



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

Sürüm 2026.1

© 2008-2025 Fast Reports Inc.

Genel bilgi

[Toolbox'a Kurulum](#)

[Sorun Giderme](#)

[Dağıtım](#)

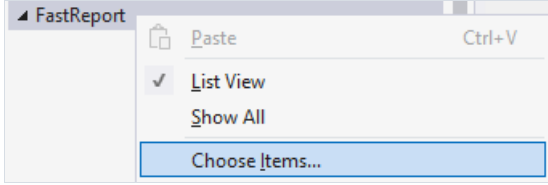
[Kaynak Kodun Derlenmesi](#)[Genel Bilgiler](#)

Toolbox'a Kurulum

FastReport.Net paketleriyle çalışırken, Toolbox bileşenleri pakete dahildir ve Visual Studio tarafından otomatik olarak yüklenmelidir. Toolbox'un manuel kurulumu da mümkündür.

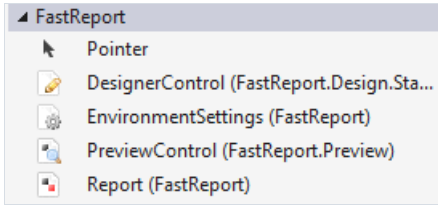
FastReport .NET bileşenlerini Visual Studio Toolbox'a manuel olarak eklemek için şu adımları izleyin:

- Toolbox'da "FastReport .NET" kategorisi varsa, silin.
- Yeni bir kategori oluşturun (bunun için Toolbox'a sağ tıklayıp "Add Tab" seçeneğini seçin) veya FastReport bileşenlerini eklemek istediğiniz mevcut bir kategoriye seçin.
- Seçtiğiniz kategoriye sağ tıklayın ve "Choose Items..." seçeneğini seçin.



- Açılan pencerede "Browse..." düğmesine tıklayın ve FastReport.dll ile FastReport.Web.dll dosyalarını seçin (bu dosyalar, FastReport .NET'in ana dizinindeki \Framework klasöründe yer alır, örneğin "C:\Program Files\FastReports\FastReport .NET Professional\Framework").
- Pencereyi OK düğmesine basarak kapatın.

Bu işlemin ardından, seçtiğiniz kategoriye aşağıdaki FastReport bileşenleri eklenecektir:



- Report
- PreviewControl
- EnvironmentSettings
- DesignerControl
- WebReport (bu bileşen yalnızca ASP.NET projesiyle çalışırken kullanılabilir).

Sorun Giderme

Tasarımcıda (örneğin, araç çubuklarının yerlerinin kayması gibi) yaşanan sorunlarda konfigürasyon dosyasının silinmesi gerekebilir. Bu dosya FastReport çalışırken oluşturulur ve aşağıdaki klasörlerde bulunur:

Windows XP:

```
C:\Documents and Settings\Kullanıcı_Adı\Local Settings\Application Data\FastReport
```

Windows Vista ve üzeri:

```
C:\Users\Kullanıcı_Adı\AppData\Local\FastReport
```

Dosya adı, kullanılan ürün türüne bağlıdır:

- FastReport .NET için: FastReport.config
- FastReport.Mono için: FastReport.Mono.config
- FastReport WPF için: FastReport.WPF.config

Konfigürasyon dosyasında saklanan bilgiler şunlardır:

- Arayüz pencerelerinin boyutları ve konumları;
- Araç çubuklarının ayarları;
- Son kullanılan veritabanı bağlantılarının parametreleri;
- (Rapor önizleme penceresinde "Email Gönder" işlevi kullanılıyorsa) Email ayarları.

Dağıtım

Uygulamanızla birlikte aşağıdaki dosyaları dağıtabilirsiniz:

- FastReport.dll (FastReport.Mono.dll, FastReport.WPF.dll) – FastReport’ın ana kütüphanesidir;
- FastReport.Web.dll – ASP.NET ortamında çalışmak için kullanılan, WebReport bileşenini içeren kütüphanedir;
- FastReport.Bars.dll – Yüzer pencereler, araç çubukları ve menülerin düzenlenmesi için kullanılan kütüphanedir (FastReport .NET’e aittir);
- FastReport.Editor.dll – Sözdizimi vurgulamalı kod düzenleyicidir. Uygulamanız rapor tasarımcısını kullanmıyorsa bu kütüphaneye gerek yoktur (FastReport .NET’e aittir);
- FastReport.WPF.RoslynPad.dll – .NET 6.0+ için sözdizimi vurgulamalı kod düzenleyicidir (FastReport WPF’e aittir);
- FastReport.Forms.WPF.dll – Form desteği sağlamak üzere kullanılan kütüphanedir (FastReport WPF’e aittir);
- FastReport.Forms.Avalonia.dll – Form desteği sağlamak üzere kullanılan kütüphanedir (FastReport Avalonia’ya aittir);
- FastReport.SkiaDrawing.dll – Skia desteğini sağlayan kütüphanedir;
- FastReport.Compat.dll (FastReport.Compat.Skia.dll) – Platformlar arası uyumluluğu sağlayan kütüphanedir;
- FastReport.DataVisualization.dll (FastReport.DataVisualization.Skia.dll) – Grafik (diagram) çizimi için kullanılan kütüphanedir;
- FastReport.xml (FastReport.Mono.xml, FastReport.WPF.xml) – FastReport’ın sınıflarına, özelliklerine ve metodlarına ilişkin açıklamaları içerir. Bu dosya kod düzenleyicide ve ipuçları panellerinde (örneğin, “Veriler” penceresinde bir işlev ya da “Özellikler” penceresinde herhangi bir özellik seçildiğinde) kullanılır. Bu dosyanın dağıtılması zorunlu değildir.

Ayrıca, raporlar dosyalarda saklanıyorsa (uygulama kaynaklarında değilse) rapor dosyalarını da dağıtmanız gerekmektedir.

FastReport NuGet Paketleri:

- **FastReport.Core (nuget.org’da demo)** – Uygulamanın temel işleyiş mantığını (gerekli verilerin alınması, raporların render edilmesi, ihracatlar vb.) içeren pakettir. FastReport.NET’te bulunan bazı işlevsellik, paketin çok platformlu olmasından dolayı mevcut değildir. .NET Framework 4.6.2 ve .NET 6 sürümlerinden itibaren çalışır.
- **FastReport.Core.Skia (nuget.org’da demo)** – SkiaDrawing için temel iş mantığını (gerekli verilerin alınması, raporların render edilmesi, ihracatlar vb.) içeren pakettir. .NET 6 sürümünden itibaren çalışır.
- **FastReport.Net (nuget.org’da demo)** – Sadece Windows ortamında, .NET Framework 4.6.2 ve .NET 6 ve üzeri sürümler için FastReport.dll kütüphanesini içeren pakettir. Bu paket farklı sürümlerde sunulmaktadır:
 - FastReport.Net.Demo – Bu paket, nuget.org ve resmi kurulum aracımızdaki deneme (Trial) sürümünde mevcuttur. Windows işletim sistemindeki farklı hedef platformlarda ürünün yeteneklerini değerlendirmeye olanak tanır. Paket içerisinde, Demo sürümüne özgü kısıtlamalar (rastgele metin parçalarının “Demo” ya da “Demo version” ile değiştirilmesi) bulunmaktadır.
 - FastReport.Net – Lisanslı FastReport .NET Standard sürümleri sahiplerine yönelik resmi kurulum aracımızda mevcuttur. Bu pakete, .NET Framework 4.6.2 ve .NET 6 ve üzeri için FastReport .NET dahildir.
 - FastReport.Net.Pro (legacy) – Bu paket, daha önce lisanslı FastReport .NET Professional ve Enterprise sürümleri sahiplerine yönelik resmi kurulum aracımızda sunulmaktaydı. .NET Framework 4.6.2 ve .NET 6 ve üzeri için FastReport .NET’i içeriyordu.
- **FastReport.WPF (nuget.org’da demo)** – .NET Framework 4.6.2 ve .NET 6 için FastReport WPF

kütüphanesini içeren pakettir.

- **FastReport.Avalonia** ([nuget.org'da demo](https://www.nuget.org/packages/FastReport.Avalonia)) – .NET 6 ve üzeri için FastReport Avalonia kütüphanesini içeren pakettir.
- **FastReport.Web** [([nuget.org'da demo](https://www.nuget.org/packages/FastReport.Web))]<https://www.nuget.org/packages/FastReport.Web>) – ASP.NET Core ortamında web uygulamalarında FastReport entegrasyon senaryoları (tarayıcıda rapor render etme, ihracat ve tarayıcıdan baskı, Online Designer ile çalışma) için kullanılan pakettir. Blazor Server bileşenlerini içerir ve yalnızca FastReport.Core ile birlikte kullanılır. .NET 6 sürümünden itibaren çalışır.
- **FastReport.Web.Skia** ([nuget.org'da demo](https://www.nuget.org/packages/FastReport.Web.Skia)) – ASP.NET Core ortamında SkiaDrawing ile web uygulamaları için FastReport entegrasyonu (tarayıcıda rapor render etme, ihracat ve tarayıcıdan baskı, Online Designer ile çalışma) senaryolarını destekleyen pakettir. Blazor Server bileşenlerini içerir ve yalnızca FastReport.Core ile birlikte kullanılır. .NET 6 sürümünden itibaren çalışır.
- **FastReport.Core3.Web** ([nuget.org'da demo](https://www.nuget.org/packages/FastReport.Core3.Web)) – ASP.NET Core ortamında web uygulamalarında FastReport entegrasyonu (tarayıcıda rapor render etme, ihracat ve tarayıcıdan baskı, Online Designer ile çalışma) senaryoları için kullanılan pakettir. Blazor Server bileşenlerini içerir ve sadece .NET 6 ve üzeri için FastReport.NET ile birlikte kullanılır.
- **FastReport.Blazor.Wasm** ([nuget.org'da demo](https://www.nuget.org/packages/FastReport.Blazor.Wasm)) – Blazor WebAssembly için Razor bileşenlerini içeren pakettir. .NET 6 sürümünden itibaren çalışır.
- **FastReport.Localization** ([nuget.org](https://www.nuget.org/packages/FastReport.Localization)) – FastReport'in yerelleştirmelerini içeren pakettir. Örneğin, Almanca veya Rusça yerelleştirme gerekiyorsa projenize ekleyin.
- **FastReport.Compat** ve **FastReport.DataVisualization** – Rapor derlemesi, MSChart desteği vb. temel iş mantığını içeren paketlerdir. Bu paketleri projenize eklemeniz gerekmez; bağımlılık paketleri olarak kullanılırlar.
- **FastReport.Data**. – FastReport'in, temel kütüphaneye dahil edilmeyen farklı veritabanları ile çalışabilmesi için kullanılan eklenti-bağlayıcı paketlerdir. Bu paketler, FastReport .NET'in (WinForms, Mono, WPF, Avalonia, ASP.NET) tüm bileşen setleri için ortak paketlerdir. Kısıtlamalar: FastReport 2021.4.0+ sürümü ve NuGet Client 3.4.4+ gereklidir.
 - [FastReport.Data.ClickHouse](https://www.nuget.org/packages/FastReport.Data.ClickHouse)
 - [FastReport.Data.Couchbase](https://www.nuget.org/packages/FastReport.Data.Couchbase)
 - [FastReport.Data.Firebird](https://www.nuget.org/packages/FastReport.Data.Firebird)
 - [FastReport.Data.Json](https://www.nuget.org/packages/FastReport.Data.Json)
 - [FastReport.Data.MongoDB](https://www.nuget.org/packages/FastReport.Data.MongoDB)
 - [FastReport.Data.MsSql](https://www.nuget.org/packages/FastReport.Data.MsSql)
 - [FastReport.Data.MySql](https://www.nuget.org/packages/FastReport.Data.MySql)
 - [FastReport.Data.OracleODPCore](https://www.nuget.org/packages/FastReport.Data.OracleODPCore)
 - [FastReport.Data.Postgres](https://www.nuget.org/packages/FastReport.Data.Postgres)
 - [FastReport.Data.RavenDB](https://www.nuget.org/packages/FastReport.Data.RavenDB)
 - [FastReport.Data.SQLite](https://www.nuget.org/packages/FastReport.Data.SQLite)

Kaynak Kodun Derlenmesi

Professional ve Enterprise sürümlerine, FastReport kütüphanelerinin kaynak kodları dahildir. Bunları uygulamanıza dahil edebilirsiniz. Aşağıda, FastReport .NET projesi örneğinde bunun nasıl yapılacağını gösteriyoruz:

- Visual Studio'da projenizi açın;
- Solution Explorer penceresini açın ve "Solution" ögesi üzerinde sağ tıklayın;
- "Add|Existing Project..." menü ögesini seçin;
- `FastReport.csproj` proje dosyasını ekleyin (bu dosya, FastReport .NET ana dizinindeki \Sources\FastReport klasöründe bulunur, örneğin: "C:\Program Files\FastReports\FastReport .NET Professional\Sources\FastReport").

