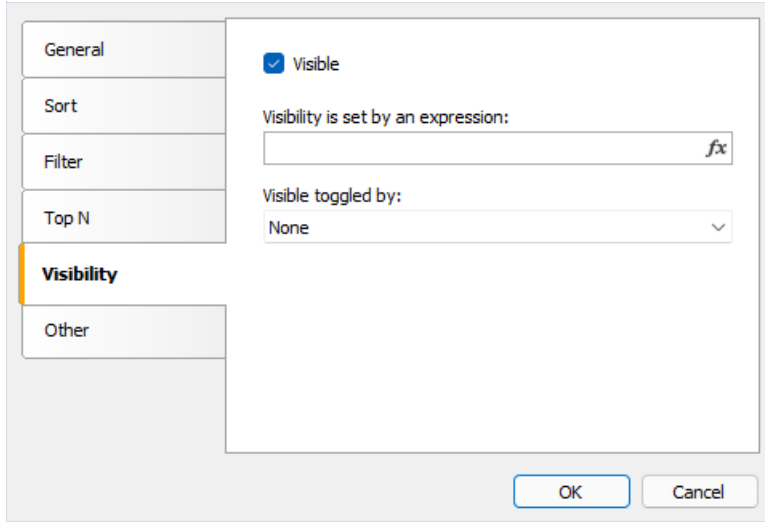
TopN, BottomN, FirstN, LastN

TopN motoru, orijinal gruptaki ilk N değeri kullanır. Elde edilen değerlerin anlamı, orijinal grubun nasıl sıralandığına bağlıdır:

- Grup, başlık değerine göre sıralandığında: ilk N değer (artan sıralama) veya son N değer (azalan sıralama) görüntülenir;
- Grup, toplam değere göre sıralandığında: en büyük N değer (azalan sıralama) veya en küçük N değer (artan sıralama) görüntülenir.

Öge görünürlüğü

Bir ögenin görünürlüğünü "Görünürlük" sekmesinden yönetebilirsiniz:



The screenshot shows a dialog box with a sidebar on the left containing the following tabs: General, Sort, Filter, Top N, **Visibility** (highlighted with an orange bar), and Other. The main area of the dialog is titled 'Visibility' and contains the following controls:

- A checked checkbox labeled 'Visible'.
- A text field labeled 'Visibility is set by an expression:' with a placeholder 'fx'.
- A dropdown menu labeled 'Visible toggled by:' with the option 'None' selected.
- 'OK' and 'Cancel' buttons at the bottom right.

- "Görünür" - başlangıç görünürlüğünü ayarlar;
- "Görünürlük bir ifade ile ayarlanır" - görünürlüğü ayarlayan, bool türünde dönen bir ifadedir. Eğer ifade boş değilse, değeri "Görünür" seçeneği yerine kullanılır;
- "Görünürlük düğme ile değiştirilir" - bu ögenin görünürlüğünü yöneten MatrixCollapseButton türündeki düğmenin adıdır. Collapsing/Expanding elements bölümüne bakınız.

Diğer ayarlar

Başlık düzenleyicisinin "Diğer" sekmesinde aşağıdaki ayarlar yer alır:

General

Sort

Filter

Top N

Visibility

Other

☐ Stepped layout

☐ Page break

☐ Merge with single subitem

Column span (0 for auto):

Row span (0 for auto):

OK

Cancel

"Stepped layout" seçeneği, iç içe geçmiş öğelerin yerleşimini blok (varsayılan) ve kademeli yerleşim arasında değiştirir. Şimdi iç içe satır başlığı (Yıl, Ay) olan bir matris örneğine bakalım. Blok yerleşimi şu şekilde görünür:

		[Name]
[Year]	[Month]	[Sum(Revenue)]
Total		[Sum(Revenue)]

Tamamlanmış rapor şu şekilde görünür:

		Andrew Fuller	Janet Leverling
2011	2		
	10	\$1,900.00	\$3,000.00
	11	\$2,000.00	\$3,100.00
	12		
2012	1		
	2	\$2,100.00	
	3		\$3,200.00
2013	1		
	2		
	3		
2014	1		
	2		
2015	1	\$1,800.00	
Total		\$7,800.00	\$9,300.00

"Year" öğesi için "Basamaklı düzen" (Stepped layout) seçeneğini etkinleştirirseniz, öğelerin yerleşimi şu şekilde değişecektir:

	[Name]
[Year]	[Sum(Revenue)]
[Month]	[Sum(Revenue)]
Total	[Sum(Revenue)]

Tamamlanmış rapor şu şekilde görünmektedir:

	Andrew Fuller	Janet Leverling
2011	\$3,900.00	\$6,100.00
2		
10	\$1,900.00	\$3,000.00
11	\$2,000.00	\$3,100.00
12		
2012	\$2,100.00	\$3,200.00
1		
2	\$2,100.00	
3		\$3,200.00
2013		
1		
2		
3		
2014		
1		
2		
2015	\$1,800.00	
1	\$1,800.00	
Total	\$7,800.00	\$9,300.00

"Stepped layout" seçeneğini etkinleştirdiğinizde, iç içe yerleştirilmiş öge, metin girintisinden sorumlu olan **Property Padding.Left** özelliğini değiştirir. Bu değeri **Özellikler (Properties)** penceresinden ayarlayabilirsiniz.

Bu seçenek, iç içe öğelere sahip olan herhangi bir öge için kullanılabilir.

"Page Break" seçeneği, öge yazdırılmadan önce yeni bir sayfa ekler. Ancak ilk öğeden önce yeni bir sayfa eklenmez.

"Merge with a single subitem" seçeneği, dinamik olarak daraltılan başlıklar durumunda kullanılır (bkz. **Collapsing/expanding elements** bölümü). Bu seçenek, "Toplam" ögesini gizlemenize olanak tanır. Aşağıda, bu seçeneğin devre dışı bırakıldığı (varsayılan) durumda tamamlanmış raporun görünümü yer almaktadır.

		Andrew Fuller
2011	Total	\$3,900.00
2012	Total	\$2,100.00
	1	
	2	
2013	3	
	Total	
2014	Total	
2015	Total	\$1,800.00
Total		\$7,800.00

ve etkinleştirildiğinde, "Year" (Yıl) ögesinin çıktısına dikkat edin:

		Andrew Fuller
2011		\$3,900.00
2012		\$2,100.00
	1	
	2	
2013	3	
	Total	
2014		
2015		\$1,800.00
Total		\$7,800.00

"Column span" ve "Row span" seçenekleri, öge yazdırılırken sütunlar ve satırlardaki hücreleri birleştirmenize olanak tanır. Varsayılan olarak, öge bu parametreleri otomatik olarak kontrol eder.

Başlık hücrelerinden erişilebilen özellikler

Matris, başlık hücrelerini yazdırmak için kullanılacak bir dizi özelliğe sahiptir. Bir özelliğe başvurmak için matrisin adı kullanılır:

```
[Matrix1.RowNo]
```

Özellik	Geri döndürülen değer	Açıklama
RowNo	int	Başlık öğesinin sıra numarası
ItemCount	int	Geçerli başlık öğesindeki alt öge sayısı

Bu özellikler, başlık hücrelerinin "Gösterilen metin" alanında kullanılabilir. Örneğin:

```
[Matrix1.RowNo].[MatrixDemo.Name]
```

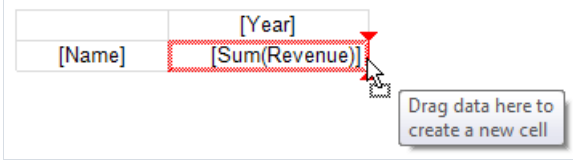
aşağıdaki metni görüntüler:

```
1. Andrew Fuller
```

Veri Alanı Yapılandırması

Adding an item

Bir ögeyi, "**Veri**" (**Data**) penceresinden **DB alanını** sürükleyerek ekleyebilirsiniz. Sürükleme sırasında, yeni ögenin veri alanının hangi bölümüne ekleneceği gösterilecektir:



Boş bir matrise öge eklediğinizde, **Property DataSource** otomatik olarak yapılandırılır.

	[Year]
[Name]	[Sum(Revenue)]

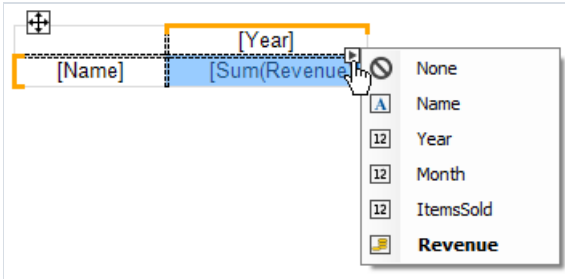
Öge Kaldırma

Bir ögenin metnini, öge menüsünden **Sil** veya **Temizle** seçeneğini seçerek temizleyebilirsiniz. Veri ögesi kendisi silinemez; bunun için, ilgili ögeyi matris başlığından silmeniz gerekir. Yukarıdaki örnekte, **[Sum(Revenue)]** ögesini silmek için "**Revenue**" ögesini başlıktan kaldırmanız gerekir.

Öge Düzenleme

Bir ögenin metnini düzenlemek için, metin düzenleyici penceresini çağırmak için çift tıklayın. Ayrıca, düzenleyiciyi Enter tuşuna basarak da çağırabilirsiniz.

Bir ögede görüntülenen DB alanını hızlıca seçmek için, fare imleci ögenin içinde olduğunda görüntülenen (smarttag) hız düğmesini kullanabilirsiniz.



Bir ögenin bağlam menüsünü kullanarak şu işlemleri de yapabilirsiniz:

- Öge düzenleyicisini çağırmak;
- Veri biçimlendirmesini özelleştirmek;
- Agrega fonksiyonunun türünü değiştirmek ([Aggregate functions](#) bölümüne bakınız);
- Yüzde hesaplaması eklemek;
- Hücreye ilerleme göstergesi veya **sparkline** eklemek.

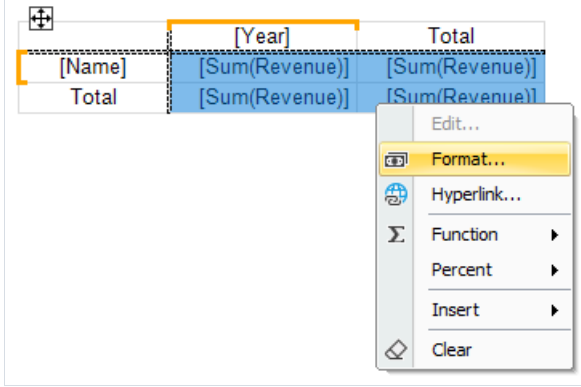
Birden Fazla Ögeyi Düzenleme

Hücrelerin toplu düzenlemesi için tüm veri alanını veya birden fazla hücreyi seçebilirsiniz. Bunu Shift tuşu veya fare

ile yapabilirsiniz:

- Bařlangıç hücrelerini seřin;
- Sol fare tuřuna tıklayın ve fareyi hareket ettirerek bir hücre grubunu seřin;

Seřilen hücreler iřin, baęlam menüsünü çağırabilir veya veritabanı alanı iřin hızlı seřim düęmesini kullanabilirsiniz:



Agregat Fonksiyonları

Agregat fonksiyonları, hücre değerlerinin toplanması ve sonuçların hesaplanması için veri hücrelerinde kullanılır. Agregat fonksiyon çağırısı aşağıdaki biçimde yapılır:

[Function(Expression)]

Kare parantezler, hücre metninde yer alan ifadeleri belirtmek için gereklidir. Normal metinle birlikte, tek bir hücrede birden fazla agregat fonksiyonu kullanabilirsiniz.

İfade genellikle veri kaynağındaki bir alanı temsil eder. Agregat fonksiyonunun kullanımına dair bir örnek:

[Sum([MatrixDemo.Revenue])]

Aşağıda, agregat fonksiyonlarının listesi verilmiştir:

Fonksiyon	Açıklama
Sum	
Min	
Max	
Avg	
Count	
CountDistinct	
StDev	
StDevP	
Var	
VarP	
First	
Last	
ValuesList	
_имя	

Kullanıcı işlevi, adının başında alt çizgi bulunan bir isimle tanımlanır. Bu işlevin kodu, raporun ana sınıfı olan **ReportScript** içinde yer almalıdır. İşlevin tanımı şu şekilde yapılır:

```
object _FuncName(List<dynamic> l)
```

Örnek olarak, aşağıda **_Sum** adlı kullanıcı işlevi gösterilmiştir:

```
public class ReportScript
{
    public object _Sum(List<dynamic> l)
    {
        dynamic value = 0;
        foreach (dynamic v in l)
            value += v;
        return value;
    }
}
```

Özel Fonksiyonlar

Özel fonksiyonlar, matris veri hücrelerinde kullanılabilir. Bu fonksiyonlar, aynı satır veya sütunda bulunan başka bir hücrenin değerini elde etmenizi sağlar.

GrandColumnTotal

Sütun toplamının genel değerini döndürür.

Parametre	Açıklama
aggregate	(İsteğe bağlı) Toplama fonksiyonu. Hücrede zaten toplama fonksiyonuna başvurulmuşsa, parametre atlanabilir.

Örnekler:

```
GrandColumnTotal(Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / GrandColumnTotal()
```

GrandRowTotal

Sıra toplamının genel değerini döndürür.

Parametre	Açıklama
aggregate	(Opsiyonel) Toplama (agrega) fonksiyonu. Eğer aynı hücrede daha önce bu agregaya başvuru yapıldıysa, parametreyi belirtmeden de kullanılabilir.

Örnekler:

```
GrandRowTotal(Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / GrandRowTotal()
```

GrandTotal

Genel toplam değerini döndürür.

parametre	açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.

Örnekler:

```
GrandTotal(Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / GrandTotal()
```

ColumnTotal

Mevcut grup için sütun toplamı değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.

Örnekler:

```
ColumnTotal(Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / ColumnTotal()
```

RowTotal

Mevcut grup için satır toplamı değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.

Örnekler:

```
RowTotal(Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / RowTotal()
```

ColumnMaxValue

Mevcut grup için sütun toplamının en yüksek değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.

Örnekler:

```
ColumnMaxValue(Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / ColumnMaxValue()
```

ColumnMinValue

Mevcut grup için sütun toplamının en düşük değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.

Örnekler:

```
ColumnMinValue(Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / ColumnMinValue()
```

RowMaxValue

Mevcut grup için satır toplamının en yüksek değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.

Örnekler:

```
RowMaxValue(Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / RowMaxValue()
```

RowMinValue

Mevcut grup için satır toplamının en düşük değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.

Örnekler:

```
RowMinValue(Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / RowMinValue()
```

FirstColumn

Sütundaki ilk hücre değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.

useInteractiveSort=false	(opsiyonel) Etkileşimli sıralama sonuçlarını dikkate alır; bu durumda, öğelerin sırası değişebilir.
useThisGroup=true	(opsiyonel) Değeri aramak için aynı seviyedeki grubu kullanır.

Dönüş değeri, fonksiyonun kullanıldığı yere bağlıdır:

- Toplam hücresi: Eğer **useThisGroup** parametresi true ise gruptaki ilk hücre değerini, aksi halde ilk toplam değerini döndürür.
- Grup hücresi: Gruptaki ilk hücre değerini döndürür.

Örnekler:

```
FirstColumn(Sum([MatrixDemo.Revenue]))
FirstColumn(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / FirstColumn()
```

FirstRow

Satırdaki ilk hücre değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.
useInteractiveSort=false	(opsiyonel) Etkileşimli sıralama sonuçlarını dikkate alır; bu durumda öğelerin sırası değişebilir.
useThisGroup=true	(opsiyonel) Değeri aramak için aynı seviyedeki grubu kullanır.

Dönüş değeri, fonksiyonun kullanıldığı yere bağlıdır:

- Toplam hücresi: Eğer **useThisGroup** parametresi true ise gruptaki ilk hücre değerini, aksi halde ilk toplam değerini döndürür.
- Grup hücresi: Gruptaki ilk hücre değerini döndürür.

Örnekler:

```
FirstRow(Sum([MatrixDemo.Revenue]))
FirstRow(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / FirstRow()
```

LastColumn

Sütundaki son hücre değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.

useInteractiveSort=false	(opsiyonel) Etkileşimli sıralama sonuçlarını dikkate alır; bu durumda öğelerin sırası değişebilir.
useThisGroup=true	(opsiyonel) Değeri aramak için aynı seviyedeki grubu kullanır.

Dönüş değeri, fonksiyonun kullanıldığı yere bağlıdır:

- Toplam hücresi: Eğer useThisGroup parametresi true ise gruptaki son hücre değerini, aksi halde son toplam değerini döndürür.
- Grup hücresi: Gruptaki son hücre değerini döndürür.

Örnekler:

```
LastColumn(Sum([MatrixDemo.Revenue]))
LastColumn(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / LastColumn()
```

LastRow

Satırdaki son hücre değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.
useInteractiveSort=false	(opsiyonel) Etkileşimli sıralama sonuçlarını dikkate alır; bu durumda öğelerin sırası değişebilir.
useThisGroup=true	(opsiyonel) Değeri aramak için aynı seviyedeki grubu kullanır.

Dönüş değeri, fonksiyonun kullanıldığı yere bağlıdır:

- Toplam hücresi: Eğer **useThisGroup** parametresi true ise gruptaki son hücre değerini, aksi halde son toplam değerini döndürür.
- Grup hücresi: Gruptaki son hücre değerini döndürür.

Örnekler:

```
LastRow(Sum([MatrixDemo.Revenue]))
LastRow(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / LastRow()
```

PreviousColumn

Sütundaki önceki hücre değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.
useInteractiveSort=false	(opsiyonel) Etkileşimli sıralama sonuçlarını dikkate alır; bu durumda öğelerin sırası değişebilir.

useThisGroup=true

(opsiyonel) Değeri aramak için aynı seviyedeki grubu kullanır.

Dönüş değeri, fonksiyonun kullanıldığı yere bağlıdır:

- **Toplam hücresi:** Eğer **useThisGroup** parametresi true ise gruptaki önceki hücre değerini, aksi halde önceki toplam değerini döndürür.
- **Grup hücresi:** Gruptaki önceki hücre değerini döndürür.

Örnekler:

```
PreviousColumn(Sum([MatrixDemo.Revenue]))  
PreviousColumn(Sum([MatrixDemo.Revenue]), true)  
Sum([MatrixDemo.Revenue]) / PreviousColumn()
```

PreviousRow

Satırdaki önceki hücre değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.
useInteractiveSort=false	(opsiyonel) Etkileşimli sıralama sonuçlarını dikkate alır; bu durumda öğelerin sırası değişebilir.
useThisGroup=true	(opsiyonel) Değeri aramak için aynı seviyedeki grubu kullanır.

Dönüş değeri, fonksiyonun kullanıldığı yere bağlıdır:

- **Toplam hücresi:** Eğer **useThisGroup** parametresi true ise gruptaki önceki hücre değerini, aksi halde önceki toplam değerini döndürür.
- **Grup hücresi:** Gruptaki önceki hücre değerini döndürür.

Örnekler:

```
PreviousRow(Sum([MatrixDemo.Revenue]))  
PreviousRow(Sum([MatrixDemo.Revenue]), true)  
Sum([MatrixDemo.Revenue]) / PreviousRow()
```

NextColumn

Sütundaki sonraki hücre değerini döndürür.

Parametre	Açıklama
aggregate	(opsiyonel) Agregat fonksiyonu. Hücrede agregata zaten başvurulduysa, parametre atlanabilir.
useInteractiveSort=false	(opsiyonel) Etkileşimli sıralama sonuçlarını dikkate alır; bu durumda öğelerin sırası değişebilir.

useThisGroup=true	(opsiyonel) Değeri aramak için aynı seviyedeki grubu kullanır.
--------------------------	--

Dönüş değeri, fonksiyonun kullanıldığı yere bağlıdır:

- **Toplam hücresi:** Eğer `useThisGroup` parametresi true ise gruptaki sonraki hücre değerini, aksi halde sonraki toplam değerini döndürür.
- **Grup hücresi:** Gruptaki sonraki hücre değerini döndürür.

Örnekler:

```
NextColumn(Sum([MatrixDemo.Revenue]))
NextColumn(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / NextColumn()
```

NextRow:

Satırdaki bir sonraki hücrenin değerini döndürür.

Parametre	Açıklama
aggregate	(isteğe bağlı) Agregat fonksiyonu. Hücredeki agregata zaten başvurulmuşsa, parametre atlanabilir.
useInteractiveSort=false	(isteğe bağlı) Öğelerin sıralamasının değişebileceği interaktif sıralama sonuçlarını dikkate alır.
useThisGroup=true	(isteğe bağlı) Değeri bulmak için aynı seviyedeki grubu kullanır.

Fonksiyonun kullanıldığı yere bağlı olarak döndürülen değer:

- **Toplam hücresi:** Eğer `useThisGroup` parametresi true ise, gruptaki sonraki hücrenin değerini döndürür; aksi halde sonraki toplamın değerini döndürür.
- **Grup hücresi:** Gruptaki sonraki hücrenin değerini döndürür.

Örnekler:

```
NextRow(Sum([MatrixDemo.Revenue]))
NextRow(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / NextRow()
```

SpecificColumn:

Belirtilen indekse sahip sütun hücresinin değerini döndürür.

Parametre	Açıklama
index	İndeks değeri
aggregate	(isteğe bağlı) Agregat fonksiyonu. Hücredeki agregata zaten başvurulmuşsa, parametre atlanabilir.

Fonksiyonun kullanıldığı yere bağlı olarak döndürülen değer şu şekildedir:

- **Toplam hücresi:** İlgili toplamın ait olduğu grubun hücresinin değerini döndürür.
- **Diğer durumlar:** Mevcut grubun sütunundaki hücre değerini döndürür.

Örnekler:

```
SpecificColumn("Andrew Fuller", Sum([MatrixDemo.Revenue]))  
SpecificColumn(2011, Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / SpecificColumn(2011)
```

SpecificRow

Satırdaki belirtilen indekse sahip hücrenin değerini döndürür.

Parametre	Açıklama
index	indeks değeri
aggregate	(İsteğe bağlı) Agregat fonksiyon. Parametre, hücredeki agregata zaten başvurulmuşsa atlanabilir.

Dönüş değeri, fonksiyonun hangi konumda kullanıldığına bağlıdır:

- **Sonuç hücresi:** Sonuca ait gruptaki hücre değerini döndürür;
- **Diğer durumlarda:** Mevcut grubun satırındaki hücre değerini döndürür.

Örnekler:

```
SpecificRow("Andrew Fuller", Sum([MatrixDemo.Revenue]))  
SpecificRow(2011, Sum([MatrixDemo.Revenue]))  
Sum([MatrixDemo.Revenue]) / SpecificRow(2011)
```

PercentOfColumnTotal

Mevcut hücre değerini sütun toplamı değerine böler.

Parametre	Açıklama
aggregate	(İsteğe bağlı) Agregat fonksiyon. Parametre, hücredeki agregata zaten başvurulmuşsa atlanabilir.

Fonksiyon Çağrısı

```
PercentOfColumnTotal(Sum([MatrixDemo.Revenue]))
```

aşağıdaki koda eşdeğerdir:

```
Sum([MatrixDemo.Revenue]) / ColumnTotal()
```

Örnekler:

```
PercentOfColumnTotal(Sum([MatrixDemo.Revenue]))  
[Sum([MatrixDemo.Revenue])] [PercentOfColumnTotal()]
```

PercentOfRowTotal

Mevcut hücre değerini satır toplamı değerine böler.

Parametre	Açıklama
aggregate	(İsteğe bağlı) Agregat fonksiyon. Parametre, hücredeki agregata zaten başvurulmuşsa atlanabilir.

Örnekler:

```
PercentOfRowTotal(Sum([MatrixDemo.Revenue]))  
[Sum([MatrixDemo.Revenue])] [PercentOfRowTotal()]
```

PercentOfGrandTotal

Mevcut hücre değerini genel toplam değerine böler.

Parametre	Açıklama
aggregate	(İsteğe bağlı) Agregat fonksiyon. Parametre, hücredeki agregata zaten başvurulmuşsa atlanabilir.

Örnekler:

```
PercentOfGrandTotal(Sum([MatrixDemo.Revenue]))  
[Sum([MatrixDemo.Revenue])] [PercentOfGrandTotal()]
```

PercentOfPreviousColumn

Mevcut hücre değerini, sütundaki önceki hücre değerine böler.

Parametre	Açıklama
aggregate	(İsteğe bağlı) Agregat fonksiyon. Parametre, hücredeki agregata zaten başvurulmuşsa atlanabilir.
useInteractiveSort=false	(İsteğe bağlı) Öğelerin sırasının değişebileceği etkileşimli sıralama sonuçlarını dikkate alın.
useThisGroup=true	(İsteğe bağlı) Değeri bulmak için aynı seviyedeki grubu kullanın.

Dönüş değeri, fonksiyonun hangi konumda kullanıldığına bağlıdır:

- **Sonuç hücresi:** Eğer `useThisGroup` parametresi `true` ise, grubun önceki hücre değeri kullanılır; aksi takdirde, önceki sonucun değeri kullanılır.
- **Grup hücresi:** Grubun önceki hücre değeri kullanılır.

Örnekler:

```
PercentOfPreviousColumn(Sum([MatrixDemo.Revenue]))
PercentOfPreviousColumn(Sum([MatrixDemo.Revenue]), true)
[Sum([MatrixDemo.Revenue])] [PercentOfPreviousColumn()]
```

PercentOfPreviousRow

Mevcut hücre değerini, satırdaki önceki hücre değerine böler.

Parametre	Açıklama
aggregate	(İsteğe bağlı) Agregat fonksiyon. Parametre, hücredeki agregata zaten başvurulmuşsa atlanabilir.
<code>useInteractiveSort=false</code>	(İsteğe bağlı) Öğelerin sırasının değişebileceği etkileşimli sıralama sonuçlarını dikkate alın.
useThisGroup=true	(İsteğe bağlı) Değeri bulmak için aynı seviyedeki grubu kullanın.

Dönüş değeri, fonksiyonun hangi konumda kullanıldığına bağlıdır:

- **Sonuç hücresi:** Eğer `useThisGroup` parametresi `true` ise, grubun önceki hücre değeri kullanılır; aksi takdirde, önceki sonucun değeri kullanılır.
- **Grup hücresi:** Grubun önceki hücre değeri kullanılır.

Örnekler:

```
PercentOfPreviousRow(Sum([MatrixDemo.Revenue]))
PercentOfPreviousRow(Sum([MatrixDemo.Revenue]), true)
[Sum([MatrixDemo.Revenue])] [PercentOfPreviousRow()]
```

Veri hücrelerinde kullanılabilen özellikler

Matris, veri hücrelerinin yazdırılmasında kullanılacak bir dizi özelliğe sahiptir. Bir özelliğe erişmek için matris adı kullanılır:

```
[Matrix1.RowIndex]
```

özellik	dönen değer	açıklama
ColumnIndex	int	Geçerli sütun indeksi
RowIndex	int	Geçerli satır indeksi
ColumnValues	object[]	Sütun başlığından değerler dizisi
RowValues	object[]	Satır başlığından değerler dizisi

Bu özellikler, belirli bir koşula bağlı olarak hücreleri renklendirirken faydalı olabilir. Ayrıca, hücreden veri kaynağı alanlarına erişebilirsiniz. Genellikle, bu durum koşullu vurgulama için gereklidir (bkz. Koşullu Vurgulama bölümü). Örneğin, veri hücresi için 2012 yılına ait değerlerin vurgulanması amacıyla aşağıdaki koşul belirtilebilir:

```
[MatrixDemo.Year] == 2012
```

Hücreyi yazdırmak için kullanılan veritabanı alanı değeri, matrisin başlığından alınır. İki başlık (satır ve sütun) bulunduğundan, hangi başlık değerlerinin öncelikli olacağını belirtmek gereklidir. Bunun kontrolünden matrisin DataRowPriority özelliği sorumludur. Varsayılan olarak, bu özellik "Rows" değerine sahiptir.

Etkileşimli Özellikler

Bu bölümde "Gelişmiş Matris" nesnesinin etkileşimli özellikleri ele alınmaktadır:

- [Başlık öğelerinin daraltılması/genişletilmesi](#);
- [Etkileşimli sıralama](#).

Öğeleri Daraltma/Genişletme

Bireysel başlık öğelerinin görünürlüğüne etkileşimli olarak yönetmek için MatrixCollapseButton türünde özel bir düğme kullanabilirsiniz. Bu düğme başlık öğesine yerleştirilir ve diğer öğelerin görünürlüğüne kontrol eder. Aşağıdaki görselde, düğme ve onun yönettiği öğe gösterilmiştir (düğme seçildiğinde kırmızı çerçeve ile işaretlenmiştir):

		[Year]
	[Month]	Total
[Name]	[Sum(Revenue)]	[Sum(Revenue)]

Önizleme penceresinde düğmeye tıkladığınızda, ilgili öğeler gizlenir veya gösterilir. Bu durumda rapor yeniden oluşturulur:

	2011	2012	2013	2014			2015	Total
	Total	Total	Total	1	2	Total	Total	Total
Andrew Fuller	\$3,900.00	\$2,100.00					\$1,800.00	\$7,800.00
Janet Leverling	\$6,100.00	\$3,200.00						\$9,300.00
Nancy Davolio	\$3,300.00	\$2,700.00	\$3,100.00				\$1,700.00	\$10,800.00
Steven Buchanan			\$3,999.00	\$4,000.00	\$4,100.00	\$8,100.00		\$12,099.00

Düğme Ekleme

Bir başlık öğesine düğme eklemek için bağlam menüsünü kullanabilirsiniz. Öğeyi seçin, sağ tıklayın ve Daraltma düğmesi (Collapse button) seçeneğini seçin. Düğme, öğenin sol tarafına eklenir.

Bir düğme eklendiğinde, düğmenin metnin üzerine binmemesi için öğenin **Property Padding.Left** değeri değiştirilir.

Düğme Özelleştirme

Bir düğme eklendiğinde, FastReport düğme ile onun yönettiği öğe arasındaki bağlantıyı otomatik olarak yapılandırır. Ancak bazı durumlarda bu bağlantıyı manuel olarak yapılandırmanız gerekebilir. Bunu yapmak için, düğmeden etkilenmesi gereken öğenin düzenleyicisini açın ve "Görünürlük / Görünürlük düğme ile değiştirilir" sekmesinde düğmenin adını belirtin.

Düğme, aynı anda birden fazla öğeyi yönetebilir.

Düğme, yönetilen öğenin üstüne yerleştirilebilir:

		[Year]
	[Month]	Total
[Name]	[Sum(Revenue)]	[Sum(Revenue)]

Veya onunla aynı seviyeye yerleştirilebilir:

	[Year]		Total
[Name]	[Sum(Revenue)]	[Sum(Revenue)]	

Kontrol edilen öğenin başlangıç durumu — görünürlüğü — öğe düzenleyicisinde "**Görünürlük / Görünür**" sekmesinde ayarlanır.

Düğme Silme

Bir düğmeyi iki şekilde silebilirsiniz:

- Düğmeyi seçin ve Delete tuşuna basın;
- Öğenin bağlam menüsünde "Daraltma düğmesi" onay kutusunun işaretini kaldırın.

Düğme Taşıma

Varsayılan olarak, düğmenin Property **Dock = Left** değeri vardır. Bu, düğmenin öğenin sol kenarına yerleştirildiği anlamına gelir. Düğmeyi yeni bir yere taşımak için Property Dock = None değerini "Özellikler" penceresinde ayarlayın.

Ayrıca, bir düğmeyi bir öğenin belirli bir konumuna sabitlemek için düğmenin **Property Anchor** özelliğini kullanabilirsiniz.

Düğmenin Görünümünü Özelleştirme

"**Border**" araç çubuğunu kullanarak, düğme simgesini özelleştirebilirsiniz: çerçeve rengi ve stili, arka plan rengi. Ayrıca, **Özellikler** penceresinde düğmenin aşağıdaki özelliklerini ayarlayabilirsiniz:

özellik	varsayılan değer	açıklama
Cursor	Hand	fare imleci tipi
Exclusive	False	Eğer true ise, yalnızca bir öğe genişletilmiş durumda olabilir.
Exportable	False	Eğer true ise, rapor dışı aktarılırken buton görüntülenecektir.
Printable	False	Eğer true ise, rapor yazdırılırken buton görüntülenecektir.
ShowCollapseExpandMenu	False	Bu ayar, düğmeye sağ tıklanıldığında "Tümünü Daralt/Tam Genişlet" seçeneklerini içeren menünün gösterilip gösterilmeyeceğini belirler.
Symbol	PlusMinus	Butonun içinde görüntülenen sembol.
SymbolSize	5	Buton sembolünün boyutu.

Etkileşimli sıralama

MatrixSortButton sıralama düğmesi, matrisin satırlarını veya sütunlarını etkileşimli olarak sıralamanızı sağlar. Düğme, alt seviyedeki bir başlık öğesine yerleştirilmelidir:

	[Year]	Total
[Name]	[Sum(Revenue)]	[Sum(Revenue)]

Önizleme penceresinde düğmeye tıkladığınızda, karşıt başlık sıralanır. Aşağıdaki örnekte, seçilen sütundaki değere göre satırlar sıralanır:

	2011	2012	2013	2014	2015	Total
Andrew Fuller	\$3,900.00	\$2,100.00			\$1,800.00	\$7,800.00
Janet Leverling	\$6,100.00	\$3,200.00				\$9,300.00
Nancy Davolio	\$3,300.00	\$2,700.00	\$3,100.00		\$1,700.00	\$10,800.00
Steven Buchanan			\$3,999.00	\$8,100.00		\$12,099.00

Düğmeye her tıklama, sıralama modunu değiştirir: artan/azalan/sıralama yok.

Düğme Ekleme

Bir düğmeyi başlık öğesine bağlam menüsünü kullanarak ekleyebilirsiniz. Ögeyi seçin, sağ tıklayın ve "Sıralama Düğmesi" seçeneğini seçin. Düğme, ögenin sağ kısmına eklenir.

Bir düğme eklediğinizde, düğmenin metnin üzerine binmemesi için ögenin **Property Padding.Right** değeri değişir.

Düğme Özelleştirme

Başlık sıralama modu, düzenleyicisinde "Sort/Interactive Sort" by Total sekmesinde ayarlanır. Aşağıdaki değerler mümkündür:

- "Hayır" - bu başlık sıralanmaz.
- "Otomatik", varsayılan moddur. Sıralama, ilk toplamın (agreganın) değeriyle yapılır.
- Toplam (agrega) adı: Eğer başlıkta birden fazla çıktı değeri varsa, sıralamak için bunlardan birini seçebilirsiniz. Aşağıdaki örnekte, satır başlığını ItemsSold değeriyle sıralamak için Sum ([MatrixDemo.ItemsSold]) agregasını seçin:

		Total
Nancy Davolio	Revenue	\$10,800.00
	ItemsSold	12
Steven Buchanan	Revenue	\$12,099.00
	ItemsSold	11
Janet Leverling	Revenue	\$9,300.00
	ItemsSold	9
Andrew Fuller	Revenue	\$7,800.00
	ItemsSold	8

Düğme Kaldırma

Bir düğmeyi kaldırmanın iki yolu vardır:

- Düğmeyi seçin ve **Delete** tuşuna basın;
- Ögenin bağlam menüsünde "**Sıralama düğmesi**" seçeneğinin işaretini kaldırın.

Düğmeyi Taşıma

Varsayılan olarak, düğmenin Property **Dock = Right** değeri vardır. Bu, düğmenin öğenin sağ kenarına yerleştirildiği anlamına gelir. Düğmeyi yeni bir konuma taşımak için Özellikler penceresinden Property Dock = None olarak ayarlayın.

Ayrıca, düğmeyi bir öge üzerinde belirli bir konuma sabitlemek için Property **Anchor** özelliğini de kullanabilirsiniz.

Düğme Görünümünü Özelleştirme

"Border" araç çubuğunu kullanarak düğme simgesini özelleştirebilirsiniz: çerçeve rengi ve stili, arka plan rengi. Ayrıca, Özellikler penceresinde düğmenin aşağıdaki özelliklerini de ayarlayabilirsiniz:

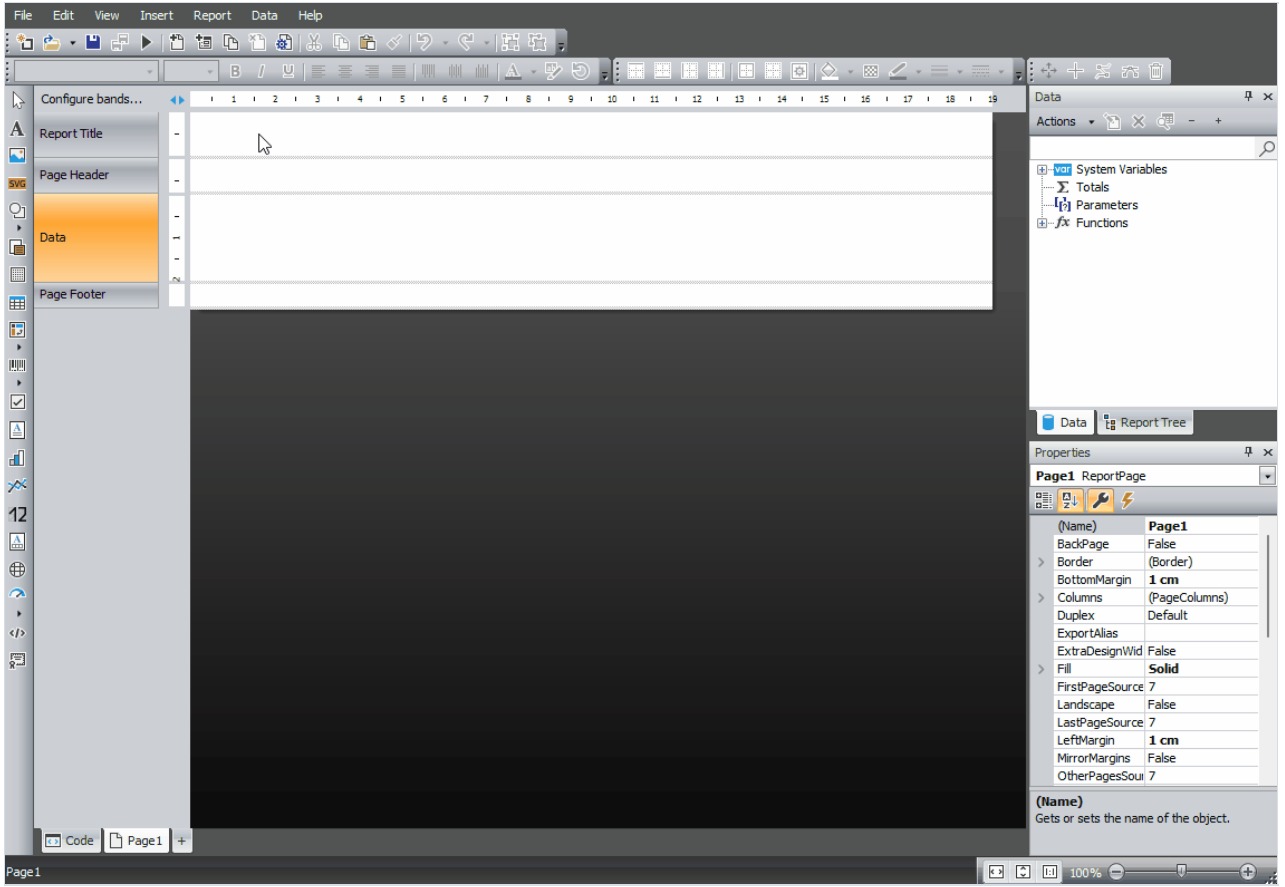
Özellik	Varsayılan değer	Açıklama
AllowInactiveSort	True	Butonun, "sıralama yapılmayan" modda inaktif bir duruma sahip olup olmayacağını belirler.
Cursor	Hand	Fare imleci tipi.
Exportable	False	Eğer true ise, rapor dışa aktarılırken buton görüntülenecektir.
InactiveSortColor	Gray	Butonun inaktif durumdaki rengi.
Printable	False	Eğer true ise, rapor yazdırılırken buton görüntülenecektir.
Symbol	Arrow	Butonun sembolü.
SymbolSize	7	Buton sembolünün boyutu.

Kullanım Örnekleri

Aşağıda, "Geliştirilmiş Matris" nesnesinin animasyonlu gifler şeklinde kullanım örnekleri yer almaktadır.

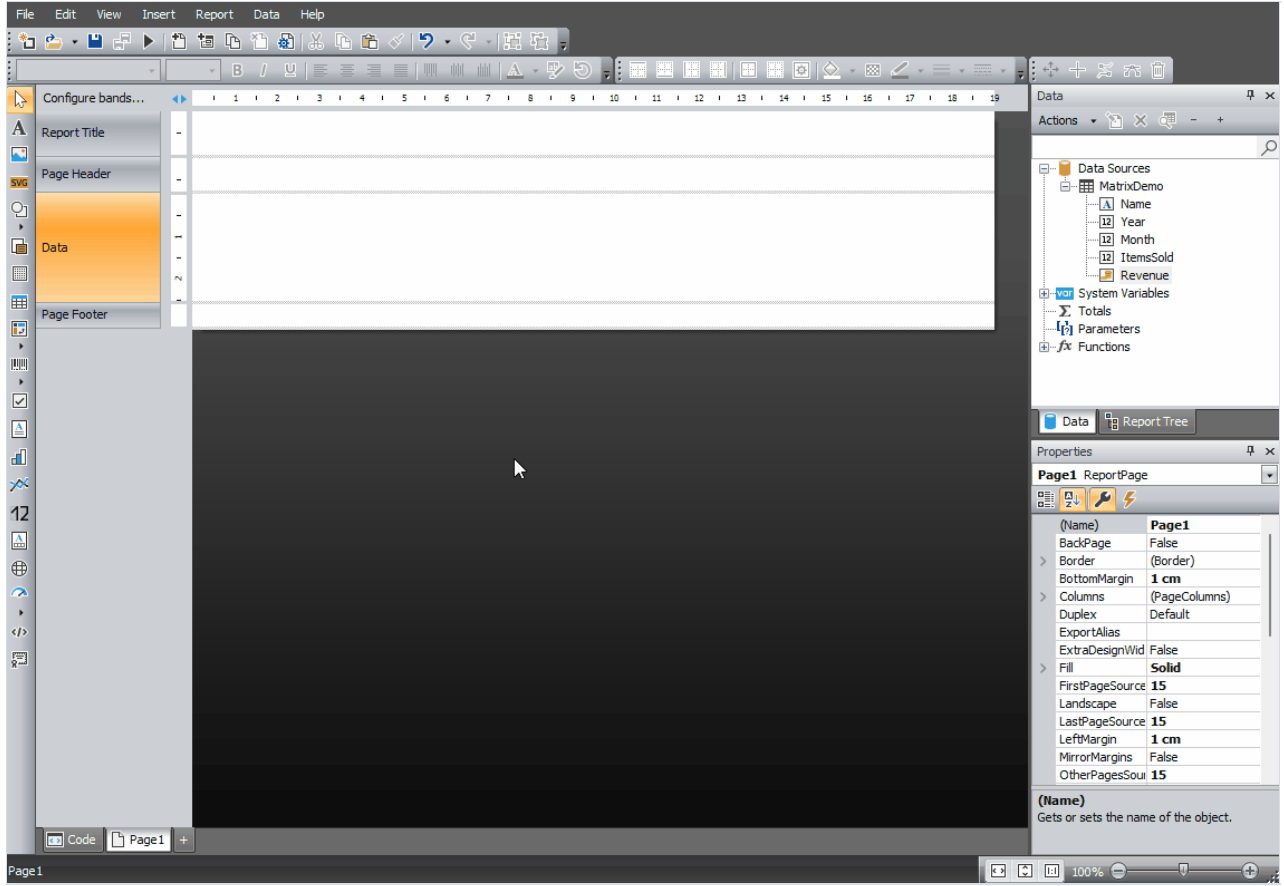
Örnek 1. Basit Matris

Basit bir matris oluşturma, stil seçimi, sonuçların eklenmesi:



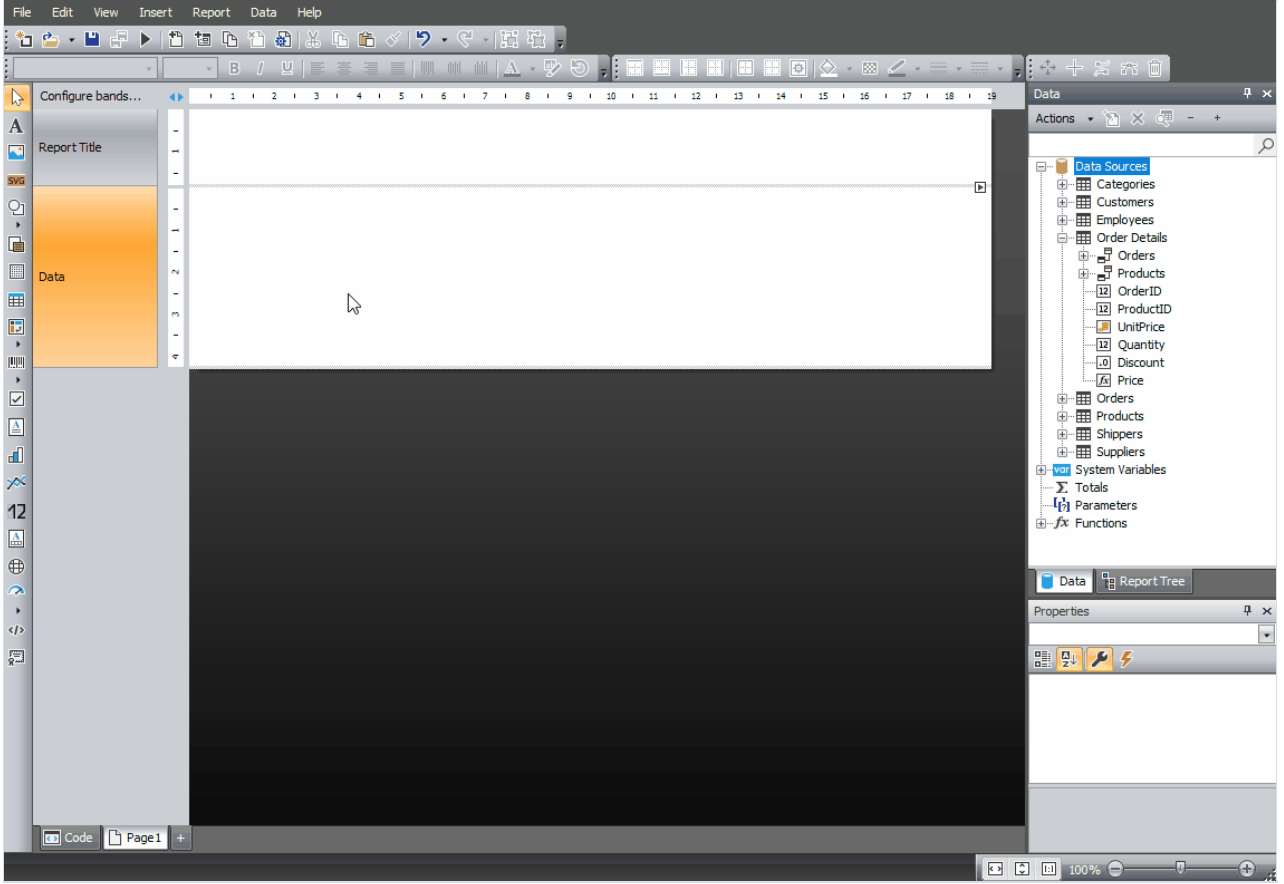
Örnek 2. Bileşik başlıklara sahip matris

Bileşik başlıklarla bir matris oluşturma, toplam ekleme, `DisplayText` özelliğini kullanma, kademeli başlık görüntüleme, daraltma/genişletme düğmeleri:



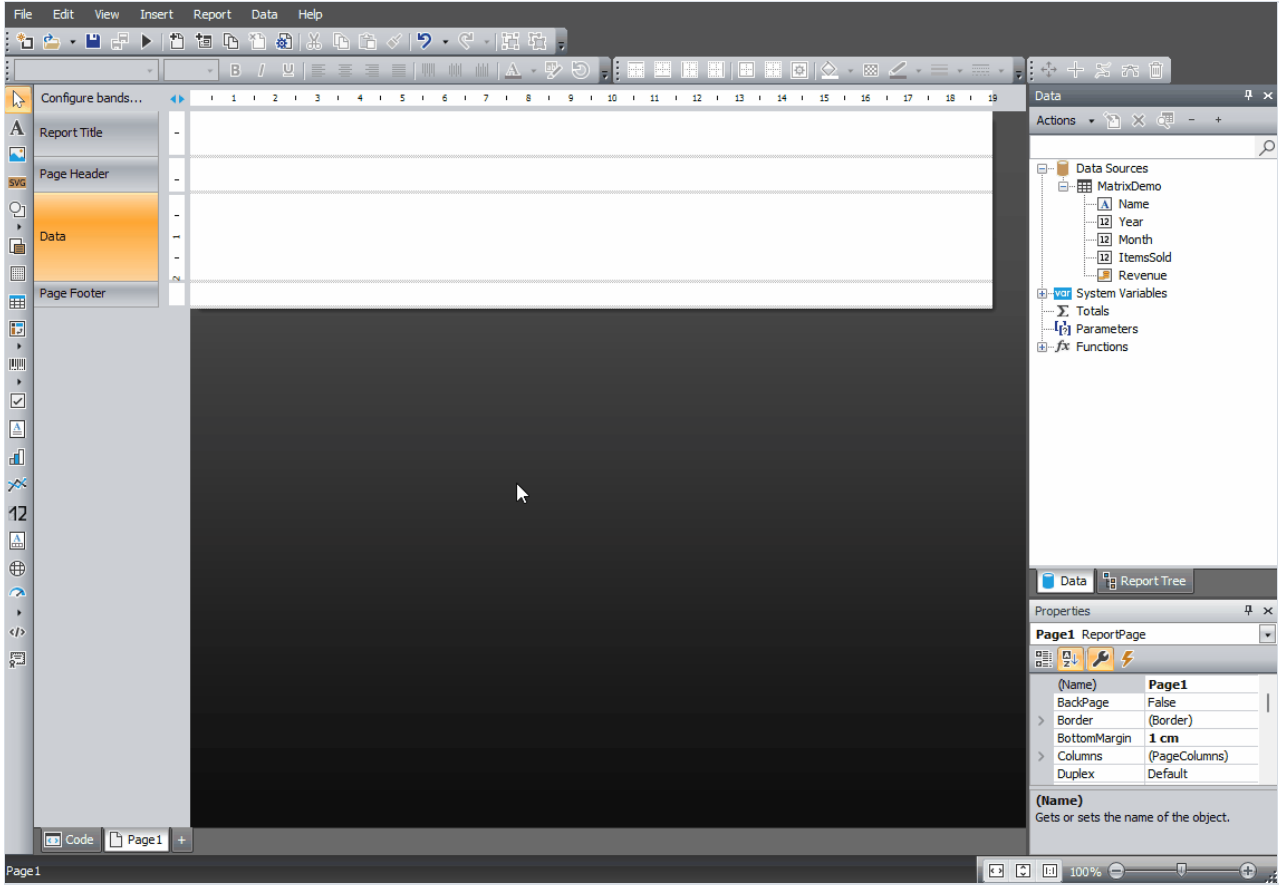
Örnek 3. TopN Gruplaması

TopN grubunun oluşturulması ve yapılandırılması (iki seçenek – basit ve özelleştirilebilir), öğelerin genişletilmesi, matrisin **ItemCount** özelliğinin kullanımı:



Örnek 4. Etkileşimli Sıralama

Etkileşimli sıralama düğmelerinin eklenmesi, veri alanına (**ItemsSold**) adlı ikinci bir veritabanı alanının eklenmesi, bu alana dayalı etkileşimli sıralamanın yapılandırılması.



İnteraktif Raporlar

Hazır FastReport raporu interaktif hale getirilebilir. Bu, kullanıcının (özellikle fare tıklaması) eylemlerine yanıt vereceği anlamına gelir. Aşağıdaki etkileşim türlerini kullanabilirsiniz:

- Rapor elemanına (bant veya nesne) tıklandığında belirli bir işlem gerçekleştirilir. Örneğin, detaylı bir rapor oluşturulur ve ayrı bir pencerede gösterilir.
- Görüntüleme penceresinin yanında, raporun yapısı gösterilir; bu yapı, rapor içinde gezinmek için kullanılabilir.

Köprüler (Hyperlink)

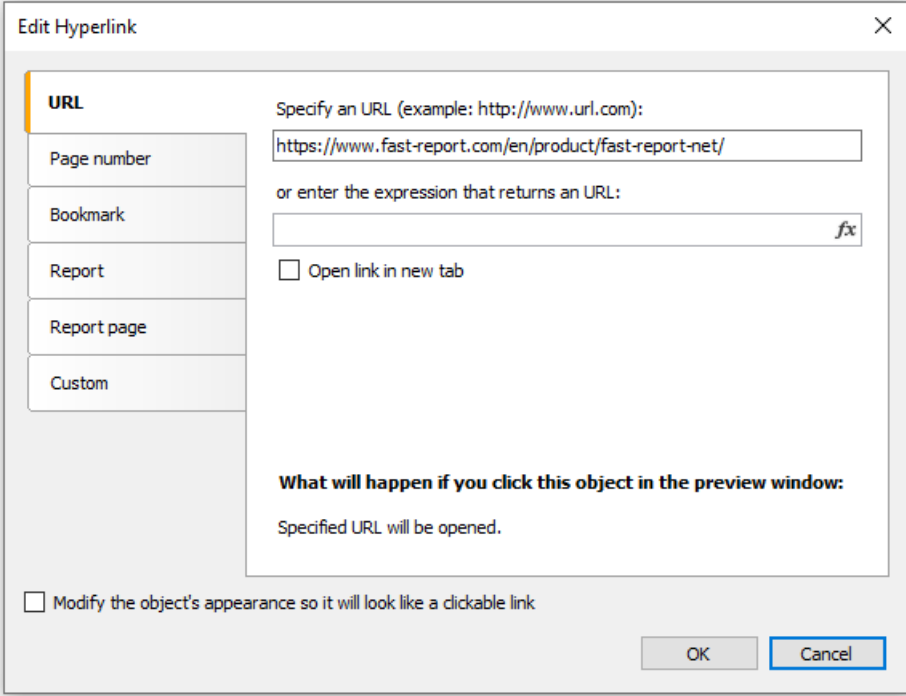
Bu bölümde, bantlar da dâhil olmak üzere raporun hemen hemen tüm nesnelerinde bulunan "Köprü" ([Hyperlink](#)) özelliği ele alınacaktır. Bu özellik, nesneye önizleme penceresinde fareyle tıklandığında gerçekleştirilecek eylemi belirlemeye yarar.

Bu tür bir nesneye tıklandığında aşağıdaki işlemlerden biri gerçekleştirilebilir:

- URL adresine geçiş yapmak;
- E-posta göndermek;
- Bir sistem komutunu çalıştırmak;
- Raporun belirli numaralı sayfasına gitmek;
- Rapor içindeki başka bir nesnede tanımlanmış yer imine gitmek;
- Ayrı bir önizleme penceresinde ayrıntılı raporu çalıştırmak;
- Kullanıcı tarafından komut dosyasında (scriptte) tanımlanmış özel bir tepkiyi çalıştırmak.

Köprü Ayarları

Köprü bağlantısını ayarlamak için, fare tıklamasına tepki verecek olan nesneyi seçin ve üzerinde farenin sağ tuşuna tıklayın. Açılan içerik menüsünden **"Köprü..."** seçeneğini seçin. Köprü düzenleyici penceresi açılacaktır:



The image shows a dialog box titled "Edit Hyperlink". On the left, there is a vertical list of options: "URL", "Page number", "Bookmark", "Report", "Report page", and "Custom". The "URL" option is selected and highlighted with an orange bar. To the right of this list, there are two input fields. The first is labeled "Specify an URL (example: http://www.url.com):" and contains the text "https://www.fast-report.com/en/product/fast-report-net/". The second is labeled "or enter the expression that returns an URL:" and is empty, with a small "fx" icon to its right. Below these fields is a checkbox labeled "Open link in new tab", which is currently unchecked. At the bottom of the dialog, there is a checkbox labeled "Modify the object's appearance so it will look like a clickable link", which is also unchecked. Below this checkbox are two buttons: "OK" and "Cancel".

Bağlantı türünü, pencerenin sol tarafındaki sekmeler yardımıyla belirleyebilirsiniz. Bağlantıyı yapılandırdıktan sonra pencerenin alt kısmında bulunan **"Nesnenin görünümünü değiştir..."** seçeneğini işaretleyebilirsiniz. Bu seçenek etkinleştirildiğinde nesnenin görünümü şu şekilde değişecektir:

- Metnin rengi mavi olacak ve altı çizilecektir;
- Fare imleci şekli (Cursor özelliği) el şeklinde olacaktır.

Bazı durumlarda köprünün önizleme ekranında görüntülenmesi ancak yazdırılırken görünmemesi gerekebilir. Bu durumu nesnenin **Printable** (Yazdırılabilir) özelliğini kapatarak kolayca gerçekleştirebilirsiniz. Bu ayarı **"Özellikler"** penceresinden yapabilirsiniz.