Diğer FastReport .NET derlemelerine olan referansları güncelleyin. Bunun için:

- Solution Explorer penceresinde "FastReport|References" ögesini genişletin;
- `FastReport.Bars` ve `FastReport.Editor` referanslarını kaldırın;
- "References" ögesi üzerinde sağ tıklayın;
- "Add Reference..." menü ögesini seçin;
- `FastReport.Bars.dll` ve `FastReport.Editor.dll` derlemelerine olan referansları ekleyin. Bu dosyalar, FastReport .NET ana dizinindeki \Framework klasöründe bulunur, örneğin: "C:\Program Files\FastReports\FastReport .NET Professional\Framework".

Kaynak koddan FastReport WPF derlemesi yapmak için "C:\Program Files\FastReports\FastReport WPF\Sources\FastReport.WPF\FastReport.WPF.sln" dosyasını açın. Çözüm, tüm FastReport WPF projelerini ve demo uygulamalarını (Samples) içerir.

Windows.Forms, WPF, Avalonia ile Çalışma

Visual Studio'da Report Bileşeninin Kullanımı

Kod İçinde Raporla Çalışma

Raporun Kaydedilmesi ve Yüklenmesi

Verilerin Kaydı

Rapora Parametre Aktarımı

Raporun Çalıştırılması

Tasarımcının Çalıştırılması

Raporun Dışa Aktarılması

FastReport .NET Ortamının Yapılandırılması

"Aç" ve "Kaydet" Diyaloglarının Değiştirilmesi

Standart İlerleme Penceresinin Değiştirilmesi

Bağlantı Dizesinin Rapora Aktarılması

SQL Metninin Rapora Aktarılması

Rapor Nesnelere Erişim

Kod ile Rapor Oluşturma

Özel Önzileme Penceresinin Kullanılması

"Bağlantı Sihirbazı"nda Tablo Listesinin Filtrelenmesi

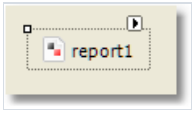
Visual Studio (WinForms)'da Report Bileşeninin Kullanımı

Tip belirlenmiş veri kaynağından veri alındığı tipik bir senaryoyu ele alalım.

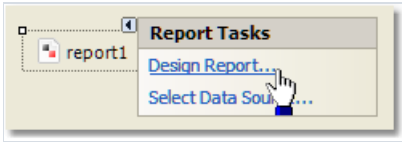
1. Windows Forms Application projesi oluşturun;
2. Projeye bir veri kaynağı ekleyin (menüden "Data | Add New Data Source..." seçeneğini kullanarak);
3. Form tasarımcısına geçin;
4. Forma, "Data" kategorisinde bulunan DataSet bileşenini ekleyin ve daha önce oluşturduğunuz tip belirlenmiş veri kaynağına bağlayın.

Rapor oluşturmak için aşağıdaki adımları izleyin:

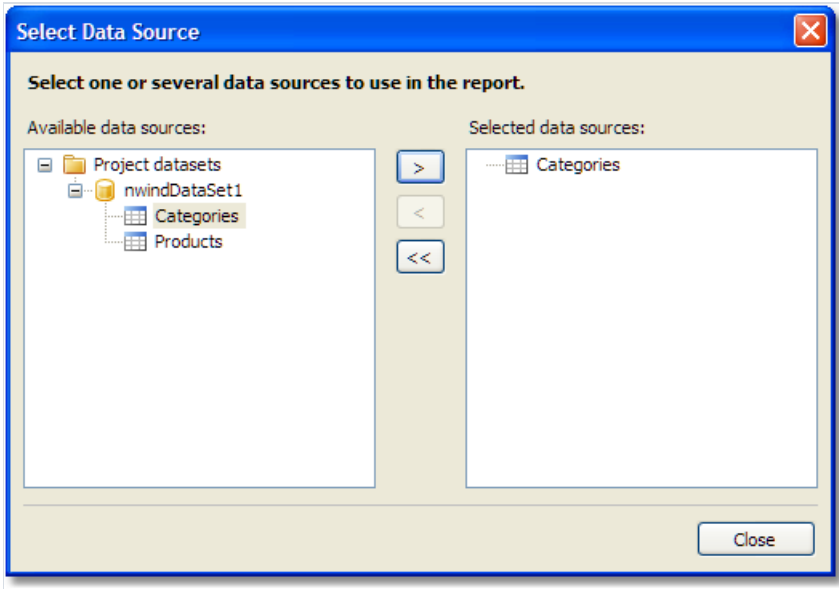
- Forma Report bileşenini ekleyin;



- Bileşene sağ tıklayın veya "smart tag" düğmesine basın ve "Design Report..." seçeneğini seçin;



- Rapor için veri kaynağını seçmeniz istenecek. İhtiyaç duyduğunuz tabloyu seçin (tek veya birden fazla);



- Raporu oluşturun. Bu konu hakkında daha fazla bilgiyi "Kullanıcı Kılavuzu"ndan okuyabilirsiniz;
- Rapor tasarımcısını kapatın;
- Forma bir Button (Düğme) ekleyin ve üzerine çift tıklayın;
- Olay işleyicisinde şu kodu yazın:

```
report1.Show();
```

- Projeyi kaydedin ve alıřtırın. Butona basıldığında rapor oluşturulacaktır.

Kod içinde rapor ile çalışma

Kod içerisinde rapor ile çalışmak için aşağıdaki adımları izlemeniz gerekmektedir:

- Rapor örneği oluşturun: Öncelikle, rapor sınıfından bir nesne türetmelisiniz.
- Rapor dosyasını yükleyin: Oluşturduğunuz rapor örneğine, rapor dosyasını (örneğin, "report1.frx") yükleyin.
- Verileri kaydedin: Uygulamada tanımlı verileri rapora kaydedin (örneğin, dataSet1 verilerini "NorthWind" adıyla).
- Parametre değerlerini aktarın (gerekirse): Rapor çalıştırılırken ihtiyaç duyulabilecek parametrelerin değerlerini rapora geçebilirsiniz.
- Raporu çalıştırın: Tüm işlemler tamamlandığında, raporu çalıştırarak sonuçları görüntüleyin.

Aşağıdaki örnek kod parçası bu adımları göstermektedir:

```
using (Report report = new Report())
{
    report.Load("report1.frx");
    report.RegisterData(dataSet1, "NorthWind");
    report.Show();
}
```

Şimdi, bu adımların her birini daha detaylı inceleyeceğiz.

Raporun depolanması ve yüklenmesi

Raporu aşağıdaki yöntemlerden biriyle saklayabilirsiniz:

Yöntem	Açıklama
Uygulamanın kaynaklarında (WinForms/Mono)	Yukarıda ele aldığımız rapor çalışma senaryosu tipik olarak bu yöntemi kullanır. Bunun sorumlusu, varsayılan olarak true olan Report nesnesinin <code>StoreInResources</code> özelliğidir. Bu yöntemin aşağıdaki artıları ve eksileri vardır: + Rapor uygulamaya gömülüdür, ek dosyaların dağıtılmasına gerek yoktur. - Raporu değiştirmek gerekirse, uygulamayı yeniden derlemek gerekir. Rapor yüklemesi, FastReport .NET'in formunuzun <code>InitializeComponent</code> metoduna eklediği kod aracılığıyla otomatik olarak gerçekleştirilir.
.FRX dosyasında	Kullanıcıların rapor dosyalarını düzenlemesine izin vermek istiyorsanız bu en uygun yöntemdir. Bu durumda, <code>StoreInResources</code> özelliğini false olarak ayarlayın. Dosyadan rapor yüklemek için Report nesnesinin <code>Load</code> metodunu kullanın: <pre>report1.Load("filename.frx");</pre>
Veritabanında	Rapor, veritabanındaki bir alanda dize veya ikili akış olarak saklanabilir. Dizeden raporu yüklemek için Report nesnesinin <code>LoadFromString</code> metodunu kullanın. Akıştan yüklemek için, aşırı yüklenmiş Load metodunu kullanın:<pre>report1.Load(stream);</pre> Bu yöntemi rapor tasarımcısında desteklemek için, "Dosya Aç" ve "Dosya Kaydet" işlemlerine verilen tepkiyi geçersiz kılmamız gerekir. Bu işlemin nasıl yapılacağı aşağıda açıklanmıştır.

Yöntem**Açıklama****C#/VB.NET sınıfı olarak**

Report sınıfı şeklinde implementasyon yapılabilir. Rapor tasarımcısında "Dosya

Farklı Kaydet..." seçeneğini seçin ve dosya türü olarak "C# Dosyası" veya "VB.NET Dosyası"nı seçin (bu, raporda seçili olan dile bağlıdır - bkz. "Kullanıcı Kılavuzu"). Oluşan dosyayı projenize ekleyebilirsiniz. Bu yöntemin artıları ve eksileri aşağıdaki gibidir: Raporla sıradan bir sınıf gibi çalışabilirsiniz; Visual Studio'nun tüm özelliklerine, hata ayıklama da dahil, erişebilirsiniz; Bu, ASP.NET'te Medium Trust moduyla uyumlu raporla çalışmanın tek yoludur; Böyle bir raporu düzenleyemezsiniz. Düzenleme için .FRX formatındaki orijinal rapor dosyasına ihtiyaç vardır; Raporu değiştirmek gerektiğinde, uygulamayı yeniden derlemeniz gerekir. Raporla çalışmak için rapor sınıfının bir örneğini oluşturun, örneğin:

```
SimpleListReport report = new SimpleListReport();  
report.Show();
```

Veri Kaydı

Eğer raporunuz uygulamadan veri kullanıyorsa (örneğin, tiplenmiş veri kaynağı veya iş nesnesi), bu verilerin rapora kaydedilmesi gerekmektedir. Bu işlem, Report nesnesinin `RegisterData` yöntemini kullanarak yapılır.

Burada incelenen tipik rapor çalışma senaryosunda verileri manuel olarak kaydetmeniz gerekmez. FastReport, formunuzun `InitializeComponent` metodunun koduna `RegisterData` yönteminin çağrısını otomatik olarak ekler.

`RegisterData` yöntemi, rapor yüklendikten sonra çağrılmalıdır:

```
report1 = new Report();
report1.Load("report.frx");
report1.RegisterData(dataSet1, "NorthWind");
```

`RegisterData` yöntemi aşırı yüklenmiştir ve aşağıdaki veri türlerinin kaydedilmesine olanak tanır:

Yöntem	Açıklama	
<code>void RegisterData(DataSet data)</code>	Veri Kaynağı Kaydı. Bu yöntem, veri kaynağındaki tüm tabloları, görünüşleri ve ilişkileri kaydeder. Bu yöntemin çağırılması, <code>RegisterData(data, "Data")</code> metodunun çağırılmasına eşdeğerdir. Dikkat: Birden fazla veri kaynağını kaydediyorsanız, <code>RegisterData(DataSet data, string name)</code> sürümünü kullanın.	
<code>void RegisterData(DataSet data, string name)</code>	Veri Kaynağı Kaydı. Bu yöntem, veri kaynağında bulunan tüm tabloları, görünüşleri ve ilişkileri kaydeder. <code>name</code> parametresinde herhangi bir ad belirtilebilir; ancak, bu adın değişmemesi ve benzersiz olması önemlidir (birden fazla veri kaynağı kaydediliyorsa). <code>name</code> adı hiçbir yerde görüntülenmez, ancak rapor nesnelerinin veri kümesine bağlanması için kullanılır.	
<code>void RegisterData(DataTable data, string name)</code>	Tablo Kaydı.	
<code>void RegisterData(DataView data, string name)</code>	Görünüm Kaydı.	
<code>void RegisterDataAsp(IDataSource data, string name)</code>	ASP.NET'te çalışırken veri kaynağı kaydı.	
<code>void RegisterData(DataRelation data, string name)</code>	Bağlantı kaydı.	
<code>void RegisterData(IEnumerable data, string name, BOConverterFlags flags, int maxNestingLevel)</code>	<code>MaxNestingLevel</code> başlangıç iç içe geçme seviyesi ile iş nesneleri ağacının kaydı. "Veriler	Rapor için verileri seç..." menüsünü kullanarak, iş nesnesi yapısını gerekli iç içe geçme derinliğine kadar açabilirsiniz.

Rapor Parametresinin Aktarılması

Rapor içinde parametreler tanımlanabilir. Konuyla ilgili daha fazla bilgi için "Kullanıcı Kılavuzu"na bakabilirsiniz. Parametre değerini aktarmak için Report nesnesinin `SetParameterValue` metodu kullanılır:

```
report1.Load("report.frx");  
report1.SetParameterValue("MyParam", 10);  
report1.Show();
```

Metot aşağıdaki şekilde tanımlanmıştır:

```
public void SetParameterValue(string complexName, object value)
```

`complexName` parametresi, parametrenin adını belirtmek içindir. İç içe geçmiş bir parametreye erişmek için, örneğin:

```
ParentParameter.ChildParameter
```

kullanarak tam adını belirtin.

Bir rapor alıřtırma

Raporu bařlatmak iin Report nesnesinin ařağıdaki yntemlerinden birini kullanın:

Yntem	Aıklama
<code>void Show()</code>	Raporu oluřturur ve nizleme penceresinde gsterir. Bu yntem ařağıdaki ile eřdeęerdir: <pre>if (Prepare()) ShowPrepared();</pre>
<code>bool Prepare()</code>	Raporu alıřtırır. Eęer rapor bařarıyla oluřturulursa, <code>true</code> dndrr. Bu yntemin ardından, <code>ShowPrepared</code> , <code>PrintPrepared</code> , <code>SavePrepared</code> veya <code>Export</code> yntemlerinden biri aęırılmalıdır: <pre>if (Prepare()) ShowPrepared();</pre>
<code>bool Prepare(bool append)</code>	Raporu alıřtırır. Eęer append parametresi <code>true</code> ise, mevcut rapora hazır rapor eklenir. Bu řekilde, birden fazla rapor oluřturulup nizleme penceresinde tek bir rapor olarak gsterilebilir: <pre>report1.Load("report1.frx"); report1.Prepare(); report1.Load("report2.frx"); report1.Prepare(true); report.ShowPrepared();</pre>
<code>void ShowPrepared()</code>	nizleme penceresinde hazırlanan raporu gsterir. Rapor, <code>Prepare</code> yntemi ile hazırlanmıř veya <code>LoadPrepared</code> yntemi kullanılarak .FPX dosyasından yklenmiř olmalıdır: <pre>if (Prepare()) ShowPrepared();</pre>
<code>void ShowPrepared(bool modal)</code>	nizleme penceresinde hazırlanan raporu gsterir. <code>modal</code> parametresi, pencerenin modal olarak gsterilip gsterilmeyeceęini belirler.
<code>void ShowPrepared(bool modal, Form owner)</code>	nceki yntemle aynıdır. <code>owner</code> parametresi, nizleme penceresinin sahibi olan pencereyi belirtir.
<code>void ShowPrepared(Form mdiParent)</code>	nceki yntemle aynı. <code>mdiParent</code> parametresi, MDI ana penceresini belirler.

Dizayncıyı Başlatma

FastReport'ın tüm sürümlerinde, programınızdan rapor tasarımcısını çağırma imkanı bulunmaktadır. Bunu yapmak için, Report nesnesinin `Design` metodunu kullanın:

```
report1 = new Report();  
report1.Load("report1.frx");  
report1.Design();
```

`Design` metodu aşırı yüklenmiştir:

Metod	Açıklama
<code>bool Design()</code>	Tasarımcı Çağrısı.
<code>bool Design(bool modal)</code>	Aynı. <code>modal</code> parametresi, tasarımcı penceresinin modal olarak gösterilip gösterilmeyeceğini belirler.
<code>bool Design(Form mdiParent)</code>	Aynı. <code>mdiParent</code> parametresi, WinForms/Mono kapsamında ana MDI penceresini belirler.
<code>async Task<bool> DesignAsync()</code>	Aynı. Tasarımcıyı asenkron olarak çağırmak, Avalonia'ya özgüdür.

Raporun Dışa Aktarımı

Hazır rapor, diğer formatlara dışa aktarılabilir. Şu anda aşağıdaki formatlar desteklenmektedir:

- PDF
- Excel 2007
- Word 2007
- PowerPoint 2007
- OpenOffice Calc / OpenOffice Writer
- RTF
- Excel XML (Excel 2003+)
- HTML
- MHT
- Görüntüler (Bmp, Png, Jpeg, Gif, Tiff, Metafile)
- SVG
- CSV
- DBF
- Json
- TXT
- ZPL
- PPML
- PostScript
- XPS
- LaTeX
- DXF
- XAML

Raporun dışa aktarılması, dışa aktarım filtresi kullanılarak yapılır. Bunun için şu adımlar izlenir:

- Raporu `Prepare` metodu ile oluşturun.
- Dışa aktarım filtresinin bir örneğini oluşturun ve özelliklerini ayarlayın.
- `Export` metodunu, rapor örneği ve dosya adı parametreleriyle çağırın.

Aşağıdaki örnek, HTML formatında rapor dışa aktarmayı göstermektedir:

```
// Raporu hazırlıyoruz
report1.Prepare();
// HTML formatında dışa aktarım örneğini oluşturuyoruz
FastReport.Export.Html.HTMLExport export = new FastReport.Export.Html.HTMLExport();
// Dışa aktarım ayarları diyalogunu gösteriyor ve raporu dışa aktarıyoruz
if (export.ShowDialog())
    export.Export(report1, @"C:\result.html");
```

Bu örnekte, dışa aktarım ayarları diyalog aracılığıyla yapılandırılmaktadır.

FastReport ortam yapılandırması

`FastReport.Utills.Config` global sınıfını kullanarak, FastReport ortamının bazı ayarlarını değiştirebilirsiniz.

Raporla ilgili ayarlar, `Config.ReportSettings` özelliğinde mevcuttur:

Özellik	Açıklama
<code>Language</code> <code>DefaultLanguage</code>	Varsayılan betik dili. Tüm oluşturulan raporlar için kullanılacaktır.
<code>bool ShowProgress</code>	İlerleme penceresinin gösterilip gösterilmeyeceğini belirler.
<code>bool ShowPerformance</code>	Rapor oluşturma süresi ve kullanılan bellek gibi performans bilgilerini önizleme penceresinde gösterip göstermeyeceğini belirler.

Tasarımcıya ilişkin ayarlar, `Config.DesignerSettings` özelliğinde mevcuttur.

Özellik	Açıklama
<code>Icon Icon</code>	Tasarımcı penceresinin simgesi.
<code>Font DefaultFont</code>	Varsayılan yazı tipi. Yeni "Metin" nesneleri için kullanılacaktır.

Önizleme penceresi ile ilgili ayarlar, `Config.PreviewSettings` özelliğinde mevcuttur.

Özellik	Açıklama
<code>PreviewButtons</code> <code>Buttons</code>	Önizleme penceresinde görünecek buton seti.
<code>int PagesInCache</code>	RAM önbelleğinde saklanan hazır raporun maksimum sayfa sayısı.
<code>bool ShowInTaskbar</code>	Görev çubuğunda pencerenin gösterilip gösterilmeyeceğini belirler.
<code>bool TopMost</code>	Diğer tüm pencerelerin üzerinde pencere gösterilip gösterilmeyeceğini belirler.
<code>Icon Icon</code>	Önizleme penceresinin simgesi.
<code>string Text</code>	Pencere başlığındaki metin. Bu özellik boş bırakılırsa, "RaporAdı - Önizleme" metni gösterilecektir.

Pencere başlığındaki metin. Bu özellik boş bırakılırsa, "RaporAdı - Önizleme" metni gösterilecektir.

Özellik	Açıklama
<code>string Address</code>	Gönderen adresi (sizin adresiniz).
<code>string Name</code>	Gönderen adı (sizin adınız).

Özellik	Açıklama
<code>string MessageTemplate</code>	Yeni e-posta oluşturulurken kullanılacak olan e-posta şablonu. Örneğin: "Merhaba, Saygılarımla, ..."
<code>string Host</code>	SMTP posta sunucusu adresi.
<code>int Port</code>	SMTP portu (genellikle 25).
<code>string UserName,</code> <code>string Password</code>	Kullanıcı adı ve şifre. Eğer posta sunucusu kimlik doğrulaması gerektirmiyorsa, bu özellikleri boş bırakın.
<code>bool AllowUI</code>	Mesaj gönderme penceresinde bu ayarları değiştirmeye izin verir. Ayarlar FastReport yapılandırma dosyasında saklanacaktır.

Arayüz ayarları, aşağıdaki özelliklerde mevcuttur.

Özellik	Açıklama
<code>UIStyle UIStyle</code>	Tasarımcı ve önizleme pencerelerinin arayüz stili.

Özelliklerin dışında, tasarımcıdaki "Aç" ve "Kaydet" diyaloglarını değiştirmeyi, ilerleme penceresini yenilemeyi ve rapora bağlantı dizesi aktarmayı sağlayan birkaç olay bulunmaktadır. Bu işlemler daha sonra açıklanacaktır.