URL'ye bağlantı

Bu tür bağlantılar kullanarak şunları yapabilirsiniz:

- Belirtilen internet adresine gidebilirsiniz;
- Sistem komutlarını çalıştırabilirsiniz, örneğin e-posta göndermek için `mailto:` kullanmak gibi.

Bu tür bağlantıya tıklandığında, bağlantının değeri parametre olarak verilerek `System.Diagnostics.Process.Start` metodu çalıştırılır.

Bağlantının değerini iki yöntemle belirleyebilirsiniz:

- Değeri doğrudan belirtme; örneğin, `"http://www.fast-report.com"`.
- Rapor çalıştırıldığında hesaplanacak bir ifade belirterek bağlantının değerini dinamik olarak oluşturabilirsiniz.

Sayfa Numarasına Bağlantı

Bu bağlantı türünü kullanarak hazırlanmış raporunuzda sayfalar arası gezinti sağlayabilirsiniz. Genellikle raporun ilk sayfasına geçiş yapmak için kullanılır. Bunun için bağlantı değeri olarak sayfa numarasını belirtmeniz gerekir (örneğin, 1).

Sayfa numarasını iki farklı şekilde belirtebilirsiniz:

- Sayfa numarasını doğrudan belirtme (örneğin, 1);
- Sayfa numarasını döndüren bir ifade belirtme. Bu ifade, rapor oluşturulurken ilgili nesne işleme anında hesaplanacaktır.

Bağlantı: İşaretlemeye Gitmek

Bu bağlantı türünü kullanarak, başka bir nesnede ya da programatik olarak tanımlanmış bir işaretleme geçiş yapabilirsiniz. HTML işaretleme dilini bilenler için belirtmek gerekir ki, işaretlemenin çalışma prensibi çapa (anchor) ile aynıdır. İşaretlemenin bir adı ve rapordaki belirli bir konumu (sayfa numarası ve sayfadaki pozisyon) bulunur. Bağlantıya tıkladığınızda, belirtilen işaretlemenin bulunduğu konuma geçiş yaparsınız.

İlk adım olarak, işaretlemenin tanımlanması gerekir. Bunu yapmak için, bağlantıya tıklandığında geçiş yapılmasını istediğiniz rapor nesnesini seçin. Neredeyse tüm rapor nesneleri bir işaretleme sahiptir; bu değeri "Bookmark" özelliğinde görebilirsiniz. Bu özelliği "Özellikler" penceresinden değiştirebilirsiniz.

Bookmark özelliği, "İfade" (Expression) türündedir ve şu şekilde kullanılabilir:

- İşaretlemenin adını bir string olarak belirtmek (tırnak işaretleri içinde):

"MyBookmark"

- İşaretleme adını döndürecek bir ifade belirtmek. Örneğin, bir veri alanını ifade olarak kullanabilirsiniz. Bu durumda ifade değeri, rapor çalıştırıldığında ilgili nesne işlenirken hesaplanır.

İşaretleme tanımlandıktan sonra, hiperlink ayar penceresinde işaretleme adını şu iki yolla belirtebilirsiniz:

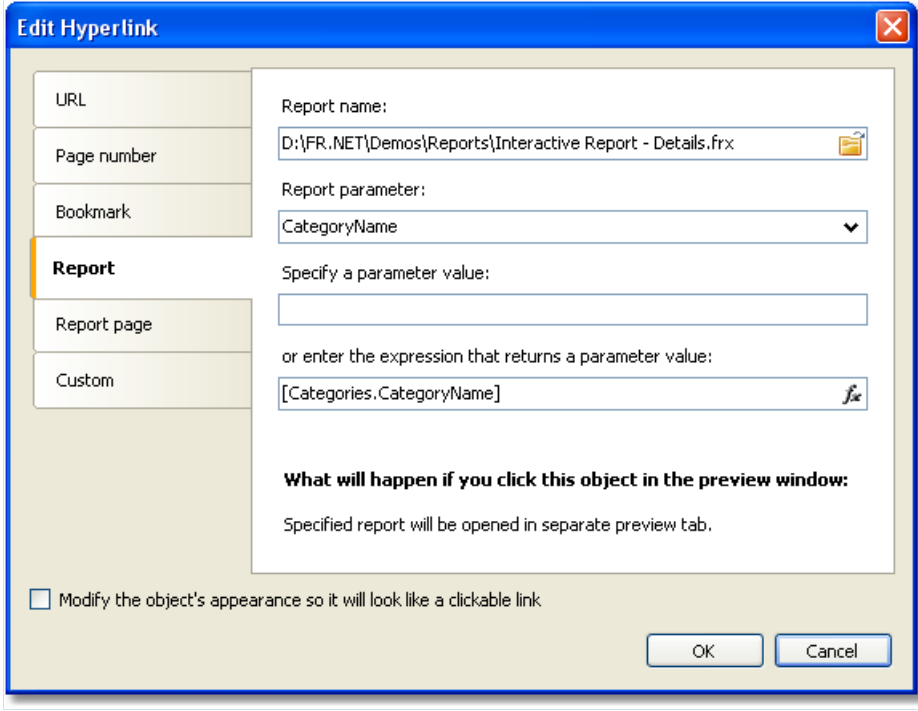
- İşaretleme adını doğrudan girebilirsiniz.
- İşaretleme adını döndürecek bir ifade girebilirsiniz. Örneğin, bu bir veri alanı olabilir; bu ifade rapor çalıştırıldığında hesaplanacaktır.

Ayrıntılı rapora bağlantı

Bu bağlantı türünü kullanarak başka bir raporu çalıştırabilir ve ayrı bir görüntüleme penceresinde gösterebilirsiniz.

Bu bağlantının ayarlarında aşağıdaki bilgileri belirtmeniz gerekmektedir:

- Detaylı rapor adı;
- Rapor parametresine atanacak değerin adı;
- Bağlantıdan rapor parametresine iletilecek değer.



The 'Edit Hyperlink' dialog box is shown with the 'Report' tab selected. It contains the following fields and options:

- URL**: (Empty text box)
- Page number**: (Empty text box)
- Bookmark**: (Empty text box)
- Report**: (Selected tab)
- Report page**: (Empty text box)
- Custom**: (Empty text box)
- Report name:**: D:\FR.NET\Demos\Reports\Interactive Report - Details.frx
- Report parameter:**: CategoryName
- Specify a parameter value:**: (Empty text box)
- or enter the expression that returns a parameter value:**: [Categories.CategoryName]
- What will happen if you click this object in the preview window:**: Specified report will be opened in separate preview tab.
- ☐ Modify the object's appearance so it will look like a clickable link
- OK** and **Cancel** buttons.

Bağlantıya tıklandığında gerçekleşecek işlemler şunlardır:

- Belirtilen rapor yüklenecektir;
- Rapor parametresine bağlantı değeri aktarılacaktır;
- Rapor oluşturularak ayrı bir görüntüleme penceresinde çalıştırılacaktır.

Rapor parametresine değer ataması iki şekilde yapılabilir:

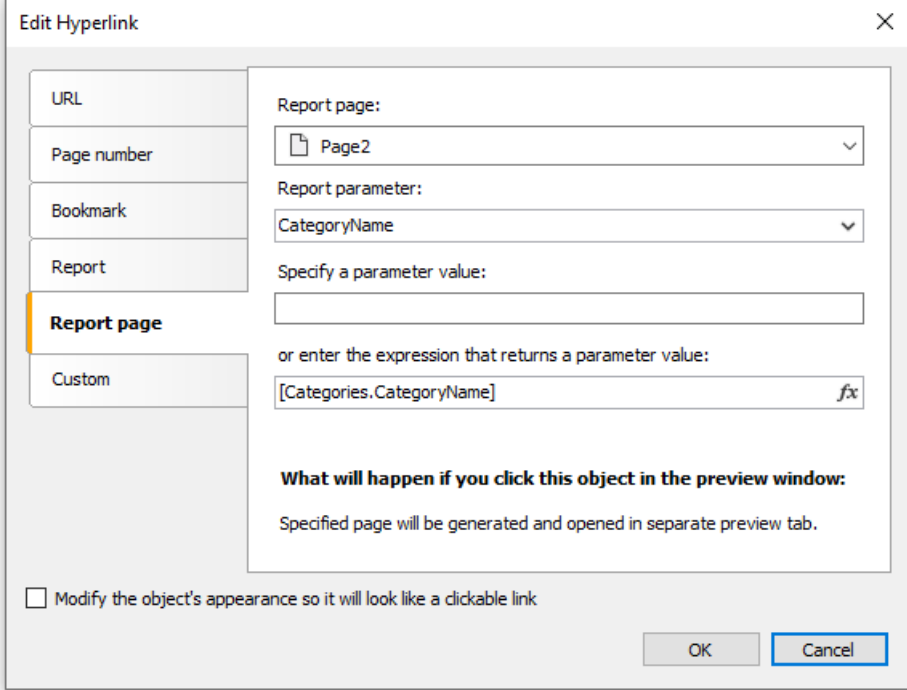
- Değeri doğrudan belirtmek;
- Değeri döndürecek bir ifade belirtmek. Belirtilen ifade, rapor çalıştırıldığında, ilgili nesnenin işleme anında hesaplanacaktır.

Detaylı Sayfaya Bağlantı

Bu tür bağlantı benzer şekilde çalışır, ancak detay raporu olarak mevcut rapordaki farklı bir sayfa kullanılır. Bunun için raporunuz en az iki sayfa içermelidir: biri ana rapor için, diğeri detaylı rapor için.

Bu bağlantı türünün ayarlarında aşağıdaki bilgileri belirtmeniz gerekir:

- Bu rapordaki sayfanın adı;
- Hiperlinkten gelecek değerin aktarılacağı rapor parametresinin adı;
- Rapor parametresine aktarılması gereken hiperlink değeri.



The image shows a dialog box titled "Edit Hyperlink". On the left, there is a sidebar with five options: "URL", "Page number", "Bookmark", "Report", and "Report page" (which is selected and highlighted with an orange bar), and "Custom". The main area of the dialog is divided into two sections. The top section contains three fields: "Report page:" with a dropdown menu showing "Page2", "Report parameter:" with a dropdown menu showing "CategoryName", and "Specify a parameter value:" with an empty text box. Below these is a label "or enter the expression that returns a parameter value:" followed by a text box containing "[Categories.CategoryName]" and a small "fx" icon. The bottom section has a bold heading "What will happen if you click this object in the preview window:" followed by the text "Specified page will be generated and opened in separate preview tab." At the bottom left, there is a checkbox labeled "Modify the object's appearance so it will look like a clickable link". At the bottom right, there are "OK" and "Cancel" buttons.

Bağlantıya tıklandığında şunlar gerçekleşir:

- Rapor parametresine hiperlink değeri aktarılır;
- Belirtilen rapor sayfası oluşturulur ve ayrı bir görüntüleme penceresinde gösterilir.

Rapor parametresinin değeri iki şekilde belirtilebilir:

- Değeri doğrudan atamak;
- Değeri döndürecek bir ifade belirlemek. Bu ifade, rapor çalıştırıldığında, ilgili nesne işlenirken hesaplanır.

Bir rapor sayfası seçildiğinde, o sayfanın **Visible** (Görünür) özelliği **false** olarak ayarlanır. Bu, ana rapor oluşturulurken bu sayfanın atlanacağı anlamına gelir.

Diğer Bağlantı

Bu bağlantı türünü kullanarak fare tıklamasına kendi tepkilerinizi belirleyebilirsiniz. Bunun için, rapor betiğinde yazılabilen nesnenin Click olay işleyicisi kullanılır. Bunu yapmak için:

- Nesneyi seçin ve **Özellikler** penceresini açın;
- Nesne olaylarını göstermek için ilgili düğmeye basın;
- **Click** olayına çift tıklayın. FastReport, **Kod** penceresine geçecek ve boş bir olay işleyicisi oluşturacaktır.

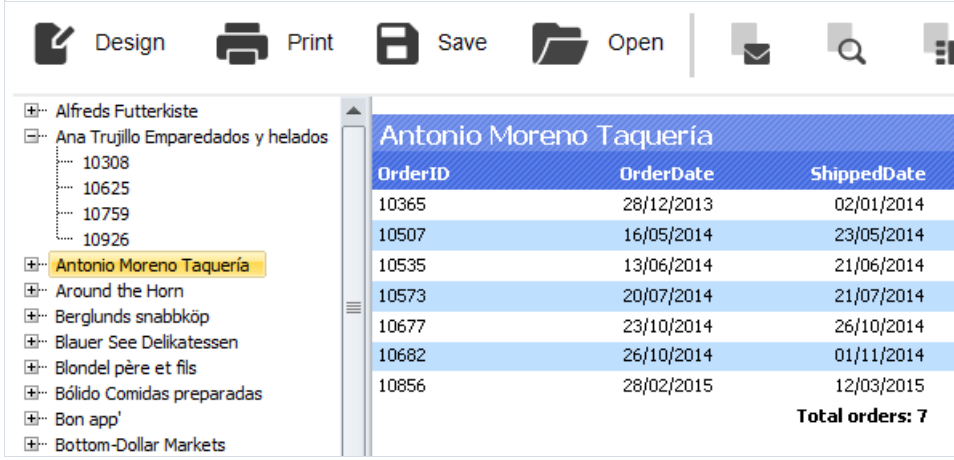
Olay işleyici kodunda istediğiniz işlemleri gerçekleştirebilirsiniz. Muhtemelen, olayı tetikleyen nesneye ve bağlantı parametrelerine erişmeniz gerekecektir. Bunun için sender parametresini kullanın:

```
private void Text2_Click(object sender, EventArgs e)
{
    // sender, tıklanan nesnedir.
    // Bağlantı değerini almak için sender'ı ReportComponentBase tipine dönüştürmeniz gerekir.
    object hyperlinkValue = (sender as ReportComponentBase).Hyperlink.Value;

    // Burada istediğiniz işlemleri gerçekleştirebilirsiniz, örneğin değeri bir ileti kutusunda gösterme:
    MessageBox.Show("Bağlantı değeri = " + hyperlinkValue.ToString());
}
```

Rapor Yapısı

Rapor yapısı (rapor ağacı veya içindekiler olarak da bilinir), görüntüleme penceresinde TreeView türündeki kontrol ögesi olarak gösterilir:



Kontrol ögesi, rapor oluşturulurken oluşturulan ağaç yapısını görüntüler. Ağaçtaki herhangi bir ögeye tıklanıldığında, raporun ilgili ögesine geçiş yapılır.

Raporun bir yapısı varsa, otomatik olarak gösterilecektir. Yapıyı göstermek ya da gizlemek için kontrol panelindeki düğmeye basılır. Ancak rapor yapısı otomatik olarak oluşturulmaz; bunun için raporun belirli ögelerini yapılandırmanız gerekir.

Rapor yapısını oluşturmak için rapor sayfalarının ve bantlarının `OutlineExpression` özelliği kullanılır. Bu özellikte, yapının ögesi için metin döndüren ifade belirtilmelidir. Bu özelliği ayarladığınız bant yazdırılırken aşağıdaki işlemler gerçekleşir:

- `OutlineExpression` özelliğinde belirtilen ifade hesaplanır;
- Hesaplanan metin rapor yapısına öge olarak eklenir;
- Alt bantlar (örneğin, master-detail raporlarda) yazdırılırken rapor yapısı bantların yapısını takip edecek şekilde oluşturulur.

`OutlineExpression` özelliğini yapılandırmak için "Özellikler" penceresinde ilgili bant öncelikle seçilmelidir.

Farklı rapor türleri için yapının nasıl yapılandırılabilirliğine ilişkin bazı yöntemler aşağıda açıklanmıştır:

- Tamamlanmış raporun sayfa listesini yapı içinde göstermek istiyorsanız, rapor sayfasının `OutlineExpression` özelliğini ayarlayın. Genellikle sayfa numarasını döndüren bir ifade kullanılır:

[PageN]

- Tek bir "Veri" bandına sahip "Liste" türündeki raporda, bu bant için `OutlineExpression` özelliğini ayarlayın. İfade olarak bantta yazdırılan herhangi bir veri alanını kullanabilirsiniz;
- İki "Veri" bandına sahip master-detail türündeki raporda, her iki bant için de `OutlineExpression` özelliğini ayarlayın. İfadeler olarak ilgili bantlarda yazdırılan veri alanlarını kullanın. Örneğin, "Kategori/Ürünler" türündeki raporda, ilk bant için kategori adı, ikinci bant için ürün adı ifade olarak belirtilmelidir;
- Gruplama içeren raporda, grup başlığı ve "Veri" bantlarının `OutlineExpression` özelliğini ayarlayın. Grup başlığı için genellikle gruplama koşulu, "Veri" bandı için ise bantta yazdırılan veri alanlarından biri kullanılır.

Kullanım örnekleri

Örnek 1. Web sayfasına bağlantı

Bu örnekte, tek bir "Metin" nesnesinden oluşan basit bir rapor oluşturacağız. Önizleme penceresinde bu nesneye tıkladığınızda, FastReport'un web sayfasına yönlendirileceksiniz.

Yeni bir rapor oluşturun ve içine bir "Metin" nesnesi ekleyin. Nesneye aşağıdaki metni yazın:

Go FastReport home page

Metin nesnesine sağ tıklayın ve açılan menüden "Köprü..." seçeneğini seçin. Bağlantıyı şu şekilde yapılandırın:

Edit Hyperlink

URL

Specify an URL (example: http://www.url.com):

<https://www.fast-report.com/en/product/fast-report-net/>

or enter the expression that returns an URL:

fx

☒ Open link in new tab

What will happen if you click this object in the preview window:

Specified URL will be opened.

☐ Modify the object's appearance so it will look like a clickable link

OK Cancel

Ardından, "Nesnenin görünümünü değiştir..." seçeneğini işaretleyerek bağlantı özelliklerinin (mavi metin rengi, alt çizgi ve fare imlecini el şekli olması) uygulanmasını sağlayın.

Raporu çalıştırın ve nesneye tıklayın. Web tarayıcısı açılarak FastReport web sayfasına yönlendirileceksiniz.

Örnek 2. Ayrıntılı Rapor Oluşturma

Bu örnekte, ürün kategorilerinin listesini içeren bir rapor oluşturacağız. Bir kategori adına tıklandığında, o kategoriye ait ürünlerin ayrıntılı listesini gösteren rapor açılacaktır.

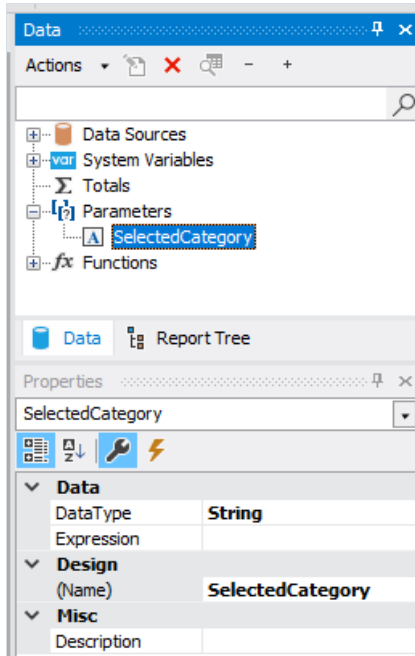
Aşağıdaki adımları gerçekleştirmelisiniz:

- Ayrıntılı raporu oluşturun.
- Ayrıntılı raporda parametre olarak kategori kimliğini (CategoryID) belirleyin.
- Ayrıntılı rapordaki satırların veri filtresini bu parametreye göre ayarlayın.
- Ana raporu oluşturun.
- Ana raporda, seçilen kategori değerini parametre olarak kullanarak ayrıntılı raporu açacak şekilde köprü (hyperlink) oluşturun.

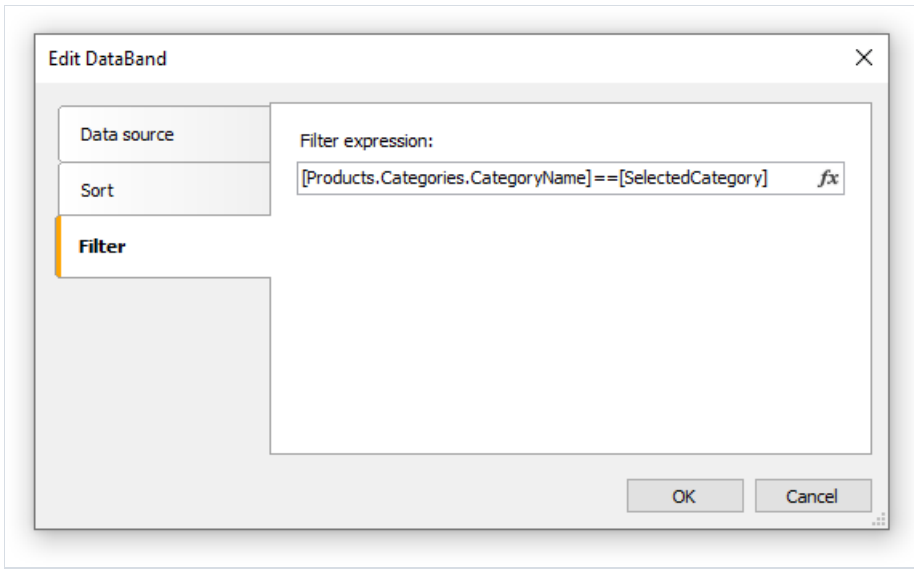
Öncelikle, ürünlerin listesini gösterecek olan ayrıntılı raporu oluşturacağız. Yeni bir rapor oluşturup veri kaynağı olarak Products tablosunu seçin ve nesneleri aşağıdaki gibi yerleştirin:

Report Title	-	Products		
Page Header	-	Product Name	Quantity Per Unit	UnitPrice
Data: Products	-	[Products.ProductName]	[Products.QuantityPerUnit]	[Products.UnitPrice]

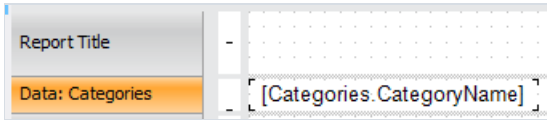
Kategoriye tanımlamak için hem Categories hem de Products tablolarında bulunan CategoryID alanını kullanmak en basit yöntemdir. Seçilen kategori değerini iletmek için bir rapor parametresi oluşturacağız. Parametre ayarlarını aşağıdaki gibi gerçekleştirin:



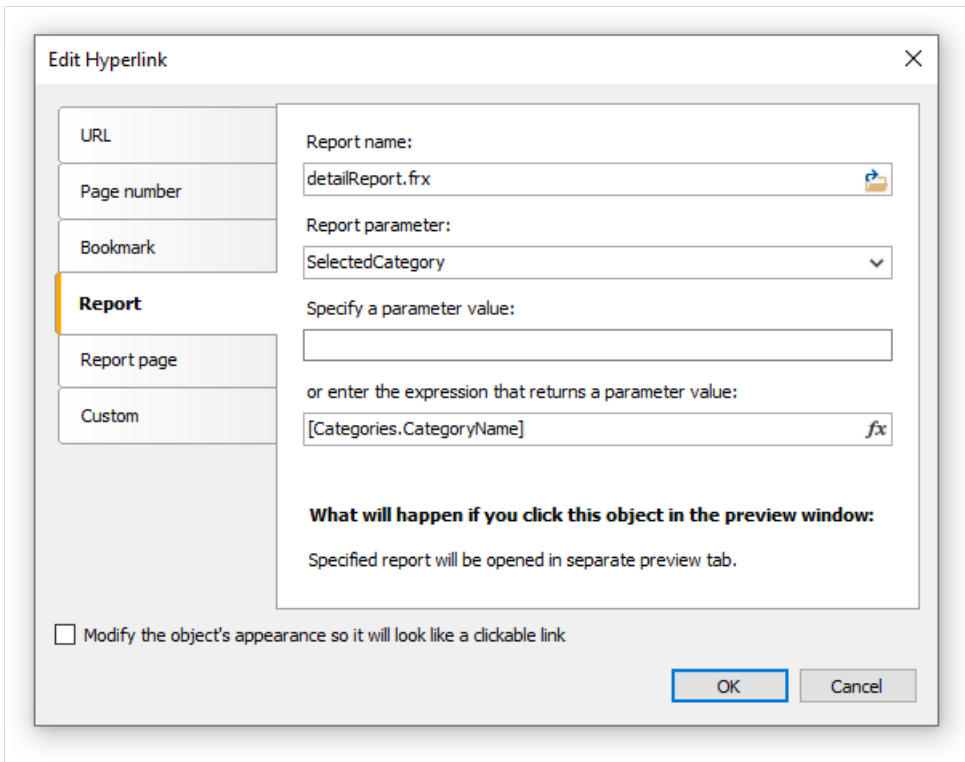
Şimdi seçilen kategoriye ait ürünlerin filtrelenmesi için gerekli koşulu belirlemelisiniz. Bunun için "Veri" bandına çift tıklayın ve "Filtre" sekmesinde aşağıdaki koşulu belirtin ("fx" düğmesine tıklayarak görsel olarak da oluşturabilirsiniz):



Şimdi ana raporu oluşturun. Yeni bir rapor oluşturun ve veri kaynağı olarak "Categories" tablosunu seçin. Nesneleri şu şekilde yerleştirin:



"Metin" nesnesine sağ tıklayın ve "Hyperlink..." menü öğesini seçin. Bağlantıyı şu şekilde ayarlayın:



Rapor adı olarak, detaylı rapor dosyasının adını seçin. Rapor parametresini, listenin sağ tarafındaki düğmeye basarak açılan menüden seçebilirsiniz. Parametre değeri olarak, [Categories.CategoryID] ifadesini belirtin.

Raporu çalıştırın ve kategoriler listesini göreceksiniz:

[Beverages](#)

[Condiments](#)

[Confections](#)

[Dairy Products](#)

[Grains/Cereals](#)

[Meat/Poultry](#)

[Produce](#)

[Seafood](#)

Bir kategorinin üzerine tıkladığınızda, detaylı bir rapor oluşturulacaktır. Bu rapor, önizleme penceresinin ayrı bir sekmesinde gösterilecektir:

Close detailed report Detailed report tab

Preview

Print Save interactive_masterreport x 3

Products

Product Name	Quantity Per Unit	UnitPrice
Pavlova	32 - 500 g boxes	17.45
Teatime Chocolate	10 boxes x 12 pieces	9.2
Sir Rodney's Marmalade	30 gift boxes	81
Sir Rodney's Scones	24 pkgs. x 4 pieces	10
NuNuCa Nuß-Nougat-	20 - 450 g glasses	14
Gumbär Gummibärchen	100 - 250 g bags	31.23
Schoggi Schokolade	100 - 100 g pieces	43.9
Zaanse koeken	10 - 4 oz boxes	9.5
Chocolade	10 pkgs.	12.75
Maxilaku	24 - 50 g pkgs.	20
Valkoinen suklaa	12 - 100 g bars	16.25
Tarte au sucre	48 pies	49.3
Scottish Longbreads	10 boxes x 8 pieces	12.5

Resimde görüldüğü gibi, sekmenin başlığı bağlantının değerine ayarlanmıştır. Bizim durumumuzda, bu, CategoryID veri sütununda bulunan sayısal değeri göstermektedir. Bu, bilgilendirici ve güzel görünmüyor. Raporumuzu, kategori numarası yerine kategori adını kullanacak şekilde değiştirelim. Bunun için şu adımları izleyin:

Detaylı raporda:

- Parametrenin **DataType** özelliğini **String** olarak değiştirin;
- "Categories" veri kaynağını rapora ekleyin. Bu, veri filtreleme sırasında **CategoryName** sütununa başvurmak için kullanılacaktır;
- "Veri" bandının filtreleme ifadesini değiştirin:

```
[Products.Categories.CategoryName] == [SelectedCategory]
```

Ana raporda:

- Köprü ayarlarını değiştirin. Artık rapor parametresine `[Categories.CategoryName]` değerini aktaracağız.

Raporu şimdi çalıştırırsak, sekmenin başlığının kategori adına ayarlandığını göreceğiz. Detaylı raporu biraz daha geliştirebiliriz. Rapor başlığında seçilen kategorinin adını yazdıracak olan bir "Text" nesnesi ekleyin.

Products		
Product Name	Quantity Per Unit	UnitPrice
Pavlova	32 - 500 g boxes	17.45
Teatime Chocolate	10 boxes x 12 pieces	9.2
Sir Rodney's Marmalade	30 gift boxes	81
Sir Rodney's Scones	24 pkgs. x 4 pieces	10
NuNuCa Nuß-Nougat-	20 - 450 g glasses	14
Gumbär Gummibärchen	100 - 250 g bags	31.23
Schoggi Schokolade	100 - 100 g pieces	43.9
Zaanse koeken	10 - 4 oz boxes	9.5
Chocolade	10 pkgs.	12.75
Maxilaku	24 - 50 g pkgs.	20
Valkoinen suklaa	12 - 100 g bars	16.25
Tarte au sucre	48 pies	49.3
Scottish Longbreads	10 boxes x 8 pieces	12.5

Bu örnekle çalışırken iki rapor oluşturduk ve aralarında birkaç kez geçiş yaptık. Bu pek konforlu değil. Görevi kolaylaştırmak için iki raporu tek bir rapora yerleştirebiliriz: ana rapor birinci sayfada, detay raporu ise ikinci sayfada olur. Bu durumda, köprü (hyperlink) aşağıdaki şekilde ayarlanmalıdır:

URL

Page number

Bookmark

Report

Report page

Custom

Report page:

Page2

Report parameter:

SelectedCategory

Specify a parameter value:

or enter the expression that returns a parameter value:

[Categories.CategoryName] *fx*

What will happen if you click this object in the preview window:

Specified page will be generated and opened in separate preview tab.

☒ Modify the object's appearance so it will look like a clickable link

OK

Cancel

Bu durumda, detay rapor sayfası olarak Page2'yi seçmemiz gerekir.

Örnek 3. Etkileşimli "Matris" Nesnesi

Bu örnekte, "Matris" nesnesinin hücreindeki bir değere tıklayarak ayrıntılı raporun nasıl oluşturulacağını inceleyeceğiz. Örnek olarak, çalışanların yıllara göre satış hacimlerini gösteren bir matrisi kullanacağız.

Matris için veri kaynağı olarak, çalışanların satışlarının yıl ve aya göre gruplandırıldığı **MatrixDemo** tablosu kullanılmaktadır:

Name	Year	Month	ItemsSold	Revenue
Nancy Davolio	1999	2	1	1000
Nancy Davolio	1999	11	1	1100
Nancy Davolio	1999	12	1	1200
Nancy Davolio	2000	1	1	1300
Nancy Davolio	2000	2	2	1400
Nancy Davolio	2001	2	2	1500
Nancy Davolio	2001	3	2	1600
Nancy Davolio	2002	1	2	1700
Andrew Fuller	2002	1	2	1800
Andrew Fuller	1999	10	2	1900
Andrew Fuller	1999	11	2	2000
Andrew Fuller	2000	2	2	2100
Janet Leverling	1999	10	3	3000
Janet Leverling	1999	11	3	3100
Janet Leverling	2000	3	3	3200
Steven Buchanan	2001	1	3	4000
Steven Buchanan	2001	2	4	4100
Steven Buchanan	2000	1	4	3999

Bu verilere dayanarak, aşağıdaki bilgileri yazdıran bir matris oluşturacağız:

- Sütun başlıklarında `MatrixDemo.Year` alanı gösterilir;

- Satır başlıklarında `MatrixDemo.Name` alanı gösterilir;
- Hücrelerde `MatrixDemo.Revenue` alanı gösterilir.

Tamamlanmış matris aşağıdaki gibi görünecektir:

Employee	Year				Total
	1999	2000	2001	2002	
Andrew Fuller	3 900,00	2 100,00		1 800,00	7 800,00
Janet Leverling	6 100,00	3 200,00			9 300,00
Nancy Davolio	3 300,00	2 700,00	3 100,00	1 700,00	10 800,00
Steven Buchanan		3 999,00	8 100,00		12 099,00
Total	13 300,00	11 999,00	11 200,00	3 500,00	39 999,00

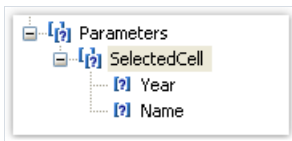
Görüldüğü gibi, hücrelerdeki değerler, çalışanın tüm yıl içindeki satışlarının toplamıdır. Şimdi matristeki bir hücreye tıklandığında daha ayrıntılı bir rapor görüntülenmesini sağlayalım. Bu rapor, "matris hücresindeki değer nereden geldi?" sorusuna yanıt vermelidir. Bizim durumumuzda ayrıntılı rapor, seçilen çalışanın seçilen yıl içindeki aylık satışlarını içerebilir.

Hücreyi, yazdırıldığı verilerle nasıl ilişkilendiririz? Matrisin her hücresinin bir adresi vardır. Bu adres, sütun ve satır başlıklarının değerlerinden oluşur. Örneğimizde hücre adresi, yıl ve çalışanın isminden oluşmaktadır. Bu bilgiler ayrıntılı rapora iletilebilir. Nasıl yapılır? Çok basit: Köprü (hyperlink) ayarlarında yalnızca raporun adını ve parametre adını belirtmeniz yeterlidir. Parametre değerini belirtmenize gerek yoktur: Matris hücreleri için FastReport, bu değerleri otomatik olarak oluşturur ve parametreye aktarır.

Diyelim ki sol üstteki, 3900 değerini içeren hücreye tıkladık. Bu, çalışan "Andrew Fuller"ın 1999 yılı satışlarının toplamıdır. Parametreye değer hangi biçimde aktarılır? FastReport, sütun ve satır değerlerini noktalı virgül kullanarak birleştirir:

1999;Andrew Fuller

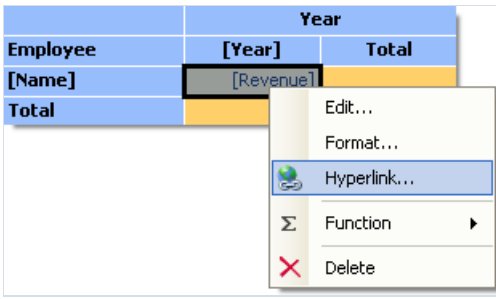
Peki, ayrıntılı raporda bu değerden yılı ve çalışan adını çıkarıp, yılı sayıya dönüştürüp bu değerleri veri filtrelemede mi kullanacağız? Kesinlikle hayır! Ayrıntılı raporda yapmanız gereken tek şey, iç içe parametreleri olan bir parametre oluşturmaktır. Bunun nasıl yapılacağı "Veriler" bölümünde anlatılmıştır. Bu durumda üst parametre aşağıdaki gibi olabilir:



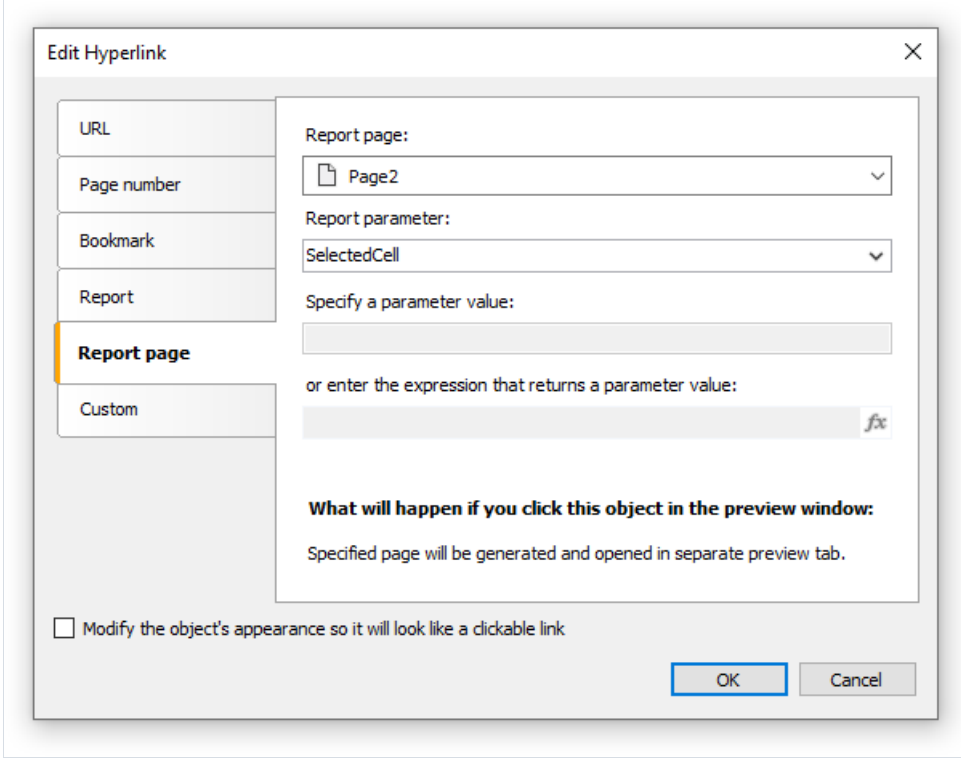
Parametre oluştururken şu hususları göz önünde bulundurun:

- Üst parametre için özel bir ayar gerekmez. Ona anlaşılır bir isim verin, yeterlidir;
- Üst parametrenin matristen gelen değerlerin sayısı kadar iç parametreye sahip olması gerekir. Burada iki değer aktarılmaktadır;
- İç parametrelerin sırası, matristen iletilen değerlerin sırası ile aynı olmalıdır. İlk önce sütun değerleri, sonra matris satırlarının değerleri aktarılır. Bizim durumumuzda ilk parametre yıl, ikinci parametre çalışanın adıdır;
- İç parametreleri istediğiniz gibi adlandırabilirsiniz, ancak matris alanlarının isimleriyle eşleşen isimleri tercih edin;
- Her iç parametre için veri tipini doğru ayarlamak çok önemlidir. Veri tipi, parametreye iletilen değer türüyle eşleşmelidir. Bizim örneğimizde ilk parametre (yıl) tam sayı tipinde (Int32), ikinci parametre (çalışanın adı) ise metin tipinde (String) olmalıdır.

Tüm bu ayrıntıları netleştirdikten sonra raporu oluşturmaya başlayalım. Matris hücresini seçin ve köprü düzenleyicisini açın:

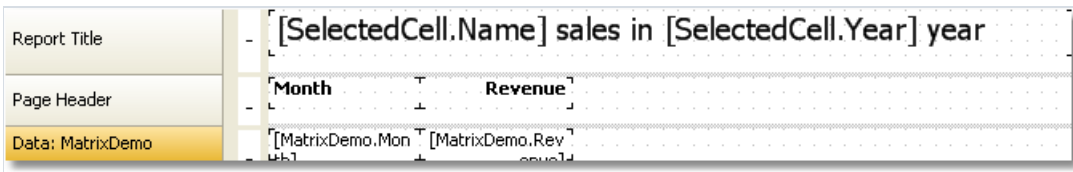


Köprü ayarlarında, parametre olarak üst parametreyi (bizim örneğimizde SelectedCell) belirtmeniz gerekir:



FastReport, değerleri `SelectedCell.Year` ve `SelectedCell.Name` iç parametrelerine kendisi aktaracaktır. Bu değerler parametrelerin veri tipine dönüştürülür; bu nedenle parametre veri tiplerini doğru ayarlamak önemlidir.

Detaylı rapor, ana raporun ayrı bir sayfasında bulunur ve matrisle aynı veri kaynağı olan MatrixDemo'yu kullanır:



Son olarak, seçilen yıl ve çalışanın satışlarını göstermek için verileri filtrelememiz gerekiyor. Bunun için Veri bandının düzenleyicisini açın ve şu filtreleme koşulunu belirtin:

```
[MatrixDemo.Year] == [SelectedCell.Year] && [MatrixDemo.Name] == [SelectedCell.Name]
```

Rapor hazırdır. Çalıştırın ve sol üst hücreye tıklayın. Ayrıntılı rapor açılacak ve aşağıdaki verileri gösterecektir:

Andrew Fuller sales in 1999 year

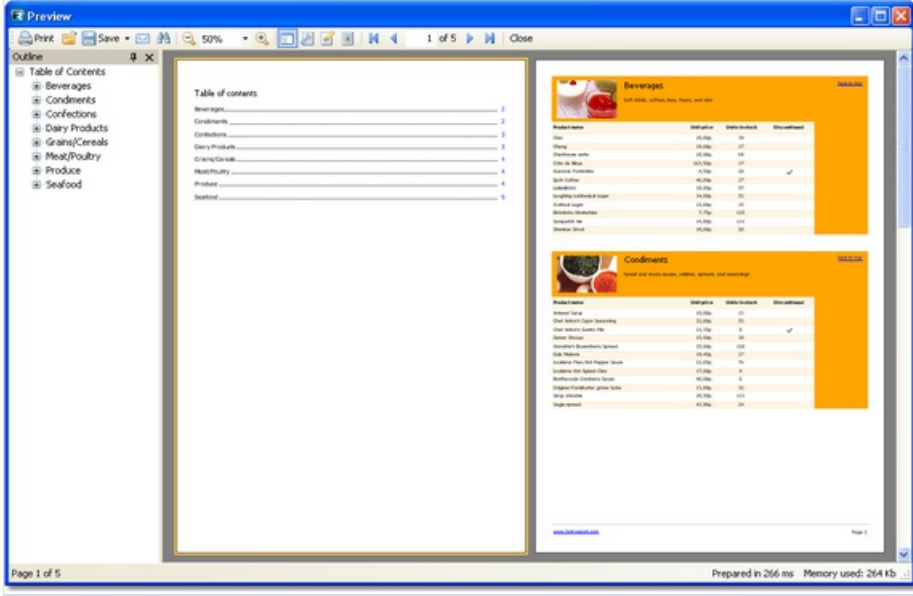
Month	Revenue
10	1 900,00
11	2 000,00

Görüldüğü gibi değerlerin toplamı, üzerine tıklanan matris hücresinin değerine karşılık gelmektedir.

Örnek 4. İçindekiler, Gezinme ve Yapıya Sahip Rapor

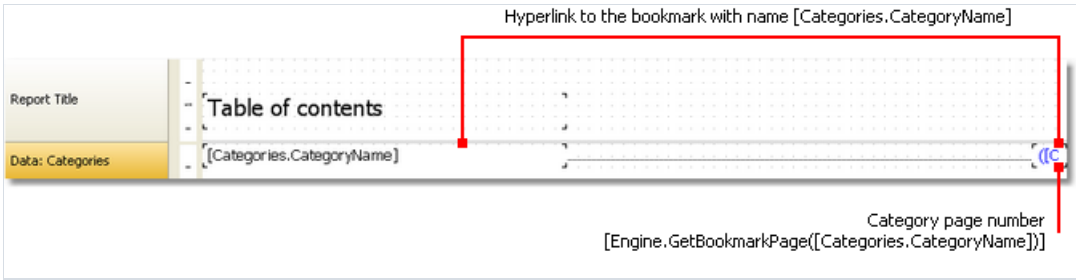
Bu örnekte, aşağıdaki özelliklere sahip bir raporun oluşturulmasını ele alacağız:

- Raporun ilk sayfasında, "İçindekiler" başlığı bulunacak ve buradaki öğelere tıklayarak ilgili sayfaya geçiş yapılabilir.
- Raporun görüntüleme penceresinin yanında, tıklanabilir rapor yapısı görüntülenecek.

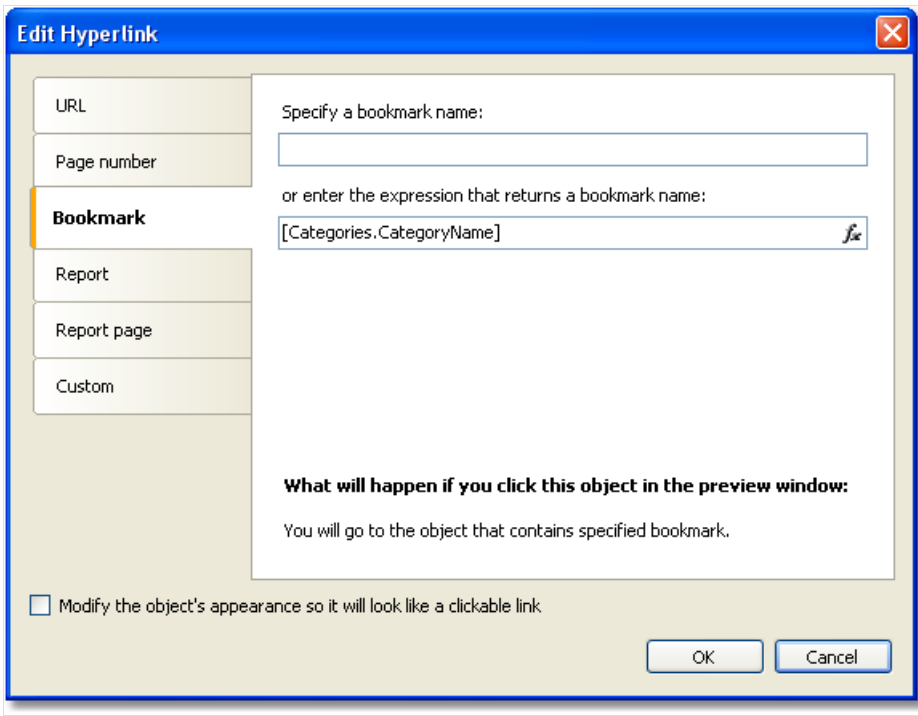


Rapor, Categories ve Products tablolarındaki verileri kullanacaktır. İçindekiler bölümünde kategorilerin listesi basılacak; raporun ana bölümünde ise kategoriler ve ürünlerin listesi yer alacaktır. Rapor şablonu iki sayfadan oluşacaktır: ilk sayfada içindekiler, ikinci sayfada raporun ana kısmı yazdırılacaktır.

İlk olarak içindekileri oluşturalım. Yeni bir rapor oluşturun ve Categories ile Products veri kaynaklarını rapora ekleyin. "Veri" bandını Categories tablosuna bağlayın ve nesneleri aşağıdaki şekilde yerleştirin:



İçindekilerdeki öğeleri interaktif hale getirmek için **Hyperlink** (Köprü) özelliğini yapılandırın:



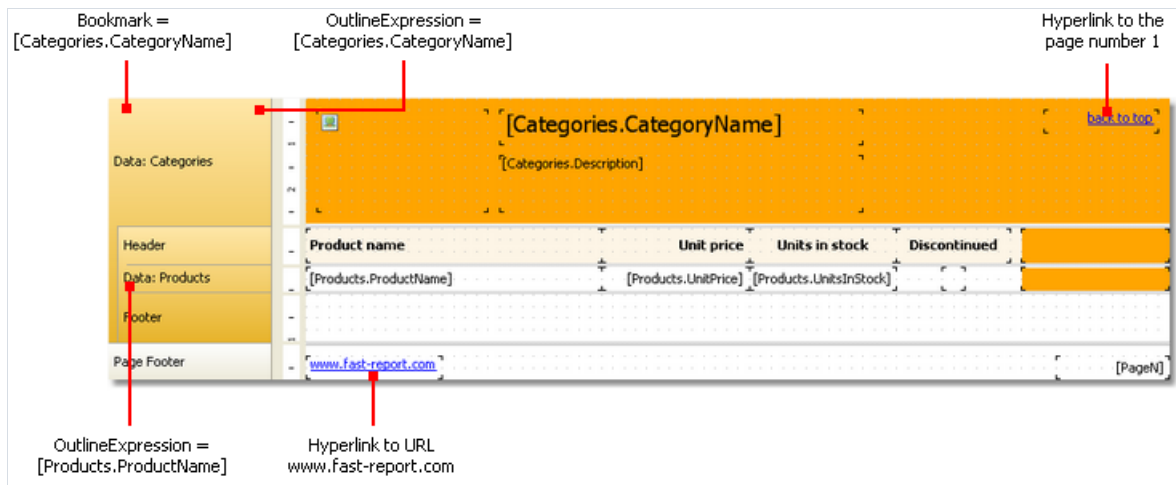
Yer imi (Bookmark) olarak kategori adını belirtin. Yer imi özelliği (Bookmark), daha sonra raporun ikinci sayfasındaki nesneler için belirlenecektir.

İçindekilerde bağlantı verilen sayfanın numarasını yazdırmak için aşağıdaki adımları uygulayın:

- Raporun çift geçişli olmasını sağlayın (bunu "Rapor|Özellikler..." menüsünden yapabilirsiniz). Bu işlem gereklidir, çünkü içindekiler sayfası diğer sayfalardan önce basılmaktadır. İçindekiler basılırken kategorilerin hangi sayfalarda yer alacağı henüz bilinmemektedir.
- Yer imine göre sayfa numarasını döndüren `Engine.GetBookmarkPage` fonksiyonunu kullanın. Bu durumda yer iminin adı `[Categories.CategoryName]` alanında tutulduğu için fonksiyonun çağırımı şu şekilde olacaktır:

```
[Engine.GetBookmarkPage([Categories.CategoryName])]
```

Raporun ikinci sayfasında aşağıdaki gibi bir master-detail (ana-detay) raporu yer alır:



İçindekilerdeki öğeye tıklandığında gidilecek yer imini ayarlamalıyız. Bunun için ilk "Veri" bandını seçin ve Bookmark özelliğini aşağıdaki ifade ile ayarlayın:

```
[Categories.CategoryName]
```

Rapor yapısını ayarlamak için aşağıdaki adımları uygulayın:

- Birinci sayfayı seçin ("Sayfa" nesnesinin kendisini). Bu işlem sayfaya geçiş yapılarak yapılabilir.
- Özellikler penceresinde `OutlineExpression` özelliğini aşağıdaki gibi ayarlayın:

```
"Contents"
```

- Raporun ikinci sayfasına geçin.
- İlk "Veri" bandını seçin ve `OutlineExpression` özelliğini aşağıdaki gibi ayarlayın:

```
[Categories.CategoryName]
```

- İkinci "Veri" bandını seçin ve `OutlineExpression` özelliğini aşağıdaki gibi ayarlayın:

```
[Products.ProductName]
```

Rapor Kalıtımı

Rapor geliştirirken, her raporda aynı verilerin – şirket bilgileri, logolar, aynı form tasarımları vb. – yer aldığı durumlarla sıkça karşılaşırız. Şimdi, onlarca rapor olduğunu ve bazı düzenlemelerin (örneğin, şirket bilgilerinin değiştirilmesi gibi) yapılması gerektiğini düşünün. Bu durumda, her raporu tek tek açıp gerekli düzeltmeleri yapmak zorunda kalırsınız.

Bu tür durumların önüne geçmek için rapor kalıtımını kullanabilirsiniz. Diyelim ki tüm raporlar için ortak olan bazı unsurlarımız var. Bu unsurlar, şirket bilgileri ve logonun bulunduğu rapor başlığı veya sayfa altbilgisi gibi öğeler olabilir. Bu öğelerin tasarımı, şirketiniz için standart olup rapordan rapora değişmez. Ortak unsurları ayrı bir rapor dosyasına (temel rapor) taşımak mümkündür.

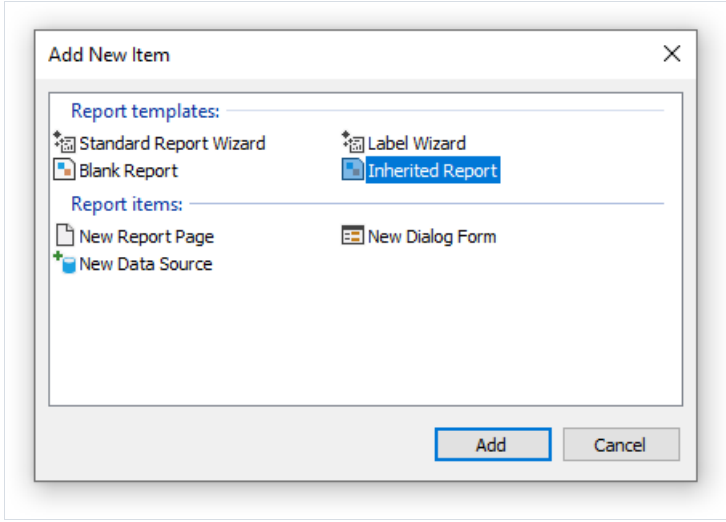
FastReport, temel rapor temel alınarak yeni bir rapor oluşturmanıza olanak tanıyan araçlar sunar. Böylece, yeni rapor temel rapordaki tüm unsurları kendi özel öğeleriyle birlikte içerir. Bu yaklaşım, kapsamlı değişiklikler sırasında zaman tasarrufu sağlar; çünkü sadece temel raporu güncellemek yeterlidir. Temel rapordan türetilen diğer tüm raporlar, yapılan değişiklikleri otomatik olarak "devralır". Bu davranış, kalıtım mekanizması tarafından sağlanır – aslında, türetilmiş bir rapor açıldığında önce temel rapor yüklenir.

Rapor Oluřturma

Kalıtımın kullanılabilmesi için řunları yapmanız gerekir:

- Temel raporu oluřturun ve dosyaya kaydedin;
- Temel rapordan kalıtılmış bir rapor oluřturun.

Kalıtılmış raporu oluřturmak için, "Dosya|Yeni..." menü seçeneğini seçin ve açılan pencerede "Kalıtılmış Rapor"u belirleyin:



Bu işlem sırasında, o ana kadar oluřturulmuş olması gereken temel rapor dosyasını seçmeniz istenecek. Seçim yapıldıktan sonra temel rapor tasarımcıya yüklenecek ve gerekli düzenlemeleri yapabileceksiniz. Görüldüğü üzere, temel rapor nesneleri "kilit" simgesiyle işaretlenmiştir:



Bu, bu tür nesnelerin silinemeyeceği, yeniden adlandırılamayacağı veya başka bir bant üzerine taşınamayacağı anlamına gelir. Yeni nesneler veya bantlar ekleyebilir, mevcut nesnelerin düzenini, boyutlarını ve konumunu değiřtirebilirsiniz. Tüm deęiřiklikleri yaptıktan sonra raporu kaydedin.

Temel Raporun Değiştirilmesi

Temel raporda değişiklik yapıldığında neler olacağını görelim. Yapabilecekleriniz:

- Temel rapordaki bir nesneyi silmek. Bu nesne, kalıtılmış rapordan da silinecektir;
- Temel rapora bir nesne eklemek. Bu nesne otomatik olarak kalıtılmış rapora eklenecektir. Ancak, temel raporda nesnenin eklendiği yer kalıtılmış raporda dolu olabilir;
- Nesnenin boyutlarını, konumunu, metnini ve düzenini değiştirmek. Tüm bu değişiklikler, eğer ilgili nesne kalıtılmış raporda değiştirilmemişse, kalıtılmış rapora yansıtılacaktır.

Son madde açıklama gerektirmektedir. İki kullanım örneğini inceleyelim:

İlk Örnek:

1. Text1 nesnesini içeren bir temel rapor oluşturun;
2. Text1 nesnesinin konumunu değiştirmeden kalıtılmış bir rapor oluşturun ve kaydedin;
3. Temel raporda Text1 nesnesini kaydırın;
4. Kalıtılmış rapor açıldığında, Text1 nesnesinin de kaydırıldığını göreceksiniz.

İkinci Örnek:

1. Text1 nesnesini içeren bir temel rapor oluşturun;
2. Kalıtılmış raporu oluşturun;
3. Kalıtılmış raporda Text1 nesnesini kaydırın ve raporu kaydedin;
4. Temel raporda Text1 nesnesini kaydırın;
5. Kalıtılmış rapor açıldığında, Text1 nesnesinin konumunun değiştirilmediğini göreceksiniz.

Bu durum, kalıtılmış raporda temel rapor nesnesinde değişiklik yapıldığından kaynaklanmaktadır. Yapılan değişiklikler kalıtılmış rapor dosyasına kaydedilmiştir. Artık temel rapordaki orijinal nesne üzerinde yapılan her türlü değişiklik kalıtılmış raporda göz ardı edilecektir. Bu örnekte, Text1 nesnesinin konumu göz ardı edilirken, diğer özelliklerde (örneğin boyut değişikliği) yapılan değişiklikler kalıtılmış raporda yansıtılmaya devam edecektir.

Bu mekanizmanın çalışma prensibi, kalıtılmış rapor dosyasının içeriğine bakıldığında daha iyi anlaşılır. Örneğin, kalıtılmış raporda değiştirilmemiş temel rapor nesnesi şu şekilde kaydedilir:

```
<inherited Name="Text1"/>
```

Nesnenin konumu değiştirilmişse şu şekilde kaydedilir:

```
<inherited Name="Text1" Left="255.15" Top="28.35"/>
```

Kalıtılmış rapor açıldığında, Text1 nesnesine temel raporda tanımlanan tüm özellikler ile kalıtılmış raporda değiştirilen özellikler yüklenecektir.

Kısıtlamalar

Rapor kalıtımı, raporların başlık ve altbilgi gibi ortak unsurlarını ayrı dosyalara çıkarmak ve yeni raporlar oluşturulurken bu dosyaları kullanmak amacıyla geliştirilmiştir. Bu nedenle, kalıtımın daha karmaşık görevler için kullanılması önerilmez, örneğin:

- Bir raporun başka bir kalıtılmış rapordan türetilmesi (yani çift kalıtım uygulanması);
- Temel raporda "Tablo" ve "Matris" gibi karmaşık nesnelerin kullanılması;
- Temel raporda betik kullanılması;
- Temel raporda parametrelerin kullanılması.