"Aç" ve "Kaydet" Diyaloglarının Değiştirilmesi

Eğer raporları veritabanında saklamaya karar verdiyseniz, rapor tasarımcısının standart "Dosya Aç" ve "Dosya Kaydet" diyalogları yerine, veritabanı ile çalışan kendi diyaloglarınızı kullanacak şekilde değiştirilmesi gerekebilir. Bu amaçla, FastReport ortam yapılandırması bölümünde açıklanan `FastReport.Utills.Config.DesignerSettings` global sınıfını kullanabilirsiniz. Bu sınıf aşağıdaki olaylara sahiptir:

`CustomOpenDialog` Olayı

Bu olay, "Dosya Aç" diyalogu gösterildiğinde tetiklenir. Olay işleyicisinde kendi diyalogunuzu göstermelisiniz. Eğer diyalog başarılı bir şekilde çalıştırıldıysa, `e.Cancel = false` olarak ayarlamalı, Seçilen dosya adını `e.FileName` parametresine döndürmelisiniz. Aşağıdaki örnek, standart dosya açma diyalogunun kullanımını göstermektedir:

```
private void CustomOpenDialog_Handler(  
    object sender, OpenSaveDialogEventArgs e)  
{  
    using (OpenFileDialog dialog = new OpenFileDialog())  
    {  
        dialog.Filter = "Rapor dosyaları (*.frx)|*.frx";  
        // Diyalog başarıyla tamamlandıysa, e.Cancel değerini false yapın  
        e.Cancel = dialog.ShowDialog() != DialogResult.OK;  
        // Seçilen dosya adını e.FileName'e atayın  
        e.FileName = dialog.FileName;  
    }  
}
```

`CustomSaveDialog` Olayı

Bu olay, "Dosya Kaydet" diyalogu gösterildiğinde tetiklenir. Olay işleyicisinde kendi diyalogunuzu göstermelisiniz. Eğer diyalog başarılı bir şekilde tamamlanırsa, `e.Cancel = false` olarak ayarlamalı, Seçilen dosya adını `e.FileName` parametresine döndürmelisiniz. Aşağıdaki örnek, standart dosya kaydetme diyalogunun kullanımını göstermektedir:

```
private void CustomSaveDialog_Handler(  
    object sender, OpenSaveDialogEventArgs e)  
{  
    using (SaveFileDialog dialog = new SaveFileDialog())  
    {  
        dialog.Filter = "Rapor dosyaları (*.frx)|*.frx";  
        // e.FileName'den varsayılan dosya adını al  
        dialog.FileName = e.FileName;  
        // Diyalog başarıyla tamamlandıysa, e.Cancel değerini false yapın  
        e.Cancel = dialog.ShowDialog() != DialogResult.OK;  
        // Seçilen dosya adını e.FileName'e atayın  
        e.FileName = dialog.FileName;  
    }  
}
```

`CustomOpenReport` Olayı

Bu olay, rapor okunurken tetiklenir. Olay işleyicisinde, `e.FileName` parametresinde belirtilen yerden raporu (`e.Report`) yüklemelisiniz. `e.FileName` `CustomOpenDialog` olay işleyicisinden dönen değeri içerir. Bu değer dosya adı, veritabanı kaydı adı vb. olabilir. Aşağıdaki örnek, bir dosyadan raporun nasıl yükleneceğini göstermektedir:

```
private void CustomOpenReport_Handler(object sender, OpenSaveReportEventArgs e)
{
    // Verilen e.FileName'den raporu yükleyin
    e.Report.Load(e.FileName);
}
```

CustomSaveReport Olayı

Bu olay, rapor kaydedilirken tetiklenir. Olay işleyicisinde, `e.FileName` parametresinde belirtilen yere raporu (`e.Report`) kaydetmelisiniz. `e.FileName` . `CustomSaveDialog` olay işleyicisinden dönen değeri içerir. Bu değer dosya adı, veritabanı kaydı adı vb. olabilir. Aşağıdaki örnek, raporun dosyaya nasıl kaydedileceğini göstermektedir:

```
private void CustomSaveReport_Handler(object sender, OpenSaveReportEventArgs e)
{
    // Verilen e.FileName'e raporu kaydedin
    e.Report.Save(e.FileName);
}
```

Standart İlerleme Penceresinin Değiştirilmesi

İlerleme penceresi, aşağıdaki işlemler sırasında gösterilir:

- Rapor oluşturma;
- Yazdırma;
- Dışa aktarma.

`FastReport.Utils.Config.ReportSettings.ShowProgress` özelliğinin değeri `false` olarak ayarlanarak ilerleme penceresini devre dışı bırakabilirsiniz. Ayrıca, kendi özel ilerleme pencerenizi de entegre edebilirsiniz. Bunun için, `FastReport.Utils.Config.ReportSettings` sınıfında tanımlı olan (bkz. FastReport Ortam Yapılandırması bölümü) aşağıdaki olayları kullanın.

Olay	Açıklama
<code>StartProgress</code>	İşlem başlamadan önce bir kez çağrılır. Bu olayda, kendi ilerleme pencerenizi oluşturmanız ve göstermeniz gerekir.
<code>Progress</code>	Her rapor sayfasının işlenmesinden sonra çağrılır. Bu olayda, mevcut ilerleme durumunu göstermeniz gerekir.
<code>FinishProgress</code>	İşlem tamamlandıktan sonra bir kez çağrılır. Bu olayda, kendi ilerleme pencerenizi yok etmeniz gerekir.

`Progress` olayına `ProgressEventArgs` tipinde `e` parametresi iletilir. Bu parametrenin aşağıdaki özellikleri vardır.

Tip, Özellik	Açıklama
<code>string Message</code>	Mesaj metni.
<code>int Progress</code>	İşlenen mevcut sayfa numarası.
<code>int Total</code>	Toplam sayfa sayısı. Bu parametre, rapor oluşturulurken 0 olabilir; çünkü bu durumda toplam sayfa sayısı bilinmemektedir.

Genellikle, `Progress` olayının işleyicisinde, `e.Message` parametresiyle iletilen metni kendi ilerleme penceresinde göstermek yeterlidir.

Rapor'a Bağlantı Dizesi Geçirme

Eğer raporunuz kendi veri kaynaklarını kullanıyorsa, rapora bağlantı dizesi geçirmeniz gerekebilir. Bu işlem üç yöntemle yapılabilir.

Birinci Yöntem: Bağlantı dizesi, rapordaki Connection nesnesine doğrudan atanır:

```
report1.Load(...);  
// Rapor yüklendikten sonra, rapor oluşturulmadan önce yapılır  
// Raporun bir bağlantısı olduğu varsayılmaktadır  
report1.Dictionary.Connections[0].ConnectionString = my_connection_string;  
report1.Show();
```

İkinci Yöntem: Bağlantı dizesi, rapor parametresi aracılığıyla iletilir. Bunun için şunları yapın:

- Rapor tasarımcısını başlatın;
- Veri penceresinde, bir rapor parametresi oluşturun (örneğin, "MyParameter" adında). Detaylar için Kullanıcı Kılavuzu'na bakınız;
- Veri penceresinde "Bağlantı" nesnesini seçin;
- Özellikler penceresine geçin ve `ConnectionStringExpression` özelliğinin değerini şu şekilde ayarlayın:

```
[MyParameter]
```

- Rapore, MyParameter parametresine bağlantı dizesini içeren değeri geçirin:

```
report1.SetParameterValue("MyParameter", my_connection_string);
```

- Üçüncü Yöntem: `FastReport.Utils.Config.ReportSettings` global sınıfındaki `DatabaseLogin` olayını kullanın (bakınız: FastReport ortam yapılandırması bölümüne). Bu olay, veritabanına her bağlantı açıldığında çağrılır. İşte bir olay işleyicisi örneği:

```
private void environmentSettings1_DatabaseLogin(  
    object sender, DatabaseLoginEventArgs e)  
{  
    e.ConnectionString = my_connection_string;  
}
```

Unutmayın, `DatabaseLogin` olayı tüm raporlar için global olarak çalışır.

SQL Metnini Rapor'a Aktarma

Rapor, "Veri Bağlantısı Sihirbazı"nda eklenen veri kaynaklarını içerebilir. Uygulama tarafından oluşturulan SQL metnini bu veri kaynağına aktarmanız gerektiğinde aşağıdaki kodu kullanabilirsiniz:

```
using FastReport.Data;
report1.Load(...);
// Rapor yüklendikten, rapor oluşturulmadan hemen sonra yapılır
// ilgili tabloyu adı (takma adı) ile buluyoruz
TableDataSource table = report1.GetDataSource("MyTable") as TableDataSource;
table.SelectCommand = "yeni SQL metni";
report1.Show();
```

#Rapor Nesnelere Erişim

Eğer raporu bir sınıf olarak saklıyorsanız (bkz. "Raporu Saklama ve Yükleme" bölümü), rapor nesnelere doğrudan erişebilirsiniz. Aşağıdaki örnekte, Text1 nesnesinin yazı tipi nasıl değiştirileceği gösterilmiştir:

```
SimpleListReport report = new SimpleListReport();  
report.Text1.Font = new Font("Arial", 12);
```

Diğer durumlarda, Report nesnesinin `FindObject` metodunu kullanmanız gerekir:

```
TextObject text1 = report1.FindObject("Text1") as TextObject;  
text1.Font = new Font("Arial", 12);
```

Rapor içerisinde tanımlı veri kaynağına erişmek için Report nesnesinin `GetDataSource` metodunu kullanın. Bu metod, kaynak takma adını (yani "Veriler" penceresinde görünen adı) parametre olarak alır:

```
DataSourceBase ds = report1.GetDataSource("Products");
```

Kod kullanarak rapor oluřturma

Ařağıda, gruplandırılmıř bir rapor oluřturan rnek gsterilmiřtir:

ReportTitle	PRODUCTS
Group Header: [Products.ProductName]. Substring(0, 1)	[[Products.ProductName].Product
Data: Products	[Products.ProductName]
Group Footer	Count: [CountOfProducts]

```
Report report = new Report();

// "Products" tablosunu kaydet
report.RegisterData(dataSet1.Tables["Products"], "Products");

// Rapor iinde kullanılabilir hale getir
report.GetDataSource("Products").Enabled = true;

// Tm kenar bořlukları 1cm olarak ayarlanmıř A4 sayfası oluřtur
ReportPage page1 = new ReportPage();
page1.Name = "Page1";
report.Pages.Add(page1);

// ReportTitle bandı oluřtur
page1.ReportTitle = new ReportTitleBand();
page1.ReportTitle.Name = "ReportTitle1";

// Ykseklięini 1.5cm olarak ayarla
page1.ReportTitle.Height = Units.Centimeters * 1.5f;

// Grup bařlıęı oluřtur
GroupHeaderBand group1 = new GroupHeaderBand();
group1.Name = "GroupHeader1";
group1.Height = Units.Centimeters * 1;

// Grup kořulunu ayarla
group1.Condition = "[Products.ProductName].Substring(0, 1)";

// Grubu page.Bands koleksiyonuna ekle
page1.Bands.Add(group1);

// Grup altbilgisini oluřtur
group1.GroupFooter = new GroupFooterBand();
group1.GroupFooter.Name = "GroupFooter1";
group1.GroupFooter.Height = Units.Centimeters * 1;

// DataBand oluřtur
DataBand data1 = new DataBand();
data1.Name = "Data1";
data1.Height = Units.Centimeters * 0.5f;

// Veri kaynaęını ayarla
data1.DataSource = report.GetDataSource("Products");

// DataBand'i bir gruba baęla
group1.Data = data1;

// "Text" nesnelerini oluřtur
// Rapor bařlıęı
```

```

// Rapor başlığı
TextObject text1 = new TextObject();
text1.Name = "Text1";

// Sınırları ayarla
text1.Bounds = new RectangleF(0, 0, Units.Centimeters * 19, Units.Centimeters * 1);

// Metni ayarla
text1.Text = "ÜRÜNLER";

// Görünümü ayarla
text1.HorzAlign = HorzAlign.Center;
text1.Font = new Font("Tahoma", 14, FontStyle.Bold);

// Bunu ReportTitle'a ekle
page1.ReportTitle.Objects.Add(text1);

// Grup
TextObject text2 = new TextObject();
text2.Name = "Text2";
text2.Bounds = new RectangleF(0, 0, Units.Centimeters * 2, Units.Centimeters * 1);
text2.Text = "[[Products.ProductName].Substring(0, 1)]";
text2.Font = new Font("Tahoma", 10, FontStyle.Bold);

// Bunu GroupHeader'a ekle
group1.Objects.Add(text2);

// DataBand
TextObject text3 = new TextObject();
text3.Name = "Text3";
text3.Bounds = new RectangleF(0, 0, Units.Centimeters * 10, Units.Centimeters * 0.5f);
text3.Text = "[[Products.ProductName]]";
text3.Font = new Font("Tahoma", 8);

// Bunu DataBand'e ekle
data1.Objects.Add(text3);

// Grup altbilgisi
TextObject text4 = new TextObject();
text4.Name = "Text4";
text4.Bounds = new RectangleF(0, 0, Units.Centimeters * 10, Units.Centimeters * 0.5f);
text4.Text = "Adet: [CountOfProducts]";
text4.Font = new Font("Tahoma", 8, FontStyle.Bold);

// Bunu GroupFooter'a ekle
group1.GroupFooter.Objects.Add(text4);

// Bir toplam ekle
Total groupTotal = new Total();
groupTotal.Name = "CountOfProducts";
groupTotal.TotalType = TotalType.Count;
groupTotal.Evaluator = data1;
groupTotal.PrintOn = group1.Footer;

// Bunu rapor toplamlarına ekle
report.Dictionary.Totals.Add(groupTotal);

// Raporu çalıştır
report.Show();

```

Hazır rapor aşağıdaki gibi görünür:

PRODUCTS

A

Aniseed Syrup

Alice Mutton

Count: 2

B

Boston Crab Meat

Count: 1

Kendi önizleme penceresini kullanma

`FastReport.Utils.Config.PreviewSettings` özelliğini kullanarak (bkz. "FastReport Ortam Yapılandırması" bölümü) standart önizleme penceresinin görünümünü ve davranışını ayarlayabilirsiniz.

Herhangi bir nedenle standart önizleme penceresi size uygun gelmiyorsa, kendi önizleme pencerenizi oluşturabilirsiniz. Bunun için, forma ekleyebileceğiniz PreviewControl kontrolünü kullanın (FastReport WPF için WpfPreviewControl, FastReport Avalonia için AvaloniaPreviewControl). Raporu kendi önizleme penceresinde göstermek için, PreviewControl'ü rapora aşağıdaki kodla bağlayın:

```
report1.Preview = previewControl1; // WinForms/Mono için
report1.WpfPreview = previewControl1; // WPF için
report1.AvaloniaPreview = previewControl1; // Avalonia için
```

Raporu oluşturup PreviewControl'de görüntülemek için, raporun `Show` metodunu kullanın:

```
report1.Show();
your_form.ShowDialog();
```

veya aşağıdaki kodu kullanabilirsiniz:

```
if (report1.Prepare())
{
    report1.ShowPrepared();
    your_form.ShowDialog();
}
```

Bu örneklerde `your_form` , PreviewControl'ün bulunduğu formunuzdur.

PreviewControl bileşeninin yöntemlerini kullanarak, çalışmasını kodla yönetebilirsiniz. Ayrıca, `ToolbarVisible` ve `StatusbarVisible` özellikleri ile standart araç çubuğu ve/veya durum çubuğunu devre dışı bırakabilirsiniz.

FastReport .NET paketine, \Demos\C#\CustomPreview örneği dahildir; bu örnekte önizleme penceresinin nasıl yönetileceği gösterilmiştir (FastReport WPF için örnek \Demos\WPF\CustomPreview klasöründe, FastReport Avalonia için ise \Demos\Avalonia\CustomPreview klasöründe bulunmaktadır).

Bağlantı Sihirbazında Tablo Listesinin Filtrelenmesi

Veri bağlantısı sihirbazı, "Veriler|Yeni Veri Kaynağı..." menüsünden çağrılabilir. Burada bağlantıyı yapılandırabilir ve bir veya daha fazla veri tablosu seçebilirsiniz. Varsayılan olarak, sihirbaz bağlantıda bulunan tüm tabloları gösterir.

Gereksiz tabloları filtrelemek isterseniz, `FastReport.Utils.Config.DesignerSettings.FilterConnectionTables` olayını kullanın. Aşağıdaki örnek, "Table 1" adlı tabloyu listeden nasıl kaldıracağınızı göstermektedir:

```
using FastReport.Utils;
Config.DesignerSettings.FilterConnectionTables += FilterConnectionTables;
private void FilterConnectionTables(
    object sender, FilterConnectionTablesEventArgs e)
{
    if (e.TableName == "Table 1")
        e.Skip = true;
}
```

ASP.NET ile çalışma

WebReport bileşenini kullanma

Web işleyicisi ayarlanıyor

Bir raporun depolanması ve yüklenmesi

Verilerin kaydedilmesi

Bir rapor parametresine değer geçirme

"Medium Trust" modunda çalışmak

Web Farm ve Web Garden mimarilerinde çalışmak

ASP.NET MVC ile çalışma

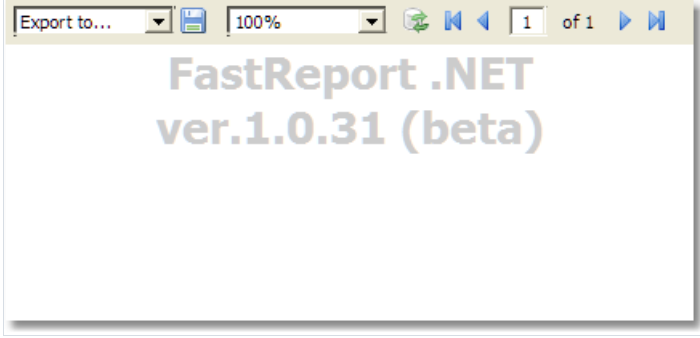
MVC'de dışa aktarma örneği

FastReport .NET ve jQuery

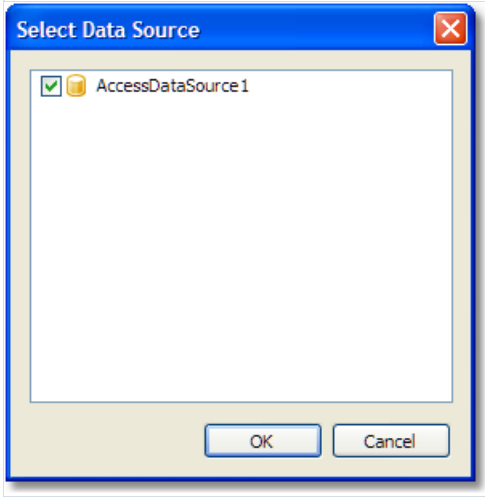
WebReport Bileşeninin Kullanımı

WebReport bileşeninin tipik kullanım senaryosunu inceleyelim.

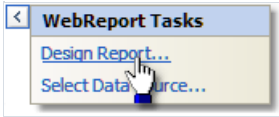
- Web projenizde veri kaynaklarının (örneğin, AccessDataSource) yapılandırıldığı varsayılmaktadır.
- WebReport bileşenini web formunuza ekleyin:



- "smart tag" menüsünden "Select Data Source..." seçeneğini seçin ve raporda kullanmak istediğiniz veri kaynağını (bir veya birden fazla) belirleyin.



- "smart tag" menüsünden "Design Report..." seçeneğini seçerek rapor tasarımcısını başlatın.



- Raporu oluşturun. Bu konuda daha fazla bilgi için "Kullanım Kılavuzu"na bakabilirsiniz.
- Tasarımcıyı kapatın.
- Projenizdeki değişiklikleri kaydedin ve projenizi çalıştırın. Hazır raporun gösterildiği bir pencere açılacaktır.

Web işleyicisi ayarlanıyor

WebReport'ün doğru çalışabilmesi için `web.config` dosyasında özel bir işleyici (handler) yapılandırması yapılması gerekmektedir. Visual Studio ortamında görsel nesne oluşturulduğunda, gerekli satırlar otomatik olarak yapılandırma dosyasına eklenir. WebReport bileşeni, uygulama çalışırken gerekli yapılandırmanın mevcut olup olmadığını kontrol eder. Eğer `web.config` dosyasında gerekli satırlar bulunmazsa, değişiklik yapmanız için talimat içeren bir hata mesajı görüntülenir.

IIS6 ile Kullanım Durumunda: `web.config` dosyası aşağıdaki satırları içermelidir:

```
<system.web>
...
<httpHandlers>
  <add path="FastReport.Export.axd" verb="*" type="FastReport.Web.Handlers.WebExport"/>
</httpHandlers>
...
</system.web>
```

IIS7 ile Kullanım Durumunda: `web.config` dosyası aşağıdaki satırları içermelidir:

```
<system.webServer>
...
<handlers>
  <add name="FastReportHandler" path="FastReport.Export.axd" verb="*"
type="FastReport.Web.Handlers.WebExport"/>
</handlers>
...
</system.webServer>
```

Projeyi başka bir sunucuya taşıdığınızda, `web.config` dosyasındaki WebReport yapılandırma satırlarının eksiksiz olduğundan emin olun.

İşleyicinin Çalışıp Çalışmadığını Kontrol Etme: Aşağıdaki URL'e erişerek WebReport işleyicisinin çalışabilirliğini test edebilirsiniz:

http://yoursite/app_folder/FastReport.Export.axd

Eğer işleyici doğru şekilde yapılandırıldıysa, FastReport sürüm numarası hakkında bilgi veren bir mesaj görüntülenecektir.

Raporun Saklanması ve Yüklmesi

Raporu aşğıdaki yöntemlerden biriyle saklayabilirsiniz:

Web Form Kodunda

Özellikler:

- Avantajlar:
 - FastReport .NET ile çalışmanın en hızlı ve basit yolu.
- Dezavantajlar:
 - Rapor, WebReport bileşeninin `ReportResourceString` özelliğinde saklanır. Bu yöntemde rapor, `ViewState` aracılığıyla istemciye gönderilir. Eğer rapor büyükse bu durum performansı olumsuz etkileyebilir.
 - Bu yöntem, medium trust moduyla uyumlu değildir.

Yükleme İşlemi: Rapor, otomatik olarak yüklenir.

.FRX Dosyasında

Bu yöntemde rapor, "App_Data" adlı özel bir klasörde saklanır. Yapılacaklar:

1. Rapor tasarımcısını başlatın.
2. Raporu oluşturun ve .FRX dosyası olarak kaydedin.
3. "Solution Explorer" penceresinde "App_Data" klasörünü seçin, sağ tıklayın ve "Add | Existing Item..." seçeneğini seçin. Yeni kaydedilen rapor dosyasını seçin.
4. WebReport bileşenini seçin ve ReportResourceString özelliğini temizleyin.
5. ReportFile özelliğini seçin ve düzenleyicisini açın. Açılan pencereden "App_Data" klasöründeki rapor dosyasını seçin.