Grafikli Raporlar

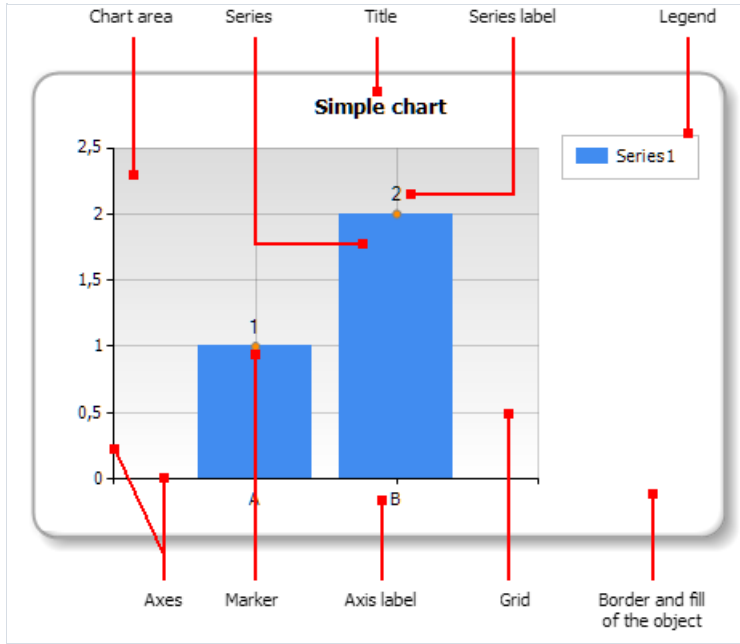
FastReport'da grafik oluşturmak için `System.Windows.Forms.DataVisualization` kütüphanesi kullanılır.

Kütüphanenin özelliklerinin kısa bir özeti:

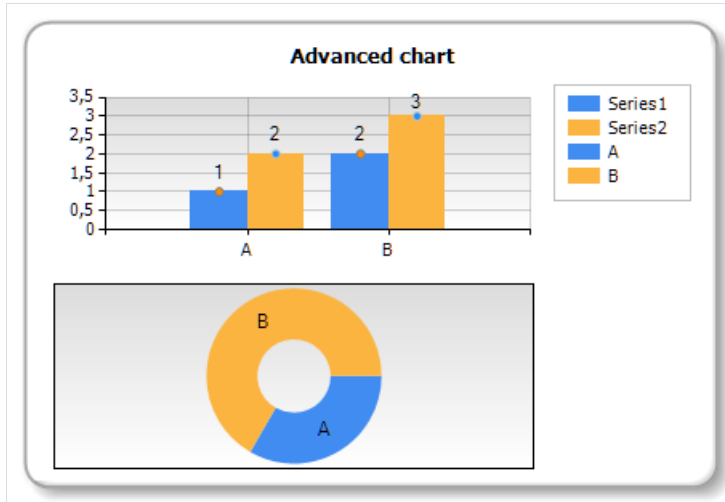
- 30'dan fazla grafik türü (histogram, alan, çizgi, balon, pasta, gül, finansal, piramit, aralık);
- 3D desteği;
- Aynı grafikte farklı tipte birden fazla seri desteği;
- Grafik üzerindeki her öğenin görünümü ve davranışı üzerinde tam kontrol.

Grafik öğeleri

Microsoft Grafik şu öğelerden oluşur:



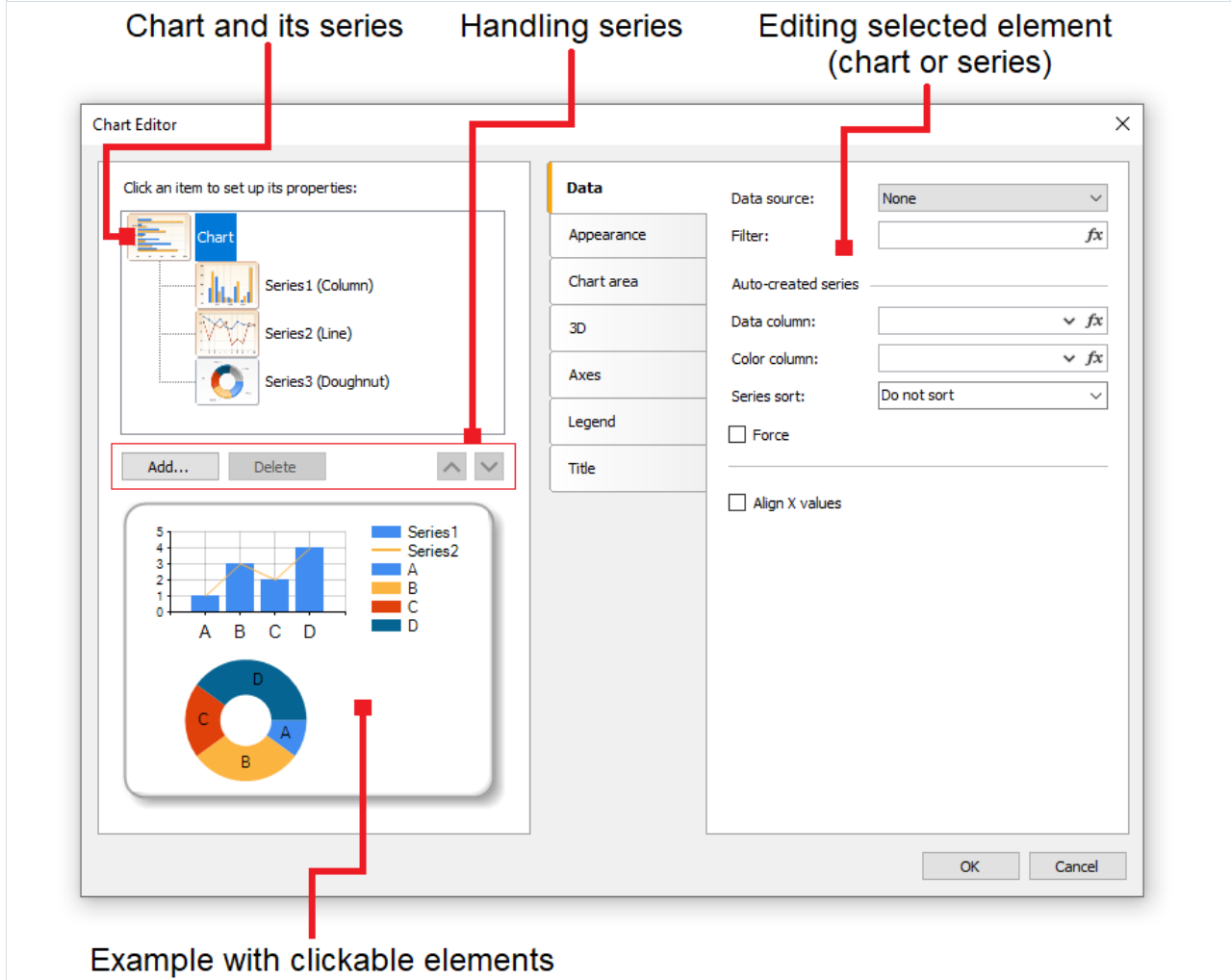
Bir grafik bir veya birden fazla grafik alanına sahip olabilir. Bir grafik alanı bir veya birden fazla seri içerebilir. Aşağıda, iki grafik alanı içeren bir grafik görebilirsiniz (ilk alan iki seri, ikinci alan bir seri içerir):



Bazı seriler (örneğin, pasta serileri) özel grafik alanı gerektirir.

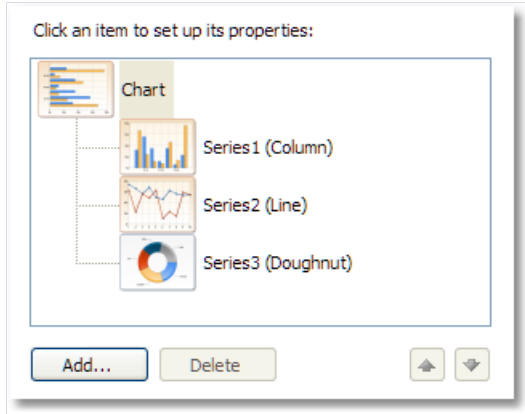
Grafik düzenleyicisi

"Grafik" nesnesi, grafik düzenleyicisinde yönetilebilen birçok ayar içerir. Düzenleyiciyi açmak için, "Grafik" nesnesine çift tıklayın:

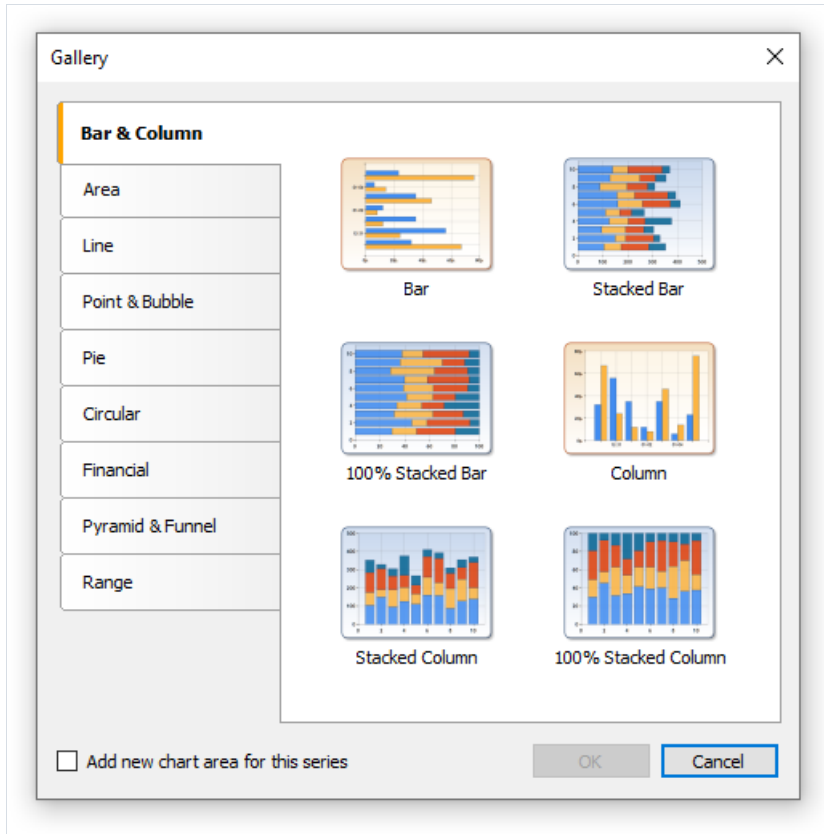


Serileri yönetme

"Grafik" nesnesi bir veya birden fazla seri içerebilir. Seri listesi, düzenleyicide görüntülenir:



Yeni bir seri eklemek için "Ekle..." butonuna basın. "Galeri" penceresini göreceksiniz:



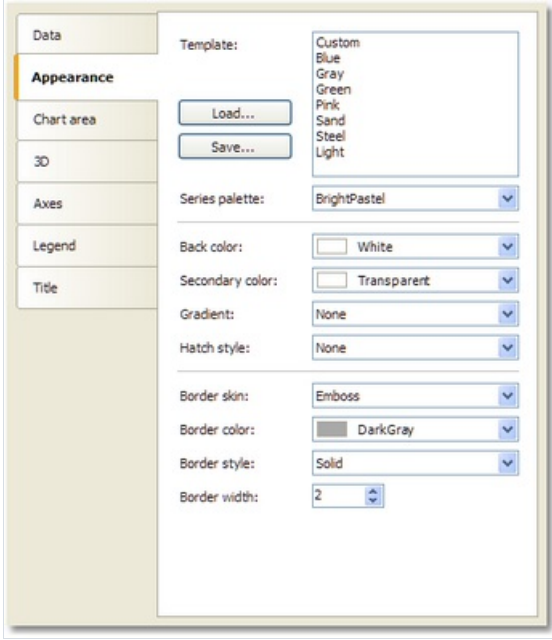
Gerekli kategoriyi seçin, ardından gerekli seri türünü seçin. Eğer seriyi kendi grafik alanına yerleştirmek istiyorsanız, "Bu seri için yeni bir grafik alanı ekle" onay kutusunu işaretleyin. Bazı seri türlerinde (örneğin, pasta, dairesel, finansal, piramidal) yeni grafik alanı, bu onay kutusunun durumuna bakılmaksızın otomatik olarak eklenir.

Seriyi silmek için "Sil" butonuna basın. Seri sırasını değiştirmek için "Yukarı" ve "Aşağı" butonlarını kullanın.

Görünümü ayarlama

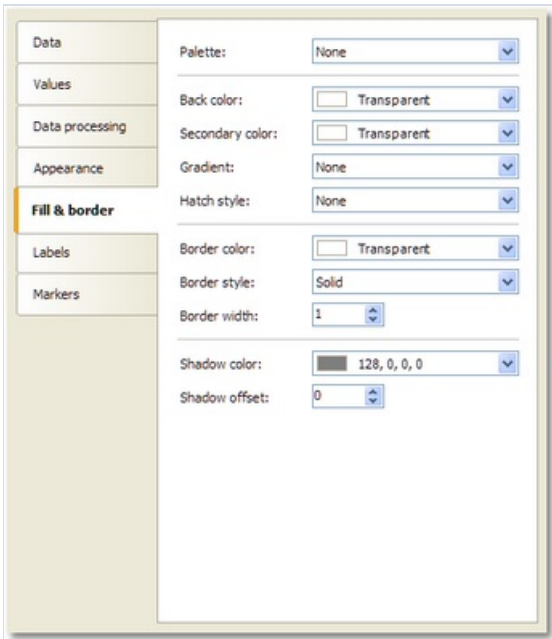
Grafik düzenleyicisini kullanarak, her grafik öğesinin görünümünü ayarlayabilirsiniz. Tüm özellikler (100'den fazla) birkaç kategoriye ayrılmıştır. Bunlardan bazıları "Grafik" nesnesine özgüdür, diğerleri ise serinin bir parçasıdır.

Seri listesinden "Grafik" nesnesini seçerseniz, aşağıdaki özellik sayfalarını göreceksiniz:



- "Özellikler" - grafik sınırı ve dolgu;
- "Grafik alanı" - sınır, dolgu, gölge;
- "3D" - 3D ayarları;
- "Eksenler" - eksenin görünümünü, başlığını, etiketlerini, ızgarayı, işaretçileri, özel etiketleri ve şeritleri ayarlama;
- "Efsane" - efsane stili, yerleşimi, sınır, dolgu, gölge ve yazı tipi;
- "Başlık" - başlığın stili, yerleşimi, sınır, dolgu, gölge, yazı tipi.

Seri listesinden seri nesnesini seçerseniz, aşağıdaki özellik sayfalarını göreceksiniz:



- "Görünüm" - seçilen seri türüne özgü bazı ayarlar;
- "Dolgu ve sınır" - seri değerlerinin dolgusu ve sınırı;
- "Etiketler" - seri etiketleri. Etiket türünü, yazı tipini, rengini ve dolgusunu seçebilirsiniz;
- "İşaretçiler" - seri işaretçileri. İşaretçi türünü, rengini ve sınırını seçebilirsiniz.

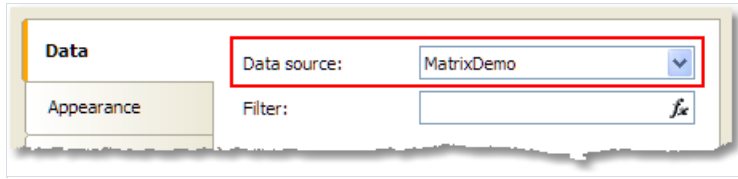
Grafiği veriye bağlama

Grafiği verilerle doldurmanın birkaç yolu vardır:

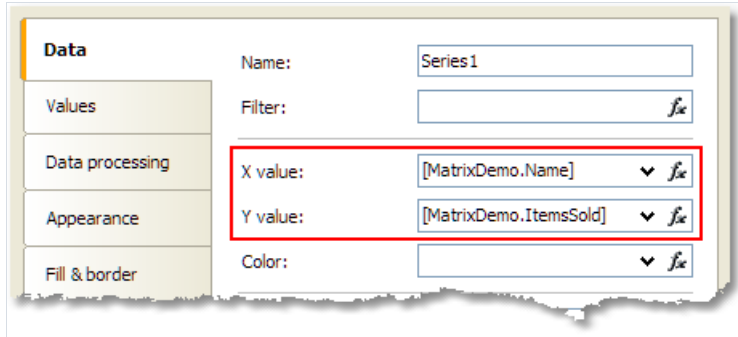
1. Veri kaynağını kullanmak. Bunun için "Grafik" nesnesi için bir veri kaynağı belirtmeniz ve her bir seriyi veri sütunlarına bağlamanız gerekir.
2. Her seri için sabit değerler kullanmak.
3. Verileri bir betik (script) aracılığıyla nesneye doldurmak.

Grafiği bir veri kaynağına bağlamak için aşağıdaki adımları izleyin:

- Seri listesinden "Grafik" nesnesini seçin;
- "Veri" sekmesine geçin;
- Veri kaynağını seçin.

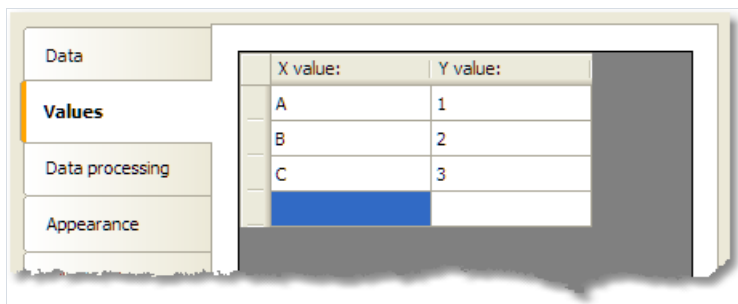


- Gerekirse, veri filtreleme ifadesini ayarlayın. Bu filtre, grafikteki tüm serilere uygulanacaktır.
- Seri listesinden seriyi seçin;
- "Veri" sekmesine geçin;
- Her seri değeri için veri sütunlarını seçin. Seri türüne bağlı olarak, iki veya daha fazla değere sahip olabilir. Çoğu seri türü iki değere sahiptir – X değeri ve Y değeri:



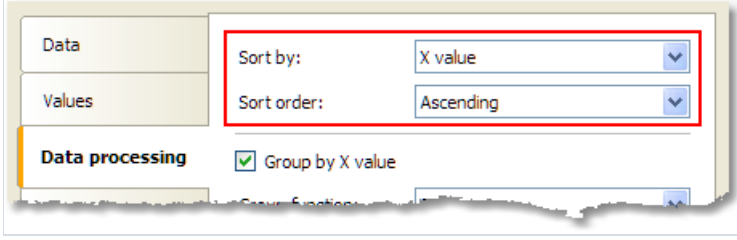
- Gerekirse, veri filtreleme ifadesini ayarlayın. Bu filtre yalnızca mevcut seriye uygulanacaktır;
- "Renk" kontrolünde, renk değeri döndüren bir veri sütunu belirtebilirsiniz.

Ayrıca seri için bir değerler listesi de verebilirsiniz. Bu durumda veri bağlantısına gerek yoktur. Bunu yapmak için, seri listesinden bir seriyi seçin ve "Değerler" sekmesine geçin. Tabloyu değerlerle doldurun:



Verileri sıralama

Varsayılan olarak, grafik nesnesi verileri doğal sırayla görüntüler. Sıralama sırasını değiştirebilirsiniz; bunu yapmak için, seri listesinden seriyi seçin ve "Veri işleme" sekmesine geçin:



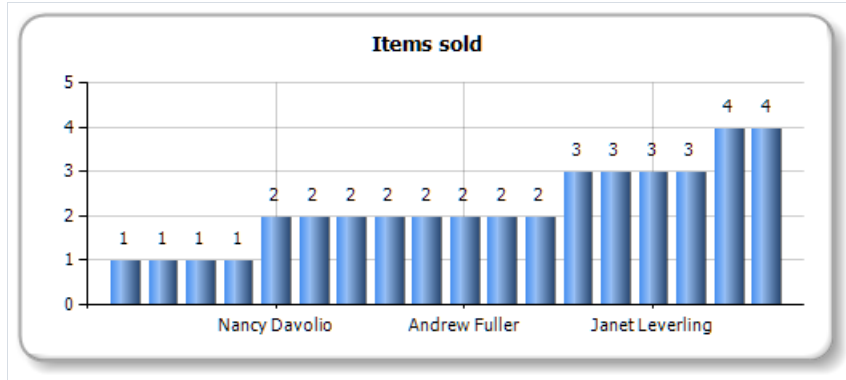
Bir sıralama modu seçebilirsiniz - sıralamayı yapma, X değerine göre sıralama ve Y değerine göre sıralama.

Verileri gruplama

Bazen, serinin birkaç aynı X değeri ile doldurulmasıyla ilgili bir problemle karşılaşırız. Örneğin, grafiklerin gösterilmesi için kullanılan MatrixDemo tablosunda aşağıdaki veriler bulunmaktadır:

Name	Year	Month	ItemsSold	Revenue
Andrew Fuller	2002	1	2	1800
Andrew Fuller	1999	10	2	1900
Andrew Fuller	1999	11	2	2000
Andrew Fuller	2000	2	2	2100
Janet Leverling	1999	10	3	3000
Janet Leverling	1999	11	3	3100
Janet Leverling	2000	3	3	3200
...				

Bu verilerle bir grafik oluşturmaya çalışırsak (örneğin, çalışanın satışları - X değerini İsim sütununa, Y değerini SatılanÜrünler sütununa ayarlarsak), aşağıdaki yanlış sonucu elde ederiz:



Bu durumda, aynı çalışanları tek bir değere gruplandırmamız gerekir. Bunu yapmak için, seri listesinden seriyi seçin ve "Veri işleme" sekmesine geçin. Grup türünü seçin - "X değeri" ve grup fonksiyonu olarak "Toplam"ı seçin:

Data processing

Appearance

Fill & border

Group by:

X value

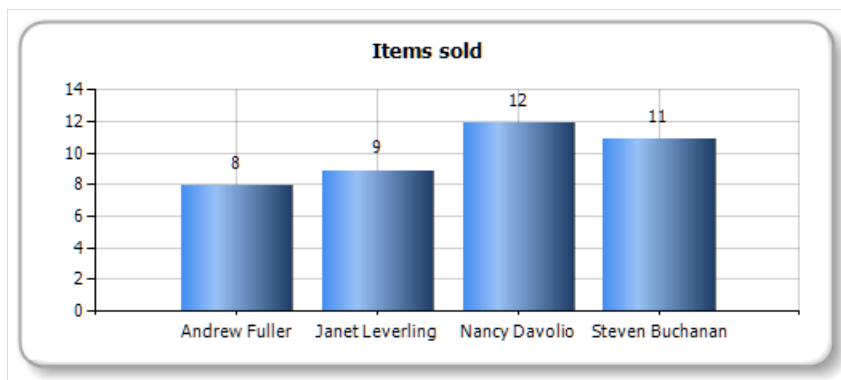
Group interval:

1,00

Group function:

Sum

Sonuç olarak, tüm aynı çalışanlar bir değere gruplandırılacak, satışları toplandıktan sonra özetlenecektir. Aşağıdaki sonucu göreceksiniz:



Değerlerin Birleştirilmesi

Bu veri işleme aracı, çeşitli kriterler kullanılarak bir serideki birden fazla değeri tek bir değere birleştirmenize olanak tanır. Aşağıdaki birleşim seçeneklerinden birini seçebilirsiniz:

Seçenek	Açıklama
En büyük N tanesini tut	Seride en büyük N değer kalır (N miktarı belirlenir). Geri kalan değerler toplanır ve tek bir değer olarak gösterilir (bu değere başlık verilir, genellikle "Diğer" olarak adlandırılır).
En küçük N tanesini tut	Seride en küçük N değer kalır (N miktarı belirlenir). Geri kalan değerler toplanır ve toplanan değere bir başlık atanmadıysa bu değer gösterilmez.
Değerden küçük	Serideki, belirlenen değerden küçük olan tüm değerler tek bir değere birleştirilir.
Yüzdeden küçük	Serideki, toplam sayının belirlenen yüzdesinden küçük olan tüm değerler tek bir değere birleştirilir.
Değerden büyük	Serideki, belirlenen değerden büyük olan tüm değerler tek bir değere birleştirilir.
Yüzdeden büyük	Serideki, toplam sayının belirlenen yüzdesinden büyük olan tüm değerler tek bir değere birleştirilir.

Böylece, en büyük 5 değeri göstermek için seriyi aşağıdaki gibi ayarlayın:

Collect data:

TopN

Value:

5

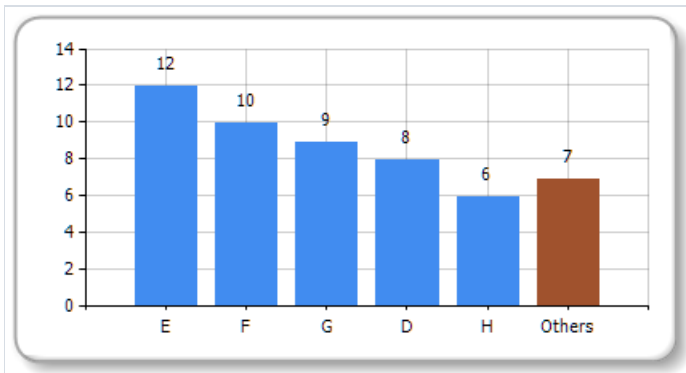
Collected text:

Others

Collected color:

Sienna

Sonuç olarak, yaklaşık şu şekilde bir görünüm elde edersiniz:



Değerleri patlatma

Pasta tipi seriler için bazı değerleri patlatabilirsiniz. Bunu yapmak için, seri listesinden seriyi seçin ve "Veri işleme" sekmesine geçin.

Explode:

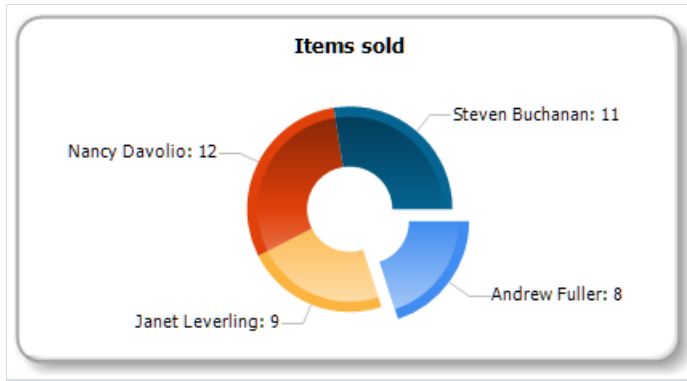
Biggest value

Value:

Aşağıdaki patlatma türlerinden birini kullanabilirsiniz: en büyük değer, en düşük değer ve belirli bir değer. Sonuncu modu seçerseniz, patlatmak istediğiniz bir değeri belirtmeniz gerekir. Bu, herhangi bir ifade olabilir (detaylar için "İfadeler" bölümüne bakınız). Örneğin, Andrew Fuller'ın değerini patlatmanız gerekiyorsa, aşağıdaki ifadeyi kullanabilirsiniz:

"Andrew Fuller"

Aşağıdaki sonucu elde edeceksiniz:



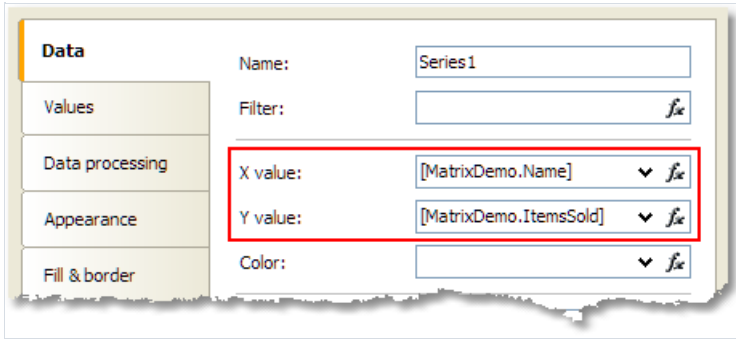
Otomatik Seri Yapılandırması

Grafiği, veri kaynağındaki verilere bağlı olarak yeni seriler oluşturacak şekilde yapılandırabilirsiniz. Otomatik serileri yapılandırmak için şu adımları izleyin:

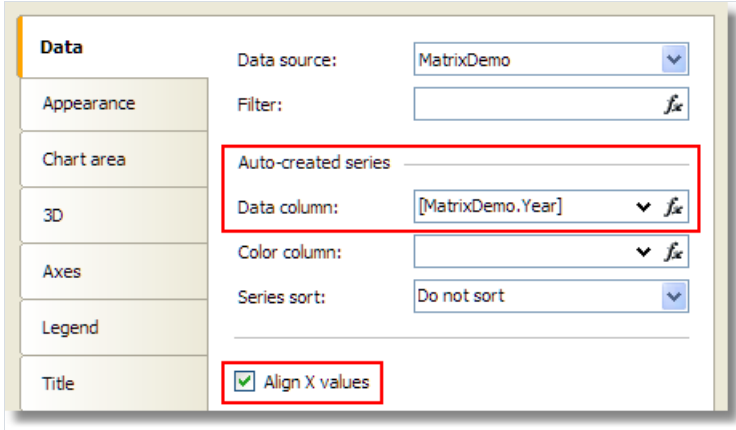
- Bir seri oluşturun ve özelliklerini yapılandırın. Bu seri, tüm yeni seriler için şablon olarak kullanılacaktır;
- **"Chart"** nesnesini seçin ve otomatik seri veri sütununu yapılandırın. Bu sütunun değeri, yeni serinin adı olacaktır. Eğer böyle bir adı taşıyan bir seri henüz yoksa, yeni seri eklenir.

Otomatik serilerin nasıl oluşturulacağını gösterelim. Çalışanların yıllık satışlarını içeren bir grafik oluşturmak için **MatrixDemo** veri tablosunu kullanacağız. Bir seri, bir yılı temsil edecektir. Bunu yapmak için:

- **MatrixDemo** veri kaynağına grafiği bağlayın;
- Bir seri oluşturun ve verilerini yapılandırın:

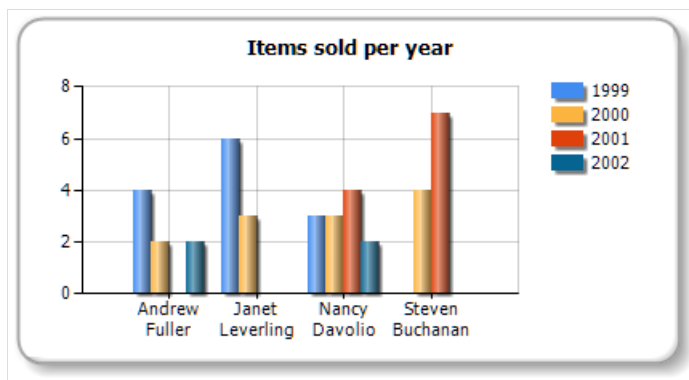


- "Veri işleme" sekmesinde "X değerine göre grupta" onay kutusunu işaretleyin. Bu gereklidir çünkü veri kaynağımızda aynı isme sahip birden fazla çalışan bulunmaktadır;
- Seri listesinde grafiği seçin ve "Veri" sekmesinde otomatik seri sütununu yapılandırın:



- Serilerimiz farklı sayıda değere sahip olabilir (çünkü bazı çalışanların ilgili yılda satışları olmayabilir). Serileri hizalamak için "X değerlerini hizala" onay kutusunu işaretleyin.

Şu sonucu elde edeceğiz:



Etkileşimli grafikler

Grafik, diğer rapor nesneleri gibi etkileşimli olabilir. Grafiği, değerine tıkladığınızda başka bir (detaylı) raporun çalıştırılıp gösterileceği şekilde ayarlayabilirsiniz. Bunu yapmak için, "Etkileşimli raporlar" bölümünde açıklandığı şekilde "Hyperlink" özelliğini ayarlamanız gerekir. Grafik, ögesine tıkladığınızda değeri kendiliğinden bir bağlantıya iletacaktır.

Hadi, FastReport demo programındaki "Charts/Etkileşimli Grafik" raporunu inceleyelim.

İki sayfadan oluşan bir rapor oluşturun. İlk sayfa bir grafik içerecek, ikinci sayfa ise bir grafik değerine tıkladığınızda gösterilecek detaylı raporu içerecek.

İlk rapor sayfasına "Grafik" nesnesini yerleştirin ve grafik düzenleyicisinde özelliklerini ayarlayın:

- Seri listesinden "Grafik" ögesini seçin ve MatrixDemo veri kaynağını seçin;
- Seri listesinden seriyi seçin ve X ve Y değerlerini ayarlayın: `X = [MatrixDemo.Name]` ,
`Y = [MatrixDemo.ItemsSold]` :

- Veri işleme" sekmesine geçin ve grup türünü seçin - "X değeri

İkinci rapor sayfasında, liste tipi bir rapor oluşturun:

- "Veri" penceresinde, SelectedEmployee adında yeni bir rapor parametresi oluşturun;
- Aşağıdaki rapor düzenini oluşturun:

ReportTitle	[SelectedEmployee] orders			
Page Header				
Data: MatrixDemo	[MatrixDemo.Name]	[MatrixDemo.Year]	[MatrixDemo.Month]	[MatrixDemo.ItemsSold]
Report Summary	Total:			[TotalItems]

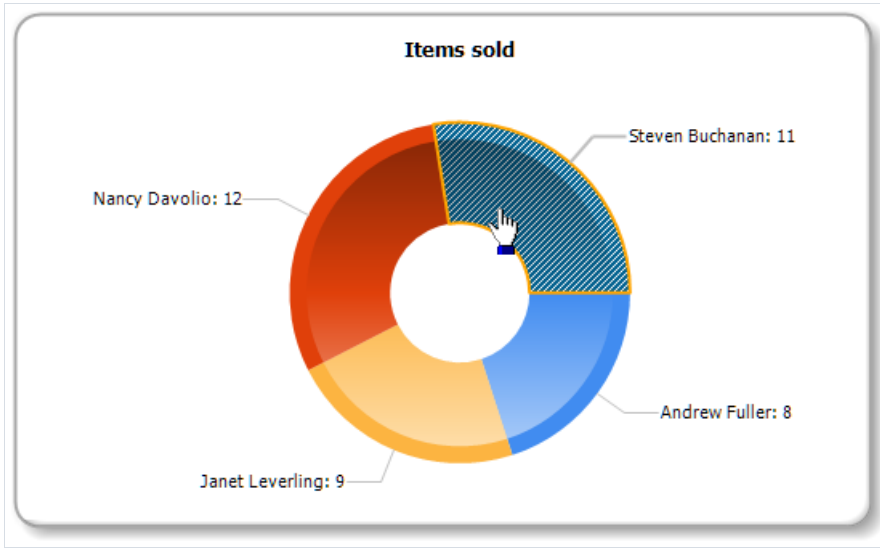
- Veri bantı düzenleyicisini açın ve aşağıdaki filtre koşulunu belirtin:

```
[MatrixDemo.Name] == [SelectedEmployee]
```

Şimdi "Grafik" nesnesinin bağlantısını ayarlayın:

- "Grafik" nesnesinin bağlam menüsünde "Hyperlink..." seçeneğini seçin;
- Bağlantı türünü seçin - "Rapor sayfası";
- İkinci rapor sayfasını seçin ve parametre adını belirtin - SelectedEmployee.

Rapor tamamlandı. Raporu çalıştırın ve fareyi herhangi bir grafik değerinin üzerine getirin. Bu değer görsel olarak seçilecektir ve fare imleci şekil değiştirecektir:



Değere tıkladığınızda, aşağıdaki detaylı raporu göreceksiniz:

Steven Buchanan orders			
Name	Year	Month	ItemsSold
Steven Buchanan	2001	1	3
Steven Buchanan	2001	2	4
Steven Buchanan	2000	1	4
Total:			11

Haritalı Raporlar

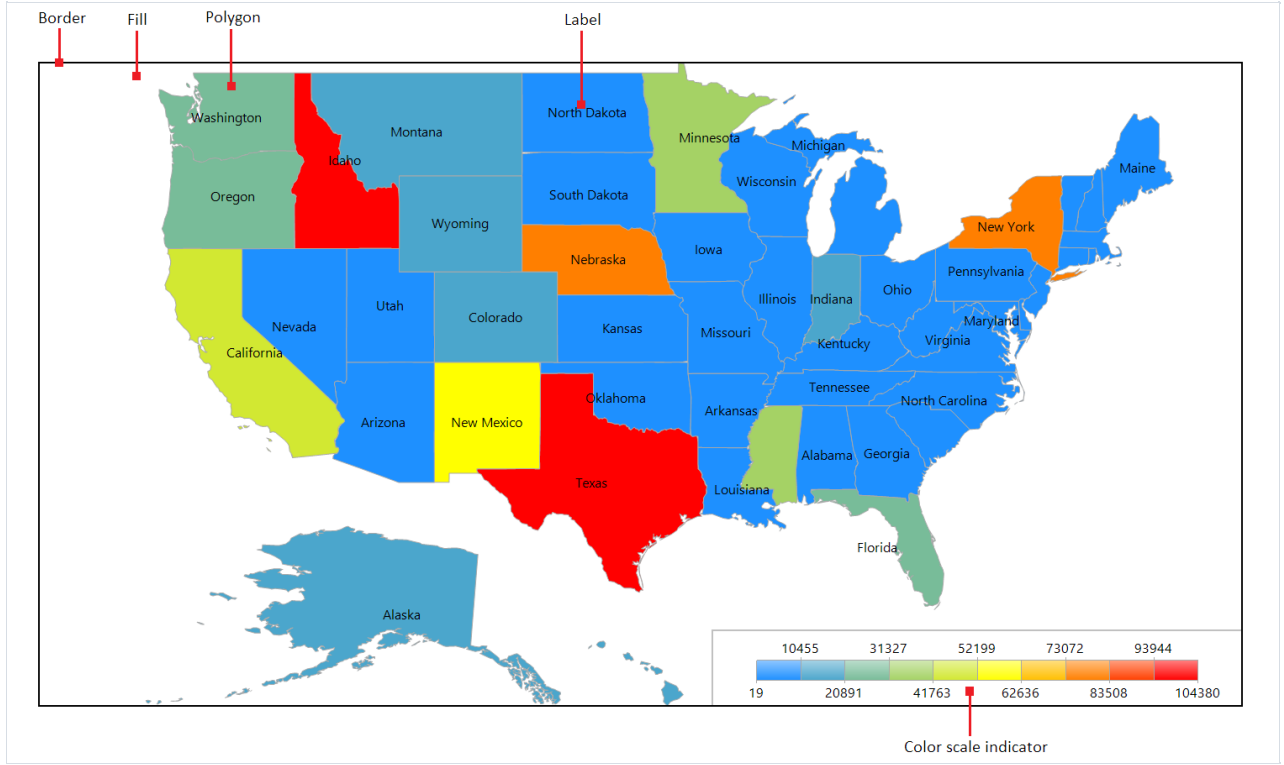
MapObject bileşeni, ESRI shapefile formatındaki iki boyutlu grafik haritaların görüntülenmesi için tasarlanmıştır. Format hakkında daha fazla bilgiyi buradan okuyabilirsiniz:

<https://tr.wikipedia.org/wiki/Shapefile>

Çalışmanın düzgün çalışması için .shp (geometri) ve .dbf (açıklama) dosyalarına ihtiyaç vardır.

Haritanın Bileşenleri

"Harita" nesnesi aşağıdaki bileşenlerden oluşmaktadır:



Bir "Harita" nesnesi bir veya birden fazla katman gösterebilir. Her katman ayrı bir haritayı içermektedir.

Görüntüleme Yönetimi

Tasarımcı modunda ve tamamlanmış raporun önizleme penceresinde, fareyi kullanarak haritanın görüntülenmesini yönetebilirsiniz:

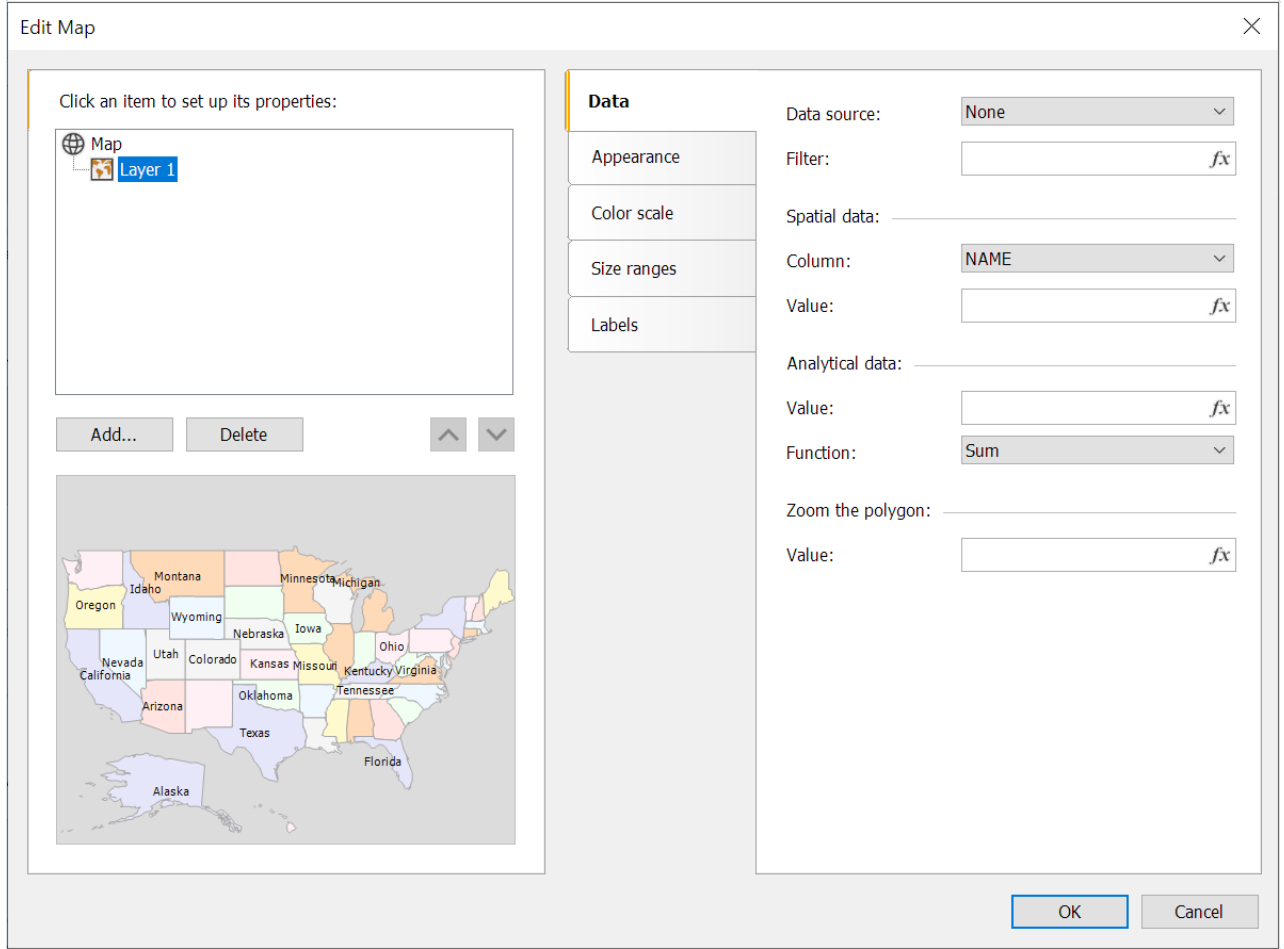
- Fare tekerleği haritanın yakınlaştırma düzeyini değiştirir;
- Fare sol tuşuna basılı tutarak haritayı hareket ettirebilirsiniz;
- Poligonun içine tıkladığınızda, açılan "Özellikler" penceresinden poligonun özelliklerini ayarlayabilirsiniz.

Yakınlaştırma için minimum ve maksimum değerler, **MinZoom** ve **MaxZoom** özelliklerinde belirlenir. Bu değerleri "Özellikler" penceresinden ayarlayabilirsiniz.

ASP.NET uygulamasında harita kontrolü yapılamaz.

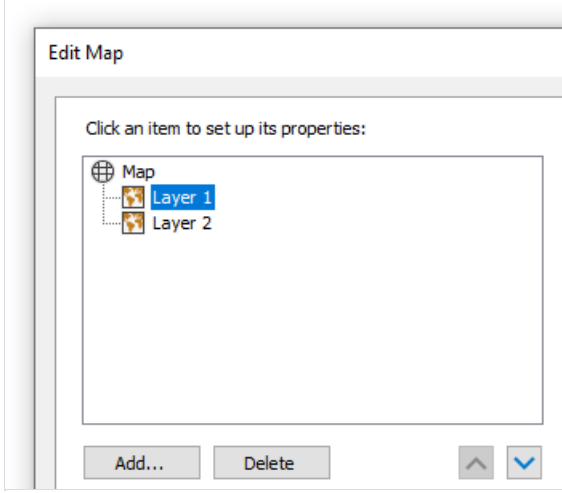
Harita Düzenleyici

"Harita" nesnesi, nesne düzenleyicisi kullanılarak yönetilebilen çok sayıda ayar içerir. Düzenleyiciyi açmak için nesneye fare ile çift tıklayın veya bağlam menüsünden "Düzenle..." seçeneğini kullanın:

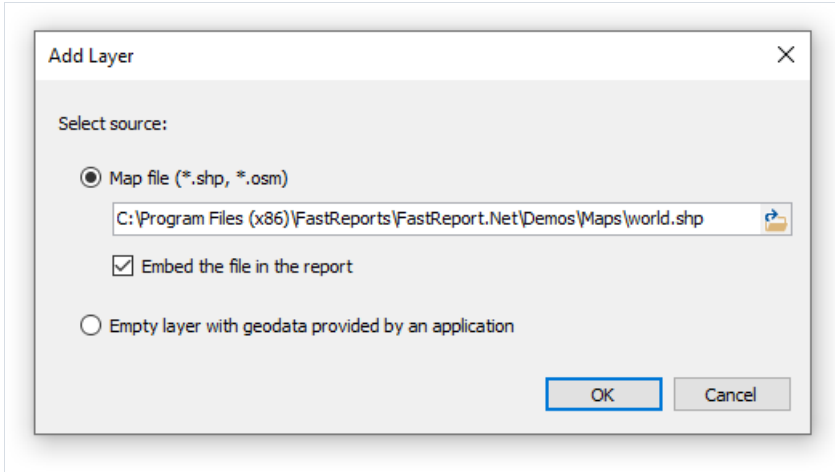


Katman Ekleme

"Harita" nesnesi bir veya birden fazla katman içerebilir. Katmanların listesi düzenleyicinin sol üst köşesinde görüntülenir:



Yeni bir katman eklemek için "Ekle..." düğmesine basın. Aşağıdaki pencere açılacaktır:



Bu aşamada katman türünü seçmeniz gerekir:

- shapefile (.shp/.dbf) dosyasından harita. Bu en sık kullanılan harita türüdür. Örneğin, dünya haritasını yazdırabilir ve satışların belirli bir değerin üstünde olduğu ülkeleri renkle vurgulayabilirsiniz;
- uygulamadan gelen kartografik veriler. Uygulamanız haritada nokta olarak gösterilecek coğrafi koordinatları (enlem ve boylamdan oluşan bir çift değer) sağlamalıdır. Nokta, verilere bağlı olarak farklı boyutta ve/veya renkte olabilir ve ayrıca bir etikete sahip olabilir. Bu katman türü genellikle ikinci katman olarak kullanılır (ilk katman, temel katman olarak shapefile dosyalarından alınır). Örneğin, temel katman belirli bir ülkenin haritasını gösterirken, ikinci katman satış yapılan şehirlerin isimleriyle birlikte noktaları gösterir. Noktanın boyutu ve rengi, şehirdeki satış düzeyini anlaşılır hale getirecek şekilde ayarlanabilir.

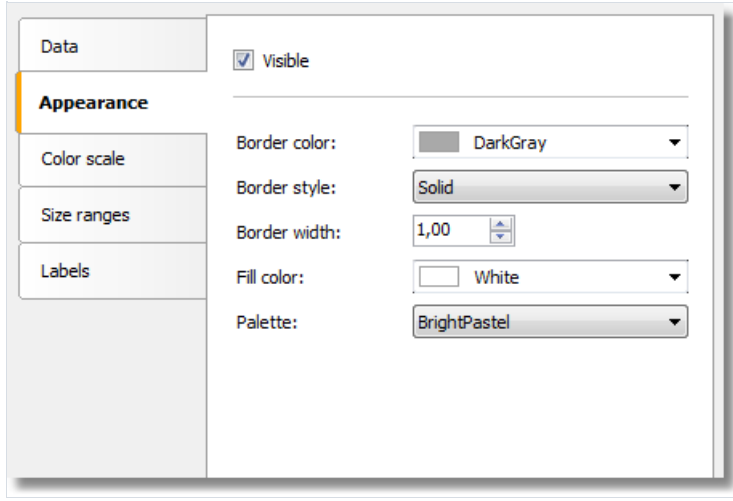
Shapefile tabanlı bir katman seçtiyseniz, harita verilerinin nasıl saklanacağını ayrıca belirtmeniz gerekir:

- veriler rapor dosyasına gömülür. Bu durumda rapor dosyasının boyutu önemli ölçüde artabilir.
- rapor dosyası shapefile dosyalarına referans verir, gömme işlemi yapılmaz. Aynı haritaları kullanan birden fazla raporunuz varsa bu mod faydalıdır.

Büyük boyutlu (30MB üzeri) veya çok sayıda poligon içeren (20.000 üzeri) haritalar raporun çalışmasını önemli ölçüde yavaşlatabilir.

Dış Görünüm Ayarları

Katmanın dış görünümünü ayarlamak için, katmanı seçtikten sonra "**Dış Görünüm**" sekmesine geçin:



The screenshot shows a software interface for configuring the appearance of a map layer. On the left is a sidebar with five tabs: 'Data', 'Appearance', 'Color scale', 'Size ranges', and 'Labels'. The 'Appearance' tab is selected and highlighted with an orange bar. To the right of the sidebar, there is a main panel. At the top of this panel is a 'Visible' checkbox, which is checked. Below this, there are several settings: 'Border color' is set to 'DarkGray' (with a color swatch), 'Border style' is set to 'Solid' (with a dropdown arrow), 'Border width' is set to '1,00' (with a numeric input field and up/down arrows), 'Fill color' is set to 'White' (with a color swatch), and 'Palette' is set to 'BrightPastel' (with a dropdown arrow).

Burada harita poligonlarının çerçeve rengini ve stilini ayarlayabilir, ayrıca renk paletini seçebilirsiniz. Analitik verilere bağlı olarak poligonları renklendirme ayarını yaptıysanız (bu konuya daha sonra değinilecektir), seçilen renk paleti dikkate alınmaz.

Gösterilen Değerlerin Ayarlanması

Haritada, örneğin dünya haritasındaki ülke isimleri gibi çeşitli yazılar görüntülenebilir. Yazıların türünü ve görünümünü, "**Yazılar**" sekmesinden ayarlayabilirsiniz:

- Eğer katman türü olarak **shapefile dosyasından gelen harita** seçilmişse, yazıların alınacağı alanı belirtmeniz gerekir. Genellikle bu alan "**NAME**" olarak belirlenmiştir. FastReport demo programıyla birlikte gelen dünya haritası için aşağıdaki alanlardan seçim yapılabilir:
 - **NAME** (örneğin, Germany)
 - **ABBREV** (örneğin, Ger.)
 - **ISO_A2** (örneğin, DE)
 - **ISO_A3** (örneğin, DEU)

Diğer haritalar için bu alanların listesi farklılık gösterebilir.

- Eğer katman türü olarak **uygulamadan gelen kartografik veriler** seçilmişse, bu pencerede yazıların gösterileceği minimum ölçek değerini belirleyebilirsiniz. Varsayılan değer **1**'dir; bu değer yazıların her zaman gösterileceği anlamına gelir.

Verilere Bağlantı

Çoğu raporda, analitik verileri — örneğin çeşitli ülkelerdeki satış seviyelerini — göstermek için "Harita" nesnesi kullanılır. Bu nedenle, bir harita katmanının bir veri kaynağına bağlanması gerekir. Bunun için harita nesnesi düzenleyicisi penceresi kullanılır. Katmanlar ağacından bir katman seçin ve "Veri" sekmesine geçin. Harita katmanı türüne uygun veriler sağlanmalıdır:

- Harita katmanı türü "ESRI shapefile" olduğunda, "Veri" sekmesi şu şekilde görünür:

The screenshot shows the 'Data' tab of a configuration window. On the left, there are tabs for 'Data', 'Appearance', 'Color scale', 'Size ranges', and 'Labels'. The 'Data' tab is active. The 'Data source' is a dropdown menu set to 'Sales'. Below it is a 'Filter' field. The 'Spatial data' section has a 'Column' dropdown set to 'NAME' and a 'Value' field set to '[Sales.Country]'. The 'Analytical data' section has a 'Value' field set to '[Sales.SalesTotal]' and a 'Function' dropdown set to 'Sum'. The 'Zoom the polygon' section has a 'Value' field.

Bu durumda uygulamanın aşağıdaki verileri sağlaması gerekir:

- isim (örneğin, ülke adı);
- sayısal değer (örneğin, söz konusu ülkedeki satış miktarı).

Örneğin, aşağıdaki alanlara ve verilere sahip **Sales** isimli bir tablonuz olsun:

Country	SalesTotal
USA	500000
Germany	1200000
Russia	300000

Bu durumda verileri aşağıdaki gibi yapılandırmanız gerekir:

- **Veri kaynağı** – Sales;
- **Mekansal veriler, Alan** – shapefile dosyasındaki ülke adını belirten alanı seçin. Genellikle bu alan **NAME** olarak geçer;
- **Mekansal veriler, Değer** – [Sales.Country] ;
- **Analitik veriler, Değer** – [Sales.SalesTotal] ;
- **Analitik veriler, İşlev** – "Toplam". Bu işlev, aynı ülkeye ait birden fazla kayıt varsa ve bunların değerleri farklıysa kullanılır.

"Poligonu büyüt" alanı, belirtilen isme sahip poligonun "Harita" nesnesinin tamamını kapsayacak şekilde

büyütülmesini sağlar. Örneğin, dünya haritasında Almanya ülkesini büyütmek istiyorsanız, bu alana "Germany" değerini (tırnak işaretleriyle birlikte) yazmalısınız.

- for map layer type "geodata from an application" the "Data" tab looks like:

The screenshot shows a configuration window for a map layer. On the left is a sidebar with tabs: 'Data' (selected), 'Appearance', 'Color scale', 'Size ranges', and 'Labels'. The main area is divided into sections. The 'Data source' is set to 'Sales'. Below it is a 'Filter' field. The 'Spatial data' section has fields for 'Latitude' (set to [Sales.Latitude]), 'Longitude' (set to [Sales.Longitude]), and 'Label' (set to [Sales.CityName]). The 'Analytical data' section has a 'Value' field (set to [Sales.SalesTotal]) and a 'Function' dropdown (set to 'Sum'). Each input field has a small 'fx' icon to its right.

Ve gereken veriler şunlardır:

- spatial data - enlem ve boylam;
- label (örneğin: bir şehir adı);
- numeric value (örneğin: bu şehirdeki satış hacmi).

Örneğin, "Satışlar" veri kaynağı bu verilere sahip olabilir:

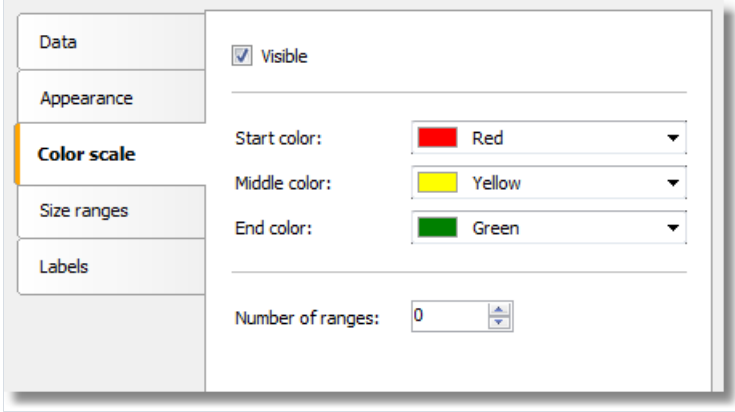
Latitude	Longitude	CityName	SalesTotal
48.13641	11.57753	München	50000
50.94165	6.95505	Köln	36000

Bu durumda verileri aşağıdaki gibi yapılandırmak gerekir:

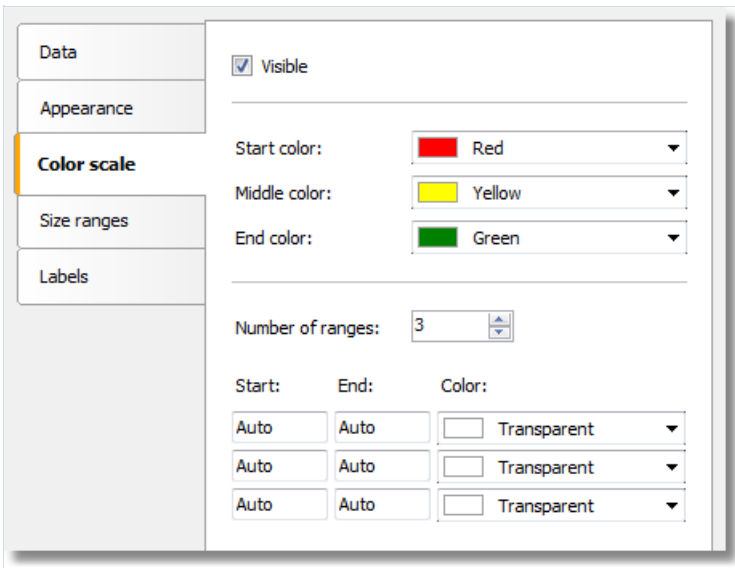
- Veri kaynağı – **Sales**
- Mekânsal veriler, enlem – **[Sales.Latitude]**
- Mekânsal veriler, boylam – **[Sales.Longitude]**
- Mekânsal veriler, etiket – **[Sales.CityName]**
- Analitik veriler, değer – **[Sales.SalesTotal]**
- Analitik veriler, fonksiyon – **"Toplam"**. Bu fonksiyon, belirli bir şehir için birden fazla kayıt olduğunda farklı değerleri toplamak amacıyla kullanılır.