Özellikler:

- Avantajlar:
 - Rapor dosyası istemciye gönderilmez.
- Dezavantajlar:
 - Bu yöntem, medium trust moduyla uyumlu değildir.

Yükleme İşlemi:

Rapor, otomatik olarak yüklenir.

Ayrıca, raporu WebReport.StartReport üzerinden yüklemek de mümkündür. Örnek kod:

```
(sender as WebReport).Report.Load(this.Server.MapPath("~/App_Data/report.frx"));
```

C#/VB.NET Sınıfı Olarak

Bu yöntemde rapor, bir rapor sınıfı olarak işlenir. Yapılacaklar:

1. Rapor tasarımcısında "File | Save As..." seçeneğini seçin ve dosya tipini "C# File" veya "VB.NET File" olarak belirleyin (raporda seçili olan dile bağlı olarak – bkz. Kullanım Kılavuzu).

2. Oluşan dosyayı projenize ekleyin. Genellikle "App_Code" klasöründe saklanması önerilir.
3. WebReport bileşeninin ReportResourceString ve ReportFile özelliklerini temizleyin.

Özellikler:

- **Avantajlar:**

- Rapor, normal bir sınıf gibi çalışır.
- Visual Studio'nun tüm özelliklerine, özellikle hata ayıklama gibi imkanlara erişim sağlanır.
- Bu yöntem, ASP.NET'te Medium Trust moduyla uyumlu olan tek yöntemdir.

- **Dezavantajlar:**

- Bu yöntemle raporu düzenlemek mümkün değildir. Düzenleme yapmak için orijinal .FRX formatındaki rapor dosyasına ihtiyaç vardır.

Çalışma Şekli:

Rapor ile çalışmak için `WebReport.StartReport` olay işleyicisini oluşturun. Olay işleyicisinde şunları yapın:

1. Rapor sınıfından bir örnek oluşturun.
2. Verileri kaydedin.
3. Bu örneği, `WebReport` bileşeninin Report özelliğine atayın.

Örnek kod:

```
SimpleListReport report = new SimpleListReport();
report.RegisterData(your_data, "your_data_name");
WebReport1.Report = report;
```

Eğer daha önce hazırlanmış bir raporu göstermek isterseniz, yine `WebReport.StartReport` olay işleyicisi ve `WebReport.ReportDone` özelliği kullanılabilir. Örnek kod:

```
(sender as WebReport).Report.LoadPrepared(this.Server.MapPath("~/App_Data/Prepared.fpx"));
(sender as WebReport).ReportDone = true;
```

Verilerin kaydedilmesi

Veri kaynağını WebReport bileşeninin "akıllı etiket" menüsünü kullanarak seçerseniz, verileri manuel olarak kaydetmeniz gerekmez. Bu durumda, FastReport.NET, veri kaynaklarının adlarını WebReport bileşeninin `ReportDataSources` özelliğinde depolar.

Bu tür bir veri kaydetme yöntemi kullanmak istemiyorsanız, bunu manuel olarak yapmanız gerekir. Bu, WebReport bileşeninin `StartReport` olayını kullanarak yapılabilir. Bu olay işleyicisinde, raporun `RegisterData` ve `RegisterDataAsp` yöntemlerini çağırabilirsiniz. Rapora `WebReport.Report` özelliği aracılığıyla erişilebilir:

```
webReport1.Report.RegisterData(myDataSet);
```

Verilerin kaydedilmesi hakkında daha fazla bilgi edinin [bu bölüm](#).

Rapor parametresinin iletilmesi

Parametre değerini iletmek için `Report` nesnesinin `SetParameterValue` metodunu kullanın. Bu metod, "Windows.Forms ile Çalışma" bölümünde incelenmiştir.

Rapor parametresini iletmek için `WebReport` bileşeninin `StartReport` olayını kullanın. Rapor, `WebReport.Report` özelliğinde mevcuttur:

```
webReport1.Report.SetParameterValue("MyParam", 10);
```

Medium Trust Modunda Çalışma

Bu modda çoğu shared-hosting sağlayıcısı hizmet verir. Aşağıdaki kısıtlamalar uygulanır:

- Rapor kodunun derlenmesi mümkün değildir;
- MS Access veri kaynakları ile çalışma mümkün değildir;
- RichObject nesnesinin kullanımı mümkün değildir;
- WinAPI çağrıları yapan veya geçici dosyalar (PDF, Open Office) oluşturan bazı dışa aktarma filtrelerinin kullanımı mümkün değildir;
- Sağlayıcıya bağlı olarak başka kısıtlamalar da olabilir.

FastReport ile bu modda çalışırken, raporun sınıf olarak saklanması yöntemini kullanın "[Raporun Saklanması ve Yüklmesi](#)" Bu durumda, raporun çalışma esnasında derlenmesine gerek yoktur.

Web Farm ve Web Garden Mimarilerinde Çalışma

FastReport rapor oluşturunca, çok sunuculu (Web Farm) veya çok işlemcili (Web Garden) mimarilerde kullanıldığında, WebReport nesneleri arasında veri senkronizasyonu için özel bir dosya depolama alanı oluşturulmasını gerektirir. \

Aşağıdaki satırları `web.config` yapılandırma dosyasına eklemeniz gerekmektedir:

```
<appSettings>
  <add key="FastReportStoragePath" value="\\FS\WebReport_Exchange"/>
  <add key="FastReportStorageTimeout" value="10"/>
  <add key="FastReportStorageCleanup" value="1"/>
</appSettings>
```

- `FastReportStoragePath` – Geçici dosyaların saklanacağı klasör yolunu belirtir; çok sunuculu mimaride her sunucunun bu klasöre erişimi olmalıdır.
- `FastReportStorageTimeout` – Rapor ön belleğinin (cache) dakika cinsinden saklanma süresidir.
- `FastReportStorageCleanup` – Rapor ön belleğinde süresi dolmuş öğelerin kontrolü için dakika cinsinden belirlenen aralıktır.

Yapılandırmanın doğruluğunu kontrol etmek için uygulamanıza şu URL üzerinden erişin:

http://yoursite/app_folder/FastReport.Export.axd

Sonuç `Cluster mode: ON` ifadesini içermelidir.

ASP.NET MVC ile Çalışma

ASPX düzeninde FastReport kullanımı, ASP.NET ile yapılan normal çalışmadan hiçbir farkı yoktur. ASPX düzeninde WebReport kullanım örneğini, \Demos\C#\MvcDemo klasöründe bulabilirsiniz.

MVC 3 sürümünden itibaren gelen Razor düzeninde WebReport kullanımına daha detaylı değinmek gerekmektedir. WebReport'un doğru çalışabilmesi için, web uygulamasının kök klasöründeki `web.config` dosyasına işlemci tanımlamalarını eklemeniz gerekmektedir. IIS7 ve üstü sunucular kullanılırken, `<system.webServer>` `<handlers>` bölümüne aşağıdaki satırı ekleyin:

```
<add name="FastReportHandler" path="FastReport.Export.axd" verb="*"
type="FastReport.Web.Handlers.WebExport" />
```

IIS6 kullanılırken ise, `<system.web>` `<httpHandlers>` bölümüne şu satır eklenmelidir:

```
<add path="FastReport.Export.axd" verb="*" type="FastReport.Web.Handlers.WebExport" />
```

Sonrasında, View'lerin bulunduğu klasördeki `web.config` dosyasında değişiklik yapmanız gerekmektedir.

`<system.web.webPages.razor>` `<namespaces>` bölümüne aşağıdaki satırları ekleyin:

```
<add namespace="FastReport" />`<br/>`<add namespace="FastReport.Web" />
```

Bu satırlar, View içerisinde FastReport nesnelerini oluşturabilmeniz ve bu nesnelerin özelliklerine doğrudan erişebilmeniz için gereklidir.

`_Layout.cshtml` dosyasında, etiketinin içine şu satırları ekleyin:

```
@WebReportGlobals.Scripts()
@WebReportGlobals.Styles()
```

Artık raporu View üzerinde görüntülemeye geçebilirsiniz. İlgili denetleyiciye gidip orada bir WebReport nesnesi oluşturun:

```
WebReport webReport = new WebReport(); // create object
webReport.Width = 600; // set width
webReport.Height = 800; // set height
webReport.Report.RegisterData(dataSet, "AppData"); // data binding
webReport.ReportFile = this.Server.MapPath("~/App_Data/report.frx"); // load the report from the file
ViewBag.WebReport = webReport; // send object to the View
```

View kodunda, uygun bir yere şu satırı ekleyin:

```
@ViewBag.WebReport.GetHtml()
```

Aynı şekilde WebReport oluřturma kodunu doğrudan View içerisinde de yazabilirsiniz.

Razor düzeninde WebReport kullanım örneğini, \Demos\C#\MvcRazor klasöründe bulabilirsiniz. Bu klasörde, önceden hazırlanmış raporun yüklenmesi dahil olmak üzere farklı rapor yükleme seçenekleri ve [StartReport](#) olayının kullanım örneği yer almaktadır.

Unutmayın, her örneğin bin klasörüne eksik olan kütüphaneleri eklemeniz gerekmektedir.

MVC'de dışa aktarma örneği

ASP.NET MVC'de Dosya İhracat Örneği FastReport .NET'in ASP.NET MVC framework ile birlikte kullanılması durumunda, formdaki bir düğmeye basıldığında istenilen çıkış formatında dosyanın kolayca oluşturulması mümkündür.

View'e Eklenecek Kod:

```
@using (Html.BeginForm("GetFile", "Home"))
{
    <input id="pdf" type="submit" value="PDF Olarak Dışa Aktar" />
}
```

- Burada `GetFile` , formun submit işleminin sonuçlarını kontrol eden metot adıdır;
- `Home` ise ilgili kontrolcünün (bu örnekte HomeController.cs dosyası) ismidir.

Kontrolcüye Eklenmesi Gereken Namespace:

```
using FastReport.Export.Pdf;
```

Kontrolcüye Eklenmesi Gereken Metot:

```
public FileResult GetFile()
{
    WebReport webReport = new WebReport();
    // Veri bağlama
    System.Data.DataSet dataSet = new System.Data.DataSet();
    dataSet.ReadXml(report_path + "nwind.xml");
    webReport.Report.RegisterData(dataSet, "NorthWind");
    // İstenen rapor dosyasını yükle
    webReport.ReportFile = this.Server.MapPath("~/App_Data/report.frx");
    // Raporu hazırla
    webReport.Report.Prepare();
    // Akışa istenilen formatta kaydet
    Stream stream = new MemoryStream();
    webReport.Report.Export(new PDFExport(), stream);
    stream.Position = 0;
    // Sonucu tarayıcıya gönder
    return File(stream, "application/zip", "report.pdf");
}
```

Burada veri bağlama ve rapor dosyasının yüklenmesi örnek olarak gösterilmiştir. Veri bağlama, rapor dosyasının tasarımcısı üzerinden de doğrudan gerçekleştirilebilir.

Eğer farklı bir format gereksinimi varsa, ilgili namespace'i ekleyip export satırında gerekli constructor'ı çağırmanız yeterlidir.

Örneğin, Excel 2007 formatı için:

```
using FastReport.Export.OoXML;  
...  
webReport.Report.Export(new Excel2007Export(), stream);  
...  
return File(stream, "application/xlsx", "report.xlsx");
```

FastReport .NET ve jQuery

FastReport .NET rapor oluşturucusunun WebReport nesnesi, çalışma sırasında jQuery kütüphanesini kullanır. Bu kütüphane oldukça popülerdir ve projenizde kullanılabilir.