Verilerin Renkle Gösterimi

Katman verilere bağlandıktan sonra, analitik bilgilerin (örneğin, farklı ülkelerdeki satış hacimleri) nasıl görüntüleneceği sorusu ortaya çıkar. En basit yöntem, ülke isimlerinin yanında satış rakamlarının da gösterildiği etiket görüntüleme ayarlarını yapılandırmaktır (bkz. "Görüntülenen Değerlerin Yapılandırılması" bölümü). Ancak, ülkeleri satış hacmine göre belirli renklerle göstermek çok daha anlaşılır bir yöntemdir. Bunun için renk skalasının yapılandırılması gerekir. Bu işlem "Renk Skalası" sekmesinden yapılabilir:



Renk skalası; minimum değer, maksimum değer ve renk olmak üzere bir dizi değerden oluşur. Bu dizilerden (aralıklardan) birden fazla olabilir. Renk skalasını yapılandırmak için kaç tane aralık olacağını belirleyip, ardından her aralık için minimum ve maksimum değerleri ve kullanılacak renkleri ayarlamak gerekir.

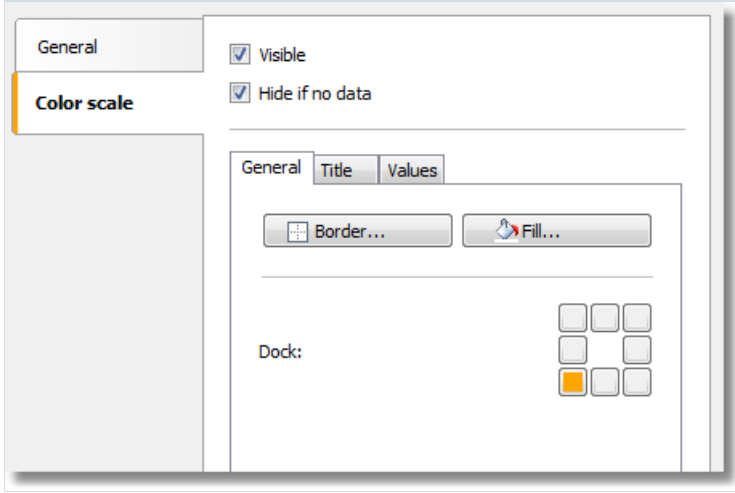


Varsayılan olarak, tüm değerler "Otomatik" ayarlıdır. Bu durumda FastReport her aralık için minimum ve maksimum değerleri otomatik olarak hesaplar ve renkleri önceden belirlenmiş olan "Başlangıç Rengi", "Orta Renk" ve "Son Renk" seçeneklerinden alır. Bu mod genellikle kullanılabilir.

Renk skalası yapılandırıldıktan sonra, haritanın alt kısmında farklı renklerde dikdörtgenlerden oluşan bir gösterge şeridi belirir:



Gösterge şeridinin görünümünü ve konumunu, katman listesinden "Harita" ögesini seçip "Renk Skalası" sekmesine geçerek yapılandırabilirsiniz.



Highlighting data using size

When using layer type "geodata from an application", the data is displayed as a small circle with a caption. The size of the circle can be bound to the data in the same way as highlighting data using colors. To do this set up the size ranges on the "Size ranges" tab:

Data		
Appearance		
Color scale		
Size ranges		
Labels		
Start size:	4	
End size:	20	
Number of ranges:	3	
Start:	End:	Size:
Auto	Auto	Auto
Auto	Auto	Auto
Auto	Auto	Auto

The size ranges have the following properties: min value, max value and size (in pixels). You can use as many ranges as you need. Set the number of ranges first and then set the properties for each range.

By default all range properties are set to "Auto", meaning FastReport calculates the minimum and maximum values for each range automatically. The auto size is chosen from two presets ("Start size", "End size"). The "Auto" mode may be suitable in most cases.

Data

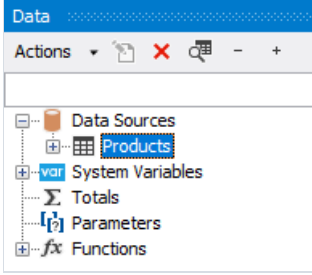
Any report prints some data. In FastReport, you can operate with the following data:

- data sources;
- system variables;
- total values;
- report parameters;
- expressions, containing any of the above mentioned data.

In this chapter, we will look at how to work with these data.

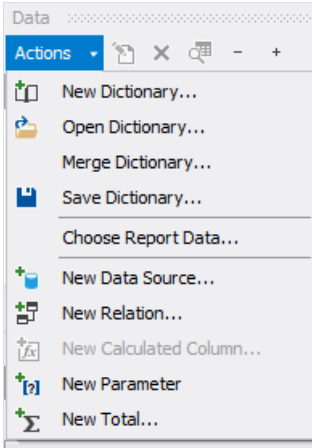
Veri Penceresi

Rapor içerisinde kullanılacak tüm verilere tek bir yerden, yani "Veri Penceresi" aracılığıyla erişilebilir. Bu pencere, "Veri|Veri Penceresini Göster..." menü seçeneği ile açılabilir.



'Veri' (Data) penceresi, tüm veri öğeleriyle işlem yapmanıza ve bunları rapor sayfasına sürükleyip bırakmanıza olanak tanır.

Tüm işlemler, araç çubuğu ve 'Eylem' (Action) menüsü yardımıyla gerçekleştirilebilir:



Bu işlemlerin bir kısmı, 'Veri' (Data) penceresinin bağlam menüsünde de mevcuttur.

Örneğin, bir veri kaynağını seçtiğinizde, bağlam menüsünü kullanarak hesaplanmış bir sütun oluşturabilir, veri kaynağını silebilir veya verilerini görüntüleyebilirsiniz.

Veri Kaynakları

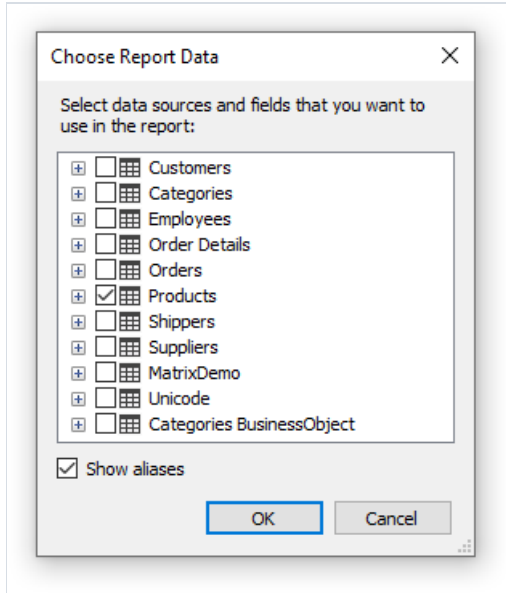
Genellikle, veri kaynağı bir veritabanı tablosunu veya bir SQL sorgusunu temsil eder. Bir raporda birden fazla veri kaynağı olabilir. Çoğu rapor için yalnızca bir veri kaynağı yeterlidir. Ancak 'master-detail' (ana-ayrıntı) türündeki raporlar, birbirine ilişkiyle bağlı iki veri kaynağı gerektirir (bu konuya bu bölümde daha sonra değineceğiz).

Bir veri kaynağı bir veya birden fazla veri sütununa sahiptir. Her sütunun belirli bir veri türü vardır. Sütunun veri türünü görmek için onu seçin ve 'Özellikler' (Properties) penceresini açın. Veri türü, 'DataType' özelliğinde belirtilmiştir. Sütun adının yanındaki simge de türünü belirlemenize yardımcı olur.

Rapor için veri kaynağını tanımlamanın iki yolu vardır:

Birinci yöntem — veri kaynağı uygulama içinde tanımlanır ve rapora kaydedilir. Bu, uygulamayı geliştiren programcının sorumluluğundadır (ayrıntılar için 'Programcı Kılavuzu'na bakınız).

Kullanıcının yapması gereken tek şey, raporda kullanılacak uygun veri kaynağını seçmektir. Bu işlem, 'Veri | Rapor Verisini Seç...' (Data | Choose Report Data...) menüsünden yapılabilir.



Bir rapora kaydedilmiş tüm veriler bu pencerede listelenir. Raporunuzda kullanmak istediğiniz verileri işaretlemeniz yeterlidir. Bu işlem, rapor üzerinde çalışırken herhangi bir anda yapılabilir.

İkinci yöntem — yeni bir veri kaynağını kendiniz oluşturmanızdır. Bu, bir veritabanı tablosu veya bir SQL sorgusu olabilir. Bu durumda, veri kaynağı tanımlı rapor dosyasına kaydedilir.

FastReport, MS SQL, Oracle, Interbase, Access gibi birçok popüler VTYS'ye (veritabanı yönetim sistemi) bağlanmayı destekler. Bunun yanı sıra, xml/xsd formatında kaydedilmiş veri dosyalarını da kullanabilirsiniz.

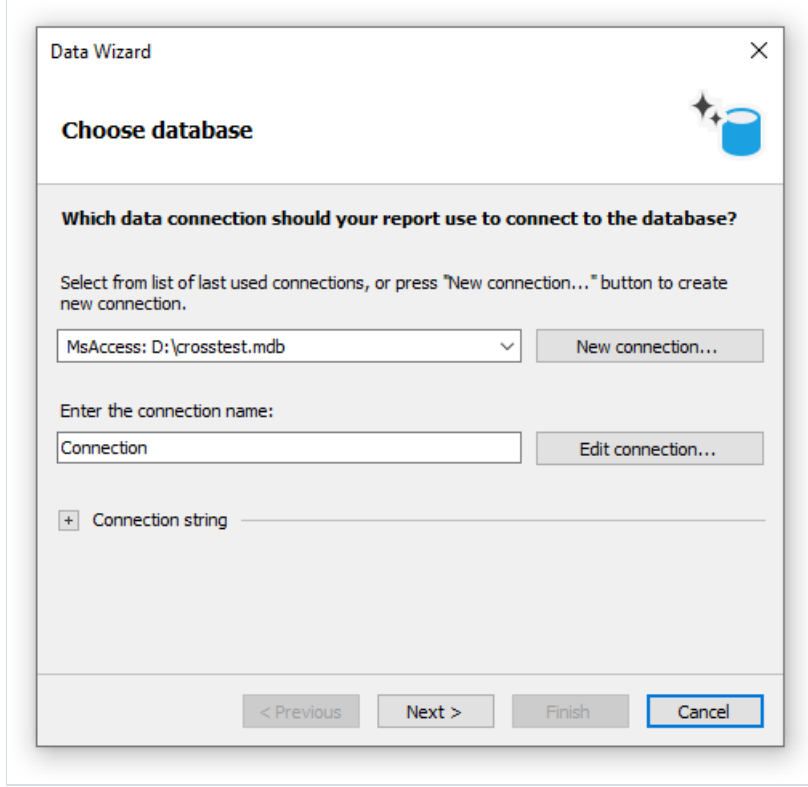
Veritabanı tablosunun içeriği rapor dosyasına kaydedilmez. Bunun yerine bağlantı dizesi ve veri kaynağının şeması saklanır.

Bağlantı dizesi, oturum açma bilgileri ve parola gibi verileri içerebileceğinden, rapor dosyasında şifrelenmiş olarak tutulur.

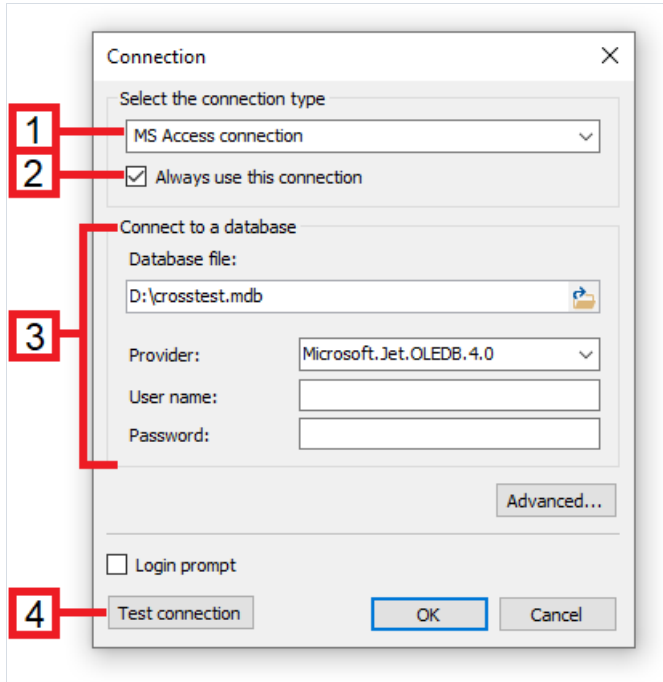
Gerektiğinde, verileri şifrelemek için kendi anahtarınızı kullanarak güvenliği artırabilirsiniz. Bu durumda, rapor dosyası yalnızca sizin programınızda doğru şekilde açılabilir.

Veri Kaynağı Oluşturma

Yeni bir veri kaynağı oluşturmak için, "Veriler|Yeni veri kaynağı..." menü seçeneğini seçin veya "Veriler" penceresinde "Eylemler" düğmesine basarak "Yeni veri kaynağı..." seçeneğini seçin. Bu işlem sonucunda "Veri Bağlantısı Sihirbazı" penceresi açılacaktır:



İlk adımda, bir veritabanı bağlantısı oluşturmanız istenir. Bunun için "Yeni bağlantı..." düğmesine basın. Bağlantı ayarlarının bulunduğu bir pencere açılacaktır:



Resimde aşağıdaki kontrol elemanları belirtilmiştir:

1. Bağlantı türü;

2. Bu kutucuk işaretlenirse, ileride varsayılan bağlantı türü olarak kullanılacaktır;
3. Seçilen bağlantı türüne ait ayarlar;
4. Bağlantıyı test etme düğmesi.

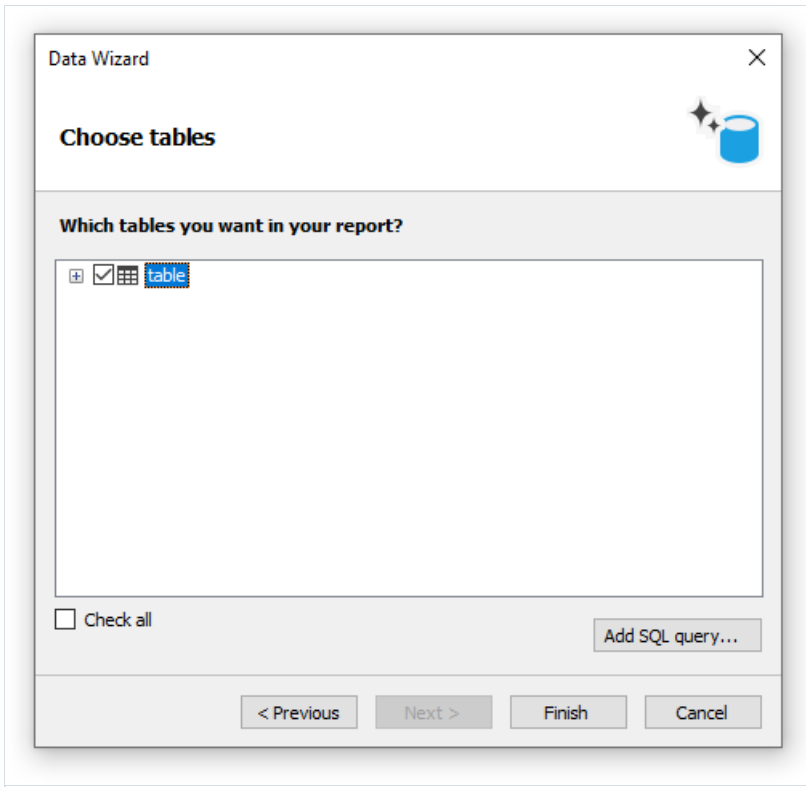
Resimde, MS Access veritabanına yapılan bir bağlantı gösterilmiştir. Farklı bir bağlantı türü seçerseniz, bağlantı ayarlarının bulunduğu alan (3) değişecektir. Örneğin, MS SQL veritabanı bağlantısı için aşağıdaki ayarlar geçerlidir:

Gerekli bağlantı türünü seçip, parametrelerini yapılandırmanız gerekmektedir. "Tamam" düğmesine bastıktan sonra pencere kapanır ve sizi bağlantı sihirbazı penceresine geri getirir.

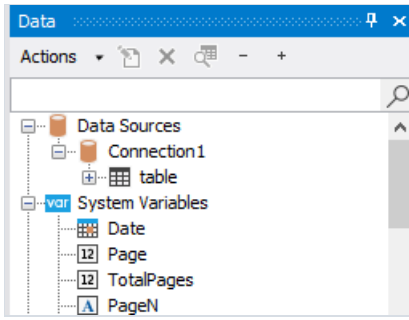
Bağlantı oluştururken bir isim belirtmeniz gerekir. Bu isim, "Veriler" penceresinde görünecektir.

Varsayılan olarak önerilen ismi kullanabilirsiniz:

Bağlantı sihirbazı penceresinde "İleri >" düğmesine basın. Bu adımda, veritabanında bulunan tabloların seçilmesi istenir:



Gerekli tabloları işaretleyip, "Hazır" düğmesiyle sihirbazı kapatın. Artık "Veriler" penceresinde, seçtiğiniz veri kaynaklarını içeren oluşturduğunuz bağlantıyı görebilirsiniz:



JSON Bağlantısı (JavaScript Object Notation)

Bu bağlantı, veri kaynağı olarak statik bir JSON dosyası veya rapor oluşturulmadan önce alınacak URL adresini sağlamaya olanak tanır.

JSON Bağlantısı Sihirbazı

FastReport tasarımcısında yeni bir bağlantı oluşturulduğunda aşağıdaki gibi görünür:

Parametre	Açıklama
Kodlama	JSON isteğinin işleneceği karakter kodlamasını belirler; eğer JSON'un alınacağı bir URL belirtilmişse, bu kodlama kullanılır.
JSON veya URL	Statik JSON verisini veya API üzerinden JSON alımını sağlayacak URL'yi belirler.
JSON Schema	JSON şemasını belirler.
Headers	JSON üzerinden API bağlantısı için gerekli HTTP başlıklarını (isteğe bağlı) belirler.

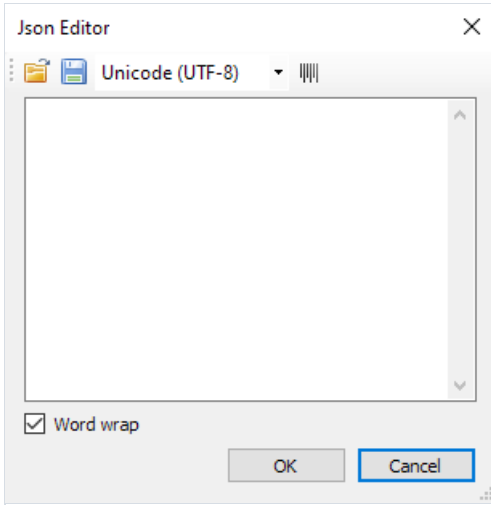
Alanların sağında, alanı ayrı bir pencerede düzenlemek için butonlar bulunur.

Eğer JSON Şeması değeri boş ise, OK veya "Bağlantıyı test et" düğmesine basıldığında JSON'a göre otomatik olarak bir şema oluşturulur.

JSON veya URL değiştirildiğinde ve şema zaten belirtilmişse, FastReport şemayı güncelleme teklifinde bulunur.

JSON Düzenleme Penceresi

Bu form, JSON'u düzenlemenize olanak tanır ve aşağıdaki gibi görünür:



Editör Araç Çubuğu Açıklaması (Soldan Sağa):

- Dosyayı Aç — JSON dosyasını açmanıza ve dosyanın içeriğini editöre yapıştırmanıza olanak tanır.
- Dosyayı Kaydet — Editördeki içeriği dosyaya kaydetmenizi sağlar.
- Kodlama — JSON dosyasının açılacağı kodlamayı belirler.
- Biçimlendirme — JSON biçimlendirmesini etkinleştirir ve JSON'un spesifikasyonlara uygunluğunu doğrulamak için validasyon yapar.

Daha sonra, JSON metnini değiştirebileceğiniz bir editör alanı gelir.

"Kelime taşımayı etkinleştir" onay kutusu, editörde kelime taşımayı açıp kapamanıza olanak tanır.

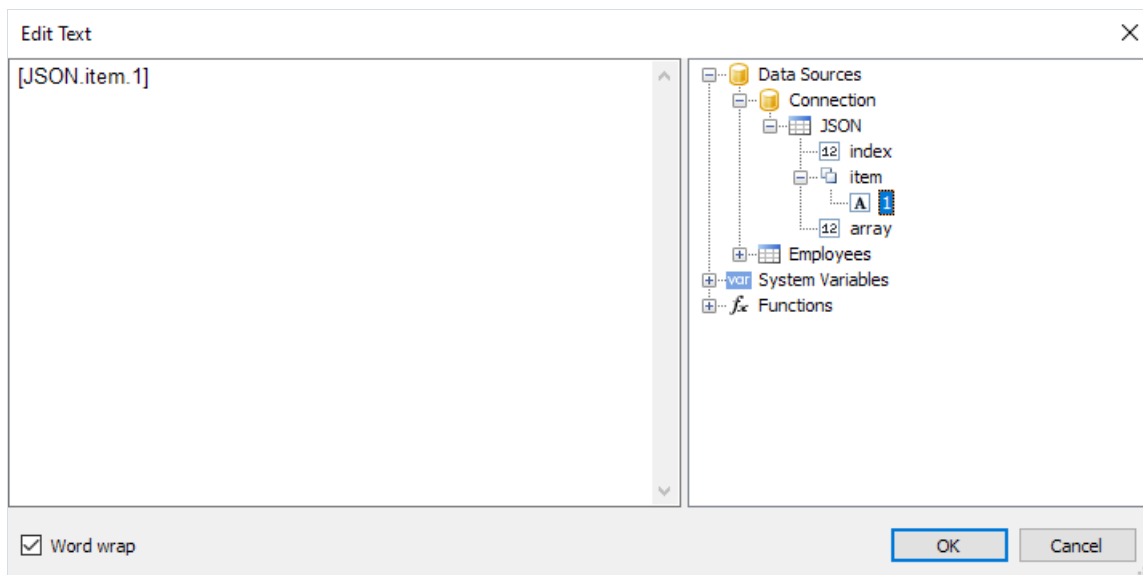
Bağlantının Çalışma Şekli

JSON bir tablo olmadığı için FastReport, JSON bağlantısını veri tablosu olarak algılamaz.

Bunun yerine, FastReport JSON içindeki her bir dizi elemanı, üç alana sahip hiyerarşik bir veri kaynağı olarak değerlendirir.

Alan	Açıklama
index	Öge numarası
item	Öge
array	Eleman dizisine bağlantı

Herhangi bir JSON dizisi DataBand'a bağlanabilir. Sonrasında, rapor nesnelerinde JSON'daki alanlar kullanılabilir.



Bağlantılarda Parametreler

Bağlantıda parametrelerin kullanılması ve Bağlantı Sihirbazı yardımıyla iletilmesi durumunda, parametreler dize (string) veri türünde olur. Ancak, tüm bağlantılar bu dize verisini, veritabanının kabul edebileceği türe düzgün bir şekilde dönüştüremeyebilir. Bu nedenle, bazı bağlantılar için, bağlantı kütüphaneleri tarafından desteklenen diğer veri türlerine dize değerini dönüştüren fonksiyonlar eklenmiştir. Aşağıda, parametre türüne göre dizenin hangi türe dönüştürüleceği belirtilmiştir. Eğer parametre değerine, aşağıdaki tablolarda yer almayan veya uygulanmamış bir tür atamak gerekirse, bu işlem yalnızca programatik olarak yapılabilir.

PostgresDataConnection

NpgsqlDbType	Type	C# Type	Örnek
Bigint		long	`123456`
Money		decimal	`123456`
Numeric		decimal	`123456`
Integer		int	`123456`
Oid		uint	`123456`
Xid		uint	`123456`
Cid		uint	`123456`
Smallint		short	`123456`
InternalChar		byte	`123456`
Real		float	`123456.12`
Double		double	`123456.12`
Boolean		bool	`True` or `1`
Bit		string	`1`
Timestamp		DateTime	`12:15:12`
TimestampTZ		DateTime	`12:15:12`
Date		DateTime	`16/02/2008`
Time		TimeSpan	`6:12:14`
Interval		TimeSpan	`6:12:14`
TimeTZ		DateTimeOffset	`05/01/2008`
Uuid		Guid	`81a130d2-502f-4cf1-a376-63edeb000e9f`
Box		NpgsqlBox	`((x1,y1),(x2,y2))`
Circle		NpgsqlCircle	<(x,y),r> (center point and radius)
Line		NpgsqlLine	{A,B,C}
Polygon		NpgsqlPolygon	((x1,y1),...)
Path		NpgsqlPath	((x1,y1),...)
LSeg		NpgsqlLSeg	((x1,y1),(x2,y2))
Point		NpgsqlPoint	(x,y)
Cidr		NpgsqlCidr	`192.168.100.128/25` IP address with mask
Inet		NpgsqlInet	`192.168.100.128/25` IP address
MacAddr		PhysicalAddress	`08:00:2b:01:02:03`
TsQuery		NpgsqlTsQuery	`'fat & rat'`
TsVector		NpgsqlTsVector	`'a fat cat sat on a mat and ate a fat rat'`
Char		string	`a` single char
Text		string	`string`
Varchar		string	`string`
Name		string	`string`
Citext		string	`string`
Bytea		string	`12AA` hex string
Varbit		string	`01101` bit string
Tid		NpgsqlTid	`12345, 123` uint number and ushort number
Array		---	Not implemented
Range		---	Not implemented
Hstore		---	Not implemented
Oidvector		---	Not implemented
MacAddr8		---	Not implemented
Int2Vector		---	Not implemented

MySqlDataConnection

MySqlDbType Type	C# Type	Örnek
Int64	long	`123456`
UInt64	ulong	`123456`
Int32	int	`123456`
Int24	int	`123456`
UInt24	uint	`123456`
UInt32	uint	`123456`
Int16	short	`123456`
UInt16	ushort	`123456`
Decimal	decimal	`123456`
NewDecimal	decimal	`123456`
Byte	sbyte	`12345`
UByte	byte	`12345`
Year	byte	`1901`
Float	float	`123456.12`
Double	double	`123456.12`
Bit	bool	`True` or '1' bool
Bool	bool	`True` or '1' bool
DateTime	DateTime	'16/02/2008 12:15:12'
Date	DateTime	'16/02/2008'
Newdate	DateTime	'16/02/2008'
Time	DateTime	`6:12:14`
Timestamp	DateTime	'16/02/2008 12:15:12'
Timestamp	DateTime	'16/02/2008 12:15:12'
Guid	Guid	`81a130d2-502f-4cf1-a376-63edeb000e9f`
VarChar	string	'string'
String	string	'string'
TinyText	string	'string'
MediumText	string	'string'
LongText	string	'string'
Text	string	'string'
VarString	string	'string'
JSON	string	'string'
Enum	string	'string'
Binary	byte[]	`12AA` hex string
VarBinary	byte[]	`12AA` hex string
Blob	byte[]	`12AA` hex string
TinyBlob	byte[]	`12AA` hex string
MediumBlob	byte[]	`12AA` hex string
LongBlob	byte[]	`12AA` hex string
Null	DBNull	
Set	---	Not implemented
Geometry	---	Not implemented

ClickHouseDataConnection

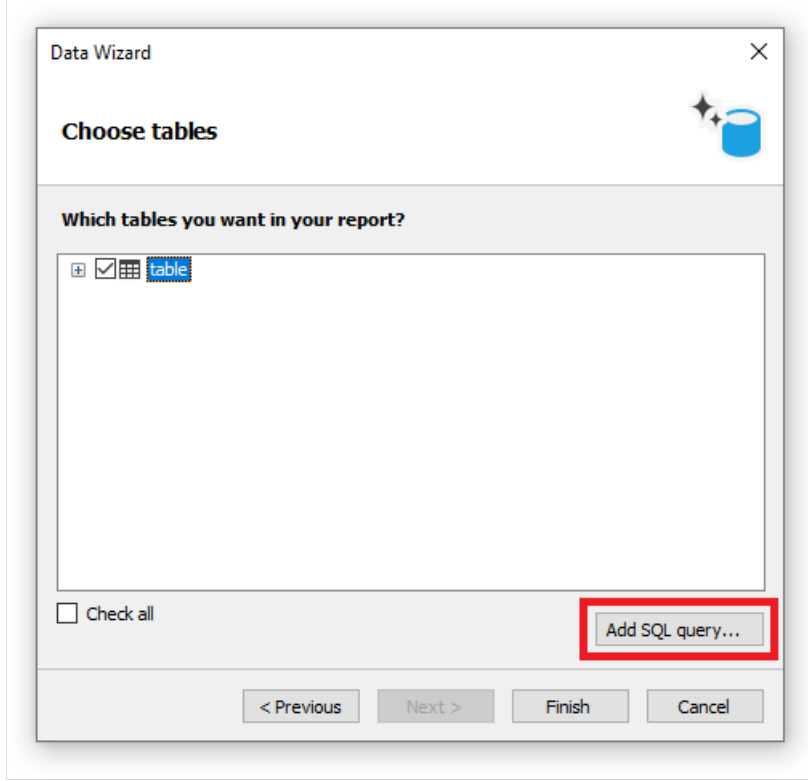
ClickHouse Type	C# Type	Örnek
Enum8	sbyte	`123456`
Int8	sbyte	`123456`
Enum16	short	`123456`
Int16	short	`123456`
Int32	int	`123456`
Int64	long	`123456`
Decimal	decimal	`123456`
Float32	float	`123456.12`
Float64	double	`123456.12`
UInt8	byte	`123456`
UInt16	ushort	`123456`
UInt32	uint	`123456`
UInt64	ulong	`123456`
Date	DateTime	'16/02/2008 12:15:12'
DateTime	DateTimeOffset	`05/01/2008`
DateTime64	DateTimeOffset	`05/01/2008`
UUID	Guid	`81a130d2-502f-4cf1-a376-63edeb000e9f`
IPv6	IPAddress	'2001:0db8:85a3:08d3:1319:8a2e:0370:7344' IP address
IPv4	IPAddress	'127.0.0.1' IP address
String	string	'string'
FixedString	string	'string'
Nothing	DBNull	
Array	---	Not implemented
Nested	---	Not implemented
Tuple	---	Not implemented
Nullable	---	Not implemented
LowCardinality	---	Not implemented

MsSqlDataConnection

SqlDbType Type	C# Type	Örnek
BigInt	long	`123456`
Bit	bool	`True` or `1`
DateTime	DateTime	`16/02/2008 12:15:12`
SmallDateTime	DateTime	`16/02/2008 12:15:12`
Date	DateTime	`16/02/2008 12:15:12`
Time	DateTime	`16/02/2008 12:15:12`
DateTime2	DateTime	`16/02/2008 12:15:12`
Char	string	`a` single char
NChar	string	`string`
NText	string	`string`
NVarChar	string	`string`
Text	string	`string`
VarChar	string	`string`
Xml	string	`<xml/>` xml string
Decimal	decimal	`string`
Money	decimal	`string`
SmallMoney	decimal	`string`
Real	float	`123456.12`
Float	float	`123456.12`
Int	int	`123456`
UniqueIdentifier	Guid	`81a130d2-502f-4cf1-a376-63edeb000e9f`
SmallInt	short	`123456`
TinyInt	byte	`123456`
DateTimeOffset	DateTimeOffset	`05/01/2008`
Variant	object	
Udt	object	
Binary	byte[]	`12AA` hex string
Image	byte[]	`12AA` hex string
Timestamp	byte[]	`12AA` hex string
VarBinary	byte[]	`12AA` hex string
Structured	---	Not implemented

SQL sorgusu oluřturma

Veri sihirbazı, veritabanında bulunan tabloları hızlıca seçmenizi sağlar. SQL sorgusu oluřturmak ise ek işlem gerektirir. Bunun için sihirbazın ikinci adımında 'Add SQL query...' (SQL sorgusu ekle...) düğmesine basın.



Sorgu sihirbazı penceresini göreceksiniz. Sorgu sihirbazı dört sayfadan oluşur. Sayfalar arasında geçiş yapmak için 'İleri' (Next) ve 'Geri' (Back) düğmelerini kullanın.

İlk adımda, bir sorgu adı belirlemeniz gerekir. Bu ad, 'Veri' (Data) penceresinde görünecektir. Benzersiz bir ad girin ve 'İleri' (Next) düğmesine basın.

Query Wizard

Enter a SQL statement

What data should return the query?

Type your SQL statement or use the Query Builder to construct it.

select * from [table]

Query Builder...

< Previous Next > Finish Cancel

İkinci adımda, SQL dilinde bir sorgu girmeniz gerekir. MSDB'nizin desteklediği dil diyalektini kullanın. Görsel sorgu oluşturmak için sorgu oluşturucuyu kullanabilirsiniz. Bunu yapmak için 'Sorgu Oluşturucu' (Query Builder) düğmesine basın. Sorgu oluşturucu daha sonra ayrıntılı olarak açıklanacaktır.

Sorgu metnini girdikten sonra 'İleri' (Next) düğmesine basın. Üçüncü adımda, sorgu parametrelerini tanımlayabilirsiniz. Sorgunuzda parametreler varsa bu gereklidir. Parametreleri bu bölümün ilerleyen kısımlarında ele alacağız.

Sihirbazın son adımında, sorgu tarafından döndürülen sütunları yapılandırabilirsiniz:

Query Wizard

Define columns

12 RowID
12 ColID
12 Cell

(Name)	RowID
Alias	RowID
BindableControl	Text
Calculated	False
CustomBindableControl	
DataType	Int32
Expression	
Format	Auto
Restrictions	None

(Name)

< Previous Next > Finish Cancel

Sorgu metninde veya parametre tanımında bir hata yaptıysanız, sihirbazın son sayfasına geçtiğinizde bir hata mesajı göreceksiniz.

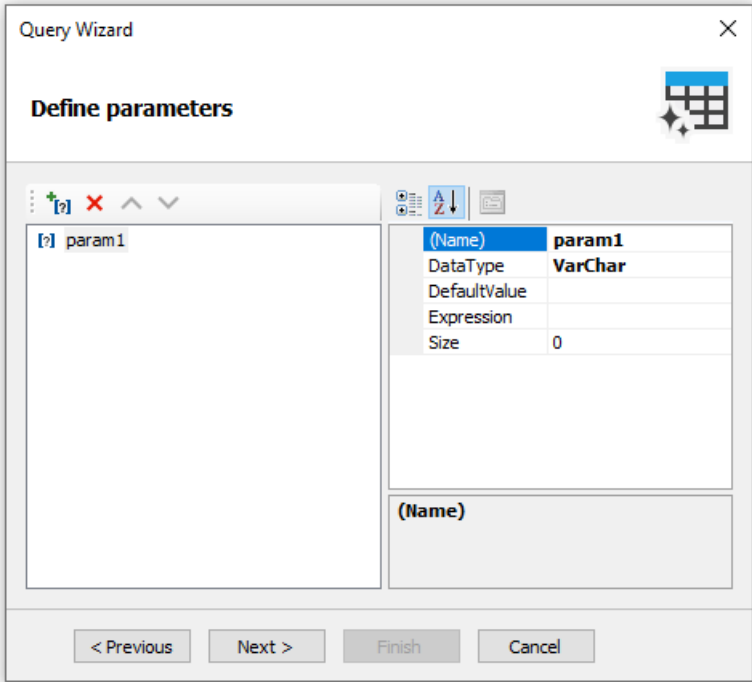
İstek Parametreleri

İstek metninde parametreler bulunabilir. Aşağıdaki isteğe bakalım:

```
select * from DVDs
where Title = @param1
```

Bu, MS SQL'in demo veritabanına yönelik bir istektir. İstek içinde **param1** adlı bir parametre tanımlanmıştır. Burada önemli bir nokta var: Farklı veritabanı yönetim sistemlerinde parametrelerin tanımlanma şekli farklılık gösterir. MS SQL için parametre, **@** sembolüyle belirtilirken; MS Access'te parametrelerin isimleri yoktur ve **?** sembolleriyle ifade edilir.

Parametre ile çalışabilmek için, en azından onun değerinin belirlenmesi gerekir. Bu işlem, yukarıda bahsedilen "Sorgu Sihirbazı"nın üçüncü adımında yapılabilir. Parametre oluşturmak için "Parametre Ekle" düğmesine tıklayın. Böylece yeni bir parametre oluşturulacaktır.



Sağdaki pencerede, parametrenin aşağıdaki özelliklerini yapılandırmanız gerekmektedir:

Özellik	Açıklama
Name	Parametre Adı. Burada, sorgu metninde kullandığınızla aynı parametre adını belirtmeniz gerekmektedir. Bazı veritabanı yönetim sistemleri (örneğin, MS Access) isimlendirilmiş parametreleri desteklemez. Bu durumda, bu özelliği değiştirmeden bırakın.
DataType	Parametrenin veri tipi.
DefaultValue	Expression özelliğinde bir değer bulunmaması veya değer hesaplanamaması durumunda (örneğin, rapor tasarım modunda sorgu işlemleri sırasında) parametreye atanacak varsayılan değeri belirtir.

Expression	Parametre deęerini döndüren ifade. İfadeler hakkında daha ayrıntılı bilgi için "İfadeler" bölümüne bakın. Bu ifade, rapor çalıştırıldığında hesaplanacaktır.
Size	Parametre veri boyutu. Bu özellik, parametre veri tipi "string" ise belirtilmelidir

Parametre özelliklerini yanlış yapılandırırsanız, sihirbazın son sayfasına geçerken hata alırsınız.

Sorgu parametresine deęer aktarımı

Parametrelerin yaygın kullanımı, kullanıcıdan örneęin şirket adı gibi belirli bilgilerin sorgulanmasıdır. Aşağıda, deęeri iki farklı yöntemle sorgu parametresine nasıl aktarabileceęinizi inceleyeceęiz.

Birinci Yöntem: Programdan Deęer Aktarmak. Bu yöntemde deęeri doğrudan programdan aktarabilirsiniz. Sorgu parametresine doğrudan deęer atamanın basit ve güvenilir bir yolu olmadığından, rapor parametresini kullanmanız gerekmektedir. Önerilen adımlar şu şekildedir:

- Rapor Parametresi Oluşturma: Rapor parametresini oluşturun (parametrelerle ilgili daha fazla bilgi için "Rapor Parametreleri" bölümüne bakınız). Rapor parametresi için, sorgu parametresinde kullanılan veri tipiyle aynı veri tipini belirleyin.
- Expression Özellięini Ayarlama: Sorgu parametresinin **Expression** özellięinde, oluşturduğunuz rapor parametresini belirtin. Bunu, Expression editörünü açarak görsel olarak yapabilirsiniz. Böylece Expression özellięi şu şekilde bir şey içerecektir:

```
[RaporParametresi]
```

- Programdan Deęer Aktarma: Programdan parametre deęerini aktarmak için Report nesnesinin `SetParameterValue` metodunu kullanın:

```
report1.SetParameterValue("RaporParametresi", 10);
```

İkinci Yöntem: Diyalog Formu Kullanmak: Bu yöntemde, raporun diyalog formunu kullanırsınız (daha fazla bilgi için "Diyalog Formları" bölümüne bakınız). Diyalog üzerinde kullanıcı tarafından veri girilebilecek kontrol öğeleri yerleştirilebilir. Örneęin, şirket adını sorgulamak için bir **TextBoxControl** kullanılabilir. Bu durumda, **Expression** özellięine aşağıdaki yazabilirsiniz:

Burada **TextBox1**, raporun diyalog formunda kullanıcı tarafından girilen deęeri içeren kontrol öğesidir.

Tam Parametre Yapılandırma Örneęi

Yukarıdaki örneęe göre tam parametre yapılandırması şu şekilde olabilir:

```
Name=param1  
DataType=VarChar  
DefaultValue= (empty string)  
Expression=TextBox1.Text  
Size=255
```

Bu adımları izleyerek, hem programatik hem de diyalog formu kullanarak sorgu parametresine deęer aktarabilirsiniz.

Bağlantıyı Düzenleme

Veri bağlantısı kurulum sihirbazı çalışırken oluşturulan bağlantı düzenlenebilir. Bunun için "Veriler" penceresindeki bağlantıyı seçin ve araç çubuğundaki "Düzenle" düğmesine tıklayın. Karşınıza daha önce gördüğünüz bağlantı sihirbazı penceresi çıkacaktır. Bağlantı ayarlarını değiştirmek için "Düzenle..." düğmesine tıklayabilirsiniz. Bağlantı türü değiştirilemez: bu, yalnızca oluşturulma anında belirlenir.

Sihirbazın ikinci sayfasında, "Veriler" penceresinde görmek istediğiniz tabloları seçebilirsiniz. Düzenlemeyi bitirdiğinizde, "Bitti" düğmesine tıklayın.

Veri Kaynağını Düzenleme

"Veri Bağlantısı Sihirbazı" kullanılarak oluşturulan veri kaynağı düzenlenebilir. Bunu yapmak için, "Veriler" penceresinden ilgili veri kaynağını seçip araç çubuğundaki "Düzenle" butonuna tıklayın. Daha önce incelediğimiz Sorgu Sihirbazı penceresi açılacaktır. Bu pencerede SQL sorgu metnini değiştirebilir, sorgu parametrelerini ve veri alanlarını ayarlayabilirsiniz.

Veri Kaynağını Silme

Veri kaynağını silmek için yine onu seçip araç çubuğundaki "Sil" butonuna tıklayın. Bu işlem, veri kaynağının fiziksel olarak silinmesine neden olmaz; sadece erişime kapatılır. "Veriler|Rapor için Veri Seç..." menüsüne giderek bu kapalı veri kaynağını yeniden etkinleştirebilirsiniz. Ancak endişe etmeyin: silinen veri kaynakları rapor dosyasına kaydedilmez ve bu nedenle rapor bir sonraki açılışta geri yüklenmez.

Mevcut bir bağlantıya veri kaynağı ekleme

Varolan bir bağlantıya (tablo veya sorgu) veri kaynağı eklemek için iki yöntem vardır:

- Bağlantı düzenleyicisini çağırın. Bu, "[Bağlantı Düzenleme](#)" bölümünde açıklandığı şekilde yapılır. Bağlantı sihirbazının ikinci sayfasında yeni bir veri kaynağı seçebilir veya oluşturabilirsiniz.
- "Bağlantı" ögesi üzerinde sağ tıklayın ve açılan menüden "Yeni Veri Kaynağı..." seçeneğini seçin. Bu da, ilk yöntemde olduğu gibi, bağlantı sihirbazının açılmasına neden olur.

Takma Adlar (aliases)

Her veri ögesinin (veri kaynağı veya alanı) kendine ait bir ismi vardır. Varsayılan olarak, bu isim veritabanında tanımlanan isimdir. Bazı durumlarda, örneğin **ProdID** gibi bir alanın isminin arkasındaki anlamı kavramak zor olabilir.

Veri öğelerinin ikinci isimleri – takma adları – da vardır. Takma adları kullanarak, bir öğeyi yeniden adlandırabilirsiniz. Örneğin, elimizde **ProdID** alanı bulunan Categories veri kaynağı varsa, aşağıdaki takma adları tanımlayabilirsiniz:

```
CATEGORY_TABLE --> Categories  
PROD_ID --> Product ID
```

Böyle bir alana şu şekilde başvurabilirsiniz:

```
[Categories.Product ID]
```

Veri ögesine başvururken, takma ad tanımlıysa onu kullanmalısınız. Bu durumda, orijinal isimle başvurmak mümkün değildir.

Veri ögesini yeniden adlandırmak için, "Veriler" penceresinde öğeyi seçin ve F2 tuşuna basın. Alternatif olarak, nesnenin bağlam menüsünden "Yeniden Adlandır" seçeneğini de seçebilirsiniz. Ardından, istediğiniz ismi yazın ve Enter tuşuna basın.

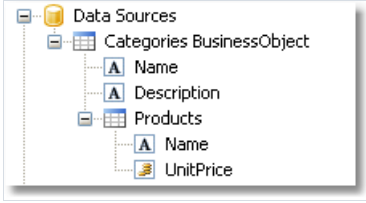
Ayrıca, "Özellikler" penceresini kullanarak da öğeyi yeniden adlandırabilirsiniz. "Veriler" penceresinde öğeyi seçin ve "Özellikler" penceresinde **Alias** özelliğinin değerini değiştirin.

Takma adı silmek (orijinal ismi geri yüklemek) için, öğeyi seçin ve bağlam menüsünden "Takma Adı Sil" seçeneğini tıklayın.

Hiyerarşik Veri Kaynakları

İncelediğimiz veri kaynakları ilişkisel olup, ilişkisel veritabanı yönetim sistemlerindeki tabloların analogudur. FastReport, aynı zamanda hiyerarşik veri kaynaklarını da destekler. Bu tür kaynaklar, raporda iş nesnelerinden veri kaydı yapılırken oluşturulur. Bu veriler yalnızca programatik olarak rapora eklenebildiğinden, onları kullanıcı perspektifinden nasıl kullanacağımıza bakalım.

Hiyerarşik veri kaynakları, normal veri kaynaklarından sadece bağlantı oluşturulmasına gerek duyulmaması açısından farklıdır. Veri kaynakları, zaten "ebeveyn-çocuk" ilişkisiyle bağlıdır. "Veri" penceresinde bu tür kaynaklar aşağıdaki gibi görünür:



Görüldüğü gibi, "Categories Business Object" adlı ebeveyn veri kaynağı, kendi içinde "Products" adlı bir çocuk veri kaynağı barındırmaktadır. Hiyerarşik veri kaynaklarıyla diğer veri kaynaklarıyla olduğu gibi çalışabilirsiniz.

İlişkiler (relations)

İki veri kaynağı arasında bir ilişki kurulabilir. İlişki, "ana-alt" (veya "ebeveyn-çocuk") ilişkisini tanımlar. Örneğin, Categories tablosundaki bir kayıt, Products tablosundaki birçok alt kayda sahip olabilir:

Categories	
CategoryID	CategoryName
1	Beverages

Products		
ProductID	CategoryID	Product name
1	1	Chai
2	1	Chang
39	1	Chartreuse verte
38	1	Côte de Blaye
24	1	Fantástica

İlişki oluşturmak için şunları belirtmek gerekir:

- ana tablo;
- alt tablo;
- ana tablodaki anahtar alanlar kümesi;
- alt tablodaki anahtar alanlar kümesi.

Örnek olarak, demo veritabanındaki 'Categories' ve 'Products' tablolarını inceleyeceğiz. Bu tablolar aşağıdaki yapıya sahiptir:

Categories	Products
CategoryID	ProductID
CategoryName	ProductName
Description	SupplierID
Picture	CategoryID
CategoriesTableAdapter	QuantityPerUnit
	UnitPrice
	UnitsInStock
	UnitsOnOrder
	ReorderLevel
	Discontinued
	EAN13
	ProductsTableAdapter

Her iki tabloda da ilişki kurulabilecek **CategoryID** alanı bulunmaktadır. Bu sayede bir kategoriye birden fazla ürün ait olabilir.

Peki, ilişkili veri kaynakları FastReport'ta nasıl kullanılır? Bunu yapmanın iki yöntemi vardır.

İlk yöntem, 'ana-ayrıntı' (master-detail) türünde raporlar oluşturmayı mümkün kılar. Bunun için iki adet 'Veri' (Data) bandı kullanılır. Ana bant (master band) ana veri kaynağına, ayrıntı bandı (detail band) ise ayrıntı veri kaynağına bağlanır.

Bizim örneğimiz şu şekilde olacaktır:



Böyle bir rapor çalıştırıldığında, her kategoriye ait ürünlerin bir listesini yazdıracaktır.



İkinci yöntem, detay veri kaynağından ana (master) veri kaynağına başvurmayı sağlar. Bunu bir örnekle göstereceğiz. Diyelim ki, tüm ürünlerin bir listesini yazdırmak istiyoruz. Bunun için "Products" tablosuna bağlı bir adet "Veri" (Data) bandına ihtiyacımız var:

"Bu amaçla, yalnızca bir adet 'Veri' bandı yeterlidir ve bu band 'Products' tablosuna bağlanmalıdır."



Böyle bir rapor, tüm kategorilerden tüm ürünleri yazdıracaktır. Diyelim ki, her ürünün yanında ait olduğu kategori adını yazdırmak istiyoruz. İlişki kullanmadan bunu yapmak daha zor olurdu. Ürünün kategorisi hakkında bildiğimiz tek şey, ID'sidir (bu, "Products" tablosundaki CategoryID sütunu ile temsil edilir). Yazdırmak istediğimiz kategori adı ise, "Categories" tablosundaki CategoryName sütununda saklanır. İlişki yardımıyla, kategori adını şu şekilde referans alabiliriz:

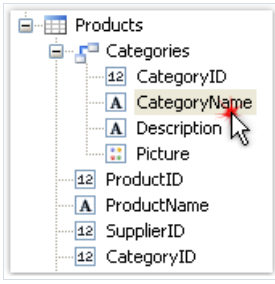
```
[Products.Categories.CategoryName]
```

"Products" tablosundaki mevcut satır için, FastReport ilgili "Categories" tablosundaki ebeveyn satırını bulacak ve CategoryName sütununun değerini döndürecektir.

Genel bir durumda, bir ebeveyn tablosunun alanına başvurmak, sınırsız sayıda tablo atağına olanak tanır:

```
[Child_table.Its_parent.Parent_of_a_parent.And _so_on.Column_name]
```

Böyle bir veri sütununu rapora eklemek için, "Veri" (Data) penceresinde "Products" tablosunu açın. Sütunları arasında "Categories" tablosuna bir bağlantı olduğunu göreceksiniz:



Yukarıda gösterilen sütunu rapora sürüklersek, şu şekilde bir "Metin" (Text) nesnesi elde ederiz:

[Products.Categories.CategoryName]

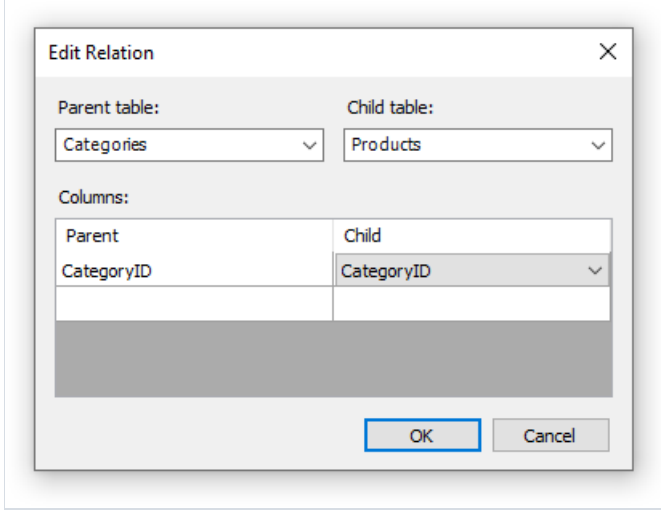
Raporumuz aşağıdaki gibi olacaktır:

Çalıştırdığımızda, aşağıdakini göreceğiz:

Alice Mutton	Meat/Poultry
Aniseed Syrup	Condiments
Boston Crab Meat	Seafood
Camembert Pierrot	Dairy Products
Carnarvon Tigers	Seafood
Chai	Beverages
Chang	Beverages

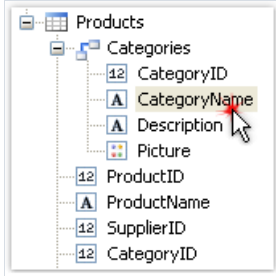
Bağlantı Oluşturma

Bağlantı oluşturmak için, "Veriler" penceresinde "Eylemler" düğmesine basın ve "Yeni bağlantı..." seçeneğini seçin. Bağlantı düzenleyicisini göreceksiniz:



Öncelikle ana ve bağlı tabloları seçmeniz gerekmektedir. Bundan sonra pencerenin alt kısmında her tabloya karşılık gelecek şekilde alan çiftlerini seçmelisiniz. Tablolar, bir veya birden fazla alan kullanılarak ilişkilendirilebilir. Alanlar ayarlandıktan sonra, bağlantı düzenleyicisini OK düğmesine basarak kapatın.

Oluşturulan bağlantıyı, bağlı veri kaynağını seçip alan listesini açtığınızda "Veriler" penceresinde görebilirsiniz. Alanlar arasında ana veri kaynağı ile olan bağlantıyı göreceksiniz:



Ana veri kaynağı alanlarını drag&drop yöntemiyle rapor sayfasına sürükleyip bırakabilirsiniz. Örneğin, resimde gösterilen alanı seçip rapor sayfasına sürükleyip bırakırsanız, aşağıdaki içeriğe sahip bir "Metin" nesnesi elde ederiz:

```
[Products.Categories.CategoryName]
```

Bağlantıyı düzenleme

Bağlantıyı düzenlemek için, alt veri kaynağının alanlar listesini açın, ilgili bağlantıyı bulun ve araç çubuğundaki "Düzenle..." düğmesine tıklayın. Bu işlem, daha önce incelediğimiz bağlantı düzenleyicisini açacaktır.

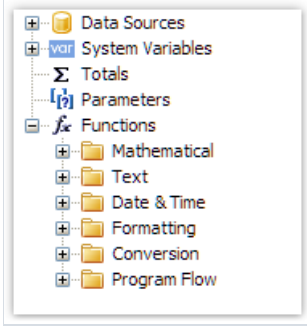
Sistem deęiřkenleri

FastReport'ta, raporun herhangi bir yerinde kullanılabilecek çeřitli deęiřkenler bulunmaktadır. Ařaęıda, tüm deęiřkenlerin tam listesi verilmiřtir.

Deęiřken	Açıklama
Date	Raporun başlangıç tarihi ve saati.
Page	Geçerli sayfa numarası.
TotalPages	Raporun toplam sayfa sayısı. Bu deęiřkeni kullanmak için, raporda çift geçiř modunu etkinleřtirmeniz gerekir. Bu ayarı "Rapor Özellikler..." menüsünden yapabilirsiniz.
PageN	"Sayfa N" řeklinde sayfa numarası.
PageNofM	"Sayfa N / M" řeklinde sayfa numarası.
Row#	Grup içindeki veri satır numarası. Bu deęer, yeni bir grup başladıęında sıfırlanır.
AbsRow#	Veri satırının mutlak numarası. Bu deęer, yeni bir grup başladıęında sıfırlanmaz.
Page#	Paketdeki hazır raporları birleřtiriyorsanız, bu deęiřken paket içindeki geçerli sayfa numarasını döndürür. Bu deęiřken bir makrodur; yani deęeri, bileřen görüntüleme penceresinde gösterildięi anda yerleřtirilir. İfadelerde kullanılamaz.
TotalPages#	Raporun toplam sayfa sayısı. Birden fazla hazır raporu bir paket halinde birleřtirirseniz, bu deęiřken paketteki sayfa sayısını döndürür. Bu deęiřkenin çalışması için çift geçiř moduna gerek yoktur. Bu deęiřken bir makrodur; yani deęeri, bileřen görüntüleme penceresinde gösterildięi anda yerleřtirilir. İfadelerde kullanılamaz.
HierarchyLevel	Hiyerarřik rapordaki mevcut seviye (bkz. "Hiyerarři Yazdır"). En üst seviye 1 deęerindedir.
HierarchyRow#	Hiyerarřik rapordaki tam satır numarası, "1.2.1" gibi bir biçimde gösterilir.

Fonksiyonlar

FastReport .NET, kullanıcıya raporda kullanılacak 60'tan fazla yerleşik fonksiyon sunar. Fonksiyonlar, çeşitli kategorilere ayrılmış olup "Veriler" penceresinde mevcuttur.



Fonksiyonlar; herhangi bir ifadeye, script kodunda kullanılabilir ve "Metin" nesnelerinde değerleri görüntülenebilir. Örneğin, bir "Metin" nesnesi şu fonksiyon çağrısını içerir:

```
[Sqrt(4)]
```

Bunun sonucu olarak 2 değeri (4'ün karekökü) yazdırılacaktır.