Razor işaretlemeyle çalışırken istemci tarayıcısında jQuery betiklerinin ve stil dosyalarının tekrarlanarak yüklenmesini önlemek için, `Layout.cshtml` dosyasında aşağıdaki satırları kullanmalısınız:

```
@WebReportGlobals.ScriptsWjQuery()  
@WebReportGlobals.StylesWjQuery()
```

tüm dosyaları yükleyen standart

```
@WebReportGlobals.Scripts()  
@WebReportGlobals.Styles()
```

satırlarının yerine. ASPX işaretlemeyle çalışırken, `ExternalJquery = true` olarak ayarlamamız gerekir (varsayılan değeri `false` 'tur).

WCF ile Çalışma

[FastReport.Service.dll Kütüphanesi](#)

[Basit WCF Servis Örneği](#)

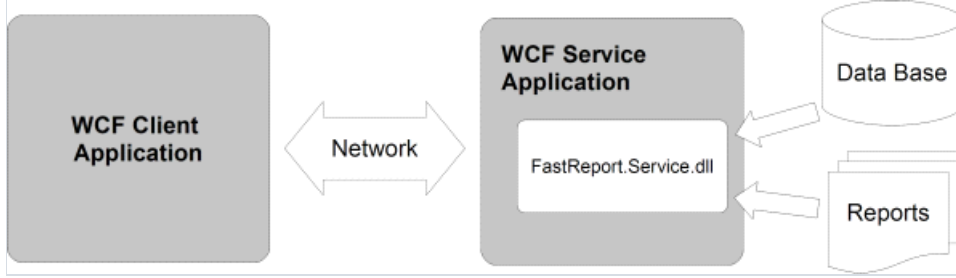
[FastReport.Service.dll Temelli Web Servisinin Oluşturulması](#)

[Windows Hizmeti Tabanlı WCF Servisinin Oluşturulması](#)

WCF servis kütüphanesi FastReport.Service.dll

FastReport .NET, .NET Framework 4.6.2 ve üzeri sürümler için geliştirilen uygulamalara entegre edilebilen **FastReport.Service.dll** kütüphanesini içermektedir.

Bu kütüphane, bir WCF Service Library olarak hazırlanmış olup, servis işlevi gören kullanıcı uygulamaları içinde kullanılmak üzere tasarlanmıştır.



Şu anda kütüphane aşağıdaki fonksiyonları içermektedir:

```
List<ReportItem> GetReportsList();
```

Mevcut raporların **ReportItem** nesneleri şeklinde listesini döndürür. Raporlar, servisin çalıştığı sunucunun sabit diskinde depolanır. Klasör hiyerarşisi göz önünde bulundurularak tüm raporlar listelenir ve dosyalar alfabetik olarak sıralanır.

```
List<ReportItem> GetReportsListByPath(string path);
```

Belirtilen **path** dizindeki raporları ReportItem nesneleri şeklinde döndürür. Bu listeye ilgili klasördeki tüm raporlar ve alt klasörler dahil edilir; dosyalar alfabetik olarak sıralanır.

```
List<GearItem> GetGearList();
```

Servisin oluşturabileceği mevcut formatların **GearItem** nesneleri şeklindeki listesini döndürür.

```
Stream GetReport(ReportItem report, GearItem gear);
```

report raporunun, **gear** formatında oluşturulması sonucunda elde edilen veri akışını döndürür. **report** ve **gear** parametreleri, daha önce alınan listelerden seçilebildiği gibi, gerekli özelliklerle yeni nesneler olarak da oluşturulabilir. Döndürülen veri akışı konumlandırma desteği sunmaz; yalnızca sıralı okuma yapılabilir.

Detaylar için öge tanımlarına bakınız:

ReportItem

```
public class ReportItem
{
    public string Path;
    public string Name;
    public string Description;
    public Dictionary<string, string> Parameters;
}
```

- **Path** – Rapor dosyasının, raporların depolandığı kök dizine göre konumunu belirtir. Rapor dosyasının uzantısı yalnızca *.frx olabilir. Bu özellik, sonraki sorgularda belirli bir raporu tanımlamak için kullanılır.
- **Name** – Raporun adı, rapor dosyasının meta verilerinden alınır. Eğer raporun meta verilerinde ad boş bırakılmışsa, isim dosya adının uzantısız haliyle aynıdır. Bu özellik, örneğin ListBox gibi kontrol elemanlarıyla etkileşimli rapor listeleri oluştururken kullanılabilir.
- **Description** – Raporun açıklamasını, rapor dosyasının meta verilerinden alır.
- **Parameters (Dictionary<string, string>)** – Rapor parametrelerini içeren sözlüktür. Rapor şablonuna aktarılabilecek parametrelerle doldurulabilir. Sadece string değerler desteklenmektedir; bu durum rapor şablonunun hazırlanmasında dikkate alınmalıdır.

GearItem

```
public class GearItem
{
    public string Name;
    public Dictionary<string, string> Properties;
}
```

- **Name** – Formatın adını belirtir. Bu alan, aşağıdaki string değerlerden birini içerebilir:

Name	Açıklama
PDF	Adobe Acrobat dosyası
DOCX	Microsoft Word 2007 dosyası
XLSX	Microsoft Excel 2007 dosyası
PPTX	Microsoft PowerPoint 2007 dosyası
RTF	Rich Text dosyası – birçok metin düzenleyici tarafından desteklenir
ODS	Open Office Spreadsheet dosyası
ODT	Open Office Text dosyası
MHT	Resimlerle birlikte sıkıştırılmış HTML dosyası, Internet Explorer'da açılabilir
CSV	Comma separated values - virgülle ayrılmış değerlerden oluşan metin tablosu
DBF	dBase veritabanı dosyası
XML	XML Excel tablosu – resimler desteklenmez

TXT	Metin dosyası
FPX	FastReport .NET hazır rapor formatı. Görüntüleme amacıyla Viewer.exe programında açılabilir veya koddaki Report.LoadPrepared(stream) metodu kullanılarak doğrudan rapor nesnesine yüklenebilir; ardından rapor Report.ShowPrepared() metodu ile ekranda gösterilebilir.

Dictionary<string, string> Properties – rapor oluşturma parametrelerinin sözlüğü. Varsayılan değerlerle desteklenen tüm parametrelerin tam listesi, sunucudan format listesi talep edildiğinde erişilebilir.

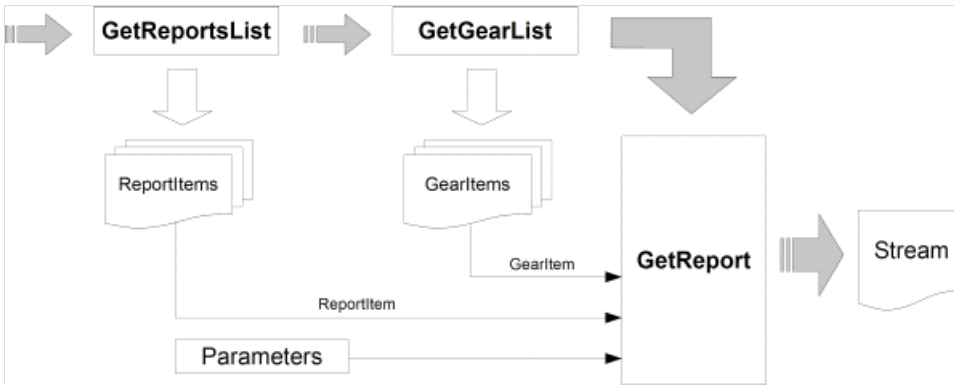
FastReport.Service.dll kütüphanesini kullanan bir servis oluştururken, uygulamanızın **App.config** veya **Web.config** dosyasına aşağıdaki ayarları eklemeniz gerekmektedir.

```
<appSettings>
  <add key="FastReport.ReportsPath" value="C:\Program files\FastReports\FastReport .NET Professional\Demos\Reports" />
  <add key="FastReport.ConnectionStringName" value="FastReportDemo" />
  <add key="FastReport.Gear" value="PDF,DOCX,XLSX,PPTX,RTF,ODS,ODT,MHT,CSV,DBF,XML,TXT,FPX" />
</appSettings>
```

FastReport.ReportsPath – rapor dosyalarının bulunduğu klasörün yolunu belirtir; bu klasördeki rapor dosyalarının listesi istemciye iletilecektir. **FastReport.ConnectionStringName** – konfigürasyon dosyasındaki <connectionStrings> bölümünde saklanan veritabanı bağlantı dizesinin adıdır. Bu değer, rapor şablonundaki dahili bağlantı dizesinin servis tarafından oluşturulma sırasında konfigürasyondan alınan dize ile değiştirilmesinde kullanılır.

FastReport.Gear – kullanılabilir formatların listesidir. Sadece gerekli formatları seçebilir ve isimlerin sırasını değiştirebilirsiniz.

FastReport.Service 'in şematik kullanımı:



Eğer hangi raporu hangi formatta alacağınızı kesin olarak biliyorsanız, servisin kullanımı şu şekilde olabilir (bu, servise yapılan istek sayısını azaltır):



Önemli noktalar (**FastReport.Service** 'de kullanılacak rapor şablonlarını oluştururken dikkat edilmesi gereken hususlar):

- Raporlardaki diyaloglar desteklenmez ve rapor oluşturma sırasında atlanır.
- Her rapor, servis tarafından rapor

oluřturulurken konfigürasyondan alınacak olan bağlantı dizesiyle değıştirilecek dahili bir DataConnection içermelidir.

FastReport.Service.dll kullanımına dair örnekler, \Demos\C#\Web\WCFWebService ve \Demos\C#\Web\WCFWebClient klasörlerinde incelenebilir. Servis konfigürasyon dosyasının örneęi:

`FastReport.Service.dll.config` .

Basit bir WCF servisi örneği

Bu örnek, programlama gerektirmeden kütüphanenin ve konfigürasyon dosyasının çalışır durumda olduğunu test etmek için hazırlanmıştır. Görevi gerçekleştirmek için Visual Studio ile birlikte gelen WcfSvcHost.exe programını kullanıyoruz:

1. Deneylerimiz için diskte herhangi bir yere, örneğin **C:\WCF\FastReport** konumunda bir klasör oluşturun.
2. Oluşturulan klasöre aşağıdaki dosyaları kopyalayın:
 - o FastReport.Service.dll
 - o FastReport.Service.dll.config
 - o FastReport.dll
 - o FastReport.Bars.dll
3. **\Data** ve **\Reports** adında iki alt klasör oluşturun.
4. **\Data** klasörüne, örneklerden alınan **\Demos\Reports\nwind.xml** veritabanı dosyasını kopyalayın.
5. **\Reports** klasörüne, **\Demos\Reports** klasörünün içeriğini kopyalayın – bu klasör, **FastReport.Service.dll** kütüphanesi ile kullanıldığında gerekli olan, veritabanı bağlantıları yerleşik test raporlarını içerir.
6. Herhangi bir metin düzenleyici ile **FastReport.Service.dll.config** konfigürasyon dosyasını açın.
7. bölümünde raporların bulunduğu klasörün yolunu aşağıdaki gibi değiştirin:

```
<add key="FastReport.ReportsPath" value="C:\WCF\FastReport\Reports" />
```