Aşağıdaki ifade 4 değerini döndürecek:

```
Sqrt(4) + 2
```

Fonksiyonların rapora görsel olarak eklenme yöntemleri şu şekildedir:

- Fonksiyon, rapor sayfasına sürüklenip bırakılabilir; bu işlemle, fonksiyon çağrısını içeren bir "Metin" nesnesi oluşturulur. Oluşturulan metin, fonksiyon parametreleri eklenerek düzenlenmelidir.
- Fonksiyon, script koduna sürüklenip bırakılabilir.
- İfade düzenleyici penceresinde, "Veriler" penceresinin bir kopyası yer alır ve buradan fonksiyon metne sürüklenip bırakılabilir.

Aşağıda, her fonksiyonun açıklaması verilecektir.

Matematiksel

Abs

Fonksiyon	Parametreler	Döner değeri
Abs	sbyte value	sbyte
Abs	short value	short
Abs	int value	int
Abs	long value	long
Abs	float value	float
Abs	double value	double
Abs	decimal value	decimal

Value değeri için mutlak büyüklüğünü döndürür.

Örnek:

$$\text{Abs}(-2.2) = 2.2$$

Acos

Fonksiyon	Parametreler	Döner değeri
Acos	double d	double

d 'nin kosinüsüne karşılık gelen açıyı (radyan cinsinden) döndürür. **d** değeri -1 ile 1 arasında olmalıdır.

Radyanları dereceye çevirmek için radyan değerini $180 / \text{Math.PI}$ ile çarpın.

Örnek:

$$\text{Acos}(0) * 180 / \text{Math.PI} = 90$$

Asin

Fonksiyon	Parametreler	Döner değeri
-----------	--------------	--------------

Asin	double d	double
-------------	----------	--------

`d` 'nin sinüsüne karşılık gelen açıyı (radyan cinsinden) döndürür. `d` değeri -1 ile 1 arasında olmalıdır.

Radyanları dereceye çevirmek için radyan değerini `180 / Math.PI` ile çarpın.

Örnek:

```
Asin(0) = 0
```

Atan

Fonksiyon	Parametreler	Dönen değer
Atan	double d	double

`d` 'nin tanjantına karşılık gelen açıyı (radyan cinsinden) döndürür.

Radyanları dereceye çevirmek için radyan değerini `180 / Math.PI` ile çarpın.

Örnek:

```
Atan(1) * 180 / Math.PI = 45
```

Ceiling

Fonksiyon	Parametreler	Dönen değer
Ceiling	double d	double
Ceiling	decimal d	decimal

`d` 'ye eşit veya daha büyük olan en küçük tam sayıyı döndürür.

Örnek:

```
Ceiling(1.7) = 2
```

Cos

Fonksiyon	Parametreler	Dönen değer
Cos	double d	double

`d` değerinin kosinüsünü döndürür. `d` değeri radyan cinsinden olmalıdır.

Dereceleri radyana çevirmek için derece değerini `Math.PI / 180` ile çarpın.

Örnek:

```
Cos(90 * Math.PI / 180) = 0
```

Exp

Fonksiyon	Parametreler	Dönen değer
Exp	<code>double d</code>	double

e sayısını (2.71828), `d` kuvvetine yükseltilmiş olarak döndürür.

Örnek:

```
Exp(1) = 2.71828
```

Floor

Fonksiyon	Parametreler	Dönen değer
Floor	double d	double
Floor	decimal d	decimal

`d` 'ye eşit veya daha küçük olan en büyük tam sayıyı döndürür.

Örnek:

```
Floor(1.7) = 1
```

Log

Fonksiyon	Parametreler	Dönen değer
<code>Log</code>	double d	double

`d` değerinin doğal logaritmasını döndürür.

Örnek:

```
Log(2.71828) = 1
```

Maximum

Fonksiyon	Parametreler	Döner değeri
Maximum	int val1, int val2	int
Maximum	long val1, long val2	long
Maximum	float val1, float val2	float
Maximum	double val1, double val2	double
Maximum	decimal val1, decimal val2	decimal

`val1` ve `val2` değeriinin en büyüğünü döndürür.

Örnek:

```
Maximum(1,2) = 2
```

Minimum

fonksiyon	parametreler	dönüş değeri
Minimum	int val1, int val2	int
Minimum	long val1, long val2	long
Minimum	<code>float val1, float val2</code>	float
Minimum	double val1, double val2	double
Minimum	decimal val1, decimal val2	decimal

`val1` , `val2` değeriinden en küçüğünü döndürür.

Örnek:

```
Minimum(1,2) = 1
```

Round

fonksiyon	parametreler	dönüş değeri
Round	double d	double

Round	<code>decimal d</code>	decimal
--------------	------------------------	---------

`d` değerini en yakın tam sayıya yuvarlar.

Örnek:

```
Round(1.47) = 1
```

fonksiyon	parametreler	dönüş değeri
Round	double d, int digits	double
Round	decimal d, int digits	decimal

`d` değerini, `digits` parametresinde belirtilen basamak sayısına yuvarlar.

Örnek:

```
Round(1.478, 2) = 1.48
```

Sin

fonksiyon	parametreler	dönüş değeri
Sin	double d	double

`d` değerinin sinüsünü döndürür. `d` değeri radyan cinsinden belirtilmelidir.

Dereceyi radyana çevirmek için, derece değerini `Math.PI / 180` ile çarpın.

Örnek:

```
Sin(90 * Math.PI / 180) = 1
```

Sqrt

fonksiyon	parametreler	dönüş değeri
Sqrt	<code>double d</code>	double

`d` değerinin karekökünü döndürür.

Örnek:

$\text{Sqrt}(4) = 2$

Tan

fonksiyon	parametreler	dönüş değeri
Tan	double d	double

d değerinin tanjantını döndürür. **d** değeri radyan cinsinden belirtilmelidir.

Dereceyi radyana çevirmek için, derece değerini $\text{Math.PI} / 180$ ile çarpın.

Örnek:

$\text{Tan}(45 * \text{Math.PI} / 180) = 1$

Truncate

fonksiyon	Параметры	dönüş değeri
Truncate	double d	double
Truncate	decimal d	decimal

d değerinin tam kısmını döndürür.

Örnek:

$\text{Truncate}(1.7) = 1$

Metinsel

Not:

- bu fonksiyonlar kendilerine verilen metin deęerini deęiřtirmez; bunun yerine, deęiřtirilmiř metni dndrrlr.
- Metindeki ilk karakterin pozisyonu 0'dır. Bu, rneęin Insert gibi bazı fonksiyonlara parametre gnderirken gz nnde bulundurulmalıdır.

Asc

Fonksiyon	Parametreler	Dnř deęeri
Asc	char c	int

Tam sayı dndrr; bu sayı, `c` karakterinin kodunu temsil eder.

rnek:

```
Asc('A') = 65
```

Chr

Fonksiyon	Parametreler	Dnř deęeri
Chr	int i	char

`i` koduna sahip karakteri dndrr.

rnek:

```
Chr(65) = 'A'
```

Insert

Fonksiyon	Parametreler	Dnř deęeri
Insert	<code>string s, int startIndex, string value</code>	string

`s` dizesine, `startIndex` konumundan bařlayarak `value` alt dizesini ekler ve yeni bir dize dndrr.

rnek:

```
Insert("ABC", 1, "12") = "A12BC"
```

Length

Fonksiyon	Parametreler	Dönüş değeri
Length	string s	int

`s` dizesinin uzunluğunu döndürür.

Örnek:

```
Length("ABC") = 3
```

LowerCase

Fonksiyon	Parametreler	Dönüş değeri
LowerCase	<code>string s</code>	string

`s` dizesindeki tüm karakterleri küçük harfe çevirir ve sonucu döndürür.

Örnek:

```
LowerCase("ABC") = "abc"
```

PadLeft

Fonksiyon	Parametreler	Dönüş değeri
PadLeft	<code>string s, int totalWidth</code>	string

`s` dizesinin soluna, `totalWidth` parametresinde belirtilen uzunluğa ulaşana kadar boşluk ekler ve yeni bir dize döndürür.

Örnek:

```
PadLeft("ABC", 5) = " ABC"
```

Fonksiyon	Parametreler	Dönüş değeri
PadLeft	<code>string s, int totalWidth, char paddingChar</code>	string

`s` dizesinin soluna, `paddingChar` karakteri ekleyerek `totalWidth` parametresinde belirtilen uzunluğa ulaşır ve yeni bir dize döndürür.

Örnek:

```
PadLeft("ABC", 5, '0') = "00ABC"
```

PadRight

Fonksiyon	Parametreler	Dönüş değeri
PadRight	string s, int totalWidth	string

s dizesinin sonuna, **totalWidth** parametresinde belirtilen uzunluğa ulaşana kadar boşluk ekler ve yeni bir dize döndürür.

Örnek:

```
PadRight("ABC", 5) = "ABC "
```

Fonksiyon	Parametreler	Dönüş değeri
PadRight	string s, int totalWidth, char paddingChar	string

s dizesinin sonuna, **paddingChar** karakteri ekleyerek **totalWidth** parametresinde belirtilen uzunluğa ulaşır ve yeni bir dize döndürür.

Örnek:

```
PadRight("ABC", 5, '0') = "ABC00"
```

Remove

Fonksiyon	Parametreler	Dönüş değeri
Remove	string s, int startIndex	string

s dizesindeki tüm karakterleri, **startIndex** konumundan başlayarak siler ve yeni bir dize döndürür.

Örnek:

```
Remove("ABCD", 3) = "ABC"
```

Fonksiyon	Parametreler	Dönen değer
Remove	string s, int startIndex, int count	string

Belirtilen **count** sayısında karakteri **s** string'inden, **startIndex** konumundan başlayarak siler ve yeni bir string

döndürür.

Örnek:

```
Remove("A00BC", 1, 2) = "ABC"
```

Replace

Fonksiyon	Parametreler	Dönen değer
Replace	string s, string oldValue, string newValue	string

s dizisinde **oldValue** alt dizisinin tüm görünümünü **newValue** alt dizisiyle değiştirir ve yeni bir dize döndürür.

Örnek:

```
Replace("A00", "00", "BC") = "ABC"
```

Substring

Fonksiyon	Parametreler	Dönen değer
Substring	string s, int startIndex	string

s dizisindeki tüm karakterleri, **startIndex** konumundan başlayarak dize sonuna kadar döndürür.

Örnek:

```
Substring("ABCDEF", 4) = "EF"
```

Fonksiyon	Parametreler	Dönen değer
Substring	string s, int startIndex, int length	string

s dizisindeki **startIndex** konumundan başlayarak **length** kadar karakter döndürür.

Örnek:

```
Substring("ABCDEF", 1, 3) = "BCD"
```

TitleCase

Fonksiyon	Parametreler	Dönen değer
-----------	--------------	-------------

TitleCase	<code>string s</code>	string
------------------	-----------------------	--------

`s` dizesindeki karakterleri "Başlık" biçimine çevirir. Bu durumda, her kelimenin ilk harfi büyük, diğer harfleri küçük yapılır.

Örnek:

```
TitleCase("john smith") = "John Smith"
```

Trim

Fonksiyon	Parametreler	Dönen değer
Trim	string s	string

`s` dizesinin başında ve sonunda bulunan gereksiz boşlukları kırpar.

Örnek:

```
Trim(" ABC ") = "ABC"
```

UpperCase

Fonksiyon	Parametreler	Dönen değer
UpperCase	string s	string

`s` dizesindeki tüm karakterleri büyük harfe çevirir ve sonucu döndürür.

Örnek:

```
UpperCase("abc") = "ABC"
```

Tarih ve Zaman

Belirli bir tarihi ayarlamak için `DateTime` yapısının yapıcıları veya `ToDateTime` (daha ayrıntılı bilgi için "Dönüştürme" bölümüne bakınız) fonksiyonu ya da `DateSerial` (aşağıda ele alınmıştır) kullanılabilir.

Örneğin, yapıcılar aracılığıyla tarih ve saat oluşturmak için aşağıdaki kod kullanılabilir:

```
new DateTime(yıl, ay, gün, saat, dakika, saniye);
```

Bu bölümdeki tarih dönüşüm fonksiyonları kullanıldığında varsayılan olarak zaman bilgisi de korunur. Tarihten zamanı kaldırmak için `Format` veya `FormatDateTime` fonksiyonlarından yararlanılabilir. Bu fonksiyonlar hakkında daha ayrıntılı bilgi "Biçimlendirme" bölümünde yer almaktadır.

Fonksiyonlarda veritabanı alanı kullanımı örneği olarak demo veritabanındaki `[Employees.BirthDate]` alanı kullanılacaktır.

```
[Employees.BirthDate] = 27.01.1986
```

AddDays

Fonksiyon	Parametreler	Dönüş Değeri
AddDays	DateTime date, double value	DateTime

`AddDays` fonksiyonu, verilen tarih değerine belirtilen gün sayısını ekler ve yeni bir tarih döndürür.

Örnekler:

```
AddDays(new DateTime(2024, 4, 2), 2) = 04.04.2024 0:00:00
AddDays.ToDateTime("2.4.2024"), 2) = 04.04.2024 0:00:00
AddDays([Employees.BirthDate], 2) = 29.01.1986 0:00:00
```

AddHours

Fonksiyon	parametreler	dönen değer
AddHours	DateTime date, double value	DateTime

Verilen tarih (`date`) değerine, belirtilen saat (`value`) miktarını ekleyerek yeni bir tarih döndürür.

Örnekler:

```
AddHours(new DateTime(2024,4,2,5,30,5), 1) = 02.04.2024 6:30:05
AddHours(ToDateTime("2.4.2024 5:30:05"), 1) = 02.04.2024 6:30:05
AddHours([Employees.BirthDate], 1) = 27.01.1986 1:00:00
```

AddMinutes

fonksiyon	parametreler	dönen değer
AddMinutes	<code>DateTime date, double value</code>	DateTime

Verilen tarih değerine (`date`) belirtilen dakika miktarını (`value`) ekler ve yeni tarihi döndürür.

Örnekler:

```
AddMinutes(new DateTime(2024,4,2,5,30,5), 10) = 02.04.2024 5:40:05
AddMinutes(ToDateTime("2.4.2024 5:30:05"), 10) = 02.04.2024 5:40:05
AddMinutes([Employees.BirthDate], 10) = 27.01.1986 0:10:00
```

AddMonths

fonksiyon	parametreler	dönen değer
AddMonths	<code>DateTime date, int value</code>	DateTime

Verilen tarih (`date`) değerine, belirtilen ay (`value`) miktarını ekler ve yeni tarihi döndürür.

Örnekler:

```
AddMonths(new DateTime(2024,4,2,5,30,5), 2) = 02.06.2024 5:30:05
AddMonths(ToDateTime("2.4.2024 5:30:05"), 2) = 02.06.2024 5:30:05
AddMonths([Employees.BirthDate], 2) = 27.03.1986 0:00:00
```

AddSeconds

fonksiyon	parametreler	dönen değer
AddSeconds	<code>DateTime date, double value</code>	DateTime

Verilen tarih (`date`) değerine belirtilen saniye (`value`) miktarını ekler ve yeni tarihi döndürür.

Örnekler:

```
AddSeconds(new DateTime(2024,4,2,5,30,5), 10) = 02.04.2024 5:30:15
AddSeconds(ToDateTime("2.4.2024 5:30:05"), 10) = 02.04.2024 5:30:15
AddSeconds([Employees.BirthDate], 10) = 27.01.1986 0:00:10
```

AddYears

fonksiyon	parametreler	döner değeri
AddYears	DateTime date, int value	DateTime

Verilen tarih (**date**) değerine belirtilen yıl (**value**) miktarını ekler ve yeni tarihi döndürür.

Örnekler:

```
AddYears(new DateTime(2024,4,2,5,30,5), 3) = 02.04.2027 5:30:05
AddYears(ToDateTime("2.4.2024 5:30:05"), 3) = 02.04.2027 5:30:05
AddYears([Employees.BirthDate], 3) = 27.01.1989 0:00:00
```

DateDiff

fonksiyon	parametreler	döner değeri
DateDiff	DateTime date1, DateTime date2	TimeSpan

Verilen iki tarih arasındaki farkı gün, saat, dakika ve saniye cinsinden hesaplar ve bu farkı yeni bir zaman aralığı olarak döndürür.

Örnekler:

```
DateDiff(new DateTime(2024,4,2,5,0,0), new DateTime(2025,1,2,5,30,5)) = -275.00:30:05
DateDiff(ToDateTime("2.1.2025 5:30:05"), ToDateTime("2.4.2024 5:00:00")) = 275.00:30:05
DateDiff(ToDateTime("2.4.2024 5:00:00"), [Employees.BirthDate]) = 13945.05:00:00
```

DateSerial

fonksiyon	parametreler	döner değeri
DateSerial	int year, int month, int day	DateTime

Belirtilen yıl (**year**), ay (**month**) ve gün (**day**) değerlerinden yeni bir **DateTime** nesnesi oluşturur.

Belirli bir tarihi ayarlamak için kullanılabilen ek bir yöntemdir.

Örnek:

```
DateSerial(2024,4,2) = 02.04.2024 0:00:00
```

Day

fonksiyon	parametreler	dönüş değeri
Day	DateTime date	int

Belirtilen tarihten ayın gününü (1-31) çıkarır.

Örnekler:

```
Day(new DateTime(2024,4,2)) = 2
Day(DateTime("2.4.2024")) = 2
Day([Employees.BirthDate]) = 27
```

DayOfWeek

fonksiyon	parametreler	dönüş değeri
DayOfWeek	DateTime date	string

Belirtilen tarihin haftanın gün adını döndürür (Pazartesi..Pazar).

Örnekler:

```
DayOfWeek(new DateTime(2024,4,2)) = "Salı"
DayOfWeek(DateTime("2.4.2024")) = "Salı"
DayOfWeek([Employees.BirthDate]) = "Pazartesi"
```

DayOfYear

Fonksiyon	Parametreler	Dönen değer
DayOfYear	DateTime date	int

Belirtilen tarihte yılın sıralı gün numarasını (1-365) döndürür.

Örnek:

```
DayOfYear(new DateTime(2024,4,2)) = 93
DayOfYear(DateTime("2.4.2024")) = 93
DayOfYear([Employees.BirthDate]) = 27
```

DaysInMonth

Fonksiyon	Parametreler	Dönen değer
DaysInMonth	int year, int month	int

Belirtilen yılın year ve ayının month içindeki gün sayısını döndürür.

Örnek:

```
DaysInMonth(2024, 4) = 30
```

Hour

Fonksiyon	Parametreler	Dönen değer
Hour	<code>DateTime date</code>	int

Belirtilen tarihten saati (0-23) döndürür.

Örnek:

```
Hour(new DateTime(2024,4,2,5,30,5)) = 5  
Hour(ToDateTime("2.4.2024 5:30:05")) = 5  
Hour([Employees.BirthDate]) = 0
```

Minute

Fonksiyon	Parametreler	Dönen değer
Minute	<code>DateTime date</code>	int

Belirtilen tarihten dakikayı (0-59) döndürür.

Örnek:

```
Minute(new DateTime(2024,4,2,5,30,5)) = 30  
Minute(ToDateTime("2.4.2024 5:30:05")) = 30  
Minute([Employees.BirthDate]) = 0
```

Month

Fonksiyon	Parametreler	Dönen değer
Month	<code>DateTime date</code>	int

Belirtilen tarihten ayı (1-12) döndürür.

Örnek:

```
Month(new DateTime(2024,4,2)) = 4
Month(ToDateTime("2.4.2024")) = 4
Month([Employees.BirthDate]) = 1
```

MonthName

Fonksiyon	Parametreler	Dönen değer
MonthName	int month	string

Belirtilen ay numarasıyla (1-12) yerleştirilmiş ay adını döndürür.

Örnek:

```
MonthName(1) = "Ocak"
```

Second

Fonksiyon	Parametreler	Dönen değer
Second	DateTime date	int

Belirtilen tarihten saniyeyi (0-59) döndürür.

Örnek:

```
Second(new DateTime(2024,4,2,5,30,5)) = 5
Second(ToDateTime("2.4.2024 5:30:05")) = 5
Second([Employees.BirthDate]) = 0
```

Year

Fonksiyon	Parametreler	Dönen değer
Year	DateTime date	int

Belirtilen tarihten yılı döndürür.

Örnek:

```
Year(new DateTime(2024,4,2)) = 2024
Year(ToDateTime("2.4.2024")) = 2024
Year([Employees.BirthDate]) = 1986
```

Biçimlendirme

Format

Sembol	Tip	Açıklama
C veya c	Para birimi	Sayı, para birimini temsil eden bir dizeye dönüştürülür. Örneğin, <code>Format("{0:C}", 10) = "10,00 ₺"</code>
D veya d	Ondalık sayı	Bu format yalnızca tamsayı türleri için kullanılabilir. Sayı, ondalık rakamlardan (0-9) oluşan bir dizeye dönüştürülür. <code>Format("{0:D}", 10) = "10"</code>
E veya e	Bilimsel	Sayı, <code>-d.ddd...E+ddd</code> veya <code>-d.ddd...e+ddd</code> biçiminde bir dizeye dönüştürülür; burada "d" (0-9) arası rakamı temsil eder. Örneğin, <code>Format("{0:E}", 10) = "1,000000E+001"</code>
F veya f	Sabit ondalık	Sayı, <code>-ddd.ddd...</code> biçiminde bir dizeye dönüştürülür; burada "d" (0-9) arasındaki rakamı temsil eder. Örneğin, <code>Format("{0:F}", 10) = "10,00"</code>
G veya g	Genel	Sayı en kısa gösterim biçimiyle dizeye dönüştürülür. Örneğin, <code>Format("{0:G}", 10) = "10"</code>
N veya n	Sayı	Sayı, <code>-d,ddd,ddd.ddd...</code> biçiminde dizeye dönüştürülür; burada "d" (0-9) arası rakamı temsil eder, "," binlik ayırıcı ve "." tam sayı ile ondalık kısmı ayırıcıdır. Örneğin, <code>Format("{0:N}", 1234.56) = "1 234,56"</code>
P veya p	Yüzde	Sayı, yüzdeyi temsil eden bir dizeye dönüştürülür. Dönüştürülen sayı, yüzde ifadesine uygun olması için 100 ile çarpılır. Örneğin, <code>Format("{0:P}", 0.15) = "15,00%"</code>
X veya x	Onaltılık	Bu format yalnızca tamsayı türleri için kullanılabilir. Sayı, onaltılık basamaklardan oluşan bir dizeye dönüştürülür. 9'dan büyük onaltılık basamakların harflerinin durumu, format belirtecinde kullanılan işaretin harf durumu ile aynıdır. Örneğin, <code>Format("{0:X}", 26) = "1A"</code>

Biçim belirtecinden hemen sonra bir sayı belirtilebilir. Ondalık sayıların biçimlendirilmiş değerinde, bu sayı ondalık kısmın basamak sayısını belirtir:

```
Format("{0:C1}", 12.23) = "12,2 ₺"
```

Eğer sayıların biçimlendirilmesi için standart yöntemler yeterli gelmezse, özel sayı biçimlendirme karakterleri kullanılabilir.

Sembol	Açıklama
0	Sıfır yer tutucu işareti. Biçimlendirilen değer bu konumda bir rakam içeriyorsa, o rakam çıktı dizisine kopyalanır. Aksi takdirde, çıktı dizisinde "0" görüntülenir.
#	Rakam yer tutucu. Eğer biçimlendirilen değer bu konumda bir rakam içeriyorsa, rakam çıkış dizisine kopyalanır. Aksi halde çıkış dizisine hiçbir şey yazılmaz.
.	Ondalık ayırıcı. Biçim dizesindeki ilk "." işareti, ondalık ayırıcının konumunu belirler.

,	Basamak gruplama ayırıcı. Biçim dizesindeki "," işareti, sayı basamak gruplarının ayırıcısının konumunu belirler.
%	Yüzde işareti. Biçimlendirme dizesinde "%" işareti kullanıldığında, sayı 100 ile çarpılır ve çıktı dizesinin ilgili konumuna "%" simgesi eklenir.
;	Bölüm ayırıcı. Biçim dizesindeki ";" işareti, pozitif, negatif ve sıfır sayılar için format bölümlerini ayırmak amacıyla kullanılır.

Kullanım örnekleri:

```
Format("{0:#,##0.00 P}", 1024.25) = "1 024,25 P"
Format("{0:00%}", 0.25) = "25%"
Format("{0:#,##0.00 P;(#,##0.00 P);Zero}", 1024.25) = "1 024,25 P"
Format("{0:#,##0.00 P;(#,##0.00 P);Zero}", -1024.25) = "(1 024,25 P)"
Format("{0:#,##0.00 P;(#,##0.00 P);Zero}", 0) = "Zero"
```

Tarih/saat biçimlendirmesi için aşağıdaki standart format belirteçleri kullanılır:

Sembol	tip	örnek
d	kısa tarih şablonu	"02.04.2024"
D	tam tarih şablonu	"2 Nisan 2024"
f	tam tarih ve saat şablonu (kısa saat şablonu)	"2 Nisan 2024 20:41"
F	tam saat şablonu	"2 Nisan 2024 20:41:33"
g	genel tarih ve saat şablonu (kısa saat şablonu)	"02.04.2024 20:41"
G	genel tarih ve saat şablonu (tam saat şablonu)	"02.04.2024 20:41:33"
t	kısa saat şablonu	"20:41"
T	tam saat şablonu	"20:41:33"

Ayrıca, özelleştirilmiş tarih/saat formatı sembollerini de kullanabilirsiniz:

sembol	açıklama
d	Ayın gününü 1 ile 31 arasında bir sayı olarak temsil eder. Tek haneli sayılar, başında sıfır olmadan biçimlendirilir.
dd	Ayın gününü 01 ile 31 arasında bir sayı olarak temsil eder.
ddd	Haftanın kısaltılmış gün adını temsil eder.
dddd	Haftanın tam gün adını temsil eder.
f veya F	Milisaniyeleri temsil eder.

h	1 ile 12 arasında bir sayı olarak saatleri temsil eder. Tek haneli sayılar, başında sıfır olmadan biçimlendirilir.
hh	Saatleri 01 ile 12 arasında bir sayı olarak temsil eder.
H	0 ile 23 arasında bir sayı olarak saatleri temsil eder. Tek haneli sayılar, başında sıfır olmadan biçimlendirilir.
HH	Saatleri 00 ile 23 arasında bir sayı olarak temsil eder.
m	Dakikaları 0 ile 59 arasında bir sayı olarak temsil eder. Tek haneli sayılar, başında sıfır olmadan biçimlendirilir.
mm	Dakikaları 00 ile 59 arasında bir sayı olarak temsil eder.
M	Ayı 1 ile 12 arasında bir sayı olarak temsil eder. Tek haneli sayılar, başında sıfır olmadan biçimlendirilir.
MM	Ayı 01 ile 12 arasında bir sayı olarak temsil eder.
MMM	Ayın kısaltılmış adını temsil eder.
MMMM	Ayın tam adını temsil eder.
s	Saniyeleri 0 ile 59 arasında bir sayı olarak temsil eder. Tek haneli sayılar, başında sıfır olmadan biçimlendirilir.
ss	Saniyeleri 00 ile 59 arasında bir sayı olarak temsil eder.
y	Yılı bir veya iki basamaklı sayı olarak temsil eder.
yy	Yılı iki basamaklı sayı olarak temsil eder.
yyyy	Yılı dört basamaklı sayı olarak temsil eder.
:	Standart zaman ayırıcını temsil eder.
/	Standart tarih ayırıcını temsil eder.

Kullanım örnekleri:

```
Format("{0:d MMM yyyy}", DateTime.Now) = "2 anp 2024"
Format("{0:dd/MM/yyyy}", DateTime.Now) = "02.04.2024"
Format("{0:d MMMM}", DateTime.Now) = "2 апреля"
Format("{0:HH:mm}", DateTime.Now) = "20:41"
Format("{0:dd/MM/yyyy HH:mm}", DateTime.Now) = "02.04.2024 20:41"
```

FormatCurrency

fonksiyon	parametreler	dönüş değeri
FormatCurrency	<code>object value</code>	string

Değer `value` 'yu Windows'un yerel ayarlarını kullanarak para birimi olarak biçimlendirir.

Örnek:

```
FormatCurrency(1.25) = "1,25 ₺"
```

fonksiyon	parametreler	dönüş değeri
FormatCurrency	object value, int decimalDigits	string

Değer `value` 'yu para birimi olarak biçimlendirir, sonucu `decimalDigits` ondalık basamak sayısına yuvarlar.

Örnek:

```
FormatCurrency(1.25, 1) = "1,3 ₺"
```

FormatDateTime

fonksiyon	parametreler	dönüş değeri
FormatDateTime	DateTime value	string

`value` değerini tarih/saat olarak biçimlendirir. Bu fonksiyon, ilgili bölümler boş ise tarih veya saati döndürmez. Boş tarih 1/1/1, boş saat ise 0:00:00 olarak kabul edilir.

Örnek:

```
FormatDateTime(new DateTime(2024,4,2)) = "02.04.2024"  
FormatDateTime(new DateTime(2024,4,2,1,30,0)) = "02.04.2024 1:30:00"  
FormatDateTime(new DateTime(1,1,1,1,30,1)) = "1:30:01"
```

fonksiyon	parametreler	dönüş değeri
FormatDateTime	DateTime value, string format	string

`value` değerini, format adlı isimlendirilmiş format kullanarak tarih/saat olarak biçimlendirir. format parametresi için izin verilen değerler:

- Long Date;
- Short Date;
- Long Time;
- Short Time.

Örnek:

```
FormatDateTime(new DateTime(2024,4,2,1,30,0), "Long Date") = "2 Nisan 2024"  
FormatDateTime(new DateTime(2024,4,2), "Short Date") = "02.04.2024"  
FormatDateTime(new DateTime(1,1,1,1,30,1), "Long Time") = "1:30:01"  
FormatDateTime(new DateTime(1,1,1,1,30,1), "Short Time") = "01:30"
```

FormatNumber

fonksiyon	parametreler	dönüş değeri
FormatNumber	object value	string

`value` değerini, Windows'un yerel ayarlarını kullanarak sayı olarak biçimlendirir.

Örnek:

```
FormatNumber(1234.56) = "1 234,56"
```

fonksiyon	parametreler	dönüş değeri
FormatNumber	object value, int decimalDigits	string

`value` değerini sayı olarak biçimlendirir ve sonucu `decimalDigits` ondalık basamak sayısına yuvarlar.

Örnek:

```
FormatNumber(1234.56, 1) = "1 234,6"
```

FormatPercent

fonksiyon	parametreler	dönüş değeri
FormatPercent	object value	string

`value` değerini, Windows'un yerel ayarlarını kullanarak yüzde olarak biçimlendirir.

Örnek:

```
FormatPercent(0.15) = "15,00%"
```

fonksiyon	parametreler	dönüş değeri
FormatPercent	object value, int decimalDigits	string

`value` deęerini yzde olarak bięimlendirir ve sonucu `decimalDigits` ondalık basamak sayısına yuvarlar.

rnek:

```
FormatPercent(0.15, 0) = "15%"
```

Dönüştürme

ToBoolean

fonksiyon	parametreler	dönüş değeri
ToBoolean	object value	bool

value değerini mantıksal türe dönüştürür.

Örnek:

```
ToBoolean(1) = true  
ToBoolean(0) = false
```

ToByte

fonksiyon	parametreler	dönüş değeri
ToByte	object value	byte

value değerini byte türüne dönüştürür.

Örnek:

```
ToByte("55") = 55
```

ToChar

fonksiyon	parametreler	dönüş değeri
ToChar	object value	char

value değerini sembole dönüştürür.

Örnek:

```
ToChar(65) = 'A'
```

ToDateTime

fonksiyon	parametreler	dönüş değeri
<code>ToDateTime</code>	object value	DateTime

`value` değerini DateTime türüne dönüştürür.

Nokta (.), virgül (,) ve eğik çizgi (/) ayırıcılarının kullanılması mümkündür.

Örnek:

```
DateTime("2.4.2024") = 02.04.2024 0:00:00
DateTime("2,4,2024") = 02.04.2024 0:00:00
DateTime("2/4/2024") = 02.04.2024 0:00:00
```

ToDecimal

fonksiyon	parametreler	dönüş değeri
<code>ToDecimal</code>	object value	decimal

`value` değerini decimal türüne dönüştürür.

Örnek:

```
ToDecimal(1) = 1m
ToDecimal("1") = 1m
```

ToDouble

fonksiyon	parametreler	dönüş değeri
<code>ToDouble</code>	object value	double

`value` değerini double türüne dönüştürür.

Örnek:

```
ToDouble(1) = 1
ToDouble("1") = 1
```

ToInt32

fonksiyon	parametreler	dönüş değeri
<code>ToInt32</code>	object value	int

alue değerini int türüne dönüştürür.

Örnek:

```
ToInt32(1f) = 1  
ToInt32("1") = 1
```

ToRoman

fonksiyon	parametreler	dönüş değeri
ToRoman	object value	string

value değerini Roma rakamlarına dönüştürür.

value değeri 3998'i aşmamalıdır.

Örnek:

```
ToRoman(9) = "IX"
```

ToSingle

fonksiyon	parametreler	dönüş değeri
ToSingle	object value	float

value değerini float türüne dönüştürür.

Örnek:

```
ToSingle(1m) = 1f  
ToSingle("1") = 1f
```

ToString

fonksiyon	parametreler	dönüş değeri
ToString	object value	string

value değerini string türüne dönüştürür.

Örnek:

```
ToString(false) = "False"
ToString(DateTime.Now) = "02.04.2024 20:41:08"
```

ToWords

fonksiyon	parametreler	dönüş değeri
ToWords	object value	string

value değerini, İngilizce dilinde para birimi cinsinden kelimelere dönüştürür.

Örnek:

```
ToWords(1024.25) = "One thousand and twenty-four dollars and 25 cents"
```

fonksiyon	parametreler	dönüş değeri
ToWords	object value, string currencyName	string

value değerini, İngilizce dilinde yazıyla para miktarı şeklinde ifade eder.

Para birimi, currencyName parametresi ile belirtilir. Bu parametrenin olası değerleri:

- "USD";
- "EUR";
- "GBP".

Örnek:

```
ToWords(1024.25, "EUR") = "One thousand and twenty-four euros and 25 cents"
```

fonksiyon	parametreler	dönüş değeri
ToWords	object value, string one, string many	string

value tamsayı değerini, İngilizce dilinde sayının yazıyla ifadesine dönüştürür.

one ve many parametrelerinde, sırasıyla tekil ve çoğul sözcük biçimleri belirlenir.

Örnek:

```
ToWords(124, "page", "pages") = "One hundred and twenty-four pages"
ToWords(1, "page", "pages") = "One page"
```

ToWordsEnGb

fonksiyon	parametreler	dönüş değeri
ToWordsEnGb	object value	string

`value` değerini, İngiliz İngilizcesinde para miktarının yazıyla ifadesine dönüştürür.

ToWords fonksiyonundan farkı şudur: varsayılan olarak GBP para birimi kullanılır; milyar ve trilyon değerlerinin tercümesinde farklılık vardır.

Örnek:

```
ToWordsEnGb(121) = "One hundred and twenty-one pounds and 00 pence"
```

fonksiyon	parametreler	dönüş değeri
ToWordsEnGb	<code>object value, string currencyName</code>	string

Sayısal değer olan `value` 'yu, parametre olarak verilen `currencyName` değerine göre, İngiliz İngilizcesinde yazıyla ifade edilmiş bir para tutarına dönüştürür. `currencyName` parametresi için olası değerler şunlardır:

- "USD";
- "EUR";
- "GBP".

Örnek:

```
ToWordsEnGb(1024.25, "EUR") = "One thousand and twenty-four euros and 25 cents"
```

fonksiyon	parametreler	dönüş değeri
ToWordsEnGb	object value, string one, string many	string

Verilen tam sayı değeri `value` , Britanya İngilizcesinde yazıyla ifade edilen biçime dönüştürülür. Parametreler `one` ve `many` , sırasıyla tekil ve çoğul kelime formlarını belirtir.

Örnek:

```
ToWordsEnGb(124, "page", "pages") = "One hundred and twenty-four pages"
ToWordsEnGb(1, "page", "pages") = "One page"
```

ToWordsRu

fonksiyon	parametreler	dönüş değeri
ToWordsRu	<code>object value</code>	string

Bu fonksiyon, sayısal değer `value` 'yu Rusça para tutarı şeklinde yazıya çevirir.

Örnek:

```
ToWordsRu(1024.25) = "Одна тысяча двадцать четыре рубля 25 копеек"
```

fonksiyon	parametreler	dönüş değeri
ToWordsRu	object value, string currencyName	string

Fonksiyon, sayısal değer olan `value` 'yu, currencyName parametresinde belirtilen para birimine göre Rusça olarak yazıyla ifade edilmiş tutara çevirir. Bu parametrenin alabileceği değerler şunlardır:

- "RUR";
- "UAH";
- "USD";
- "EUR".

Örnek:

```
ToWordsRu(1024.25, "EUR") = "Одна тысяча двадцать четыре евро 25 евроцентов"
```

Fonksiyon	Parametreler	Dönüş değeri
ToWordsRu	object value, bool male, string one, string two, string many	string

Tam sayı değerini (`value`) Rusça yazıyla ifade eder. Eğer isim erkek cinsiyetteyse, male parametresine true değeri atanır. `one` , `two` ve `many` parametrelerinde sırasıyla 1, 2 ve 5 sayıları için kelime formları belirtilir (örnek: 1 "лист", 2 "листа", 5 "листов").

Örnek:

```
// "страница" kelimesi dişil cinsiyette olduğu için - parametre male = false
ToWordsRu(124, false, "страница", "страницы", "страниц") = "Сто двадцать четыре страницы"

// "лист" kelimesi eril cinsiyette olduğu için - parametre male = true \
ToWordsRu(124, true, "лист", "листа", "листов") = "Сто двадцать четыре листа"
```

Koşullar

Choose

fonksiyon	parametreler	dönüş değeri
Choose	double index, params object[] choice	object

`choice` dizisinden `index` numaralı öğeyi döndürür. Dizideki ilk öğenin indeksi 1'dir.

Örnek:

```
Choose(2, "one", "two", "three") = "two"
```

IIf

fonksiyon	parametreler	dönüş değeri
IIf	bool expression, object truePart, object falsePart	object

Eğer `expression` `true` ise, `truePart` değerini döndürür; aksi halde `falsePart` değerini döndürür.

Örnek:

```
IIf(2 > 5, "true", "false") = "false"
```

Switch

fonksiyon	parametreler	dönüş değeri
Switch	params object[] expressions	object

`expressions` parametresine "koşul-değer" çiftleri gönderilir. Fonksiyon, sırasıyla verilen koşulları değerlendirir ve ilk koşulun true (doğru) olduğu durumda, o koşula karşılık gelen değeri döndürür.

Örnek:

```
// returns one of the following values - "a greater than 0",  
// "a less than 0", "a equals to 0", depending on "a" value  
Switch(  
    a > 0, "a greater than 0",  
    a < 0, "a less than 0",  
    a == 0, "a equals to 0")
```

IsNull

fonksiyon	parametreler	dönüş değeri
IsNull	<code>string name</code>	bool

`name` parametresi, veritabanı sütun adı, parametre adı veya toplam adı olarak kullanılabilir ve çift tırnak içerisinde yazılmalıdır.

Eğer `name` ile belirtilen nesnenin değeri null ise, metot `true` döndürür; aksi halde `false` döndürür.

Örnek:

```
IsNull("Parameter")
```

Toplam Değerler

Çoğu raporda, grup toplamı, listedeki satır sayısı gibi özet bilgiler gösterilmelidir. FastReport'ta bu amaçla kullanılan "toplam değerler" sayesinde belirli bir veri aralığı için fonksiyon hesaplanabilir.

Toplam için aşağıdaki parametreler belirtilir:

- Toplam fonksiyon türü;
- Fonksiyonun hesaplanacağı veri aralığı;
- Hesaplanacak ifade. (Adet fonksiyonu için ifade belirtilmez);
- İfade – koşul. Bu koşul sağlanırsa fonksiyon hesaplanır. (Bu parametre isteğe bağlıdır.)

Veri aralığı şu şekilde belirlenir: Fonksiyon her satır için hesaplanacak "Veriler" bandı belirtilir. Ayrıca, toplamın yazdırılacağı band tanımlanır. Bu band yazdırıldıktan sonra toplam değeri sıfırlanır.

Aşağıda toplam fonksiyonların listesi verilmiştir:

Fonksiyon	Açıklama
Toplam	Belirtilen ifade için veri aralığının toplamını hesaplar.
Minimum	Belirtilen ifade için veri aralığının minimum değerini hesaplar.
Maksimum	Belirtilen ifade için veri aralığının maksimum değerini hesaplar.
Ortalama	Belirtilen ifade için veri aralığının ortalama değerini bulur.
Adet	Veri aralığındaki satır sayısını döndürür.
Benzersiz Adet	Veri aralığındaki farklı satır sayısını döndürür.

Sonuç Oluřturma

Toplam fonksiyonunun kullanımını bir örnekle inceleyeceęiz. 'Categories' ve 'Products' adlı iki tabloyu kullanan bir 'master-detail' (ana-ayrıntı) raporu oluřturalım:



Hazırlanan rapor ařaęıdaki gibi olacaktır:

Beverages		Condiments	
Product Name	Units In Stock	Product Name	Units In Stock
Chai	39	Aniseed Syrup	13
Chang	17	Chef Anton's Cajun Seasoning	53
Chartreuse verte	69	Chef Anton's Gumbo Mix	0
Côte de Blaye	17	Genen Shouyu	39
Guaraná Fantástica	20	Grandma's Boysenberry Spread	120
Ipoh Coffee	17	Gula Malacca	27
Lakkalikööri	57	Louisiana Fiery Hot Pepper Sauce	76
Laughing Lumberjack Lager	52	Louisiana Hot Spiced Oliva	4
Outback Lager	15	Northwoods Cranberry Sauce	6
Röhnbräu Klosterbier	125	Original Frankfurter grüne Soße	32
Sasquatch Ale	111	Sirup d'érable	113
Steeleye Stout	20	Veggie-spread	24

Bu rapora, her kategori için stokta bulunan toplam birim sayısını (UnitsInStock veri sütununun toplamı) yazdıracak bir toplam ekleyelim. Toplam, 'Data Footer' bandında yazdırılacaktır.

Toplam değeri yazdırmak için önce onu oluřturmanız gerekir. Bunun için 'Veri' (Data) penceresindeki 'Eylem' (Action) düęmesine basın ve 'Yeni toplam' (New total) öęesini seçin.

Bir dięer yöntem ise, veri ağacında 'Toplamlar' (Totals) öęesine sağ tıklayıp 'Yeni toplam' (New total) menü öęesini seçmektir. Toplam düzenleyici penceresi görünecektir.

Koşula göre toplam hesaplama

Önceki örnekte, toplam tüm veri aralığı için hesaplanıyordu. Toplam düzenleyicisinde bir koşul belirterek aralığı kısıtlayabiliriz. Toplam, yalnızca koşulun true döndürdüğü aralıktaki satırlar için hesaplanacaktır. Örneğin, aşağıdaki koşulu belirtebiliriz:

Edit Total

Total

Total name: TotalUnits

Function: Sum

Data column or expression: [Products.UnitsInStock]

Evaluate on each row of the band: Data 1: Products

Evaluate if the following condition is met: ![Products.Discontinued]

Print on the band: DataFooter 1: Products

Options

- ☒ Reset after print
- ☐ Reset if band is repeated
- ☐ Include invisible rows

OK Cancel

Bu, yalnızca Discontinued bayrağı ayarlanmamış ürünler için toplamın hesaplanması gerektiği anlamına gelir.

Kümülatif Toplamın Yazdırılması

Örneğimizde, toplamlar "Veri Altbilgisi" bandı yazdırıldıktan sonra sıfırlanıyordu. Bu durum, editörde toplamın bu bandın yazdırılmasından sonra sıfırlanması gerektiğini belirtmemizden kaynaklanıyordu. Böylece, her ürün kategorisi kendi toplam değerlerini yazdırıyordu.

Eğer "Yazdırdıktan sonra sıfırla" onay kutusu kapatılırsa, toplam yazdırıldıktan sonra sıfırlanmayacaktır. Bu, kümülatif toplam olarak adlandırılır.

Aynı anda hem normal toplamaları hem de kümülatif toplamaları yazdırmanız gerekiyorsa, benzer ayarlara sahip başka bir toplam oluşturun ve "Yazdırdıktan sonra sıfırla" onay kutusunu kapatın.

Sayfa Sonuçları

Sayfanın altbilgisinde basılacak olan sonucu oluşturmak için, "Banda Yazdır" alanına bantı – sayfa altbilgisi – belirtiniz.

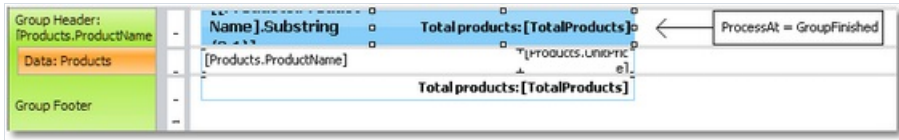
Başlıkta Toplamın Yazdırılması

Genellikle toplam değerler, bantların alt bölümlerinde (veri altı, grup altı vb.) yazdırılır. Bu, yazdırmanın doğal sırasıdır: o anda toplam zaten hesaplanmış ve kullanılmaya hazırdır. Ancak, çoğu zaman toplamı, örneğin grup başlığında yazdırmak gerekebilir. Bunu yapmaya çalıştığımızda sıfır değeri görürüz. Çünkü o anda toplam henüz hesaplanmaya başlanmamıştır; değeri grup alt bölümünde hazır olacaktır.

Bu problemi çözmek için FastReport, "ertelenmiş baskı" mekanizması sunar. Text nesnesinin ProcessAt adlı özelliği vardır ve bu özellik aşağıdaki değerlerden birini alabilir:

Değer	Açıklama
Default	Normal mod, varsayılan değer.
ReportFinished	Nesnenin değeri, rapor tamamlandığında güncellenir.
ReportPageFinished	Rapor sayfasındaki tüm bantlar tamamlandığında nesnenin değeri güncellenir.
PageFinished	Nesnenin değeri, sayfa altbilgisinin baskısı sırasında güncellenir.
ColumnFinished	Nesnenin değeri, kolon altbilgisinin baskısı sırasında güncellenir.
DataFinished	Nesnenin değeri, veri altbilgisinin baskısı sırasında güncellenir.
GroupFinished	Nesnenin değeri, grup altbilgisinin baskısı sırasında güncellenir.

Çalışma prensibini bir örnekle açıklayalım. Grup başlığına, grubun toplamını yazdıran "Metin" adlı bir nesne yerleştirelim. ProcessAt özelliğini GroupFinished olarak ayarlayalım:



Rapor çalıştırıldığında aşağıdakiler gerçekleşir:

- Grup başlığı yazdırılacaktır. Bu sırada FastReport, sıfır (yanlış) toplamı yazdıracak ve bu yeri son raporda kaydedecektir;
- Grup verilerinin tüm satırları yazdırılacaktır;
- Grup altbilgisi yazdırılacaktır. Bu noktada FastReport, daha önce kaydedilen nesneyi alacak ve tekrar işleyecek, bu sefer doğru toplam değeri ile.

Hazır rapor şu şekilde görünür:

A		Total products: 2
Alice Mutton		\$39.00
Aniseed Syrup		\$10.00
		Total products: 2
B		Total products: 1
Boston Crab Meat		\$18.40
		Total products: 1
C		Total products: 9
Camembert Pierrot		\$34.00
Carnarvon Tigers		\$62.50
Chai		\$18.00
		\$19.00

Benzer şekilde, raporun tamamına ait toplamı rapor başlığında (`ProcessAt = ReportFinished` olarak ayarlayın) veya sayfa toplamını sayfa başlığında (`ProcessAt = PageFinished` olarak ayarlayın) yazdırmak mümkündür.

Ertelenmiş baskı mekanizması, dosya önbellesi etkinleştirildiğinde çalışmaz (menü "Rapor|Özellikler...", "Dosya önbellesini kullan" işareti).

Rapor Parametreleri

Rapor içerisinde parametreler tanımlanabilir. Parametre, değeri hem rapor içinde hem de dışında tanımlanabilen bir değişkendir (raporu çağıran program, parametre değerlerini rapora iletebilir – detaylar için "Programcı Kılavuzu"na bakınız). Parametre ifadelerde kullanılabilir, "Metin" türündeki nesnelerde görüntülenebilir.

Parametrelerin en yaygın kullanım şekilleri:

- Parametre ile belirlenen koşula göre verilerin filtrelenmesi;
- Rapor içerisinde parametre değerinin yazdırılması.

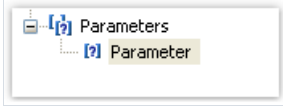
Parametrenin aşağıdaki özellikleri vardır:

Özellik	Açıklama
Name	Parametre adı, nokta dışında herhangi bir karakter içerebilir.
DataType	Parametre veri tipi.
Expression	Parametre değerini döndüren ifade. İfadeler hakkında daha fazla bilgi için "İfadeler" bölümüne bakınız. Bu ifade, parametreye erişildiğinde hesaplanacaktır.
Value	Parametre değeri. Bu özellik tasarımcıda erişilebilir değildir ve programatik olarak doldurulabilir.

Name ve DataType özelliklerini yapılandırmalısınız. Expression özelliği boş bırakılabilir. Bu durumda, parametre değeri programatik olarak iletilmelidir.

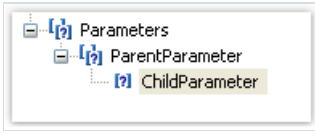
Parametre oluřturma

Bir parametre oluřturmak iin, "Veri" penceresinde "Rapor Parametreleri" gesini sein ve baėlam mensnden "Yeni Parametre" seeneėini belirleyin:



F2 tuřuna basarak istenen parametre adını girin, ardından "zellikler" penceresine geip parametrenin veri tipini (**DataType** zelliėi) belirtin.

Parametreler i ie oluřturulabilir. Bir i parametre oluřturmak iin, st parametreyi sein ve baėlam mensnden "Yeni Parametre" seeneėini belirleyin:



Bu sayede hem st parametreye hem de i parametreye eriřim saėlanır. İ ie gemiř seviye sayısı sınırsızdır.

Raporlarda Parametrelerin Kullanımı

Bir parametreye, ifadenin içinde köşeli parantezler kullanarak başvurulabilir:

```
[Parameter name]
```

İç içe (nested) bir parametreye aşağıdaki şekilde başvurulmalıdır:

```
[Parent parameter.Child parameter]
```

Parametrenin belirli bir türe sahip olması nedeniyle (DataType özelliğinde belirtilmiştir), o veri tipine uygun işlemleri gerçekleştirebilirsiniz. Örneğin, string tipindeki parametreler ifadelere şu şekilde dahil edilebilir:

```
[StringParameter].Substring(0, 2)
```

Bir parametre kullanım örneğine bakalım. Diyelim ki, **Employees** (Çalışanlar) tablosunu yazdıran bir raporumuz var. Belirtilen numaraya sahip çalışanın (EmployeeID alanı) bilgilerini yazdırmak istiyoruz. Bu durumda, "Çalışan Numarası" adlı bir parametre oluşturacağız. Parametrenin tipini, EmployeeID alanının tipine uygun olarak **Int32** olarak belirleyeceğiz. Belirtilen numaraya sahip çalışanı filtrelemek için, "Veriler" bandı düzenleyicisinde yer alan "Filtre" sekmesine aşağıdaki ifadeyi ekleyeceğiz:

```
[Employees.EmployeeID] == [EmployeeID]
```

Programınızdan rapora parametre değeri aktarmak için ise şu kodu kullanabilirsiniz:

```
report1.SetParameterValue("EmployeeID", 2);
```

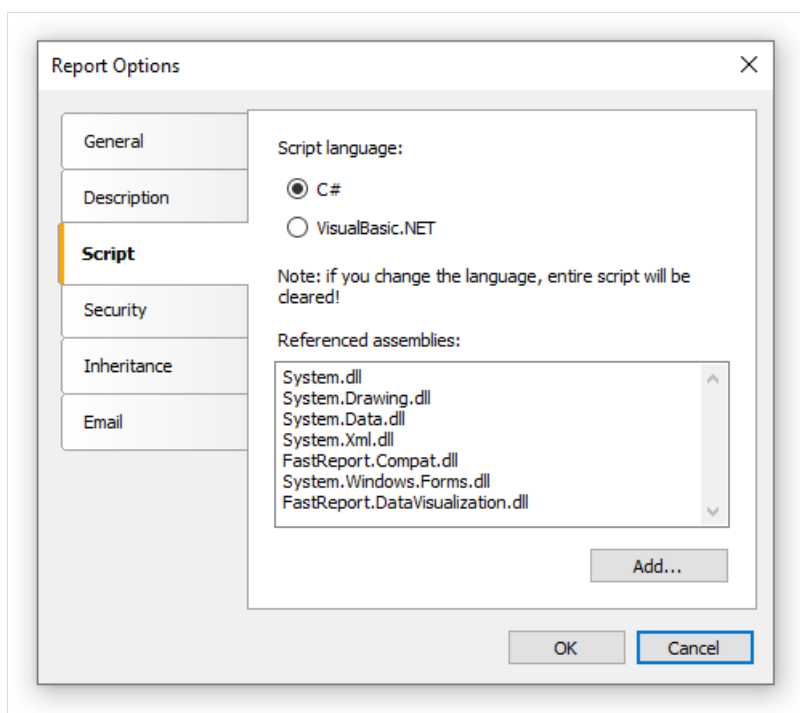
Expressions

In many places in FastReport, expressions are used. For example, the "Text" object can contain expressions in square brackets.

An expression is a code in C# or VB.NET language, which returns any value. For example:

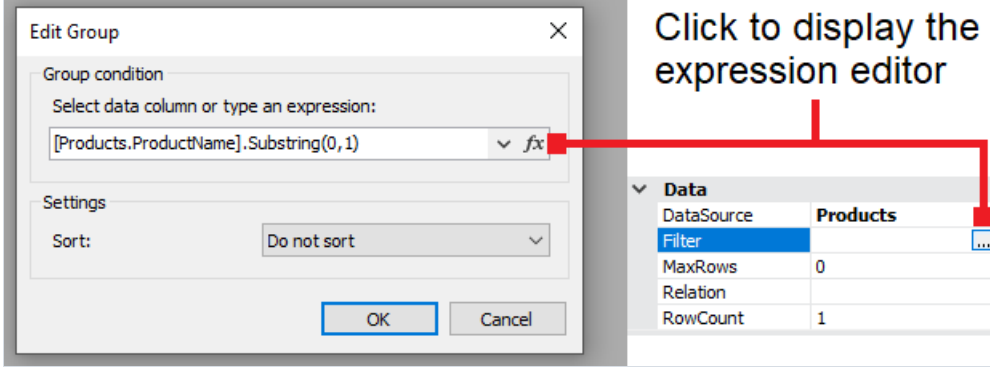
```
2 + 2
```

An expression should be written in a language chosen as a script in a report. By default, it is C#. You can change the language in the "Report|Options..." menu by choosing the "Script" element in a window.

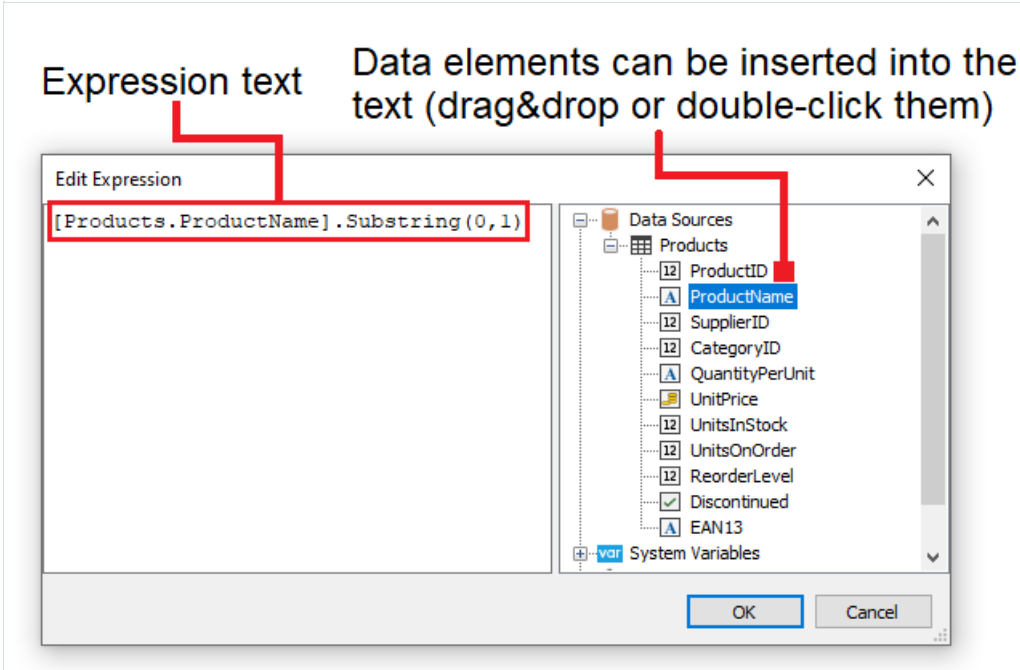


İfade Düzenleyici

İfadenin yazımını kolaylaştırmak için ifade düzenleyicisini kullanın. FastReport arayüzünde ifadelerin girilebildiği yerlerde çağrılabilir:



İfade düzenleyici, ifadenin metnini girebileceğiniz ve "Veri" penceresinden öğeleri ekleyebileceğiniz bir penceredir:



Rapor nesnelere erişim

Rapor nesnelere erişmek için (örneğin, "Text" nesnesine) nesnenin adını kullanın. Aşağıdaki örnek, Text1 nesnesinin yüksekliğini döndürür:

```
Text1.Height
```

Rapor özelliklerine erişmek için `Report` değişkenini kullanın. Aşağıdaki örnek, raporun yüklendiği dosyanın adını döndürür:

```
Report.FileName
```

Ayrıca, nesnenin iç içe geçmiş özelliklerine de erişebilirsiniz. Aşağıdaki örnek, raporun adını döndürür:

```
Report.ReportInfo.Name
```

.NET Fonksiyonlarının Kullanımı

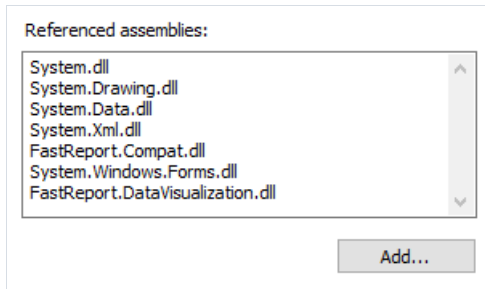
Herhangi bir .NET nesnesini ifadelere dahil edebilirsiniz. Aşağıdaki örnek, `Max` fonksiyonunun kullanımını göstermektedir:

```
Math.Max(5, 10)
```

Varsayılan olarak, rapor aşağıdaki .NET derlemelerini kullanır:

```
System.dll
System.Drawing.dll
System.Windows.Forms.dll
System.Data.dll
System.Xml.dll
```

Bu derlemelerde tanımlı tüm .NET nesnelere erişiminiz vardır. Başka bir derlemeye erişmeniz gerekiyorsa, rapor derlemeleri listesine o derlemenin adını ekleyin. Bu işlem, "Rapor|Özellikler..." menüsünde, "Script" seçeneği aracılığıyla yapılabilir:



Örneğin, raporda tanımlı bir fonksiyonu kullanmak istiyorsanız, rapor derlemeleri listesine uygulamanızın derlemesine (.exe veya .dll) referans ekleyin. Sonrasında, fonksiyona uygulamanızın namespace'i kullanılarak erişebilirsiniz. Örneğin, uygulamada aşağıdaki fonksiyon tanımlıysa:

```
namespace Demo
{
    public static class MyFunctions
    {
        public static string Func1()
        {
            return "Hello!";
        }
    }
}
```

Rapor içinde bu fonksiyona şu şekilde erişebilirsiniz:

```
Demo.MyFunctions.Func1()
```

Eğer script'e `using Demo` direktifi eklerseniz, fonksiyona erişim şeklini kısaltabilirsiniz:

```
MyFunctions.Func1()
```

Rapor scriptinde tanımladığınız değişkenlere veya fonksiyonlara erişmek için, doğrudan değişken veya fonksiyon adını kullanın:

```
myPrivateVariableThatIHaveDeclaredInScript  
MyScriptFunction()
```

İfade içinde, değer döndüren fonksiyonlar kullanılabilir.

Rapor verilerine erişim

Standart dil öğelerinin yanı sıra, ifadelerde aşağıdaki rapor elemanları kullanılabilir:

- Veri kaynaklarının alanları;
- Sistem değişkenleri;
- Toplam değerler;
- Rapor parametreleri.

Bu öğelerin tamamı "Veriler" penceresinde bulunur. Bunlarla ilgili detaylı bilgiyi "Veriler" başlıklı bölümü inceleyerek edinebilirsiniz. Herhangi bir öğeyi kullanmak için ifadenin içine köşeli parantez içine alabilirsiniz. Örneğin:

[Page] + 1

Bu ifade, bir sonraki basılacak sayfanın numarasını döndürür. İfadede, raporun mevcut sayfa numarasını veren **Page** adlı sistem değişkeni kullanılmış ve köşeli parantez içine alınmıştır.

Veri Kaynaklarına Başvurma

Veri kaynaklarının alanlarına erişim için aşağıdaki yazım biçimi kullanılır:

```
[KaynakAdı.AlanAdı]
```

Kaynak adı ile alan adı arasında nokta kullanılır, örneğin:

```
[Employees.FirstName]
```

Eğer veri kaynağına bir ilişki (relation) kullanılarak erişiliyorsa, kaynak adı bileşik olabilir. İlişkiler hakkında daha fazla bilgi "Veriler" bölümünde anlatılmıştır. Örneğin, ilişkili veri kaynağının alanına şu şekilde erişilebilir:

```
[Products.Categories.CategoryName]
```

Alanların ifadelere nasıl dahil edildiğine dair bir örneğe bakalım:

```
[Employees.FirstName] + " " + [Employees.LastName]
```

Burada önemli bir noktaya değinmek gerekir. Her alanın belirli bir veri tipi vardır – bu, alanın Özellikler penceresinde DataType (Veri Tipi) özelliğinde belirtilir (Önceden Veri penceresinde ilgili alan seçilerek görülebilir). Bir alanın tipine bağlı olarak, ifadelerde nasıl kullanılabileceği belirlenir. Yukarıdaki örnekte, hem ad hem de soyad alanları string (metin) tipindedir; bu nedenle, bu şekilde kullanılmaları uygundur. Aşağıdaki örnekte ise, Employees.Age (yaş) gibi sayısal tipte bir alanı kullanmaya çalışmak hata verecektir:

```
[Employees.FirstName] + " " + [Employees.Age]
```

Bu hata, bir metin ile sayının doğrudan toplanamamasından kaynaklanır. Bunun için sayının açıkça string'e dönüştürülmesi gerekir:

```
[Employees.FirstName] + " " + [Employees.Age].ToString()
```

Bu örnekte, Employees.Age alanına sanki tamsayı bir değişkenmiş gibi davranılmıştır. Gerçekte de öyledir. Bildiğimiz üzere, tüm ifadeler çalıştırılabilir koda derlenir. Derleyicinin standart dışı gördüğü tüm işlemler (sistem değişkenlerine ve veri alanlarına erişim gibi), derleyicinin anlayabileceği farklı bir biçime dönüştürülür. Örneğin, yukarıdaki ifade şu biçime derlenecektir:

```
(string)(Report.GetColumnValue("Employees.FirstName")) + " " +  
(int)(Report.GetColumnValue("Employees.Age")).ToString()
```

Görüldüğü üzere, FastReport derleme sırasında veri alanlarına erişimi şu şekilde değiştirir:

```
[Employees.FirstName] --> (string)(Report.GetColumnValue("Employees.FirstName"))
```

```
[Employees.Age] --> (int)(Report.GetColumnValue("Employees.Age"))
```

Yani, veritabanı alanlarını belirli bir tipe sahip değişkenler gibi ifadelerde kullanabiliriz. Örneğin, aşağıdaki ifade çalışan adının ilk karakterini döndürecektir:

```
[Employees.FirstName].Substring(0, 1)
```

Sistem deęiřkenlerine eriřim

İfadelerde ařaęıdaki sistem deęiřkenleri kullanılabilir (bunlar "Veri" penceresinde mevcuttur):

Deęiřken	.NET veri türü	Açıklama
Date	DateTime	Papor bařlangıç tarihi ve saati.
Page	int	Geçerli sayfa numarası.
TotalPages	int	Raporun toplam sayfa sayısı. Bu deęiřkeni kullanmak için raporda çift geçiř modunu etkinleřtirmek gerekir. Bu ayarı "Rapor Özellikler..." menüsünden yapabilirsiniz.
PageN	string	Sayfa numarası řu formatta: "Sayfa N".
PageNofM	string	Sayfa numarası řu formatta: "Sayfa N'den M'e".
Row#	int	Grup içindeki veri satır numarası. Bu deęer, yeni grup bařladıęında sıfırlanır.
AbsRow#	int	Verilerin mutlak satır numarası. Bu deęer, yeni grup bařladıęında sıfırlanmaz.

Her deęiřkenin belirli bir veri tipi vardır; bu nedenle ifadelerde nasıl kullanılacaęı buna baęlıdır. İřte tarih kullanılan bir ifadenin örneęi:

```
[Date].Year
```

Bu ifade, geçerli yılı döndürür. Çünkü **Date** deęiřkeni **DateTime** tipindedir, dolayısıyla Year özellięine eriřebiliriz. Benzer řekilde geçerli ayı almak için [Date].Month kullanılabilir.

FastReport, sistem deęiřkenine yapılan bařvuruyu (örneęin Date deęiřkeni için) řu řekilde dönüřtürür:

```
((DateTime)Report.GetVariableValue("Date"))
```

Toplam Değerlere Başvurma

Toplam değere başvurmak için, onun adını kullanın:

```
[TotalSales]
```

FastReport, toplam değere yapılan başvuruyu aşağıdaki forma dönüştürür:

```
Report.GetTotalValue("TotalSales")
```

Görüldüğü gibi, burada veri tipi belirtilmemiştir. Bunun nedeni, toplam değer `FastReport.Variant` tipinde olmasıdır. Bu tip, herhangi bir türe otomatik olarak dönüştürülebilir; dolayısıyla, toplam değer doğrudan her türlü ifadede kullanılabilir. Örneğin:

```
[TotalSales] * 0.2f
```

Rapor Parametrelerine Eriřim

Bir rapor parametresine bařvurmak iin, onun adını kullanın:

```
[Parameter1]
```

Parametreler i ie olabilir. Bu durumda, ebeveyn parametrenin adı ve alt parametrenin adı, bir nokta ile ayrılarak belirtilir:

```
[ParentParameter.ChildParameter]
```

Parametreler, veri alanları ve sistem deėiřkenleri gibi, belirli bir veri tipine sahiptir. Bu tip, parametrenin **DataType** özelliėinde tanımlanır. Parametrenin veri tipi, ifadedeki nasıl kullanılabileceėini belirler.

FastReport, rapor parametresine yapılan bařvuruyu (örneėin, string parametre iin) ařaėıdaki řekle dönüřtürür:

```
((string)Report.GetParameterValue("Parameter1"))
```

Script

Script is a higher-level programming language, which is part of the report. Script can be written in one of the following .NET languages:

- C#
- VisualBasic.NET

A script is applicable in many places. Using the script, you can do the following:

- perform data handling, which cannot be done via regular means of the FastReport engine;
- control the printing of report pages and bands on the page;
- control the interaction between elements on dialogue forms;
- control the formation of dynamic "Table" objects;
- and many more.

In order to see the report's script, switch to the "Code" tab in the designer:

Main class **Data elements can be dragged into the script**

The screenshot shows the FastReport designer interface. The main window displays the 'Code' tab with C# script. The script includes using statements for System, System.Collections, System.ComponentModel, System.Windows.Forms, System.Drawing, System.Data, and FastReport namespaces. It defines a public class 'ReportScript' with a private void method 'Picture1_Click'. Red arrows point from the text 'Main class' to the 'ReportScript' class definition. Another red arrow points from the text 'Data elements can be dragged into the script' to the 'Data Sources' pane on the right, which shows a tree view of data elements including 'Employees', 'EmployeeID', 'LastName', and 'FirstName'. Below the main window, there are three labels with red arrows pointing to specific parts of the interface: 'Code' tab points to the 'Code' tab in the bottom pane; 'The event handler that was created in the "Properties" window' points to the 'Picture1_Click' event handler in the 'Properties' window; 'In the "Properties" window, you can create event handlers by double-clicking the event' points to the 'Click' event in the 'Properties' window.

"Code" tab **The event handler that was created in the "Properties" window** **In the "Properties" window, you can create event handlers by double-clicking the event**

Script language can be set in the "Report[Options...]" menu. This is supposed to be done just after you have created a new report, because when changing the language, the existing script gets deleted.

Report Options

General

Description

Script

Security

Inheritance

Email

Script language:

☒ C#

☐ VisualBasic.NET

Note: if you change the language, entire script will be cleared!

Referenced assemblies:

System.dll

System.Drawing.dll

System.Data.dll

System.Xml.dll

FastReport.Compat.dll

System.Windows.Forms.dll

FastReport.DataVisualization.dll

Add...

OK

Cancel

Genel Bilgi

Diğer rapor oluşturucuların aksine, FastReport'taki betik yalnızca sizin yazdığınız kodu içerir. Betikte şunları yapabilirsiniz:

- Betik ana sınıfına (ReportScript) kendi değişkenlerinizi, metotlarınızı ve özelliklerinizi ekleyebilirsiniz;
- Rapor nesnelerinin olayları için olay işleyicileri oluşturabilirsiniz;
- Gerekirse betiğe yeni sınıflar ekleyebilirsiniz. Sınıf, ana sınıf olan `ReportScript` sınıfından önce ya da sonra eklenebilir.

Yapamayacaklarınız:

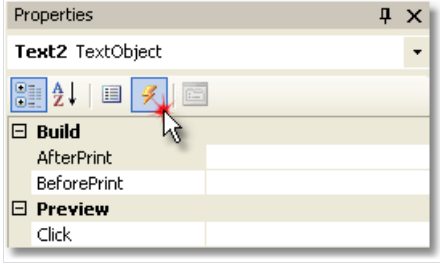
- `ReportScript` ana sınıfını silemez, yeniden adlandıramaz ya da görünürlüğünü (erişim düzeyini) değiştiremezsiniz;
- Bu ana sınıfın bulunduğu ad alanını (namespace) yeniden adlandıramazsınız.

Rapor çalıştırıldığında aşağıdaki işlemler gerçekleşir:

- FastReport, rapordaki nesnelerin adlarıyla aynı ada sahip değişkenleri betiğe ekler. Bu işlem betik derlenmeden önce yapılır ve betikte doğrudan nesne adlarıyla onlara erişebilmenizi sağlar;
- Raporda bulunan ifadeler, betiğe fonksiyonlar olarak eklenir;
- Betik boş değilse derlenir;
- Betiğe dolaylı olarak eklenen değişkenler başlatılır;
- Betikte tanımlanan olay işleyicileri, ilgili rapor nesnelerine bağlanır;
- Rapor çalıştırılır.

Olay İşleyicileri (Event Handlers)

Betik (script), esas olarak nesnelerin olay işleyicilerini oluşturmak için kullanılır. Olay işleyicisi oluşturmak için ilgili nesneyi seçin. "Özellikler" penceresinde olaylar listesini görüntülemek için düğmeye tıklayın:



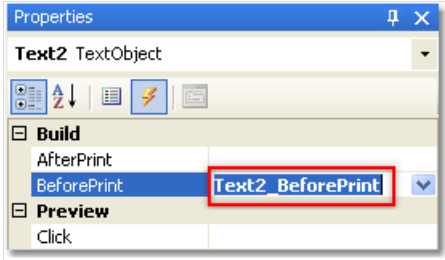
İlgilendiğiniz olayı seçin ve üzerine çift tıklayın. FastReport, rapor koduna boş bir olay işleyicisi ekleyecektir:

```
private void Text2_BeforePrint(object sender, EventArgs e)
{
}
```

"Rapor" (Report) nesnesinin de olayları vardır. Bu nesneyi aşağıdaki şekillerde seçebilirsiniz:

- "Rapor Ağacı" (Report Tree) penceresinden "Report" ögesini seçin;
- "Özellikler" penceresindeki nesne açılır listesinden "Report" ögesini seçin.

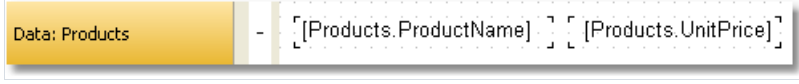
Bir olay işleyicisini silmek için, "Özellikler" penceresinden ilgili olayı seçin, olay adı metnini vurgulayın ve Delete tuşuna basın.



Olaylar

Raporu mümkün olan en esnek şekilde yönetebilmek için, her rapor nesnesi birden fazla olaya sahiptir ve bu olaylara betik içerisinden bir metod (olay işleyicisi) atanabilir. Örneğin, "Veri" bandına bağlı bir olay işleyicisinde, kayıtları filtreleyebilir yani belirli koşullara göre bandı gizleyebilir veya gösterebilirsiniz.

Bir raporun oluşturulma sürecini ve bu süreçte oluşan olayları inceleyelim. Örnek olarak, bir sayfa, bir "Veri" bandı ve bu band üzerinde iki "Metin" nesnesi içeren basit bir raporu ele alalım:



Raporun başında, "Rapor" nesnesine ait **StartReport** olayı tetiklenir. Sayfa oluşturulmadan önce, sayfaya ait **StartPage** olayı çağrılır. Bu olay, rapor şablonundaki her sayfa için bir kez çalışır (hazır rapordaki sayfa sayısı ile karıştırılmamalıdır!). Bizim örneğimizde şablonda yalnızca bir sayfa olduğundan, hazır raporda kaç sayfa olursa olsun bu olay sadece bir kez tetiklenir. Ardından, hazırlanan raporda sayfa gerçekten oluşturulduğu anda **CreatePage** olayı gerçekleşir.

Sonrasında "Veri" bandındaki satırların yazdırılması süreci başlar. Bu süreç şu şekilde ilerler:

1. Bandın **BeforePrint** olayı tetiklenir;
2. Band üzerindeki tüm nesnelerin **BeforePrint** olayları tetiklenir;
3. Tüm nesneler veri ile doldurulur;
4. Band üzerindeki tüm nesnelerin **AfterData** olayları tetiklenir;
5. Bandın **BeforeLayout** olayı tetiklenir;
6. Band üzerindeki nesnelerin (özellikle esnek boyutlu olanların) konumlandırılması yapılır, bandın yüksekliği hesaplanır ve gerekiyorsa band esnetilir;
7. Bandın **AfterLayout** olayı tetiklenir;
8. Eğer band mevcut sayfa boşluğuna sığmazsa yeni bir sayfa oluşturulur;
9. Band ve üzerindeki tüm nesneler, oluşturulan sayfaya yazdırılır;
10. Bandın **AfterPrint** olayı tetiklenir;
11. Band üzerindeki tüm nesnelerin **AfterPrint** olayları tetiklenir.

Veri kaynağında veri olduğu sürece bu yazdırma süreci devam eder. Veriler tükendiğinde, örneğimizdeki raporun oluşturulması tamamlanır ve sayfanın **FinishPage** olayı, ardından "Rapor" nesnesinin **FinishReport** olayı çağrılır.

Bu şekilde, farklı nesnelere ait olayları kullanarak rapor oluşturmanın hemen her aşamasını kontrol etmek mümkündür. Olayları doğru kullanmanın anahtarı, yukarıda belirtilen on bir adımda açıklanan bant yazdırma sürecini tam olarak anlamaktır.

Çoğu işlem yalnızca bandın **BeforePrint** olayı kullanılarak gerçekleştirilebilir – bu olayda nesneye yapılan tüm değişiklikler anında uygulanır. Ancak bu olayda, band esnek boyutluysa hangi sayfaya yazdırılacağını analiz etmek mümkün değildir – çünkü bandın yüksekliği 6. adımda hesaplanır. Bu bilgiye **AfterLayout** (7. adım) veya **AfterPrint** (10. adım) olayları ile ulaşılabilir, ancak sonuncusunda band zaten yazdırılmış olacağından nesne üzerinde yapılacak işlemler sonuç vermez.

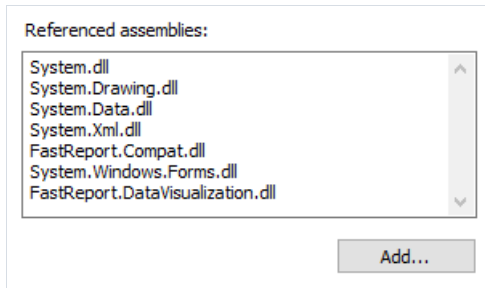
Özetle, her olayın ne zaman tetiklendiğini net bir şekilde bilmeniz ve hedefinize uygun olanları kullanmanız gerekir.

.NET Nesnelerine Erişim

Script içinden aşağıdaki derlemelerde tanımlı olan tüm .NET nesnelerine erişebilirsiniz:

```
System.dll
System.Drawing.dll
System.Windows.Forms.dll
System.Data.dll
System.Xml.dll
```

Bunun yanı sıra, FastReport derlemelerinde tanımlı olan tüm nesneleri de kullanabilirsiniz. Başka bir derlemeye erişmeniz gerekiyorsa, raporun derleme listesine bu derlemenin adını eklemeniz gerekir. Bunu "Rapor | Özellikler..." menüsünden yapabilir, açılan pencerede "Script" ögesini seçerek gerçekleştirebilirsiniz:



Örneğin, script içinde uygulamanızda tanımlı bir fonksiyonu kullanmak istiyorsanız, uygulamanın derlemesine (.exe veya .dll) raporun derleme listesine bir başvuru (referans) eklemelisiniz. Bu işlemten sonra fonksiyona, uygulamanızın ad alanını (namespace) kullanarak erişebilirsiniz. Uygulamada aşağıdaki fonksiyon tanımlıysa:

```
namespace Demo
{
    public static class MyFunctions
    {
        public static string Func1()
        {
            return "Hello!";
        }
    }
}
```

Script içinden bu fonksiyona şu şekilde erişebilirsiniz:

```
string hello = Demo.MyFunctions.Func1();
```

Eğer script'e `using Demo` yönergesini eklerseniz, fonksiyona daha kısa yoldan da erişebilirsiniz:

```
string hello = MyFunctions.Func1();
```

Rapordaki nesnelere erişim

Rapor nesnelere (örneğin, "Text" nesnesine) erişmek için nesnenin adını kullanın. Aşağıdaki örnek, `Text1` nesnesinin yüksekliğini döndürür:

```
float height = Text1.Height;
```

Raporun "yerel" ölçü biriminin ekran pikseli olduğunu unutmayın. Bu, `Left`, `Top`, `Width`, `Height` gibi nesne özelliklerine erişirken dikkate alınmalıdır. Piksel ile santimetre arasında dönüşüm yapmak için `Units` sınıfında tanımlı sabitleri kullanın:

```
float heightInPixels = Text1.Height;  
float heightInCM = heightInPixels / Units.Centimeters;  
Text1.Height = Units.Centimeters * 5; // 5 cm
```

Report ve Engine Nesneleri

Raporun içinde bulunan nesnelere ek olarak, betikte iki değişken tanımlanmıştır: **Report** ve **Engine**.

Report değişkeni, geçerli rapora başvuran bir referans döndürür.

Aşağıdaki tabloda Report nesnesine ait yöntemlerin (metotların) listesi verilmiştir:

Metot	Açıklama
object Calc(string expression)	fadeyi hesaplar ve sonucu döndürür. Bu yöntem ilk kez çağrıldığında, ifade derlenir, bu da bir miktar zaman gerektirir.
object GetColumnValue(string complexName)	Veri kaynağındaki alanın değerini döndürür. Alan adı, "Kaynak.Alan" biçiminde belirtilmelidir. Eğer alanın değeri null ise, bu değer varsayılan bir değere dönüştürülür (0, boş string, false vb.).
object GetColumnValueNullable(string complexName)	Veri kaynağındaki alanın değerini döndürür. Önceki yöntemden farklı olarak, null değerleri varsayılan değerlere dönüştürmez.
Parameter GetParameter(string complexName)	Belirtilen ada sahip rapor parametresini döndürür. İç içe geçmiş bir parametreye erişmek için ad bileşik olabilir: MainParam.NestedParam.
object GetParameterValue(string complexName)	Belirtilen ada sahip rapor parametresinin değerini döndürür.
void SetParameterValue(string complexName, object value)	Belirtilen ada sahip rapor parametresinin değerini ayarlar.
object GetVariableValue(string complexName)	Sistem değişkeninin değerini döndürür, örneğin Date (Tarih).
object GetTotalValue(string name)	"Veri" penceresinde tanımlanmış olan bir toplamın değerini, adıyla birlikte döndürür.
DataSourceBase GetDataSource(string alias)	Rapor içinde tanımlanmış olan veri kaynağını, adıyla birlikte döndürür.

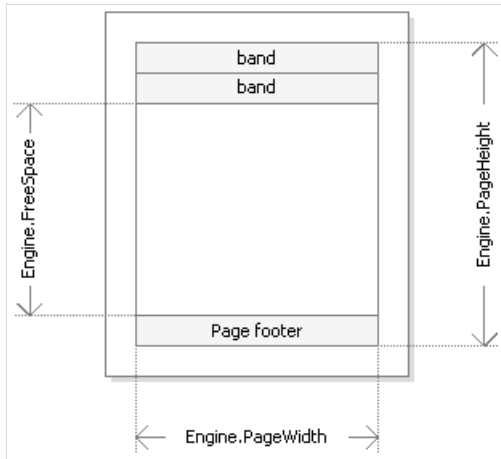
Engine nesnesi, raporun oluşturulmasını yöneten bir motoru temsil eder. Motorun özellikleri ve yöntemleri kullanılarak, sayfada bantların (bandların) yerleştirilme süreci kontrol edilebilir.

Aşağıda **Engine** nesnesine ait kullanılabilecek özellikler verilmiştir:

Özellik	Açıklama
float CurX	X eksenı boyunca geçerli koordinat kayması. Bu özelliğe değer atayarak yazdırılan nesnelerin konumunu kaydırabilirsiniz.

float CurY	Y eksenini boyunca geçerli yazdırma konumu. Bu özelliğe değer atayarak yazdırılan nesnelerin konumunu kaydırabilirsiniz.
int CurColumn	Çok sütunlu bir raporda geçerli sütunun numarası. İlk sütunun numarası 0'dır.
int CurPage	Geçerli yazdırılan sayfanın numarası. Bu değere, Page adlı sistem değişkeni üzerinden erişilebilir.
float PageWidth	Sayfa genişliği eksi sol ve sağ kenar boşluklarının toplamı.
float PageHeight	Sayfa yüksekliği eksi üst ve alt kenar boşluklarının toplamı.
float PageFooterHeight	Sayfa altbilgisinin (ve ona bağlı tüm alt bantların) yüksekliği.
float ColumnFooterHeight	Sütun altbilgisinin (ve ona bağlı tüm alt bantların) yüksekliği.
float FreeSpace	Sayfadaki boş alanın boyutu.
bool FirstPass	Raporun ilk (veya tek) geçişi gerçekleştiriliyorsa true değerini döndürür. Geçiş sayısı, Report.DoublePass özelliğinden alınabilir.
bool FinalPass	Raporun son (veya tek) geçişi gerçekleştiriliyorsa true değerini döndürür.

Aşağıdaki şekilde, bir rapor sayfasının görüntüsü ve sayfanın çeşitli boyutlarını döndüren özelliklerin adları gösterilmektedir.



Engine.PageWidth ve **Engine.PageHeight** özellikleri, sayfanın fiziksel boyutlarından genellikle daha küçük olan yazdırma alanının boyutunu belirler.

Yazdırma alanının boyutunu, rapor sayfasının **LeftMargin**, **TopMargin**, **RightMargin**, **BottomMargin** özellikleriyle ayarlanan kenar boşlukları belirler.

Engine.FreeSpace özelliği, sayfadaki boş alanın yüksekliğini belirler. Sayfada "Sayfa Altbilgisi" (Footer) bandı varsa, **FreeSpace** hesaplanırken bu bandın yüksekliği dikkate alınır.

Her bandın yazdırılmasından sonra sayfadaki boş alan azalır; bu da **FreeSpace** hesabında dikkate alınır.

Hazır rapor sayfaları nasıl oluşturulur? FastReport çekirdeği, sayfada yeterli boş alan olduğu sürece bantları art arda yerleştirir. Sayfada yeterli boş alan kalmadığında, varsa "Sayfa Altbilgisi" bandı yazdırılır ve yeni bir boş sayfa oluşturulur. Daha önce belirtildiği gibi, her bandın çıktısından sonra sayfadaki boş alan (yükseklik olarak) azalır.

Ayrıca yeni bir bandın çıktısı, **X** ve **Y** eksenindeki geçerli konumdan başlar. Bu konum, **Engine.CurX** ve **Engine.CurY** özellikleriyle belirlenir. Her band yazdırıldığında, **CurY** değeri otomatik olarak bandın yüksekliği kadar artırılır. Yeni bir sayfa oluşturulduğunda, **CurY** konumu sıfırlanır (**CurY = 0**). Çok sütunlu raporlarda **CurX** konumu da değiştirilebilir.

Engine.CurX ve **Engine.CurY** özellikleri yalnızca okunabilir değil, aynı zamanda yazılabilir özelliklerdir. Bu da, uygun olaylar içinde bu konumları elle ayarlayarak bandları manuel olarak kaydırabileceğiniz anlamına gelir. Bu özelliklerin kullanımına ilişkin örnekler için "Örnekler" bölümüne bakınız.

Boyut döndüren özelliklerle çalışırken, FastReport'ta kullanılan ölçü biriminin ekran pikseli (screen pixels) olduğunu unutmayın.

Engine nesnesinde aşağıdaki yöntemler (metotlar) tanımlanmıştır:

Metot	Açıklama
void AddOutline(string text)	Rapor yapısına bir öge ekler (bkz. "Etkileşimli Raporlar" bölümü) ve geçerli konumu eklenen ögeye kaydırır.
void OutlineRoot()	Geçerli konumu yapı köküne kaydırır.
void OutlineUp()	Geçerli konumu bir üst seviyeye kaydırır.
void AddBookmark(string name)	Yer imi ekler (bkz. "Etkileşimli Raporlar" bölümü).
int GetBookmarkPage(string name)	Belirtilen ada sahip yer iminin bulunduğu sayfa numarasını döndürür.
void StartNewPage()	Rapora sayfa sonu ekler. Rapor çok sütunluysa, yeni bir sütun eklenir.

AddOutline , **OutlineRoot** ve **OutlineUp** metodları kullanılarak rapor yapısı programatik olarak oluşturulabilir. Genellikle bu işlem, her bandta ve rapor sayfasında bulunan **OutlineExpression** özelliği sayesinde otomatik olarak gerçekleştirilir.

AddOutline metodu, mevcut düğüme bir alt düğüm ekler ve bu yeni düğümü geçerli düğüm olarak ayarlar. Bu ögeyle, raporun geçerli sayfası ve sayfadaki geçerli konum ilişkilendirilir. Böylece, **AddOutline** metodunu art arda birkaç kez çağırırsanız aşağıdaki gibi bir "basamak" yapısı elde edilir:

```
Item1
  Item2
    Item3
```

Geçerli ögeyi kontrol etmek için **OutlineUp** ve **OutlineRoot** metodları kullanılır. **OutlineUp** metodu, işaretçiyi bir üst seviyede bulunan ögeye taşır. Örneğin:

```
Engine.AddOutline("Item1");
Engine.AddOutline("Item2");
Engine.AddOutline("Item3");
Engine.OutlineUp();
Engine.AddOutline("Item4");
```

Bu yapı aşağıdaki gibi olacaktır:

```
Item1
  Item2
    Item3
    Item4
```

`OutlineRoot` metodu, geçerli öğeyi yapının köküne taşır. Örneğin:

```
Engine.AddOutline("Item1");
Engine.AddOutline("Item2");
Engine.AddOutline("Item3");
Engine.OutlineRoot();
Engine.AddOutline("Item4");
```

Bu durumda yapı şu şekilde olur:

```
Item1
  Item2
    Item3
Item4
```

Yer imleriyle çalışmak için `Engine` nesnesinin `AddBookmark` ve `GetBookmarkPage` metodları kullanılır. Yer imleri genellikle rapor nesnelerinde bulunan `Bookmark` özelliği aracılığıyla otomatik olarak eklenir.

`AddBookmark` metodu kullanılarak yer imi programatik olarak eklenebilir. Bu metod, geçerli sayfa ve yazdırma konumuna bir yer imi oluşturur.

`GetBookmarkPage` metodu, belirtilen yerimin bulunduğu sayfa numarasını döndürür. Bu metod genellikle içindekiler tablosu oluşturulurken sayfa numaralarını göstermek amacıyla kullanılır. Bu durumda, raporun iki geçişli olarak yapılandırılması gerekir.

Veri Kaynaklarına Eriřim

FastReport ifadelerinin aksine (bkz. "İfadeler" bölümü), betik içerisinde rapor verilerine köşeli parantezler kullanılarak erişilemez. Bunun yerine, alan değerini döndüren `Report` nesnesinin `GetColumnValue` metodu kullanılır:

```
string productName = (string)Report.GetColumnValue("Products.Name");
```

Görüldüğü gibi, veri kaynağı adı ve alanı nokta (`.`) ile ayrılarak belirtilmelidir. Eğer bir ilişki (relation) üzerinden bir veri kaynağına erişiliyorsa, veri kaynağı adı bileşik olabilir. İlişkiler hakkında daha fazla bilgi "Veriler" bölümünde yer almaktadır. Örneğin, ilişkili bir veri kaynağındaki alana aşağıdaki şekilde erişilebilir:

```
string categoryName = (string)Report.GetColumnValue("Products.Categories.CategoryName");
```

Çalışmayı kolaylaştırmak için "Veriler" penceresini kullanabilirsiniz. Bu pencereden betiğe veri öğelerini sürükleyerek bırakabilirsiniz; bu durumda FastReport, elemana erişim için gerekli kodu otomatik olarak oluşturur.

Bir veri kaynağının kendisine erişmek için `Report` nesnesinin `GetDataSource` metodunu kullanın:

```
DataSourceBase ds = Report.GetDataSource("Products");
```

`DataSourceBase` sınıfının özellikleri ve metotları hakkında bilgi almak için FastReport .NET Class Reference yardım sistemine başvurabilirsiniz. Genellikle bu nesne betik içerisinde şu şekilde kullanılır:

```
// veri kaynağına referans al
DataSourceBase ds = Report.GetDataSource("Products");
// veri kaynağını başlat
ds.Init();
// veri kaynağındaki tüm kayıtları dolaş
while (ds.HasMoreRows)
{
    // geçerli kayıt için alan değerini al
    string productName = (string)Report.GetColumnValue("Products.Name");
    // bu değer ile işlemler yap...
    // ...
    // bir sonraki kayda geç
    ds.Next();
}
```

Sistem deęiřkenlerine eriřim

Sistem deęiřkenine eriřmek iin `Report` nesnesinin `GetVariableValue` metodunu kullanın:

```
DateTime date = (DateTime)Report.GetVariableValue("Date");
```

Sistem deęiřkenlerinin listesini "Veriler" penceresinde gorebilirsiniz. Bu pencereden deęiřkenleri betięe surukleyerek bırakabilirsiniz; bu durumda FastReport, deęiřkene eriřim iin gerekli kodu otomatik olarak oluřturur.

Toplam Değerlere Erişim

Toplam değere erişmek için Report nesnesinin `GetTotalValue` metodunu kullanın:

```
float sales = Report.GetTotalValue("TotalSales");
```

Topamlar listesini "Veri" penceresinde görebilirsiniz. Bu pencereden toplamlara script içine sürükleyip bırakarak FastReport otomatik olarak bu toplama erişim kodunu oluşturur. \

Toplam değerin veri tipi `FastReport.Variant` türündedir. Bu tür, herhangi bir ifadede doğrudan kullanılabilir çünkü `FastReport.Variant` tipi otomatik olarak istenilen türe dönüştürülür. Örneğin:

```
float tax = Report.GetTotalValue("TotalSales") * 0.2f;
```

Toplam değere, hesaplandığı anda erişilebilir. Genellikle bu, toplamın yer aldığı bandın raporda yazdırıldığı andır.

Rapor Parametrelerine Eriřim

Rapor parametresine eriřmek için `Report` nesnesinin `GetParameterValue` metodunu kullanın:

```
int myParam = (int)Report.GetParameterValue("MyParameter");
```

Parametreler iç içe olabilir. Bu durumda, üst parametrenin adını ve ardından nokta kullanarak alt parametrenin adını belirtin:

```
Report.GetParameterValue("ParentParameter.ChildParameter");
```

Parametrelerin belirli bir veri tipi vardır. Bu, parametrenin `DataType` özelliğinde belirtilir. Parametreye eriřirken bunu dikkate almanız gerekir. Parametrelerin listesini "Veriler" penceresinde görebilirsiniz. Bu pencereden parametreleri betiğe sürükleyebilirsiniz; bu durumda FastReport, parametreye eriřim kodunu otomatik olarak oluşturur.

Parametre deęerini deęiřtirmek için `Report` nesnesinin `SetParameterValue` metodunu kullanın:

```
Report.SetParameterValue("MyParameter", 10);
```

Kullanım örnekleri

Örnek 1. Nesnenin Görünümünü Değiştirme

Bu örnekte, nesnede yazdırılan değere bağlı olarak metin renginin nasıl değiştirileceğini göstereceğiz. Aşağıdaki öğeleri kullanacağız:

- `BeforePrint` olayı;
- betik içerisinde veritabanı alanına erişim.

Aşağıdaki gibi basit bir rapor oluşturun:

Data: Products	-	[Products.ProductName]	[Products.UnitPrice]
----------------	---	------------------------	----------------------

Ürünün fiyatını yazdıran nesneye **BeforePrint** olayı için bir olay işleyicisi oluşturun:

```
private void Text2_BeforePrint(object sender, EventArgs e)
{
    if (((Decimal)Report.GetColumnValue("Products.UnitPrice")) > 20)
        Text2.TextColor = Color.Red;
}
```

`Products.UnitPrice` alanının değerini betiğe eklemek için, "**Veriler**" penceresinden bu alanı sürükleyip betiğe bırakın. Bu işlem sonucunda betiğe aşağıdaki satır otomatik olarak eklenecektir:

```
((Decimal)Report.GetColumnValue("Products.UnitPrice"))
```

Raporu çalıştırdığınızda, fiyatı 20'den büyük olan tüm ürünlerin kırmızı renkle vurgulandığını göreceksiniz:

Chai	18,00
Chang	19,00
Aniseed Syrup	10,00
Chef Anton's Cajun Seasoning	22,00
Chef Anton's Gumbo Mix	21,35
Grandma's Boysenberry Spread	25,00

Aynı etki, koşullu biçimlendirme kullanılarak da elde edilebilir (bu konu hakkında daha fazla bilgi için "[Koşullu Biçimlendirme](#)" bölümüne bakınız).

Örnek 2. Veri bandındaki çift satırların vurgulanması

Bu örnekte, "Veri" bandındaki çift satırların dolgu renginin nasıl değiştirileceğini göstereceğiz. Aşağıdakileri kullanacağız:

- Bandın `BeforePrint` olayı,
- Betik içinden sistem değişkeni `Row#` 'a erişim.

Aşağıdaki gibi basit bir rapor oluşturun:

Data: Products	-	[Products.ProductName]	[Products.UnitPrice]
----------------	---	------------------------	----------------------

Band için bir `BeforePrint` olayı işleyicisi oluşturun:

```
private void Data1_BeforePrint(object sender, EventArgs e)
{
    if (((Int32)Report.GetVariableValue("Row#")) % 2 == 0)
        Data1.FillColor = Color.Gainsboro;
}
```

`Row#` sistem değişkeni, yazdırılmakta olan bandın satır numarasını döndürür. Bu değışkene betikte erişmek için, onu "Veriler" penceresinden sürükleyip bırakabilirsiniz. Bu durumda betiğe aşağıdaki satır eklenecektir:

```
((Int32)Report.GetVariableValue("Row#"))
```

Raporu çalıştırdığınızda, bandın çift satırlarının gri renkle vurgulandığını göreceksiniz.

Chai	18,00
Chang	19,00
Aniseed Syrup	10,00
Chef Anton's Cajun Seasoning	22,00
Chef Anton's Gumbo Mix	21,35

Benzer bir etkiyi bandın `EvenStyle` özelliğini kullanarak da elde edebilirsiniz. Bu konuda daha fazla bilgi için ["Satırların dönüşümlü vurgulanması"](#) bölümüne bakınız.

Örnek 3. Verilerin Filtrelenmesi

Bu örnekte, "Veri" bandındaki satırların belirli bir koşula göre nasıl gizleneceğini göstereceğiz. Aşağıdakileri kullanacağız:

- Bandın **BeforePrint** olayı,
- Betik içinde veritabanı alanına erişim.

Aşağıdaki gibi basit bir rapor oluşturun:

Data: Products	-	[Products.ProductName]	[Products.UnitPrice]
----------------	---	------------------------	----------------------

Band için **BeforePrint** olayı işleyicisi oluşturun:

```
private void Data1_BeforePrint(object sender, EventArgs e)
{
    if (((Decimal)Report.GetColumnValue("Products.UnitPrice")) > 20)
        Data1.Visible = false;
}
```

Bu durumda, ürünün fiyatı **20'den büyük** olan satırlar gizlenecektir:

Chai	18,00
Chang	19,00
Aniseed Syrup	10,00
Konbu	6,00
Genen Shouyu	15,50
Pavlova	17,45

Aynı etki, "Veri" bandı düzenleyicisinde yapılandırılabilen veri filtreleme koşulu kullanılarak da elde edilebilir.

Örnek 4. Toplamların Hesaplanması

Bu örnekte, toplam bir değerin programatik olarak nasıl hesaplanacağını göstereceğiz. Şu bileşenleri kullanacağız:

- Band'ın `BeforePrint` olayı,
- Betik (script) içinden veritabanı alanına erişim,
- Rapor içinde yazdırılacak bir yerel değişken.

Aşağıdaki gibi bir rapor oluşturun:

Data: Products	-	[Products.ProductName]	[Products.UnitPrice]
Report Summary	-		Total: [sum]

Betikte `summa` adında bir değişken tanımlayın ve band için `BeforePrint` olay işleyicisi oluşturun:

```
public class ReportScript
{
    private decimal summa;

    private void Data1_BeforePrint(object sender, EventArgs e)
    {
        summa += (Decimal)Report.GetColumnValue("Products.UnitPrice");
    }
}
```

`Products.UnitPrice` tablosunun alanı, "Veriler" penceresinden betiğe sürüklenerek eklenebilir.

Röd Kaviar	15,00
Longlife Tofu	10,00
Rhönbräu Klosterbier	7,75
Lakkalikööri	18,00
Original Frankfurter grüne Soße	13,00
Total:	2222,71

Raporu çalıştırırsanız aşağıdakine benzer bir sonuç görürsünüz:Aynı etki, **toplamlar (taller)** kullanılarak da elde edilebilir. Daha fazla bilgi için "[Veriler](#)" bölümüne bakınız.

Örnek 5. Yazdırma Konumunun Kaydırılması

Bu örnekte, Engine nesnesinin özelliklerini kullanarak bantları elle nasıl kaydırabileceğimizi göstereceğiz. Şunları kullanacağız:

- Bandın `BeforePrint` olayı;
- Engine nesnesi.

Aşağıdaki gibi basit bir rapor oluşturun:

Data: Products	-	[Products.ProductName]	[Products.UnitPrice]
----------------	---	------------------------	----------------------

Band için `BeforePrint` olay işleyicisini oluşturun:

```
private void Data1_BeforePrint(object sender, EventArgs e)
{
    Engine.CurX = ((Int32)Report.GetVariableValue("Row#")) * 10;
}
```

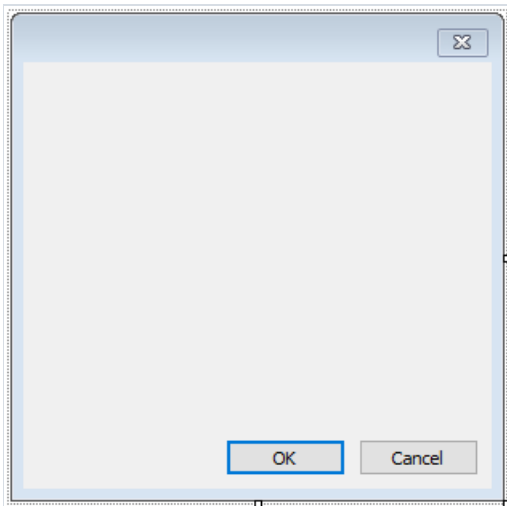
Raporu çalıştırdığınızda, aşağıdakini göreceksiniz:

Chai	18,00
Chang	19,00
Aniseed Syrup	10,00
Chef Anton's Cajun Seasoning	22,00
Chef Anton's Gumbo Mix	21,35

Dialogue forms

Apart from an ordinary page, a report can contain one or several dialogue forms. Such dialogue form will be shown when the report is started. In the dialogue, you can enter any type of information, needed for creating a report. Also, a dialogue may be used to filter data, which is shown in the report.

For adding a dialogue into a report, press the  button on the designer's toolbar. A new dialogue looks as follows:



A report, having one or several dialogues, works like this:

- when started, the report shows the first dialogue;
- if the dialogue is closed with the help of the "OK" button, then the next dialogue is shown; otherwise, the report stops working;
- after all the dialogues have been shown, the report is built.

Kontrol Öğeleri

Diyalog formunda aşağıdaki kontrol öğeleri kullanılabilir:

İkon	Ad	Açıklama
	ButtonControl	Bir butonu temsil eder.
	CheckBoxControl	İki durumu olan bir onay kutusudur (açık/kapalı).
	CheckedListBoxControl	Satırların onay kutuları içeren bir listesini temsil eder.
	ComboBoxControl	Bir açılır listeyi temsil eder.
	DataGridViewControl	Tablo şeklinde verileri görüntüler.
	DataSelectorControl	İki listeyi gösterir ve satırları bir listeden diğerine taşımaya olanak tanır.
	DateTimePickerControl	Tarih veya zaman girişi yapılmasına olanak tanır.
	GroupBoxControl	Diğer kontrol öğelerini gruplandırmaya olanak tanır.
	LabelControl	Açıklayıcı bir etiket temsil eder.
	ListBoxControl	Satır listesini temsil eder.
	ListViewControl	İkonlu satır listesini temsil eder.
	MaskedTextBoxControl	Maske girişi kullanılarak değerin girilmesine olanak tanır.
	MonthCalendarControl	Takvimi temsil eder.
	NumericUpDownControl	Sayısal değer girişi yapmaya olanak tanır.
	PanelControl	Diğer kontrol öğelerini gruplandırmaya olanak tanır.
	PictureBoxControl	Resmi görüntüler.
	RadioButtonControl	İki durumu olan (açık/kapalı) bağımlı anahtar temsil eder.
	RichTextBoxControl	RTF formatındaki biçimlendirilmiş metni görüntüler.

	TextBoxControl	Tek satırlı veya çok satırlı metin düzenleme alanını temsil eder.
	TreeViewControl	Verileri hiyerarşik şekilde görüntüler.

DataSelectorControl hariç, tüm kontrol öğeleri .NET Framework'de bulunan standart kontrol öğelerinin tam karşılıklarıdır. İsim çakışmalarını önlemek amacıyla, kontrol öğelerinin isimlerinin başında Control öneki bulunur. Örneğin, ButtonControl kontrol öğesi, .NET Framework'deki Button öğesine karşılık gelir.

Koddaki Bir Öğeye Erişim

Bir öğeye, öğenin adı kullanılarak erişilebilir:

```
TextBoxControl1.Text = "my text";
```

Aslında, FastReport kontrol öğesi, .NET Framework'ün standart kontrol öğesi üzerine bir sarmalayıcıdır. Bu, standart öğenin birçok fakat tüm özelliklerini uygulamaz. Eğer uygulanan özellikler yeterli gelmiyorsa, standart öğeye şu yollarla erişebilirsiniz:

- **Control Özelliğini Kullanarak:** Bu özellik, `System.Windows.Forms.Control` tipinde bir nesne döndürür:

```
(TextBox1.Control as TextBox).ShortcutsEnabled = false;
```

- **"Control" Ekini Kaldırarak Oluşan Özelliği Kullanarak:** Örneğin, `TextBoxControl` öğesi için, `TextBox` özelliği `System.Windows.Forms.TextBox` tipinde standart öğeyi döndürür:

```
TextBox1.TextBox.ShortcutsEnabled = false;
```

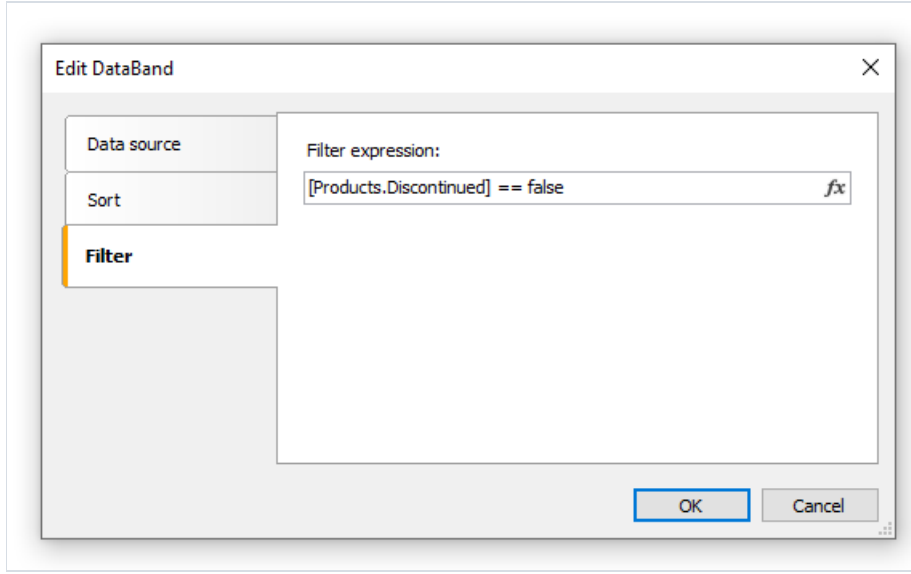
Kontrol öğelerinin özellikleri ve metodları hakkında detaylı bilgi için MSDN referansına bakabilirsiniz.

Veri Filtreleme

Diyalogların kullanım yollarından biri, raporda görüntülenen verilerin filtrelenmesidir. Örneğin, tüm çalışanların listesini yazdıran bir rapor mevcuttur. Diyalog kullanılarak, kullanıcıya bunlardan bir veya birkaçını seçme imkanı sunulabilir; ardından rapor oluşturulurken seçilen çalışanların gösterilmesi için veriler filtrelendir.

Filtreleme kullanımı için, orijinal raporun tüm verileri yazdırması gerekir. "Filtreleme" adı, rapor oluşturma sürecinde gereksiz verilerin atılacağını varsayar.

Veri filtrelemeyi düzenlemenin en basit yolu, "Data" bandının **Filter** özelliğini kullanmaktır. Band editöründe, örneğin, filtreleme koşulu belirtilebilir:



Diyalog kullanılarak, kullanıcıdan bir değer istenip bu değer filtre ifadesinde kullanılabilir. "Basit Filtre" örneğine, "Örnekler" bölümünde bakınız.

Bu yöntem, tek bir basit değer istenmesi gerektiğinde kullanılabilir. Ancak, kullanıcıya bir değer listesi gösterip onlardan bir veya birkaçını istemek gerektiğinde bu uygulama oldukça karmaşık hale gelir. Görünüşte basit bir görev – kullanıcıya ListBoxControl öğesinde çalışan listesini gösterip, bir veya birkaçını seçmelerini sağlamak – aslında şu işlemleri gerektiren bir script kullanmayı zorunlu kılar:

- Veri kaynağını adı ile alır;
- Verileri başlatır;
- Kaynaktaki mevcut değerlerle ListBoxControl listesini doldurur;
- Çalışanlar seçildikten sonra filtreleme koşulunu oluşturur.

FastReport, bu işlemlerin tamamını herhangi bir kod yazmadan otomatik olarak gerçekleştirmeyi sağlar. Bunun için, şu anda inceleyeceğimiz otomatik filtreleme yöntemi kullanılır.

Otomatik filtreleme – nasıl çalışır

Kontrol ögesi, DataColumn özelliği aracılığıyla veri alanına bağlanır. Eğer öge, bir değer listesini görüntüleyebiliyorsa (örneğin, ListBoxControl), belirtilen veri alanındaki tüm değerlerle otomatik olarak doldurulur; bu, diyalog başlatıldığında gerçekleşir. Daha sonra kullanıcı, diyalog üzerinde çalışır, öğeden bir veya daha fazla değer seçer ve diyalogu kapatır. Bu noktada, DataColumn özelliğinde belirtilen veri kaynağına, seçilen değerleri içeren bir filtre uygulanır.

Bu yöntemin avantajı, herhangi bir raporda hiç kod satırı yazmadan kullanılabilmesidir.

Otomatik filtreleme, aşağıdaki kontrol öğeleri tarafından desteklenir:

Icon	Name
	CheckBoxControl
	CheckedListBoxControl
	ComboBoxControl
	DataSelectorControl
	DateTimePickerControl
	ListBoxControl
	MaskedTextBoxControl
	MonthCalendarControl
	NumericUpDownControl
	RadioButtonControl
	TextBoxControl

Filtre İşlemi

Varsayılan olarak, FastReport kontrol ögesinin değerine eşit olan değerleri içeren veri satırlarını filtreler. Bu, kontrol ögesinin **FilterOperation** özelliği kullanılarak ayarlanır. Bu özellikte, veri filtrelemesi sırasında kullanılacak işlem belirtilir. Aşağıdaki işlemleri kullanabilirsiniz:

İşlem	Eşdeğer	Eylem
Equal	=	Kontrol ögesinin değerine eşit olan değerleri seç.
NotEqual	< >	Kontrol ögesinin değerine eşit olmayan değerleri seç.
LessThan	<	Kontrol ögesinin değerinden küçük olan değerleri seç.
LessThanOrEqual	< =	Kontrol ögesinin değerinden küçük veya eşit olan değerleri seç.
GreaterThan	>	Kontrol ögesinin değerinden büyük olan değerleri seç.
GreaterThanOrEqual	> =	Kontrol ögesinin değerinden büyük veya eşit olan değerleri seç.

Örneğin, kontrol ögesinin **FilterOperation** özelliği **LessThanOrEqual** olarak ayarlanmışsa ve kontrol ögesine **5** değeri girilmişse, filtrelenen alandaki değeri 5'ten küçük veya eşit olan tüm veri satırları seçilir.

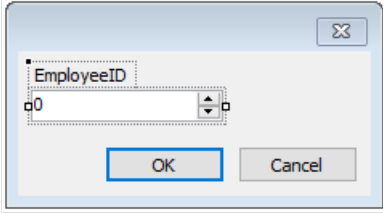
Dize tipindeki veriler için, bir dizeden diğerine girişimi kontrol eden ek işlemler kullanılabilir:

İşlem	Eylem
Contains	Kontrol ögesindeki değeri içeren satırları seç.
NotContains	Kontrol ögesindeki değeri içermeyen satırları seç.
StartsWith	Kontrol ögesindeki değerden başlayan satırları seç.
NotStartsWith	Kontrol ögesindeki değerden başlamayan satırları seç.
EndsWith	Kontrol ögesindeki değer ile biten satırları seç.
NotEndsWith	Kontrol ögesindeki değer ile bitmeyen satırları seç.

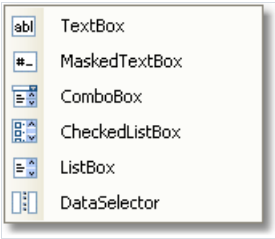
Örneğin, kontrol ögesinin **FilterOperation** özelliği **StartsWith** olarak ayarlanmışsa ve kontrol ögesine **A** değeri girilmişse, filtrelenen alandaki değeri **A** ile başlayan tüm veri satırları seçilir.

Rapor'a filtre ekleme

FastReport'un diyalog tasarımcısı, veri filtrelemeyi destekleyen kontrolleri eklemek için oldukça kullanışlı özelliklere sahiptir. Bunun için, "Data" penceresinden veri sütununu sürükleyip diyalog formunun üzerine bırakın. Bu işlem sırasında, FastReport bir başlık (LabelControl kontrolü) ve veri filtreleme için kullanılacak asıl kontrolü otomatik olarak oluşturur:



Kontrol tipi, veri sütununun türüne bağlıdır. Eğer sütun string (metin) türündeyseniz, ekledikten sonra hangi kontrol tipini kullanmak istediğiniz sorulacaktır.



Aynı veri sütununa bağlı iki benzer kontrol eklediyseniz, FastReport FilterOperation (Filtre İşlemi) özelliği yardımıyla veri aralığını otomatik olarak yapılandırır. İlk kontrol için FilterOperation = GreaterThanOrEqual (BüyükEşit), ikinci kontrol için ise LessThanOrEqual (KüçükEşit) olarak ayarlanır. Bu işlem, eklediğiniz sütun metin (string) türünde değilse gerçekleştirilir.

Bu nedenle, herhangi bir rapora veri filtreleme eklemek için aşağıdaki adımları uygulamanız gerekir:

- Raporun içine yeni bir diyalog ekleyin.
- Raporu filtrelemek istediğiniz veri sütununu diyalogun üzerine bırakın.

Değer aralığına göre filtreleme

Bu filtreleme yöntemi, örneğin maliyet gibi nicel özelliklere sahip değerlerle çalışırken kullanışlıdır. Böylece, belirli bir değer altında veya üstünde olan ürünleri filtreleyebilirsiniz. Giriş elemanına girilen değerin nasıl yorumlanacağını belirtmek için, daha önce bahsedilen FilterOperation özelliği kullanılır.

Aynı veri alanına bağlı iki kontrol elemanı kullanarak ve her biri için farklı FilterOperation ayarları belirleyerek, veri aralığının başlangıç ve bitişini tanımlayabilirsiniz. İlk eleman için FilterOperation özelliğini GreaterThanOrEqualTo olarak ayarlamanız, ikinci eleman için ise LessThanOrEqualTo olarak ayarlamanız gerekmektedir.

Bağlantılı Alana Göre Filtreleme

Bildiğimiz gibi, iki veri kaynağı arasında bağlantı kurulabilir. Bu konu hakkında detaylı bilgi için "[Veriler](#)" bölümüne bakınız. Bağlantı sayesinde, diğer bağlantılı kaynaktan gelen bir alan kullanılarak kaynak veriler filtrelenebilir.

Diyelim ki, diyaloga bir ListBoxControl kontrolü eklediniz ve `DataColumn` özelliğine şu veri alanını belirttiniz:

```
Products.Categories.CategoryName
```

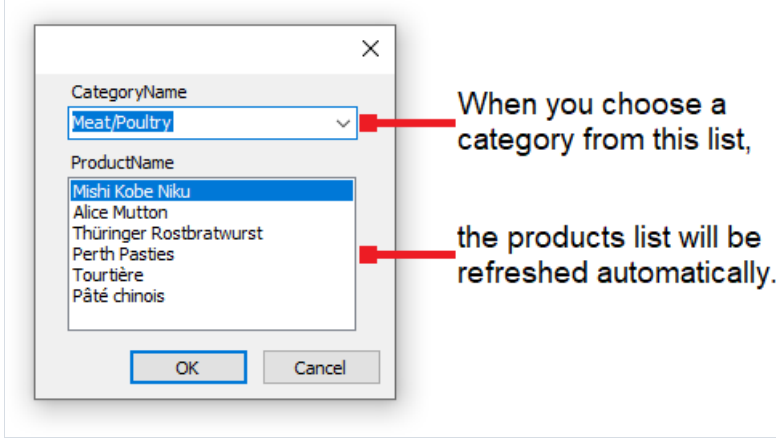
Filtreleme nasıl çalışacaktır?

- Kontrol doldurulurken, `Categories` veri kaynağının `CategoryName` alanı kullanılacaktır;
- Veri filtreleme sırasında, `Products` veri kaynağındaki ve `[Products.Categories.CategoryName]` alanı kullanıcının seçtiği değerlerden birini içeren satırlar filtrelenecektir.

```
the [Products.Categories.CategoryName] contains one of the values, selected by the user
```

Kademeli Liste ile Filtreleme

Kademeli liste, bir listedeki seçeneklerin, kullanıcının başka bir listede yaptığı seçime göre değiştiği bir listedir. Örneğin, bir formda iki listeniz var – biri kategoriler, diğeri ürünler için. İlk listeden bir kategori seçtiğinizde, ikinci listede seçilen kategoriye ait ürünler görüntülenir.



Kademeli bir liste oluşturmak için, aralarında ana-detay ilişkisi bulunan iki veri kaynağı kullanmanız gerekir (veri kaynakları ve ilişkiler hakkında daha fazla bilgi için "Data" bölümüne bakın). Ana listeyi, ana veri kaynağındaki bir sütuna bağlayın; detay listesini ise detay veri kaynağındaki bir sütuna bağlayın. Ayrıca, ana listenin DetailControl (DetayKontrolü) özelliğini detay listesine ayarlayın.

Kod ile filtrelemeyi yönetme

Otomatik filtreleme yetenekleri çoğu durum için yeterli olsa da, filtrelemeyi manuel olarak yönetme seçeneğiniz de bulunmaktadır. Bu amaçla aşağıdaki özellikler ve metotlar kullanılır.

AutoFill özelliği, kontrolün doldurulmasını yönetir. Bu özellik, örneğin ListBoxControl gibi veri listesi gösterebilen kontrollerde kullanılır. Diyalog gösterilmeden önce FastReport, bu tür kontrolleri verilerle doldurur. Varsayılan olarak bu özellik **true** değerindedir. Eğer kapatırsanız, kontrol otomatik olarak doldurulmaz ve kendiniz **FillData** metodunu çağırarak doldurmanız gerekir:

```
ListBox1.FillData();
```

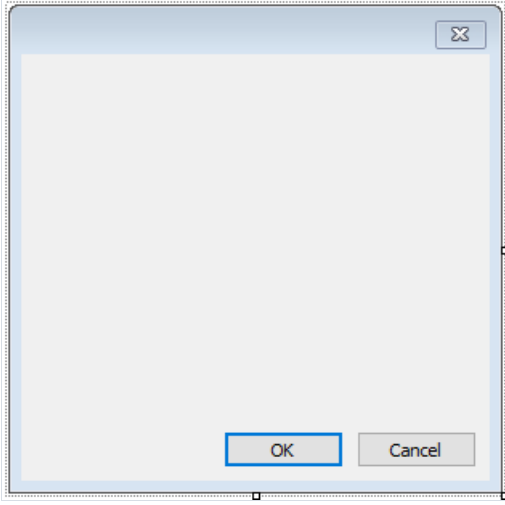
AutoFilter özelliği, verilerin filtrelenmesini yönetir. Bu özellik tüm kontrol elemanları tarafından kullanılır. Diyalog "OK" butonuna basılarak kapatıldıktan sonra FastReport verileri otomatik olarak filtreler. Varsayılan olarak bu özellik **true** değerindedir. Eğer kapatırsanız, filtreleme otomatik olarak yapılmaz ve kendiniz **FilterData** metodunu çağırarak filtrelemeniz gerekir:

```
ListBox1.FilterData();
```

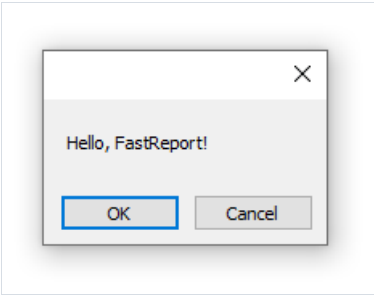
Kullanım örnekleri

Örnek 1. Diyalog "Hello, FastReport!"

Bu örnekte yapacağımız tek şey, bir karşılama mesajı gösterecek bir diyalog oluşturmaktır. Yeni bir rapor oluşturun ve rapora bir diyalog ekleyin. Bunu yapmak için araç çubuğundaki düğmeye tıklayın:



Diyaloga bir LabelControl kontrolü yerleştirin ve Özellikler penceresinde yer alan **Text** özelliğini ayarlayın:

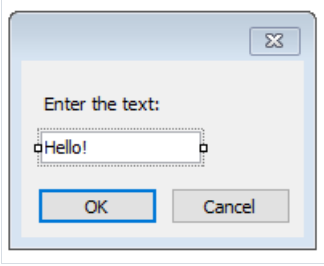


Raporu çalıştırdığınızda diyalog penceresini göreceksiniz. Diyalogu "OK" düğmesine basarak kapatırsanız rapor oluşturulur; "İptal" düğmesi veya başlıktaki çarpı ile kapatırsanız rapor çalışmayı durdurur ve tasarım moduna geri dönersiniz.

Örnek 2. Kullanıcıdan Bir Metin İsteme

Bu örnekte, kullanıcıdan rastgele bir metin girmesini isteyen bir diyalog oluşturacağız ve ardından girilen değeri raporda yazdıracağız.

Yeni bir rapor oluşturun ve içine bir diyalog ekleyin. Diyalog üzerine LabelControl ve TextBoxControl kontrollerini yerleştirin.



Bu durumda, girdiğimiz değer TextBoxControl'ün `Text` özelliğinde yer alır. Bu değeri raporda yazdırmak için, "Rapor Başlığı" bandına yeni bir "Metin" nesnesi ekleyin ve içine aşağıdakini yazın:

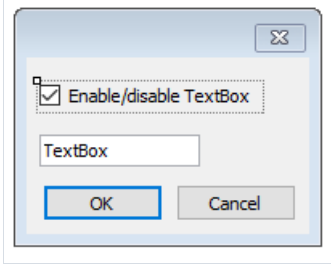
You have entered: [TextBox1.Text]

Burada TextBox1, TextBoxControl kontrolünün adıdır.

Örnek 3. Diyalog Elemanlarının Yönetimi

Script ve kontrollerin olaylarını kullanarak, kontrolleri tıpkı Visual Studio'da olduğu gibi yönetebilirsiniz. Aşağıda bir örnekle, CheckBoxControl'ün TextBoxControl'ün erişilebilirliğini nasıl yönettiğini göstereceğiz.

Yeni bir rapor oluşturun ve içine bir diyalog ekleyin. Diyalog üzerine, aşağıdaki şekilde gösterildiği gibi CheckBoxControl ve TextBoxControl kontrollerini yerleştirin:



Şimdi CheckBoxControl'ü seçin, "Özellikler" penceresini açın ve düğmeye tıklayın. Onay kutusunun durumu değiştiğinde tetiklenen **CheckedChanged** olayına çift tıklayın. FastReport bu olay için boş bir olay işleyicisi oluşturacaktır. Aşağıdaki kodu bu alana yazın:

```
private void CheckBox1_CheckedChanged(object sender, EventArgs e)
{
    TextBox1.Enabled = CheckBox1.Checked;
}
```

Raporu çalıştırdığımızda, onay kutusu yardımıyla TextBoxControl'ü etkinleştirebilir veya devre dışı bırakabiliriz.

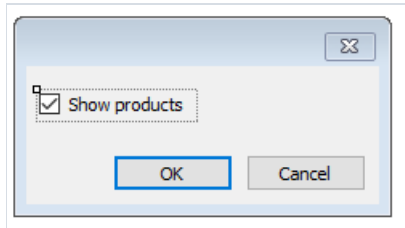
Örnek 4. Rapor Nesnelerinin Yönetimi

Her kategori içindeki ürünlerin listesini yazdıran bir rapor örneğini inceleyelim:

Beverages	
Soft drinks, coffees, teas, beers, and ales	
Product name	Unit price
Chai	18,00
Chang	19,00
Chartreuse verte	18,00
Côte de Blaye	263,50
Guaraná Fantástica	4,50
Ispoh Coffee	46,00
Lakkaiköörni	18,00
Laughing Lumberjack Lager	14,00
Outback Lager	15,00
Rhinbrau Klosterbier	7,75
Sasquatch Ale	14,00
Stout	18,00

Condiments	
Sweet and savory sauces, relishes, spreads, and seasonings	
Product name	Unit price
Aniseed Syrup	10,00

Yalnızca kategorileri yazdırıp ürünlerin yazdırılmasını nasıl durdurabileceğimizi diyalog yardımıyla göstereceğiz. Bunun için rapora bir diyalog ekleyin:



"OK" düğmesine çift tıklayın. FastReport, **Click** olayı için boş bir olay işleyicisi oluşturacaktır. Bu alana aşağıdaki kodu yazın:

```
private void btnOk_Click(object sender, EventArgs e)
{
    Data2.Visible = CheckBox1.Checked;
}
```

Ürün listesini yazdıran bandın görünürlüğünü kontrol edeceğiz. Örneğimizde bu, "Data2" adını taşıyan banddır. Rapor çalıştırıldığında ve onay kutusu işaretli değilse, aşağıdaki sonucu elde ederiz:



Beverages

Soft drinks, coffees, teas, beers, and ales



Condiments

Sweet and savory sauces, relishes, spreads, and seasonings



Confections

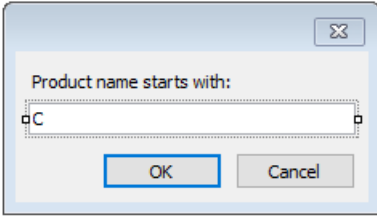
Desserts, candies, and sweet breads

Örnek 5. Basit Filtre

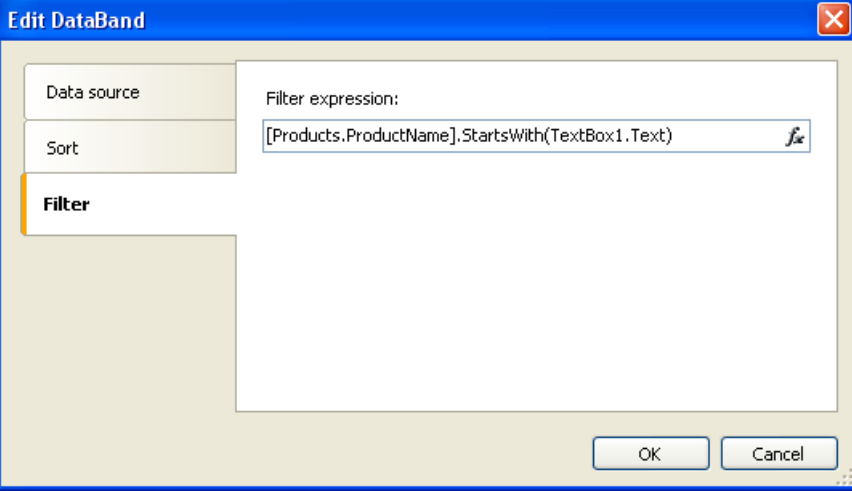
Aşağıdaki rapora bakalım, bu rapor bir ürün listesi yazdırır:

Page Header	Product Name
Data: Products	[Products.ProductName]

Bu örnekte, ürünlerin listesini ürün adının ilk harfine göre nasıl filtreleyeceğimizi göstereceğiz. Bu esnada, otomatik veri filtreleme özelliklerini kullanmayacağız. Bunun için rapora yeni bir diyalog ekleyin ve üzerine iki kontrol yerleştirin - LabelControl ve TextBoxControl.



Şimdi "Veri" bandı düzenleyicisini açın ve aşağıdaki filtre ifadesini belirtin:



Raporu çalıştırın ve her şeyin düzgün çalıştığından emin olun:

×

Product name starts with:

C

OK

Cancel

Product Name

Chai
Chang
Chef Anton's Cajun
Chef Anton's Gumbo Mix
Camarvon Tigers
Côte de Blaye
Chartreuse verte
Chocolade
Camembert Pierrot

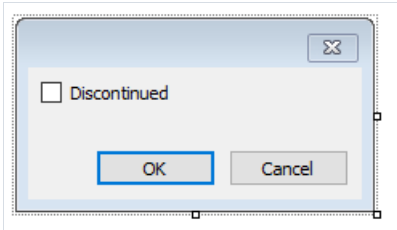
Örnek 6. Otomatik Filtreleme

Bu örnekte, Products tablosundaki ürün listesini yazdıran rapora nasıl filtre ekleneceğini göstereceğiz. Filtreleme, Products.Discontinued alanına göre yapılacaktır.

Raporun kendisi aşağıdaki gibidir:

Header	Product name	Discontinued
Data: Products	[Products.ProductName]	✓

Araç çubuğundaki butona tıklayarak rapora yeni bir diyalog ekleyin ve "Veriler" penceresinden Products.Discontinued alanını üzerine sürükleyin:



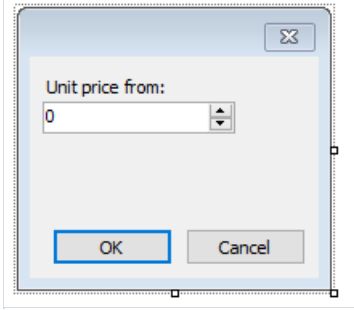
Yapmamız gereken tek şey bu – sadece 2 fare tıklaması yaptık. FastReport, kontrol öğesini otomatik olarak veri alanına bağladı.

Raporu çalıştırın ve Discontinued onay kutusunu işaretleyin. Daha sonra "Tamam" butonuna basın, ve sadece Discontinued işareti olan ürünleri içeren raporu göreceksiniz:

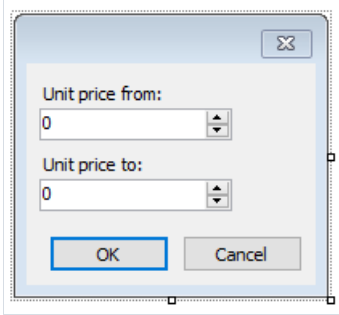
Product name	Discontinued
Alice Mutton	✓
Chef Anton's Gumbo Mix	✓
Guaraná Fantástica	✓
Mishi Kobe Niku	✓
Perth Pasties	✓
Rössle Sauerkraut	✓
Singaporean Hokkien Fried Mee	✓
Thüringer Rostbratwurst	✓

Örnek 7. Aralıkta Otomatik Filtreleme

Önceki örnekteki raporda, belirlenen aralıktaki fiyatlara sahip ürünlerin nasıl yazdırılacağını gösterelim. Bunun için rapora bir iletişim penceresi ekleyip üzerine Products.UnitPrice alanını sürükleyip bırakacağız. Ardından ögenin metnini düzenleyeceğiz:



Şimdi benzer şekilde başka bir Products.UnitPrice alanı ekleyin ve başlığını düzenleyin:



Yapmamız gereken tek şey bu. Geri kalan tüm işlemleri FastReport gerçekleştirdi: ögeleri veri alanına bağladı ve FilterOperation özelliklerini ayarladı. İlk ögenin FilterOperation değeri GreaterThanOrEqual, ikincisinin ise LessThanOrEqual olarak ayarlandı.

Raporu çalıştırın ve örneğin 20 ile 30 arasındaki değerleri girin. "Tamam" düğmesine bastığınızda rapor oluşturulacak ve belirlediğimiz aralıktaki fiyatlara sahip ürünleri içerecektir.

Product name	UnitPrice
Chef Anton's Cajun Seasoning	22,00
Chef Anton's Gumbo Mix	21,35
Fløtemysost	21,50
Grandma's Boysenberry Spread	25,00
Gravad lax	26,00
Gustaf's Knäckebröd	21,00
Louisiana Fiery Hot Pepper Sauce	21,05
Maxilaku	20,00
Nord-Ost Matjeshering	25,89
Pâté chinois	24,00
Queso Cabrales	21,00
Sirop d'érable	28,50
Tofu	23,25
Uncle Bob's Organic Dried Pears	30,00

Örnek 8. İlişkili veri kaynağının alanına göre filtreleme

Bu örnekte, veri kaynağını filtrelemek için ilişkili kaynakta bulunan bir alanı kullanacağız.

Basit liste türünde, ürün listesini yazdıran bir raporu ele alalım. Her ürünün yanında, o ürünün ait olduğu kategori adı yazdırılır. Bu, bağlantı aracılığıyla yapılır:

```
[Products.Categories.CategoryName]
```

Rapor aşağıdaki gibi görünmektedir:

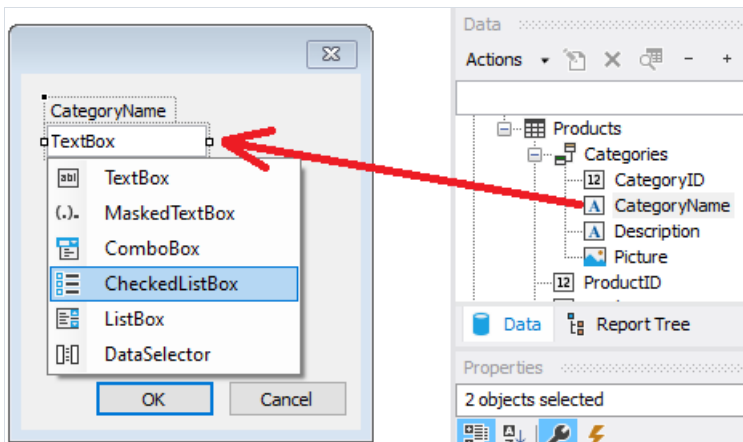
Report Title	PRODUCT CATALOG		
Header	Product name	Category name	Unit price
Data: Products	[Products.ProductName]	[Products.Categories.CategoryName]	[Products.UnitPrice]

Rapor çalıştırıldığında aşağıdakileri göreceğiz:

PRODUCT CATALOG		
Product name	Category name	Unit price
Alice Mutton	Meat/Poultry	39,00
Aniseed Syrup	Condiments	10,00
Boston Crab Meat	Seafood	18,40
Camembert Pierrot	Dairy Products	34,00
Carnarvon Tigers	Seafood	62,50
Chai	Beverages	18,00
Chang	Beverages	19,00
Chartreuse verte	Beverages	18,00
Chef Anton's Cajun Seasoning	Condiments	22,00
Chef Anton's Gumbo Mix	Condiments	21,35
Chocolade	Confections	12,75

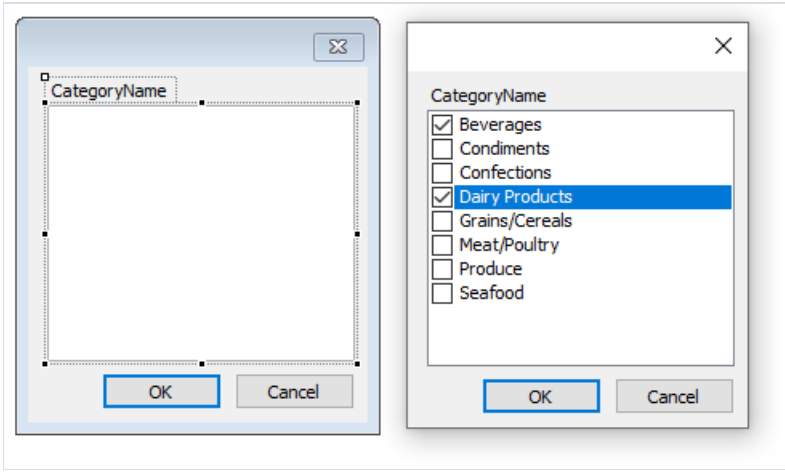
Şimdi kategori adına göre filtreleme ekleyelim. Bunun için yeni bir diyalog ekleyin ve üzerine

`Products.Categories.CategoryName` alanını sürükleyin:



Kontrol öğesi oluşturulurken tipini seçmeniz istenecek. `CheckedListBoxControl` öğesini seçin. Raporu çalıştırırsanız,

aşağıdaki diyalogu göreceksiniz:



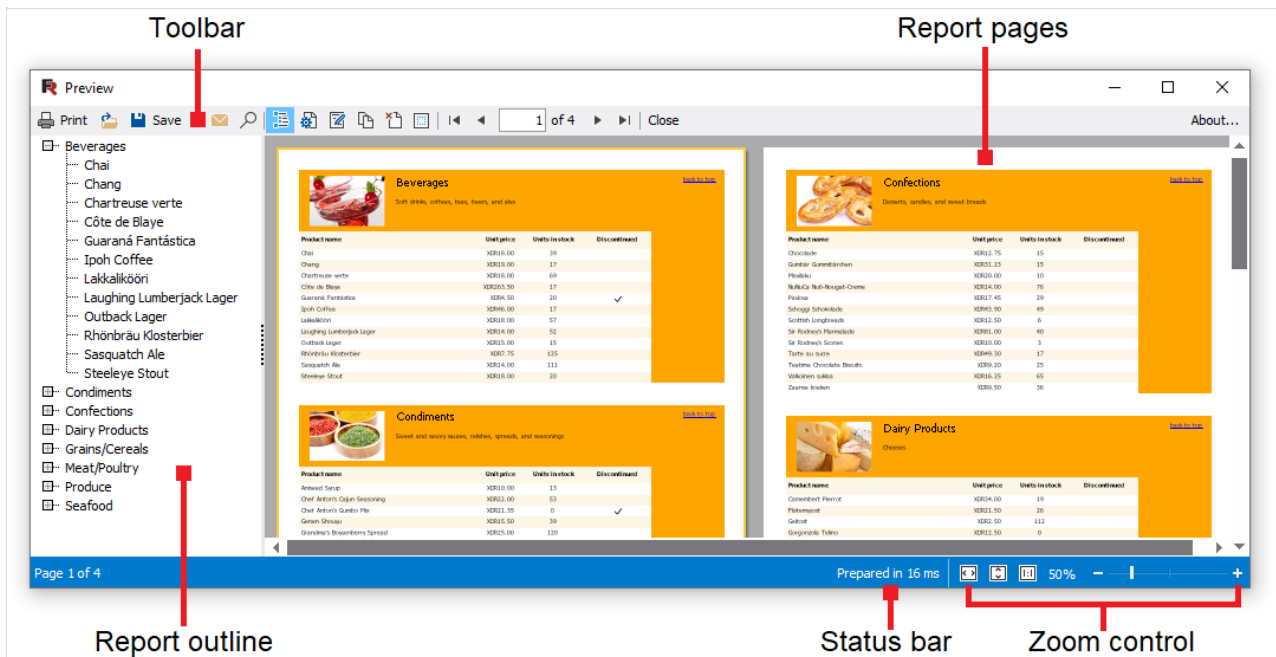
Birkaç öge seçin ve "Tamam" düğmesine basın. Sonrasında veriler filtrelenecek ve aşağıdaki raporu göreceksiniz:

PRODUCT CATALOG		
Product name	Category name	Unit price
Camembert Pierrot	Dairy Products	34,00
Chai	Beverages	18,00
Chang	Beverages	19,00
Chartreuse verte	Beverages	18,00
Côte de Blaye	Beverages	263,50
Fløtemysost	Dairy Products	21,50
Geitost	Dairy Products	2,50
Gorgonzola Telino	Dairy Products	12,50
Guaraná Fantástica	Beverages	4,50
Gudbrandsdalsost	Dairy Products	36,00
Ipoh Coffee	Beverages	46,00
Istiklal Çikolata	Beverages	18,00

Görüldüğü gibi, yalnızca seçili kategorilere ait ürünler kalmıştır.

Preview, print, export

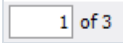
A built report can be shown on the screen, printed on the printer or exported into one of the supported formats. All these can be done in the preview window:



On the toolbar, you can find the following buttons:



Button	Description
	Print the report.
	Open the prepared report file in FPX format.
	Save the report in one of the supported formats.
	Send the report by email.
	Text search in the report.
	Shows or hides report outline.
	Page settings.
	Edit current report page.
	Watermark settings.

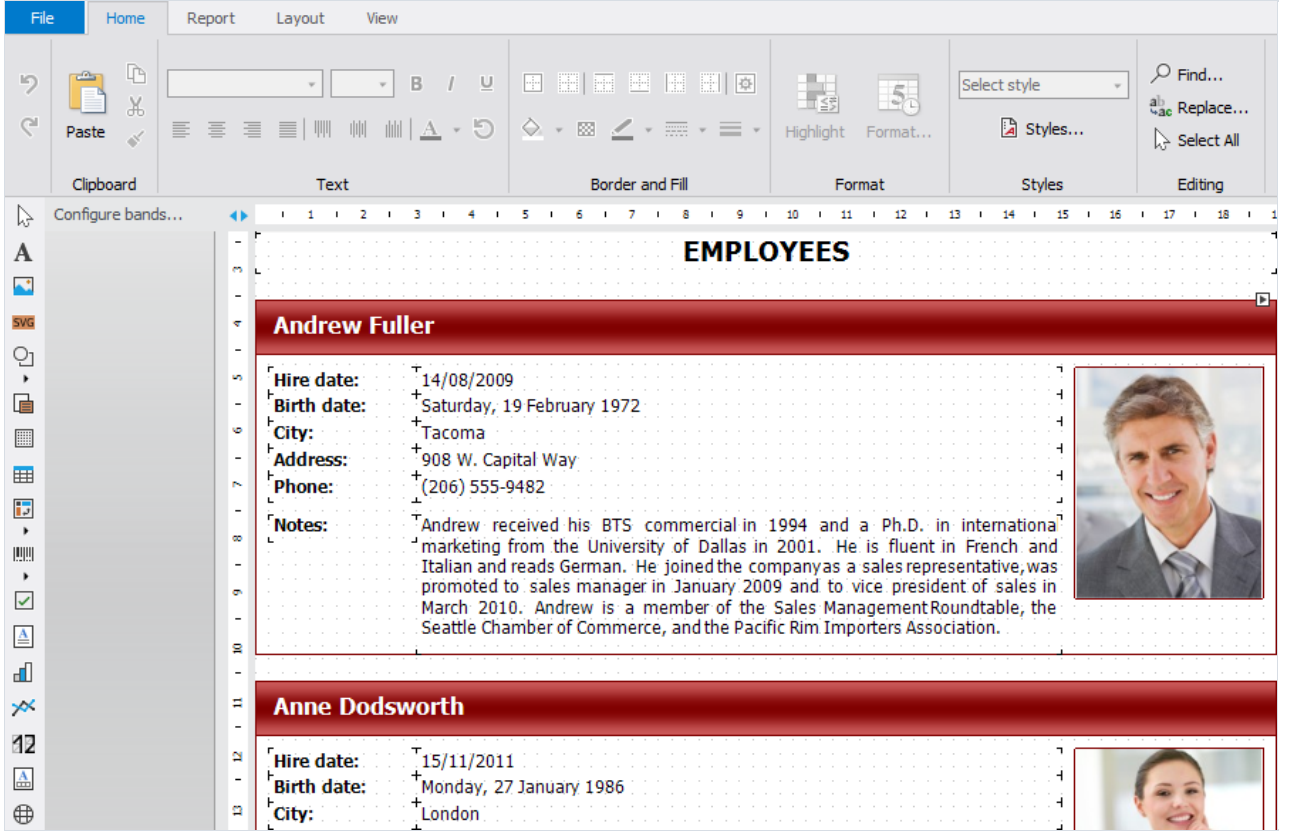
Button	Description
	Navigate to the first page.
	Navigate to the previous page.
	Navigate to the indicated page. Enter the page number and press Enter.
	Navigate to the next page.
	Navigate to the last page.

You can use the following keyboard control:

Key	Description
Ctrl+P	Print the report.
Ctrl+F	Text search.
Arrows	Scroll the preview.
PageUp, PageDown	Page up/down.
Home	Navigate to the first page.
End	Navigate to the last page.
Esc	Close the preview window.

Raporun düzenlenmesi

Hazır rapor sayfasını düzenlemek için, önizleme penceresindeki düğmesine  tıklayın. Bu işlemle, mevcut sayfa rapor tasarımcısında açılır ve bu sayfada istediğiniz tüm değişiklikleri yapabilirsiniz:



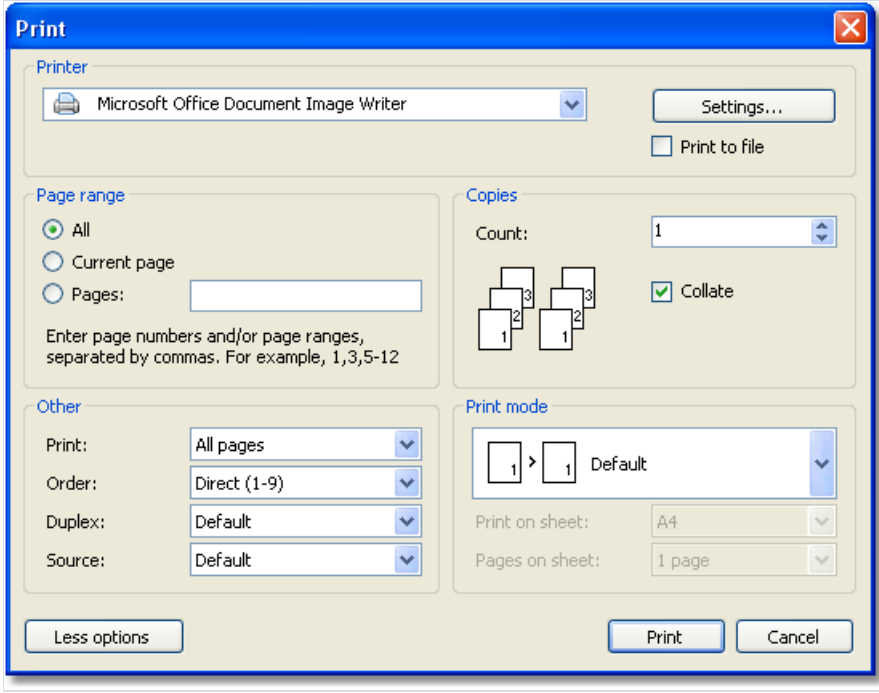
The screenshot displays the Fast-Report editor interface. The top menu bar includes File, Home, Report, Layout, and View. The Home tab is active, showing a ribbon with various tool groups: Clipboard, Text, Border and Fill, Format, Styles, and Editing. The main workspace shows a report design for 'EMPLOYEES'. The report has a red header band with the title 'EMPLOYEES'. Below the header, there are two data bands, each containing a table of employee information. The first data band is for 'Andrew Fuller' and the second is for 'Anne Dodsworth'. Each data band includes fields for Hire date, Birth date, City, Address, Phone, and Notes. A small portrait photo of each employee is displayed to the right of their respective data bands. The left sidebar shows the 'Configure bands...' panel with various icons for report elements like text, images, and tables.

EMPLOYEES	
Andrew Fuller	
Hire date:	14/08/2009
Birth date:	Saturday, 19 February 1972
City:	Tacoma
Address:	908 W. Capital Way
Phone:	(206) 555-9482
Notes:	Andrew received his BTS commercial in 1994 and a Ph.D. in international marketing from the University of Dallas in 2001. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager in January 2009 and to vice president of sales in March 2010. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association.
Anne Dodsworth	
Hire date:	15/11/2011
Birth date:	Monday, 27 January 1986
City:	London

Düzenlemeyi tamamladıktan sonra tasarımcıyı kapatın. Bu durumda, rapor sayfasındaki değişiklikleri kaydetmeniz istenecektir.

Raporun Yazdırılması

Bir raporu yazıcıdan yazdırmak için (veya Ctrl+P kısayol tuş kombinasyonunu) tıklayın. Ekranda bir yazdırma iletişim penceresi görünecektir:



Bu iletişim penceresindeki ayarları aşağıda inceleyelim:

- "Daha Fazla/Daha Az" butonu: Tüm iletişim penceresini ya da sadece temel ayarları gösterir. Varsayılan olarak pencere basitleştirilmiş biçimde görüntülenir. \
- "Yazıcı" grubu: Buradan yazıcı seçilebilir, ayarları yapılabilir ("Ayarlar" butonu) ve dosyaya yazdırma seçilebilir. \
- "Sayfalar" grubu: Hangi sayfaların yazdırılacağı seçilir (tüm sayfalar, geçerli sayfa veya belirtilen sayfa numaraları). \
- "Kopyalar" grubu: Kopya sayısı belirlenebilir ve kopyalarda sayfa sırası seçilebilir ("Kopyalara göre sırala"). \
- "Diğer" grubu: Seçilen aralıktaki hangi sayfaların yazdırılacağı seçilebilir (tüm, çift, tek sayfalar), yazdırma sırası belirlenebilir (doğrudan, ters), çift taraflı yazdırma ayarları yapılabilir ("Çift Taraflı" – eğer yazıcınız destekliyse) ve kağıt kaynağı (yazıcı çekmecesini) seçilebilir. \
- "Yazdırma modu" grubu: Yazdırma modlarından biri seçilebilir.

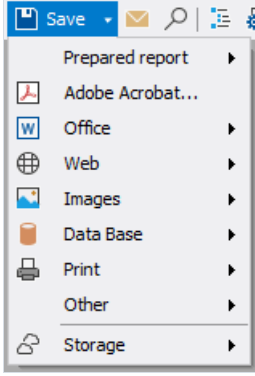
Mod	Açıklama
Normal	Yazıcı, raporda belirtilen kağıt boyutunda yazdırır. Raporun her bir sayfası, yazdırılan bir yaprağa karşılık gelir.
Büyük sayfaları böl	Bu mod, A3 formatındaki bir raporu A4 kağıda yazdırmanız gerektiğinde kullanılır. Bu durumda, raporun bir sayfası iki ayrı baskı sayfasına bölünür. Bu modu kullanırken, "Sayfayı yazdır" listesinden yazdırmak istediğiniz kağıt boyutunu seçmeniz gerekir.
Ölçeklendirme	Bu mod, A4 formatındaki bir raporu A3 kağıda yazdırmanız gerektiğinde kullanılır. Tek bir baskı sayfasına 1, 2, 4 veya 8 rapor sayfası yazdırılabilir. Bu modu kullanırken, "Sayfayı yazdır" listesinden yazdırmak istediğiniz kağıt boyutunu ve "Sayfa sayısı" listesinden bir sayfada yazdırılacak rapor sayfası adedini belirtmeniz gerekir.

"Yazdır" düğmesine bastıktan sonra raporun yazdırılması başlar. Eğer "Dosyaya yazdır" seçeneği işaretliyse, dosya adı istenir ve rapor bu dosyaya kaydedilir (PRN uzantılı dosya, yazıcıya gönderilen bilgilerin bir kopyasını içerir).

Raporu Dışa Aktarma

FastReport, oluşturulan raporun düzenleme, arşivleme, e-posta ile gönderme gibi çeşitli amaçlarla farklı formatlarda dışa aktarılmasına olanak tanır.

Dışa aktarma işlemini seçmek için, önizleme penceresinde "Kaydet" düğmesine tıklayın ve dışa aktarılacak formatı seçin:



Şu anda aşağıdaki formatlara dışa aktarma desteklenmektedir:

- PDF
- Ofis formatları:
 - Excel 2007
 - Word 2007
 - PowerPoint 2007
 - OpenOffice Calc / OpenOffice Writer
 - RTF
 - Excel XML (Excel 2003+)
- Web formatları:
 - HTML
 - MHT
- Görüntü formatları:
 - Bmp, Png, Jpeg, Gif, Tiff, Metafile
 - SVG
- Veritabanı formatları:
 - CSV
 - DBF
 - Json
- Yazdırma formatları:
 - TXT
 - ZPL
 - PPML
 - PostScript
 - XPS
- Diğer formatlar:
 - LaTeX
 - DXF

- o XAML

Ayrıca raporun, Dropbox, Box, Google Drive, OneDrive, Simple Storage Service gibi bulut hizmetlerine veya bir FTP sunucusuna kaydedilmesi de mümkündür.

FPX Formatında Kaydetme

.FPX uzantılı dosya FastReport'un "yerel" formatıdır. Bu formatın avantajları şunlardır:

- **Raporun kalite kaybı olmadan saklanması** – daha önce kaydedilmiş dosya açıldığında, yazdırma, dışa aktarma, düzenleme gibi tüm işlemler gerçekleştirilebilir.
- **Kompakt depolama formatı** – XML formatında olup ZIP ile sıkıştırılmıştır.
- Gerekirse, rapor dosyası ZIP formatını destekleyen herhangi bir arşiv programıyla açılabilir ve herhangi bir metin düzenleyicisinde elle düzenlenebilir.

Bu formatın tek dezavantajı, görüntülenmesi için **FastReport .NET** uygulamasının kurulu olması gerektirir.

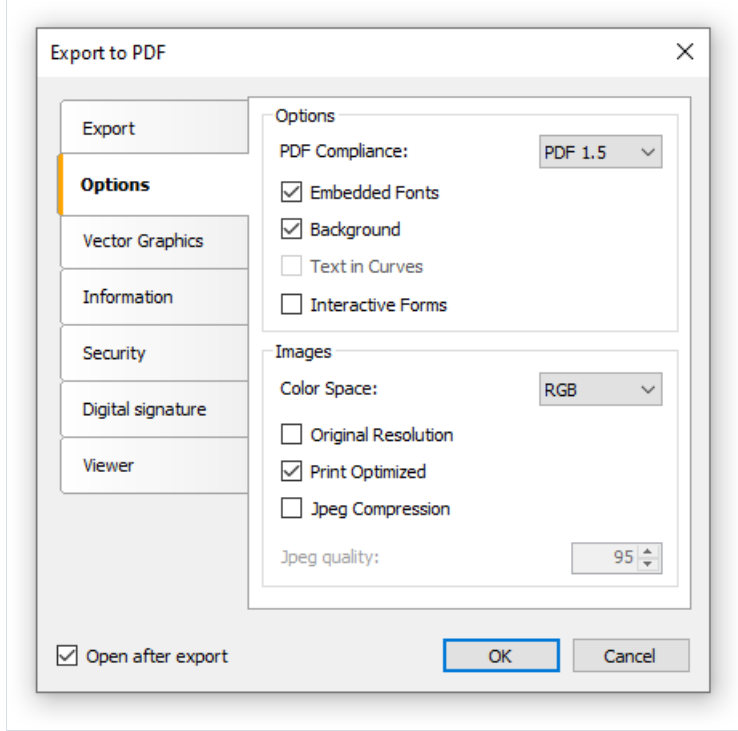
FPX formatında kaydetmek için, önizleme penceresinde **"Kaydet"** düğmesine tıklayın ve dosya türü olarak **"Hazır rapor"** seçin. Daha önce kaydedilmiş bir dosyayı açmak için **"Aç"** düğmesine tıklayın.

Adobe Acrobat (PDF) formatına dışa aktarma

PDF (Portable Document Format) – Adobe Systems firması tarafından oluşturulmuş, platformdan bağımsız bir elektronik belge formatıdır. Görüntüleme için ücretsiz Acrobat Reader yazılımı kullanılır. Bu format oldukça esnektir – gerekli yazı tiplerini, vektörel ve raster (piksel tabanlı) görselleri dahil etme imkânı sunar. Görüntüleme ve sonrasında yazdırma amaçlı belgelerin iletimi ve saklanması için çok uygundur. \

Dışa aktarma yöntemi – katmanlı.

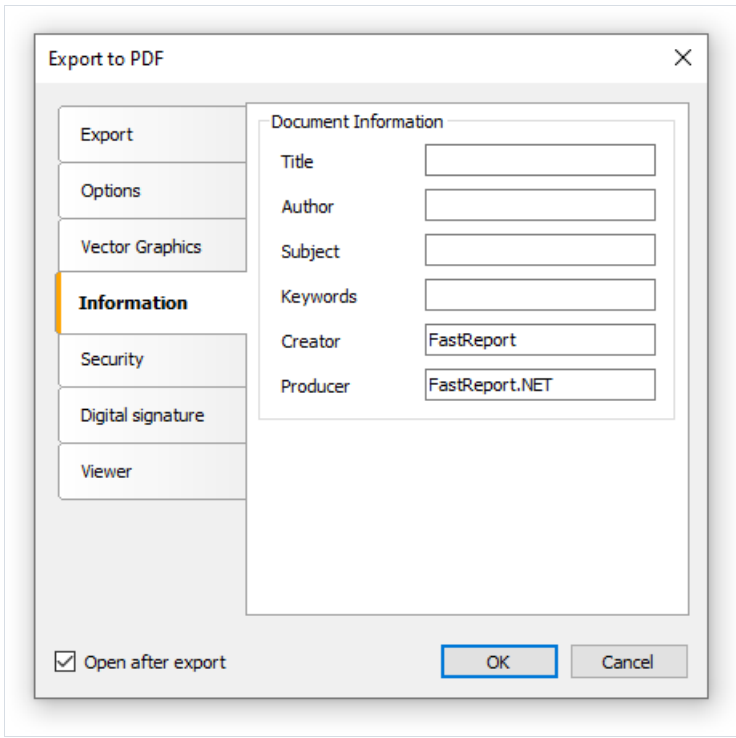
PDF formatına dışa aktarma sırasında, çıktı dosyasının ayarlarını yapabileceğiniz bir iletişim penceresi sunulur:



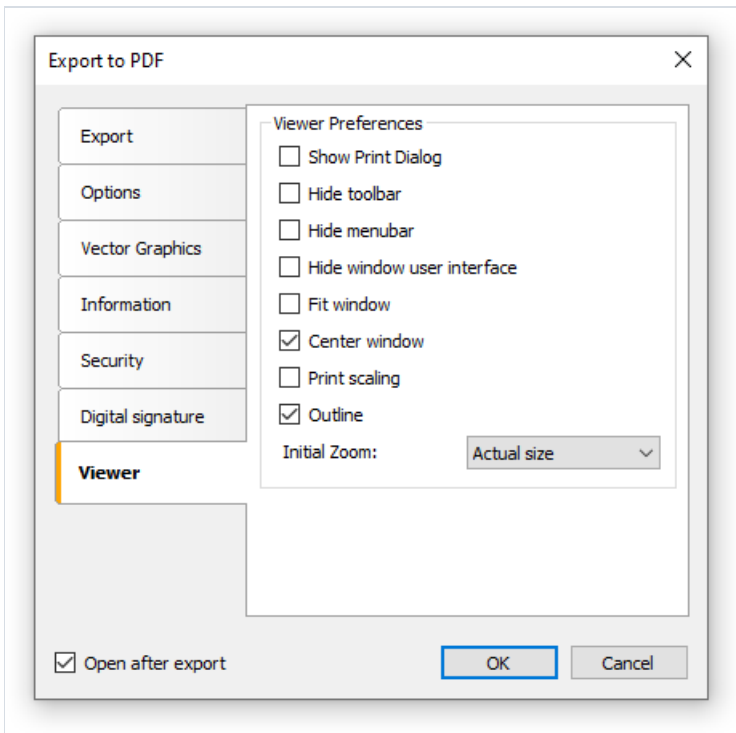
Dışa aktarma ayarları:

- "Yoğunlaştırılmış" - çıkış dosyası sıkıştırılmıştır. Bu, dosya boyutunu küçültür ancak dışa aktarma süresini artırır;
- "Gömülü yazı tipleri" - raporda kullanılan tüm yazı tipleri PDF dosyasına dahil edilecektir. Bu, dosya boyutunu önemli ölçüde artıracaktır;
- "Arka plan" - sayfa filigranı bir resim olarak dışa aktarılacaktır. Bu, dosya boyutunu önemli ölçüde artıracaktır;
- "Yazdırma optimizasyonu" - tüm grafik nesnelerinin (resimler, grafikler gibi) yüksek çözünürlükte yazdırma için çıkışı.

"Bilgi" sekmesinde, belge bilgi alanlarını doldurabilirsiniz:



"Verici" sekmesinde, Adobe Acrobat belge görüntüleyicisi ile ilgili bazı seçenekleri ayarlayabilirsiniz:

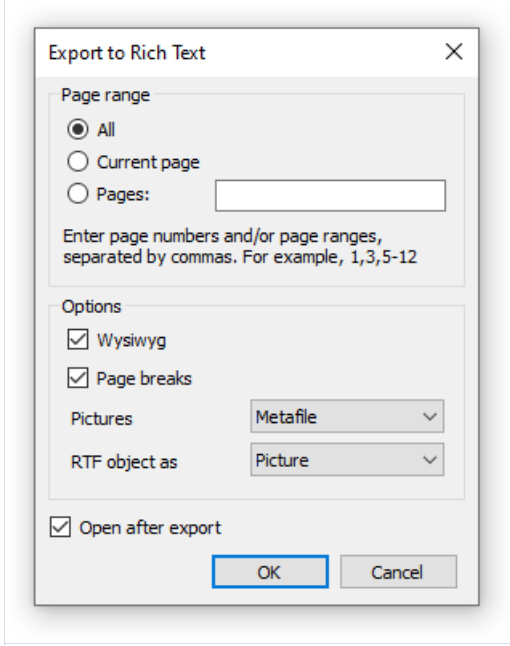


RTF'ye Dışa Aktarma

RTF (Rich Text Format), Microsoft firması tarafından metin belgelerinin değişimi için standart bir format olarak geliştirilmiştir. Günümüzde RTF belgeleri, çoğu modern metin düzenleyici ve işletim sistemiyle uyumludur.

Dışa aktarma yöntemi – tablolama yöntemidir.

RTF formatına dışa aktarma sırasında, çıktı dosyasının ayarlarını yapılandırmak için bir iletişim penceresi sunulur:



Dışa Aktarma Ayarları:

- Wysiwyg – raporun görünümüne en yüksek düzeyde sadakat sağlar. Bu seçenek kapatıldığında, ortaya çıkan tabloda satır ve sütun sayısını azaltmak için optimizasyon yapılır.
- Sayfa Sonları – RTF dosyasında sayfa sonlarını etkinleştirir.
- Resimler – çıktı dosyasındaki grafik görüntülerin formatını seçme imkanı sağlar. Varsayılan olarak meta dosya (.EMF) formatı önerilir. Bu, "Grafik" ve "Şekil" gibi nesnelerin en yüksek kalitede dışa aktarılmasına olanak tanır.

Dışa Aktarmadan Sonra Aç – dışa aktarma işlemi tamamlandığında, sonuç dosyası işletim sisteminde varsayılan olarak tanımlı RTF görüntüleyici (örneğin, MS Word) ile otomatik olarak açılır.

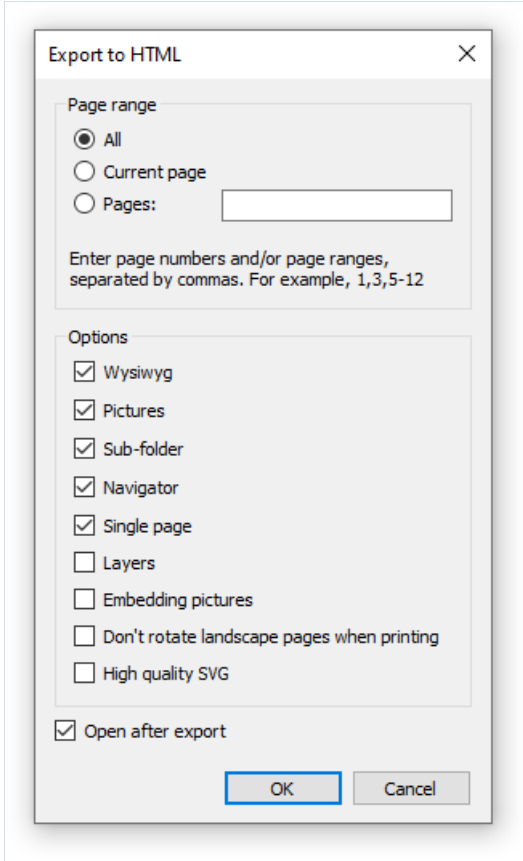
Not: Dosyanın görünümü ve boyutu, rapor şablonuna büyük ölçüde bağlıdır (bkz. "[Rapor Geliştirme Önerileri](#)" bölümü).

HTML'ye Dışa Aktarma

HTML (Hypertext Markup Language), Internet üzerindeki belgeleri biçimlendirmek için standart dil olarak kabul edilir. HTML, teknik ve bilimsel belgelerin paylaşımı için, dizgi konusunda uzman olmayan kişiler tarafından da kullanılabilecek şekilde geliştirilmiştir. Göreceli olarak basit ama görsel olarak düzgün belgeler oluşturmak için kullanılır. Ayrıca HTML, köprü metin (hypertext) desteği içerir.

Dışa aktarma yöntemi: Tablo temelli.

HTML'ye dışa aktarma sırasında, dışa aktarılacak belgenin ayarlarını yapabileceğiniz bir diyalog penceresi sunulur:



Dışa aktarma ayarları:

- Wysiwyg – dışa aktarma sonucu, raporun görseline mümkün olduğunca benzer olur;
- Resimler – grafik görüntülerin dışa aktarılmasını sağlar;
- Dosyalar için ayrı klasör – tüm ek dosyalar `.files` uzantılı ayrı bir klasöre kaydedilir;
- Gezinme aracı – sayfalar arasında hızlı geçiş yapmayı sağlayan özel bir gezgin oluşturulur;
- Tek sayfa – tüm sayfalar tek bir dosyada kaydedilir;
- Dışa aktarmadan sonra aç – oluşturulan dosya, işletim sisteminde varsayılan HTML görüntüleyici program ile hemen açılır.

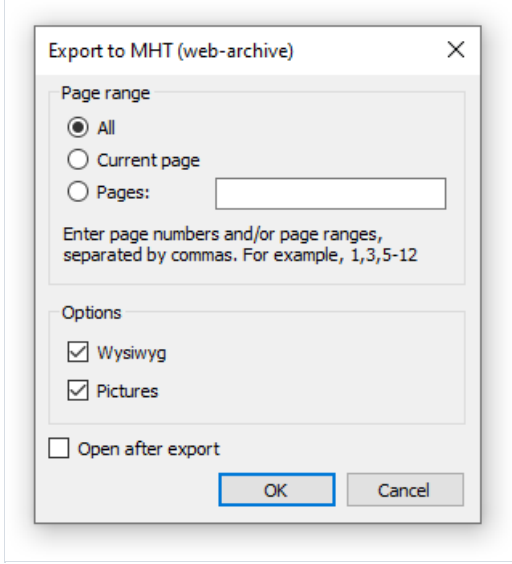
Belgenin görünümü ve boyutu, büyük ölçüde rapor şablonuna bağlıdır (bkz. "[Rapor geliştirme önerileri](#)" bölümü).

MHT'ye (web arşivi) dışa aktarma

MHT formatı (web arşivi), MIME kodlamasına dayanır ve bir sayfanın metin ve grafik içeriğini tek bir dosyada saklamaya olanak tanır. Sayfaların MHT formatında kaydedilmesi Internet Explorer tarafından desteklenir.

Dışa aktarma yöntemi – tablo şeklindedir.

MHT formatına dışa aktarma sırasında, çıkış belgesinin ayarlarını yapılandırmak için bir iletişim penceresi sunulur:



Dışa aktarma ayarları:

- Wysiwyg – dışa aktarma sonucu, rapora mümkün olduğunca benzer görünecektir;
- Resimler – grafik içeriklerin dışa aktarımını etkinleştirir;

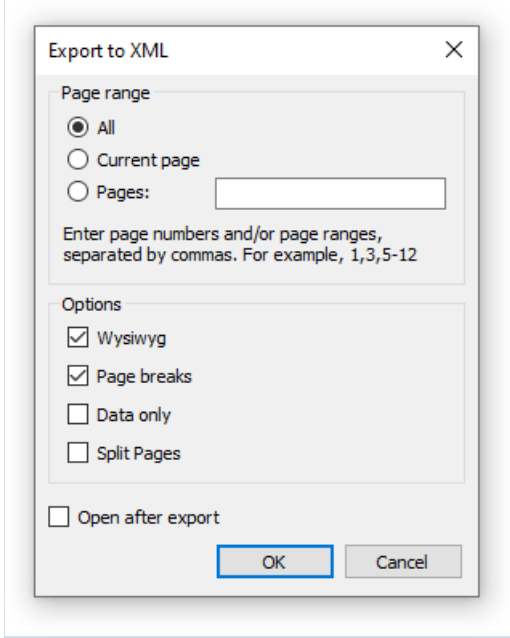
Oluşan dosyanın görünümü ve boyutu, rapor şablonuna bağlıdır ("[rapor geliştirme ile ilgili Öneriler](#)" bölümüne bakınız).

Excel (XML) Dışa Aktarma

Excel – Microsoft Office paketine dahil olan, elektronik tablo oluşturma ve düzenleme uygulamasıdır. \

Dışa aktarma yöntemi – tablo temelli.

Excel'e dışa aktarma sırasında, çıkış belgesinin ayarlarının yapılabileceği bir diyalog penceresi sunulur:



Dışa Aktarma Ayarları:

- Wysiwyg – raporun görünümüne en yüksek düzeyde sadık kalınarak dışa aktarım yapılır. Bu seçenek devre dışı bırakıldığında, sonuç tablo içerisindeki satır ve sütun sayısını azaltmaya yönelik optimizasyon uygulanır.
- Sayfa Sonları – sayfa sonları Excel dosyasına dahil edilir.
- Dışa aktarma sonrası aç – dışa aktarılan dosya, işlem tamamlandıktan sonra otomatik olarak Excel programında açılır. \

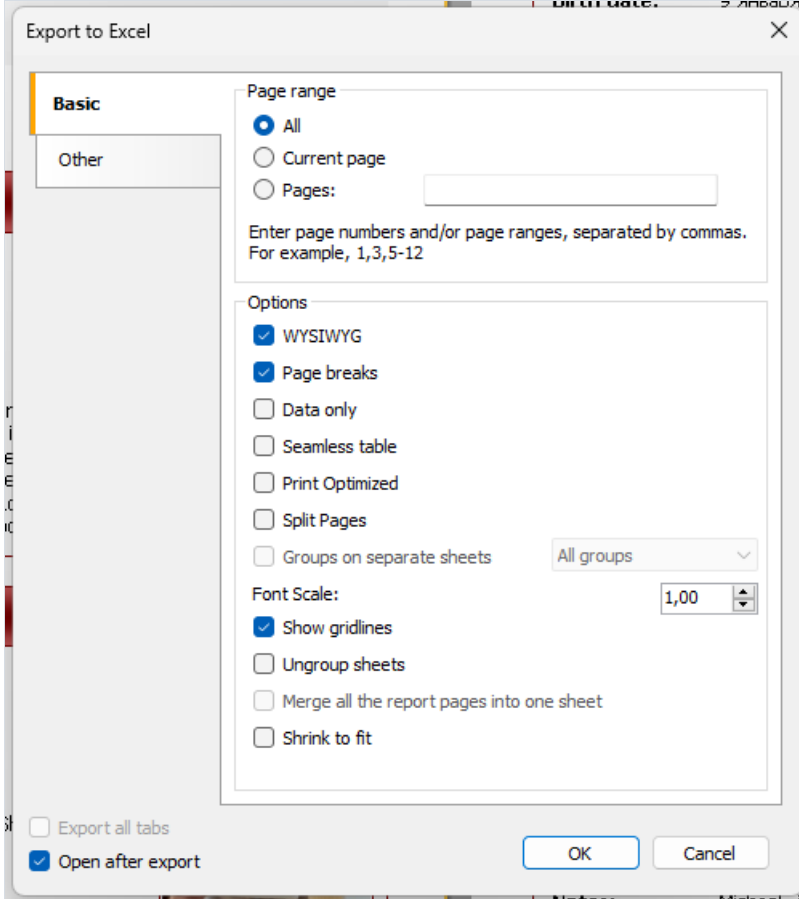
Not: Dosyanın görünümü ve boyutu, büyük ölçüde rapor şablonuna bağlıdır (bkz. "Rapor Geliştirme Önerileri" bölümü).

Excel 2007'ye Dışa Aktarma

Excel 2007, Microsoft Office 2007 paketine dahil bir elektronik tablo uygulamasıdır.

Dışa aktarma yöntemi – tablolama esaslıdır.

Excel'e dışa aktarma sırasında, çıktı belgesinin ayarlarını yapılandırmak için bir diyalog penceresi sunulur:

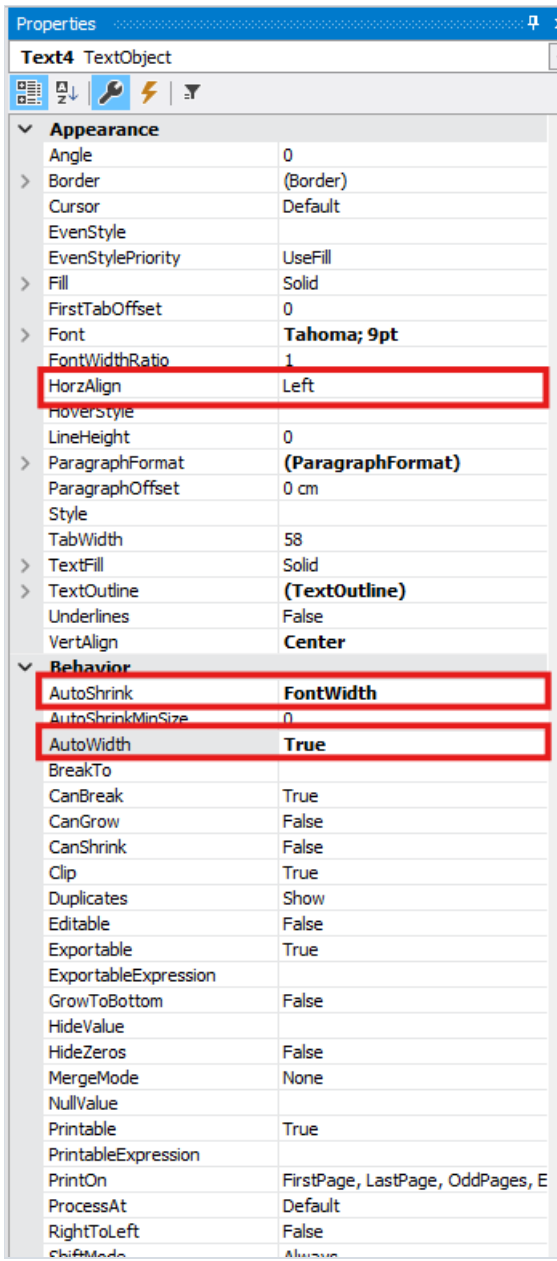


Dışa Aktarma Ayarları:

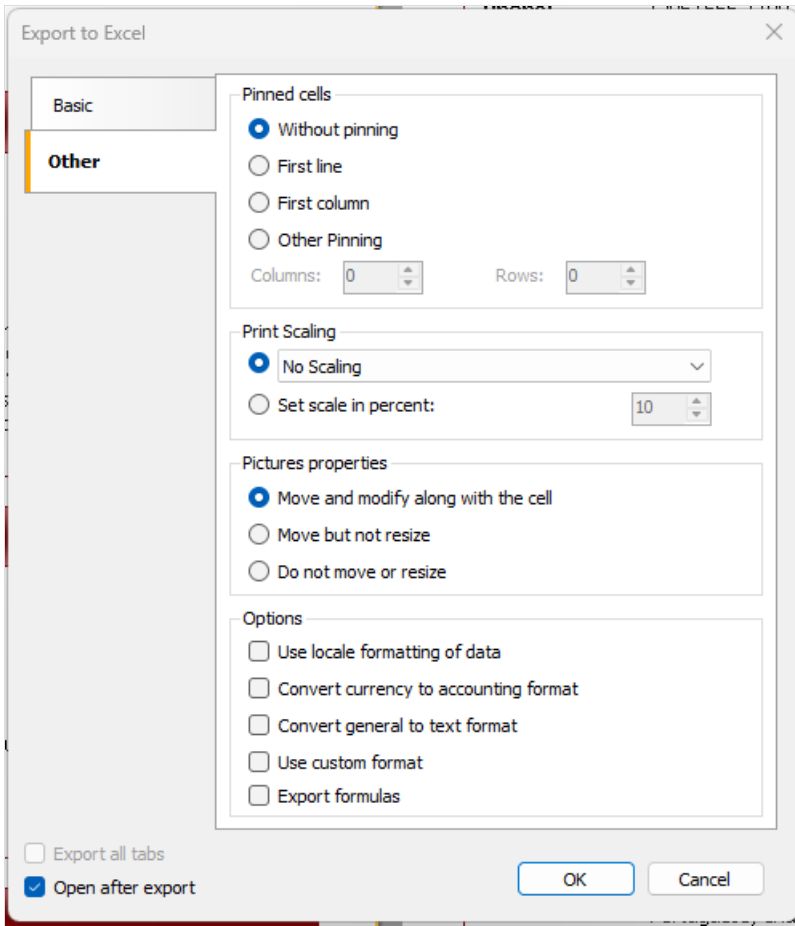
- Wysiwyg – raporun görsel görünümüne maksimum düzeyde uygunluk sağlar. Bu seçenek devre dışı bırakıldığında, çıktı tablosundaki satır ve sütun sayısını azaltmak için optimizasyon uygulanır;
- Sayfa Sonları – sayfa sonlarını etkinleştirir;
- Sayfaları Ayır – hazırlanan rapordaki her sayfa ayrı bir Excel sayfası olarak dışa aktarılır;
- Grupları Ayrı Sayfalarda Göster – her grubun verilerini ayrı bir sayfada sunar. Belgede yalnızca bir grup varsa ya da hiç grup yoksa, açılır listede yalnızca “Tüm Gruplar” seçeneği görünür;
- Sayfaları Gruplandırmayı Kaldır – birlikte gruplandırılan sayfalar senkronize çalışır: biri üzerinde yapılan değişiklikler diğer tüm seçili sayfalara da uygulanır. Varsayılan olarak tüm sayfalar gruplanmıştır;
- Otomatik Genişlik Ayarı – Excel'de hücre genişliği için “Otomatik Sığdır” özelliğini etkinleştirir;
- Tüm Sekmeleri Dışa Aktar – etkileşimli raporlarda, yeni sekmelerde açılan detaylı raporların tüm sekmelerini dışa aktarır;
- Dışa Aktarmadan Sonra Aç – dışa aktarılan dosya işlem tamamlandıktan sonra Excel'de otomatik olarak açılır.

Otomatik genişlik ayarı, yalnızca aşağıdaki metin nesnesi özellikleri belirli değerlerdeyse çalışır:

- AutoWidth = true (Otomatik genişlik);
- AutoShrink = ShrinkToText (Yazı tipine göre otomatik daraltma);
- HorzAlign, Justify dışındaki herhangi bir hizalama değerine sahip olmalıdır.



“Diğer” sekmesi altındaki bölümler, Excel’e dış aktarma sırasında verilerin görünümü ve davranışı üzerinde hassas kontrol sağlar. Her bölüm, verilerin Excel’de nasıl biçimlendirileceğini ve gösterileceğini belirlemek için çeşitli seçenekler sunar:



Hücre Sabitleme bölümü, Excel sayfasının belirli alanlarının sabitlenme biçimini seçmenizi sağlar. Bu, tablo içinde kaydırma veya yakınlaştırma yapıldığında belirli satır veya sütunların görünürlüğünü korumak açısından faydalıdır.

Yazdırma Ölçekleme bölümü, verilerin yazdırılırken sayfalara nasıl dağıtılacağını belirlemek için ölçekleme ve sayfa yerleşimini yapılandırma imkânı sağlar.

Görsel Özellikleri bölümü, Excel hücrelerine yerleştirilen görsellerin davranışlarını yapılandırmaya olanak tanır. Sunulan seçenekler sayesinde, hücrelerin yeniden boyutlandırılması veya taşınması durumunda görsellerin nasıl hareket edeceği ve boyutlandırılacağı kontrol edilebilir.

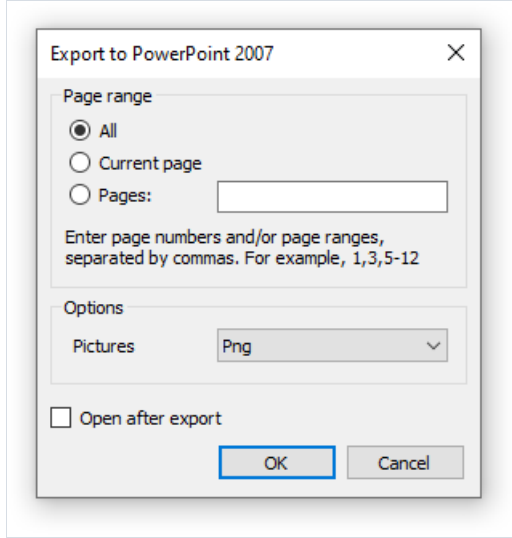
Rapor şablonunun yapısı, çıktı dosyasının görünümünü ve boyutunu önemli ölçüde etkileyebilir (bkz. "[Rapor Tasarımı İçin Öneriler](#)" bölümü).

PowerPoint 2007'ye Aktarma

PowerPoint 2007, Microsoft Office 2007 paketine dâhil olan bir sunum hazırlama uygulamasıdır.

Dışa aktarma yöntemi – katmanlıdır.

PowerPoint 2007'ye dışa aktarma sırasında, çıktı dosyasının ayarlarını belirlemek için bir iletişim penceresi açılır:



Dışa Aktarma Ayarları:

- Resimler – grafik görüntülerin hangi formatta kaydedileceğini seçme imkânı;
- Dışa aktarımdan sonra aç – dışa aktarma işlemi tamamlandıktan sonra ortaya çıkan dosya PowerPoint programında otomatik olarak açılır.

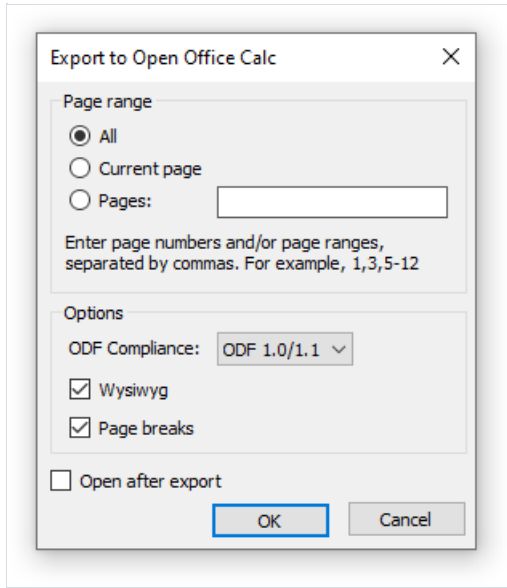
OpenOffice Calc'e Dışa Aktarma

OpenDocument Formatı, düzenlenebilir ofis belgelerinin (notlar, raporlar, kitaplar gibi metin belgeleri; elektronik tablolar, çizimler, veritabanları, sunumlar) saklanması ve paylaşılması için açık bir dosya formatıdır. Bu standart, OASIS endüstri topluluğu tarafından geliştirilmiş olup, başlangıçta OpenOffice.org tarafından oluşturulan XML tabanlı bir formata dayanır. 1 Mayıs 2006 tarihinde ISO/IEC 26300 uluslararası standardı olarak kabul edilmiştir. \

FastReport, OpenDocument biçiminde (.ods uzantılı) tablo dışa aktarmayı destekler. Bu dosyalar, ücretsiz OpenOffice ofis paketiyle açılabilir.

Dışa aktarma yöntemi: Tablo tabanlıdır.

Dışa aktarma sırasında, çıktı belgesinin ayarlarının yapılabildiği bir iletişim penceresi sunulur:



Dışa aktarma ayarları:

- Wysiwyg – Raporun görsel yapısına en yakın çıktıyı elde etmeyi sağlar. Bu seçenek kapatıldığında, sonuç tabloyu daha az satır ve sütunla optimize etmeye yönelik işlemler yapılır;
- Sayfa sonları – Sayfa sonlarının dahil edilmesini sağlar;

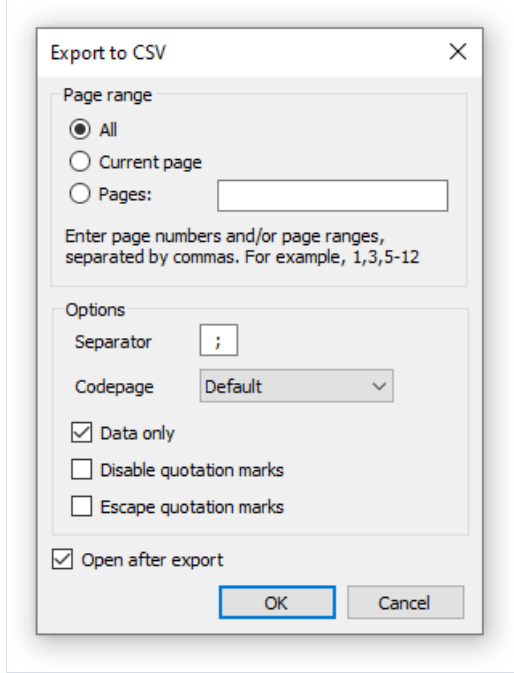
Dışa aktarmadan sonra aç – Dışa aktarma işlemi tamamlandıktan sonra ortaya çıkan dosya otomatik olarak açılır.

CSV'ye Dışa Aktarma

CSV dosyası, tablo şeklinde biçimlendirilmiş ve her sütun değerinin bir ayırıcıyla ayrıldığı, her yeni satırın yeni bir satırdan başladığı şekilde düzenlenmiş verileri içerir. Bu format, çeşitli tablo düzenleyicilerine aktarılabilir.

Dışa aktarma yöntemi – tablolama esaslıdır.

CSV'ye dışa aktarma sırasında, çıktı belgesinin ayarlarını yapılandırmak için bir iletişim penceresi sunulur:



Dışa Aktarma Ayarları:

- Ayırıcı – ayırıcı karakterin seçimi;
- Kodlama – belgenin karakter kodlamasının seçimi. "Varsayılan" kodlama, Windows'un geçerli kod sayfasına karşılık gelir;
- Sadece veriler – yalnızca "Veri" bandı üzerinde bulunan nesneler dışa aktarılır;

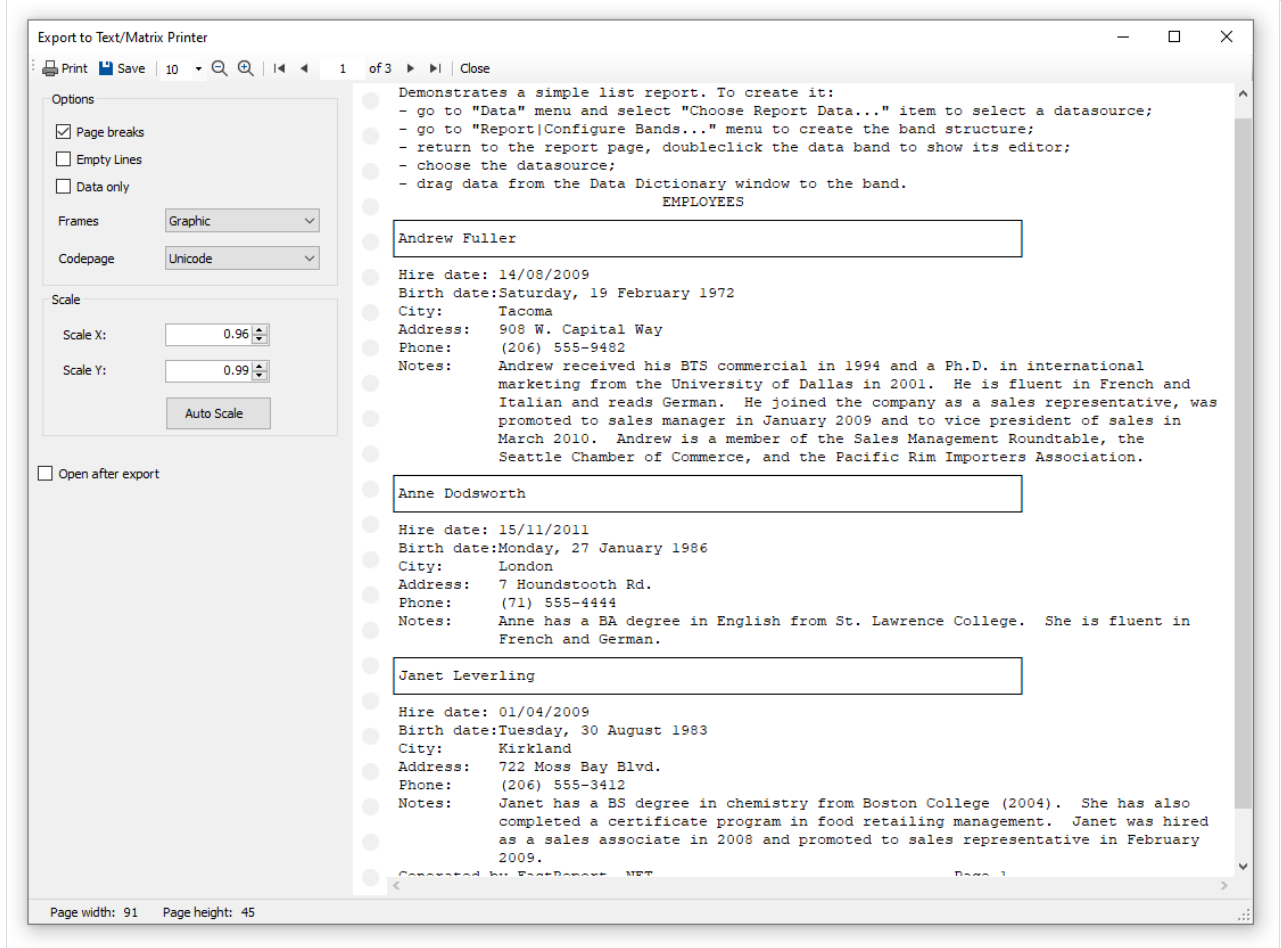
Dışa aktarmadan sonra aç – dışa aktarma tamamlandıktan sonra, ortaya çıkan dosya Excel'de otomatik olarak açılır.

TXT'ye Aktarma (Metin Formatı)

Düz metin dosyası, rapordaki bilgileri bu formatın özelliklerine göre en iyi şekilde optimize ederek ve dönüştürerek içeren bir dosyadır. Elde edilen belge, bir dosya olarak kaydedilebilir veya nokta vuruşlu yazıcıda yazdırılabilir.

Aktarma yöntemi – tablolama.

Metin biçimine aktarım sırasında, çıktı dosyasının ayarlarını yapabileceğiniz bir iletişim penceresi sunulur:



Aktarma Ayarları:

- Sayfa sonları – metin dosyasına sayfa sonlarının dahil edilmesini sağlar;
- Boş satırlar – çıktı dosyasına boş satırların dahil edilmesini sağlar;
- Yalnızca veriler – yalnızca "Veri" bandı üzerindeki nesneler dışa aktarılır;
- Çerçeveler – çerçeve tipinin seçimi (grafiksel / metinsel / çerçevesiz);
- Kodlama – metin dosyasının karakter kodlamasının seçimi;
- X Ölçeği – yatay ölçekleme katsayısı;
- Y Ölçeği – dikey ölçekleme katsayısı;
- Otomatik ayarlama – bilgilerin kaybını önlemek amacıyla X ve Y katsayılarının otomatik olarak belirlenmesi.

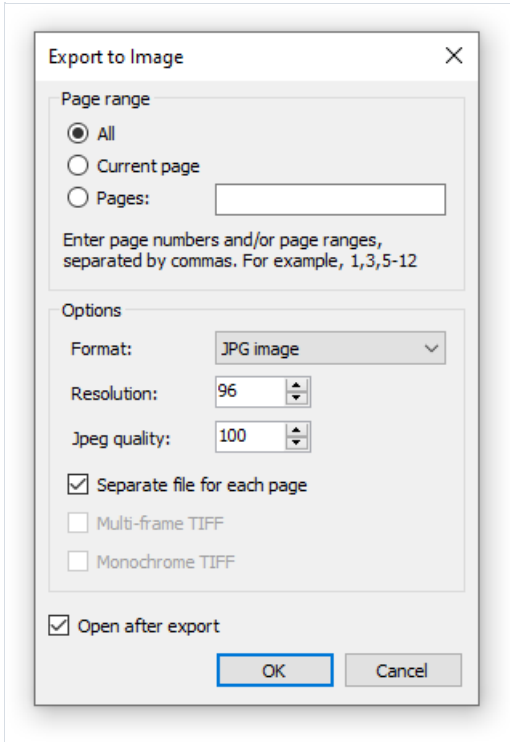
Görsel dosyalara dışa aktarma

FastReport, bilgileri aşağıdaki grafik formatlarında dışa aktarmanıza olanak tanır:

- BMP
- PNG
- JPG
- GIF
- TIFF
- Meta dosyası (EMF, WMF)

Dışa aktarma yöntemi – çizimdir.

Bir görsel dosyasına dışa aktarma sırasında, görselin ayarlarını yapılandırmak için bir iletişim penceresi sunulur:



Dışa aktarma ayarları:

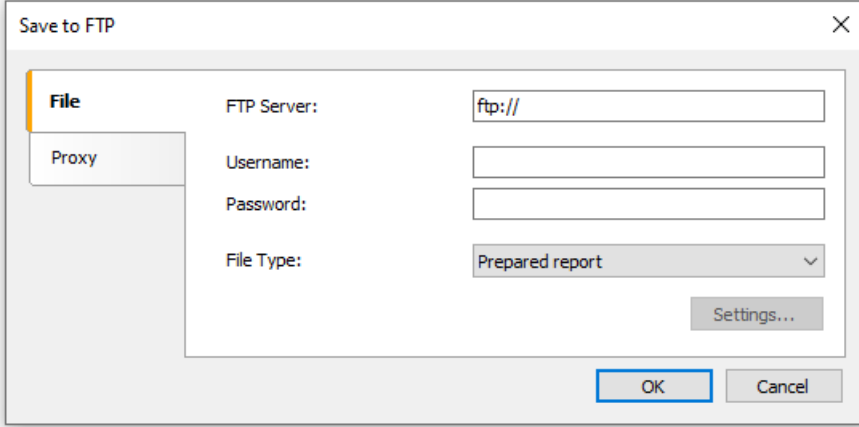
- Format – Görüntü formatı;
- Çözünürlük (dpi) – Görüntünün çözünürlüğü. TIFF formatına dışa aktarımda, dikey ve yatay çözünürlük için farklı değerler belirtilebilir;
- Jpeg Kalitesi – JPG dosyasının sıkıştırma oranı. Bu seçenek yalnızca Jpeg formatına dışa aktarımda kullanılır;
- Her sayfa için ayrı dosya – Bu seçenek etkinleştirildiğinde, rapordaki her sayfa ayrı bir dosya olarak dışa aktarılır. Dosya adı, seçilen ada sayfa numarası eklenerek oluşturulur;
- Çok sayfalı TIFF – Bu seçenek etkinleştirildiğinde, birden fazla sayfa içeren tek bir TIFF dosyası oluşturulur. Bu seçenek yalnızca TIFF formatına dışa aktarımda kullanılır. Bu durumda “Her sayfa için ayrı dosya” seçeneği yok sayılır;
- Siyah-beyaz TIFF – Bu seçenek etkinleştirildiğinde, siyah-beyaz bir TIFF dosyası oluşturulur. Bu seçenek yalnızca TIFF formatına dışa aktarımda kullanılır.

Not: “Her sayfa için ayrı dosya” seçeneği devre dışıyken birden fazla sayfanın tek dosyada dışa aktarılması, yüksek kaynak kullanımı gerektirebilir.

FTP'ye Dışa Aktarma

FastReport .NET'in önizleme penceresinde, oluşturulan raporu bir FTP sunucusuna kaydetme imkânı vardır. Rapor, desteklenen herhangi bir formata da aktarılabilir ve ardından FTP'ye kaydedilebilir.

Bunun için "Kaydet" düğmesine tıklayın ve "FTP..." seçeneğini seçin. FTP sunucusuna kaydetme penceresi açılacaktır:

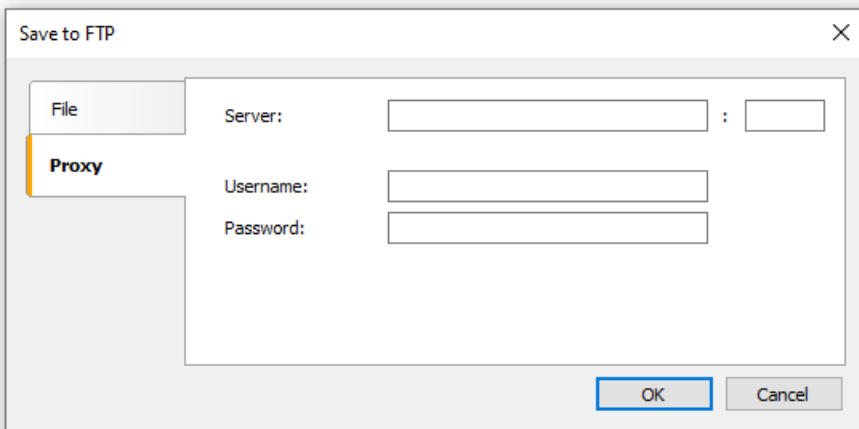


"Dosya" sekmesinde şu alanlar bulunur:

- FTP Sunucusu: Dosyanın kaydedileceği FTP sunucusunun URL adresi bu alana girilmelidir.
- Kullanıcı Adı ve Şifre: İlgili FTP sunucusundaki kullanıcı adınızı ve şifrenizi bu alanlara girin.
- Dosya Türü: Dosyanın kaydedileceği formatı seçebileceğiniz açılır bir listedir (Hazır rapor ya da desteklenen dışa aktarma biçimlerinden biri).

Herhangi bir dışa aktarma biçimi seçildiğinde, "Ayarlar..." düğmesi aktif olur. Bu düğmeye tıklayarak seçilen dışa aktarma biçiminin ayarlarını yapılandırabileceğiniz pencereye geçebilirsiniz.

Eğer bir proxy sunucusu kullanıyorsanız, "Proxy" sekmesinde proxy sunucusunun URL adresini, port numarasını, kullanıcı adınızı ve şifrenizi girebilirsiniz:



Gerekli ayarları yaptıktan ve dosya formatını seçtikten sonra "**Tamam**" düğmesine basmanız yeterlidir — dosya FTP sunucusuna kaydedilecektir.

Dropbox'a Aktarma

Hazırlanan bir rapor, FastReport.NET önizlemesinden Dropbox'a kaydedilebilir. Rapor, Dropbox'a kaydedilmeden önce desteklenen formatlardan birine dışa aktarılabilir.

Bir raporu Dropbox'a kaydetmeden önce, Dropbox hesabınızda bir uygulama oluşturulmalıdır. Bunu yapmak için Dropbox hesabınıza giriş yapın ve şu adımları izleyin:

- "Diğer" düğmesine tıklayın: bu düğme Dropbox ana sayfasının en altında bulunur.
- Açılır listeden "Geliştiriciler" seçeneğini seçin: geliştiriciler için olan sayfaya yönlendirileceksiniz.
- "Uygulama Konsolu"na gidin: bu sizi uygulama listesinin bulunduğu sayfaya yönlendirir.
- "Uygulama Oluştur" düğmesine tıklayın: Dropbox e-posta adresinizi kontrol edecektir. "E-posta Gönder" düğmesine tıklayın. E-posta kutunuzda "Onayla" düğmesini içeren bir e-posta bulacaksınız: e-posta adresinizi onaylamak için bu düğmeye tıklayın.

Son olarak, "Yeni bir Dropbox Platformu uygulaması oluştur" sayfasına yönlendirileceksiniz.

"Dropbox API uygulaması"nı seçin ve "Uygulamanız Dropbox'ta ne tür verileri saklamalı?" sorusuna "Dosyalar ve veri depoları" seçeneğini seçerek yanıt verin.

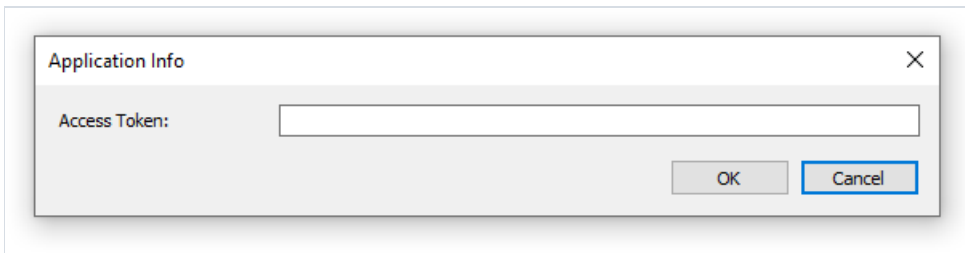
"Uygulamanız yalnızca kendi özel klasörüne sınırlandırılabilir mi?" sorusuna sunulan iki yanıttan birini seçerek cevap verin.

Bu sayfa, uygulamanın adını girmenizi de gerektirir.

"Uygulama oluştur" düğmesine tıkladıktan sonra sistem uygulamayı oluşturacaktır.

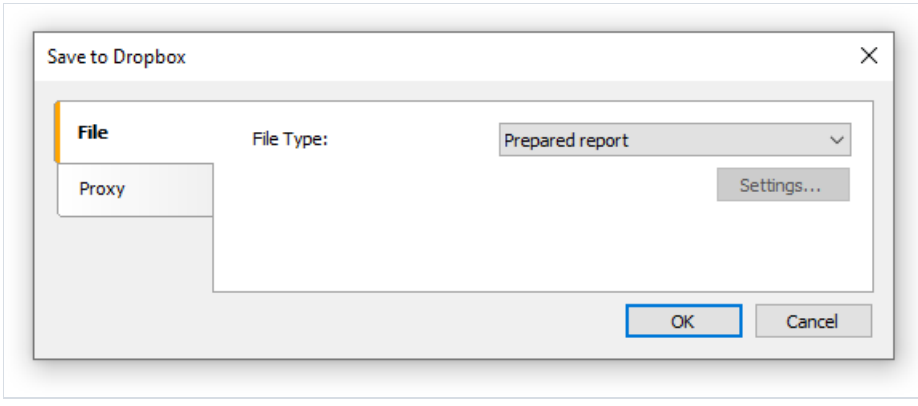
Sonuç olarak, uygulama ayarları sayfası açılır. Burada, Dropbox'a dışa aktarım yaparken gerekli olan "Uygulama anahtarı (App key)" ve "Uygulama gizli anahtarı (App secret)" bilgilerini görebilirsiniz.

Artık FastReport.NET önizlemesine gidip "Kaydet" düğmesine basarak ve "Dropbox..." seçeneğini seçerek dosyayı Dropbox'a dışa aktarabilirsiniz. Dropbox'a ilk kez dışa aktarım yapıldığında, "Uygulama Bilgisi" penceresi görüntülenecektir:



Yukarıda elde edilen "Uygulama Anahtarı"nı (App key) ve "Uygulama Gizli Anahtarı"nı (App secret) girin. "Tamam" düğmesine tıkladıktan sonra, FastReport.NET bu değerleri kaydeder ve bir sonraki kullanımdan itibaren tekrar kullanır.

"Dropbox'a Kaydet" penceresinin iki sekmesi vardır: Dosya ve Proxy:

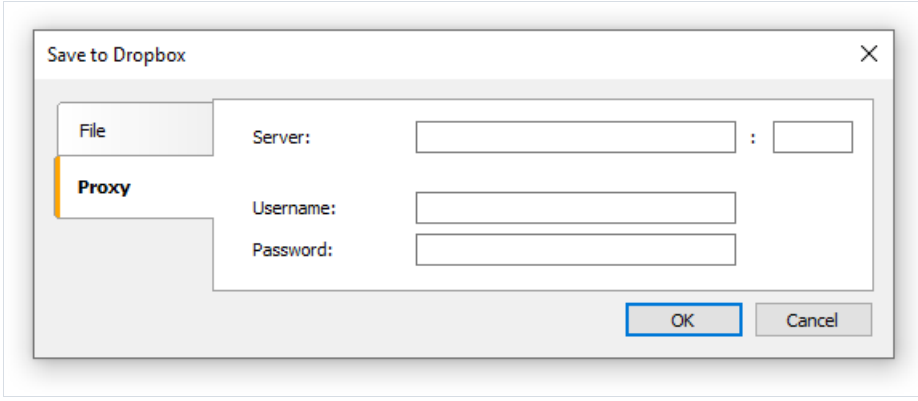


Dosya sekmesi şu alanları içerir:

- Dosya Türü: Açılır listeden dosya formatını (hazırlanan rapor veya dış aktarma formatlarından biri) seçin.

Bir dış aktarma formatı seçildiğinde, "Ayarlar..." düğmesi etkinleşir ve seçilen dış aktarma formatı için ayar penceresini açar.

Bir proxy sunucusu kullanılıyorsa, proxy sunucusunun URL adresi, portu, kullanıcı adı ve şifresi Proxy sekmesinde ayarlanır:



Tüm ayarlar yapıldıktan sonra, dosyayı Dropbox'a kaydetmek için "Tamam" düğmesine tıklayın.

Google Drive'a Dışa Aktarma

FastReport .NET önizleme modunda, oluşturulmuş raporu Google Drive'a kaydetme imkanı bulunmaktadır. Rapor, desteklenen herhangi bir formata önceden dışa aktarılabilir.

İlk olarak Google Drive'da bir uygulama oluşturmanız gerekir. Bunun için şu adrese gidin:

<https://code.google.com/apis/console>

- Bu sayfada lisans sözleşmesi şartlarını kabul edin.

Ardından proje yönetim sayfasına yönlendirilirsiniz. Burada "Services" sekmesine geçin ve Drive API'yi etkinleştirin. Sonra "API Access" sekmesine geçin ve "Create an OAuth 2.0 client ID" düğmesine tıklayın.

Branding Information bölümünde uygulama adını girin ve Next butonuna basın.

Client ID Settings bölümünde aşağıdakileri seçin:

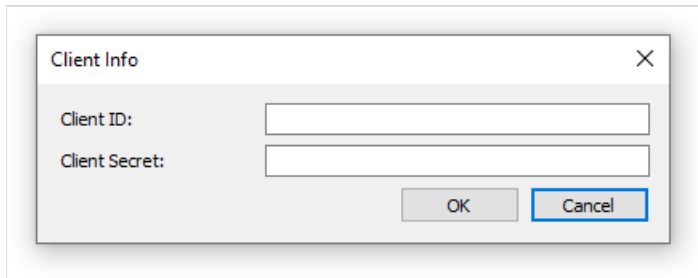
- Application type: "Installed application"
- Installed application type: "Other"

Create Client ID butonuna tıklayın.

Sonuç olarak, API Access sayfasında "Client ID for installed applications" başlığı altında Client ID ve Client Secret bilgileri görüntülenir. Bu iki bilgi ileride gerekecektir.

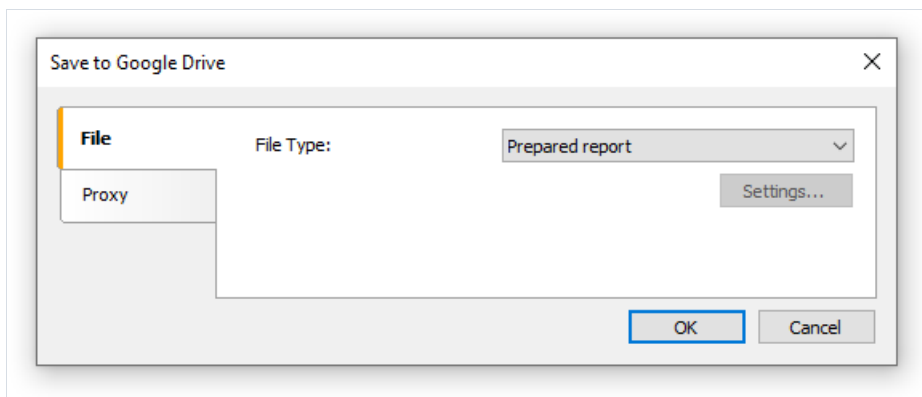
Şimdi FastReport .NET önizleme moduna geçip dosyayı Google Drive'a dışa aktarabilirsiniz. Bunun için "Kaydet" butonuna basın ve "GoogleDrive..." seçeneğini tıklayın.

Google Drive'a ilk dışa aktarma girişiminde, "İstemci Bilgisi" adlı bir pencere açılır:



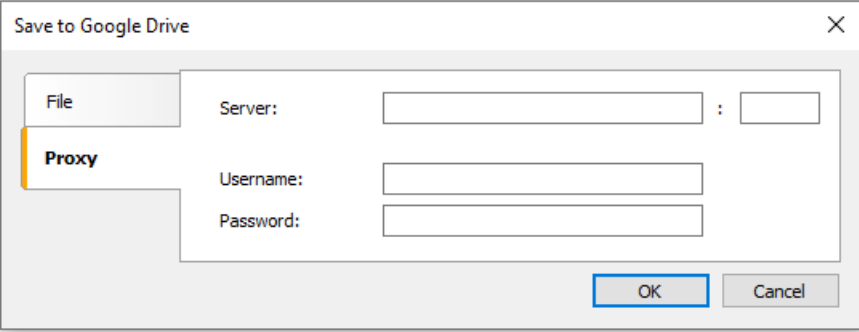
Buraya önceden alınmış Client ID ve Client Secret bilgilerini girin. "Tamam" butonuna bastıktan sonra FastReport bu bilgileri kaydeder ve bir daha girmeniz gerekmez.

Daha sonra Google Drive'a Kaydetme penceresi açılır:



"Dosya" sekmesinde, kaydedilecek dosya türünü (hazır rapor veya dışa aktarım formatlarından biri) seçebilirsiniz. Dışa aktarma formatı seçildiğinde, "Ayarlar..." düğmesi aktif hale gelir. Bu düğme ile ilgili dışa aktarım ayarlarına erişebilirsiniz.

Bir proxy sunucu kullanıyorsanız, "Proxy" sekmesinde sunucu URL'si, port, kullanıcı adı ve parolayı girebilirsiniz.



The image shows a "Save to Google Drive" dialog box. It has a title bar with a close button (X). On the left, there are two tabs: "File" and "Proxy". The "Proxy" tab is selected and highlighted with an orange bar. The main area of the dialog contains three input fields: "Server:" followed by a text box and a port box (separated by a colon), "Username:" followed by a text box, and "Password:" followed by a text box. At the bottom right, there are two buttons: "OK" and "Cancel".

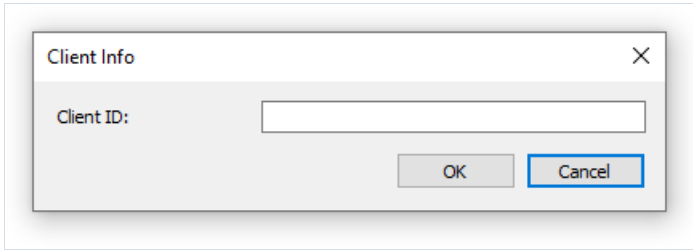
Ayarları girdikten ve formatı seçtikten sonra "Tamam" butonuna tıklayın. Dosya Google Drive'a kaydedilecektir.

OneDrive'a Dışa Aktarma

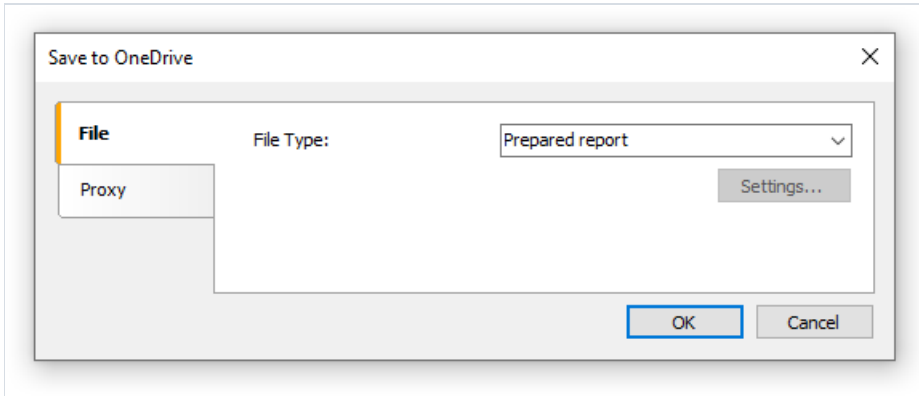
FastReport .NET önizleme modunda, oluşturulan raporu OneDrive'a kaydetme özelliği bulunmaktadır. Rapor ayrıca desteklenen formatlardan birine dışa aktarılıp ardından OneDrive'a da kaydedilebilir.

Öncelikle OneDrive'da bir uygulama oluşturulmalıdır. Bunun için OneDrive sayfasında "Geliştiriciler" bağlantısına tıklanır. Geliştirici merkezi sayfasında "Uygulamalarım" seçilir, ardından "Uygulama oluştur" butonuna tıklanır. Uygulama adı girilir ve dil seçilir. "Kullanım koşulları" ile "Gizlilik bildirimi" okunur ve "Kabul ediyorum" düğmesine basılır. Sonuç olarak, uygulama ayarlarının bulunduğu bir sayfa açılır. Burada gerekli olan "İstemci Kimliği" (Client ID) ve "İstemci Sırrı" (Client Secret) görüntülenir. Geri yönlendirme etki alanı belirtilir ve "Kaydet" düğmesine tıklanır.

Artık FastReport .NET önizleme moduna geçilerek dosya OneDrive'a dışa aktarılabilir. Bunun için "Kaydet" butonuna tıklanır ve "OneDrive..." seçeneği belirlenir. OneDrive'a ilk dışa aktarma denemesinde "İstemci Bilgileri" penceresi açılır:



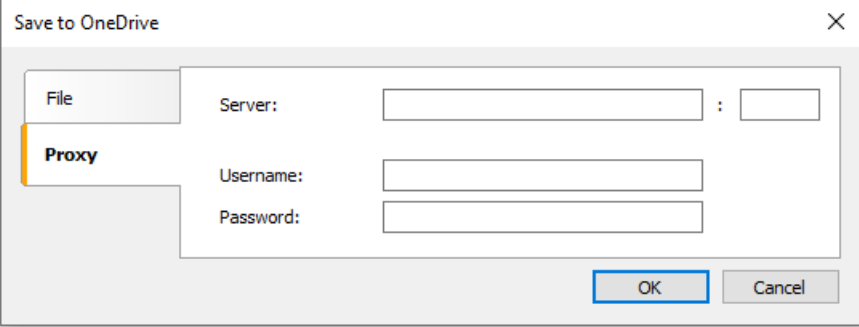
Bu pencerede daha önce alınan İstemci Kimliği (Client ID) ve İstemci Sırrı (Client Secret) girilir. "Tamam" butonuna tıklanmasının ardından FastReport bu bilgileri kaydeder ve tekrar girilmesine gerek kalmaz.



Sonrasında OneDrive'a kaydetme penceresi açılır:

Dosya sekmesinde, kaydedilecek dosyanın türü seçilebilir (Hazır rapor veya dışa aktarma biçimlerinden biri). Herhangi bir dışa aktarma seçildiğinde "Ayarlar..." butonu etkin hale gelir. Bu butona basıldığında, seçilen dışa aktarma türünün ayar penceresi açılır.

Eğer bir proxy sunucusu kullanılıyorsa, **Proxy** sekmesinde proxy sunucusunun URL'si, portu, kullanıcı adı ve parolası girilebilir.



The image shows a 'Save to OneDrive' dialog box. It has a title bar with a close button (X). On the left, there is a sidebar with two tabs: 'File' and 'Proxy'. The 'Proxy' tab is selected and highlighted with an orange bar. The main area of the dialog contains three input fields: 'Server:' followed by a text box and a port box (indicated by a colon), 'Username:' followed by a text box, and 'Password:' followed by a text box. At the bottom right, there are two buttons: 'OK' and 'Cancel'. The 'OK' button is highlighted with a blue border.

Ayarlar yapıldıktan ve format seçildikten sonra "Tamam" butonuna tıklanır ve dosya OneDrive'a kaydedilir.

Rapor Geliştirme İçin Öneriler

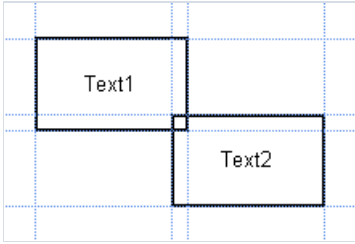
Bu bölümde, verilerin farklı formatlara aktarımı için hazırlanacak raporların geliştirilmesine yönelik öneriler sunulmaktadır.

FastReport, rapor oluştururken nesnelerle çalışmak için geniş olanaklar sunar; bu da her türlü raporun hızlı bir şekilde hazırlanmasında ve ardından yazıcıda çıktısının alınmasında önemli bir avantaj sağlar. Yazdırılan belge, ekranda nasıl görünüyorsa aynı şekilde görünecektir; bu da FastReport rapor oluşturucusunun ana hedefidir.

Ancak bu geliştirme özgürlüğünün bir dezavantajı, oluşturulan belgenin farklı veri formatlarına aktarımındaki zorluklardır. Bu formatların çoğu, bilgileri sunarken belirli sınırlamalara sahiptir. Birçok format, verilerin tablo şeklinde sunulmasını gerektirir. HTML, XLS, RTF ve CSV gibi formatlardan söz edilmektedir. Bu gibi tablolaştırma temelli formatlarda hücrelerin çakışması veya üst üste gelmesi (özellikle HTML ve RTF’te) kabul edilemez.

Aktarım filtreleri, nesnelerin aktarımında bu gereksinimleri mümkün olduğunca dikkate alır. Bu, nesne çakışmalarını ve optimal yerleşimlerini dikkate alan özel algoritmalar sayesinde gerçekleştirilir. Nesnelerin çakıştığı yerlerde, çıktı tablosunda yeni satırlar ve sütunlar oluşturulur. Bu, aktarım sonucunun orijinal rapora mümkün olduğunca benzemesi için gereklidir. Rapor içerisinde çok fazla çakışan nesne olması, çıktı tablosundaki satır ve sütun sayısının artmasına neden olur, bu da aktarım sürecini yavaşlatır ve sonuç dosyasının kullanılmasını zorlaştırır.

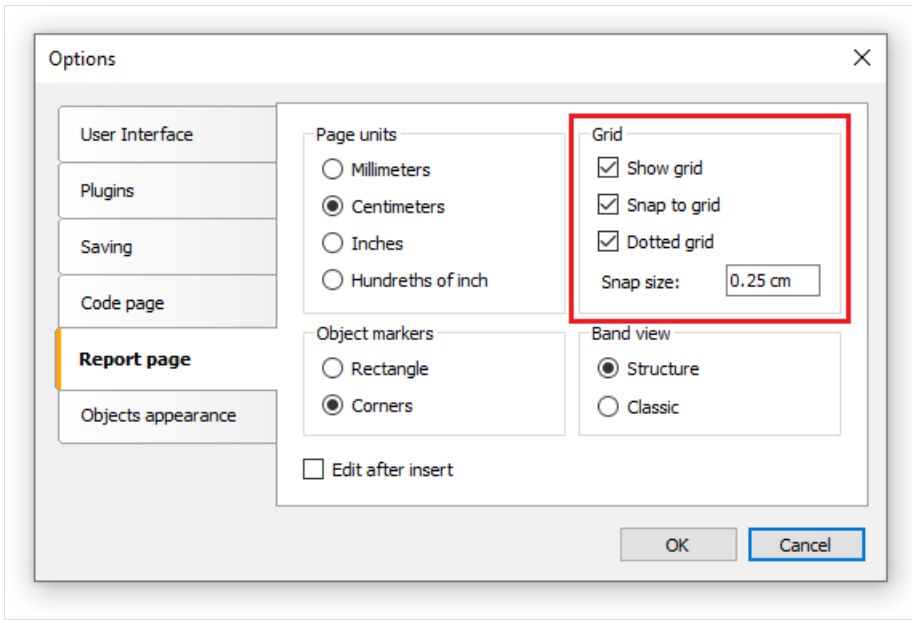
Bir formatta aktarım kalitesi, büyük ölçüde rapor şablonunun doğru bir şekilde hazırlanmasına bağlıdır. Aşağıdaki örneği ele alalım:



Bu raporun geliştirilmesi sırasında, aynı bantta yer alan iki nesnenin çakışmasına izin verilmiştir. Rapor oluşturulurken 150 kayıt olduğu varsayılırsa, RTF formatına aktarım sırasında çıktı tablosunda 450 satır oluşturulur (her bir nesne için 150 satır ve çakışma için 150 satır). Çakışma giderildiğinde, satır sayısı 300’e düşecektir. Rapor büyüdükçe ve nesne sayısı arttıkça, bu fark daha da büyür ve bu durum çıktı dosyasının boyutuna da doğrudan yansır.

Raporlarda tablo oluştururken, komşu hücrelerin sınırlarının birbirine değdiğinden emin olun, ancak birbirini geçmemeli veya üst üste binmemelidir. Aktarım filtresi algoritması hücreleri kesmeye çalışsa da, sonuç beklenenden farklı olabilir (görmek istediğinizin aynısını elde edemeyebilirsiniz). Nesneleri dikey ve yatay olarak aynı hizaya yerleştirmeye özen gösterin.

Hücrelerin çakışmasını önlemek için, nesnelerin ızgaraya hizalanması faydalı olabilir. Tasarımcı ayarlarında "ızgaraya hizalama" seçeneğinin etkin olduğundan emin olun. Hizalamayı kolaylaştırmak için ızgara aralığı artırılabilir. Izgara adımı ve hizalama ayarları, tasarımcı menüsünde "Görünüm | Ayarlar..." bölümünden yapılabilir:



Metinleri çerçevelemek için, ayrı grafik nesneleri (çizgiler, dikdörtgenler vb.) yerine "Metin" nesnesinin çerçeve özelliğini kullanmanız önerilir. Şeffaf metin nesneleri altında arka plan nesneleri kullanmaktan kaçının.

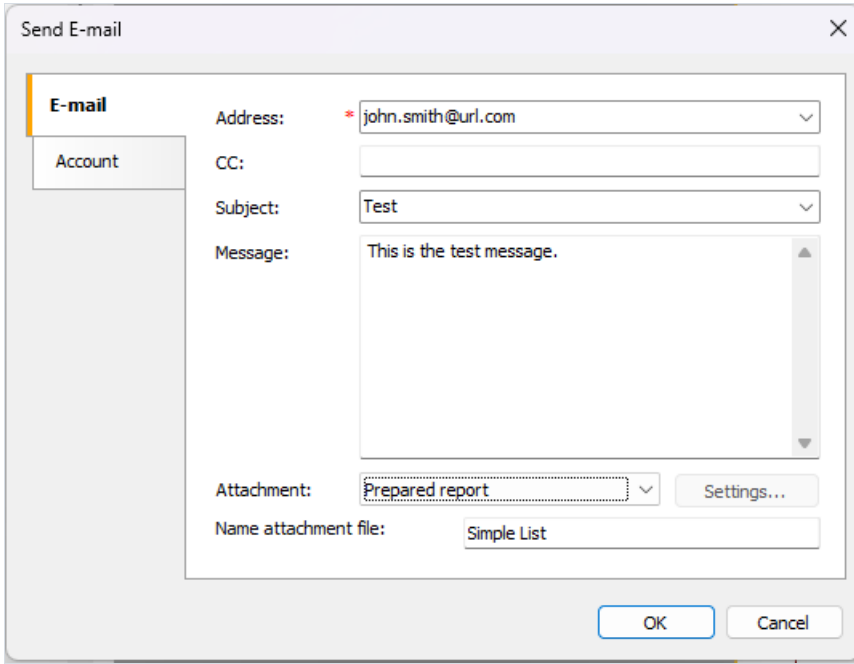
Bu basit kuralların uygulanması, verileri tabloya dayalı (veya tablo tabanlı) biçimlerde sunan tüm formatlara aktarıldığında raporun istenilen gibi görünmesini sağlar.

Raporu e-posta ile gönderme

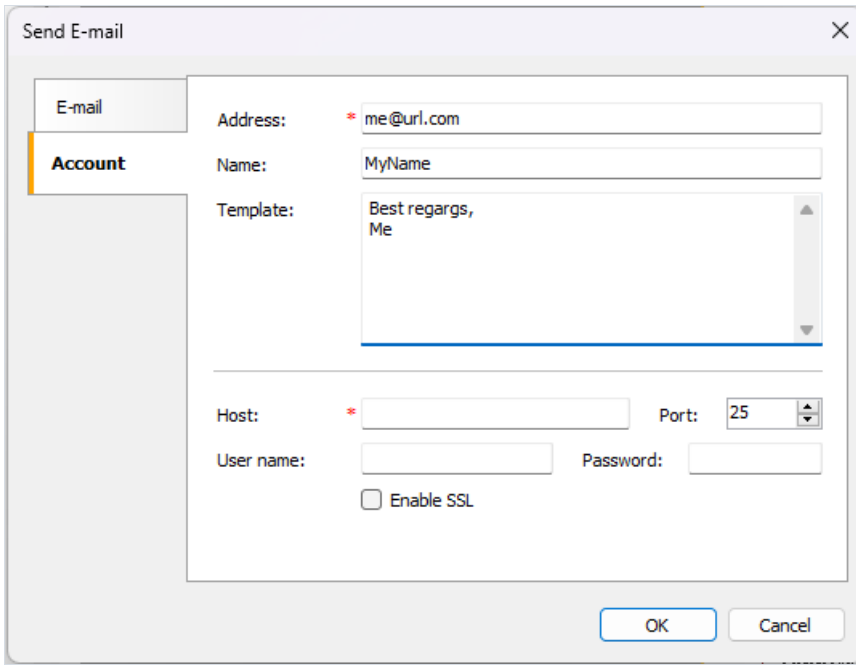
FastReport, oluşturulan raporu istediğiniz formatta e-posta ile göndermenize olanak tanır. Gönderme işlemi iki modda çalışır:

Mod	Açıklama
SMTP	Varsayılan çalışma kipidir. E-posta göndermek için herhangi bir yardımcı programa gerek yoktur.
MAPI	Bu çalışma kipi, programatik olarak <code>Config.EmailSettings.UseMAPI = true</code> özelliği belirtilerek ya da <code>EnvironmentSettings</code> bileşeni kullanılarak <code>EnvironmentSettings.EmailSettings.UseMAPI = true</code> özelliği ayarlanarak etkinleştirilebilir. \ E-posta göndermek için sistemde yüklü olan bir e-posta programı kullanılır. Bu programın MAPI ile uyumlu olması gerekir.

E-posta gönderebilmek için alıcının adresini belirtmeniz gerekir. Ayrıca, "Konu" ve "Mesaj" alanlarını doldurmanız önerilir ancak zorunlu değildir. Pencerenin alt kısmında, e-postaya eklenecek rapor dosyasının formatını seçin:

A screenshot of the 'Send E-mail' dialog box. The dialog has a title bar 'Send E-mail' with a close button. On the left, there is a sidebar with 'E-mail' and 'Account' tabs. The 'E-mail' tab is active. The main area contains fields for 'Address:' (with a red asterisk and the value 'john.smith@url.com'), 'CC:', 'Subject:' (with the value 'Test'), and 'Message:' (with the text 'This is the test message.'). Below these fields, there is an 'Attachment:' dropdown menu showing 'Prepared report' and a 'Settings...' button. At the bottom, there is a 'Name attachment file:' field with the value 'Simple List'. At the very bottom of the dialog are 'OK' and 'Cancel' buttons.

Eğer SMTP kipini kullanıyorsanız, e-posta gönderiminden önce e-posta hesabı ayarlarını yapılandırmanız gerekir. Bu işlem yalnızca bir kez yapılır. Tüm ayarlar FastReport yapılandırma dosyasında saklanır ve sonraki kullanımlarda otomatik olarak uygulanır. Ayarlar "Ayarlar" sekmesinde yer alır. Zorunlu alanlar kırmızı yıldızla işaretlenmiştir:

A screenshot of a 'Send E-mail' dialog box. The dialog has a title bar with a close button. On the left, there are two tabs: 'E-mail' and 'Account'. The 'Account' tab is selected. The main area contains several fields: 'Address:' with a red asterisk and the value 'me@url.com'; 'Name:' with the value 'MyName'; 'Template:' with a text area containing 'Best regargs, Me'; 'Host:' with a red asterisk and an empty field; 'Port:' with a dropdown menu showing '25'; 'User name:' with an empty field; 'Password:' with an empty field; and an 'Enable SSL' checkbox which is unchecked. At the bottom right, there are 'OK' and 'Cancel' buttons.

Send E-mail

E-mail

Account

Address: * me@url.com

Name: MyName

Template: Best regargs, Me

Host: * Port: 25

User name: Password:

☐ Enable SSL

OK Cancel

Eğer sunucunuz kimlik doğrulama (authentication) gerektiriyorsa, "Giriş" (Login) ve "Parola" (Password) alanlarını da doldurmanız gerekir.

Minimum system requirements

Minimum system requirements for installing and using FastReport .NET:

- Operating system MS Windows 7-11, Windows Server 2012-2019;
- CPU: 1 GHz;
- RAM: 512 MB;
- You also need the installed [.NET Framework version 4.6.2 or higher](#).

İletişim ve Teknik Destek

Ürün kullanımıyla ilgili sorularınızı her zaman [e-posta](#), veya sitedeki [formlar aracılığıyla](#) iletebilirsiniz.

Ürünümüzle ilgili önerilerinizi de memnuniyetle karşılarız.