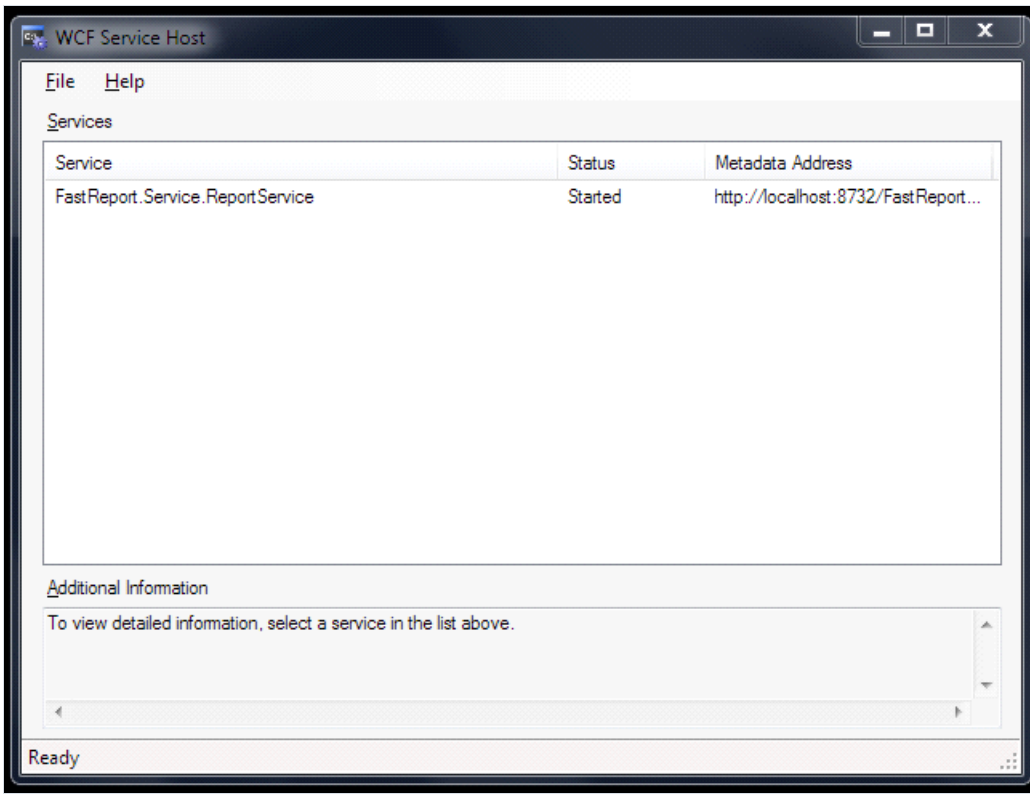
8. bölümünde veritabanımıza bağlantı dizesini aşağıdaki şekilde değiştirin:

```
<add name="FastReportDemo" connectionString="XsdFile=;XmlFile=C:\WCF\FastReport\Data\nwind.xml"/>
```

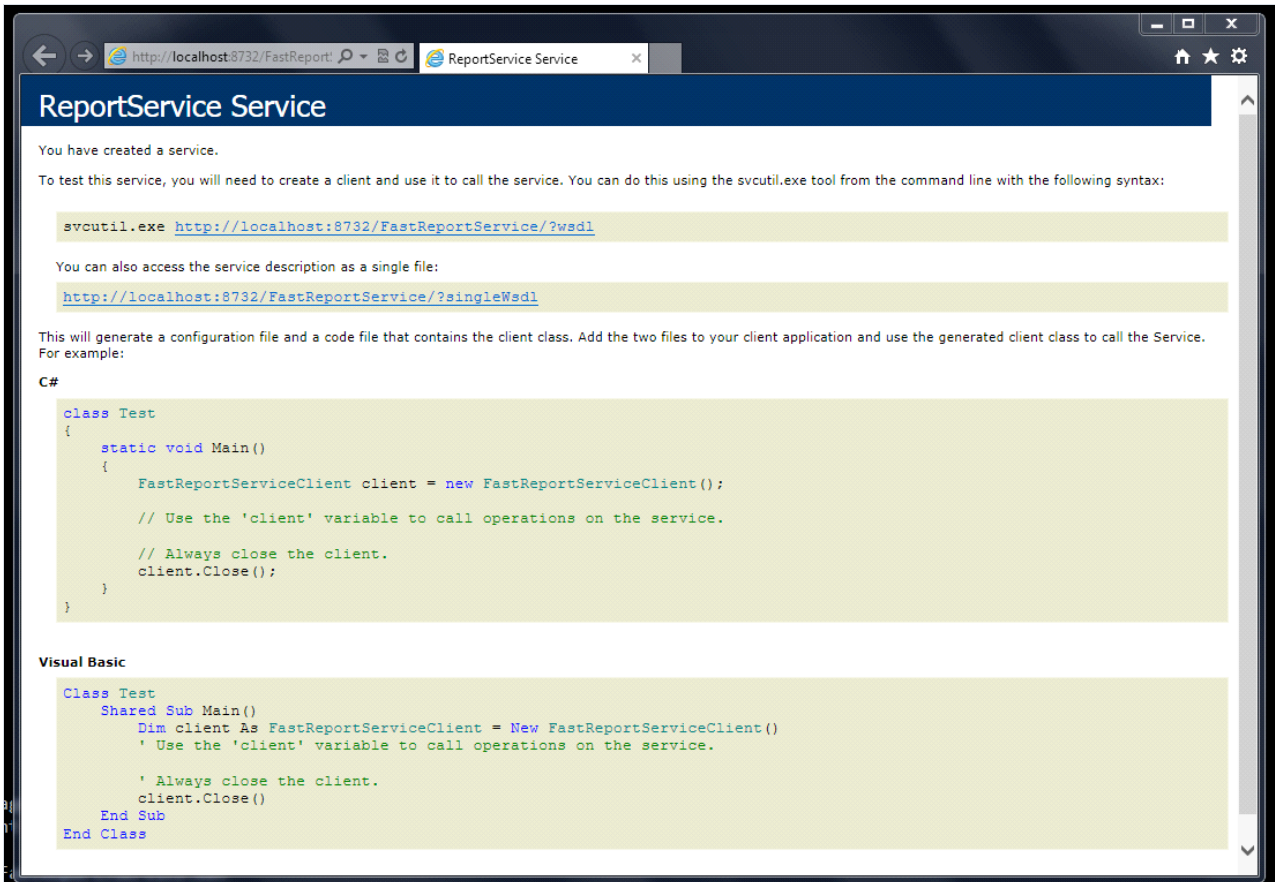
9. Aşağıdaki satırı içeren **service.bat** dosyasını oluşturun:

```
"C:\Program Files\Microsoft Visual Studio 10.0\Common7\IDE\WcfSvcHost.exe"  
/service:"C:\WCF\FastReport\FastReport.Service.dll"  
/config:"C:\WCF\FastReport\FastReport.Service.dll.config"
```

10. Explorer üzerinden yönetici haklarıyla (Run as administrator) **service.bat** dosyasını çalıştırın. Bildirim alanında (system tray) WCF Service Host ikonu belirecektir. İkona çift tıklayın; aşağıdaki içeriğe sahip bir pencere açılmalıdır.



11. Tarayıcınızı açıp şu adrese gidin: <http://localhost:8732/FastReportService/>



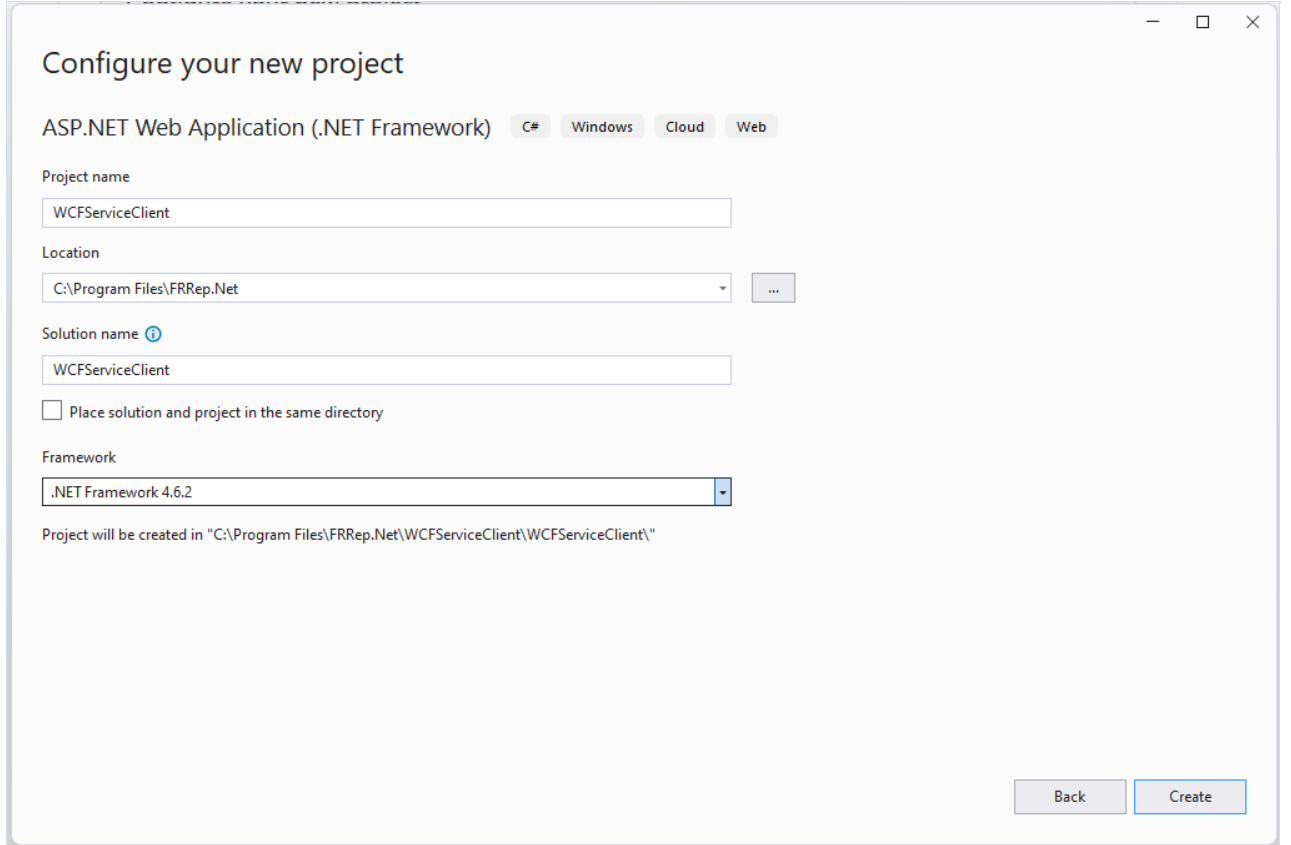
Servis port numarasını, konfigürasyon dosyasında şu satırı değiştirerek ayarlayabilirsiniz:

```
<add baseAddress="http://localhost:8732/FastReportService/" />
```

FastReport.Service.dll kullanarak Web Servisi oluřturma

FastReport.Service.dll kullanarak web servisi oluřturmak olduka basit bir iřlemdir. Ařağıda, .NET Framework 4.6.2 (veya daha st) altında alıřan temel bir ASP.NET Web Application projesi oluřturup FastReport.Service.dll kullanarak web servisini yapılandırma adımlarını bulabilirsiniz

Visual Studio'yu aın ve .NET Framework 4.6.2 (veya daha st) iin bir ASP.NET Web Application projesi oluřturun.



Configure your new project

ASP.NET Web Application (.NET Framework) C# Windows Cloud Web

Project name
WCFSERVICEClient

Location
C:\Program Files\FRRep.Net

Solution name ⓘ
WCFSERVICEClient

☐ Place solution and project in the same directory

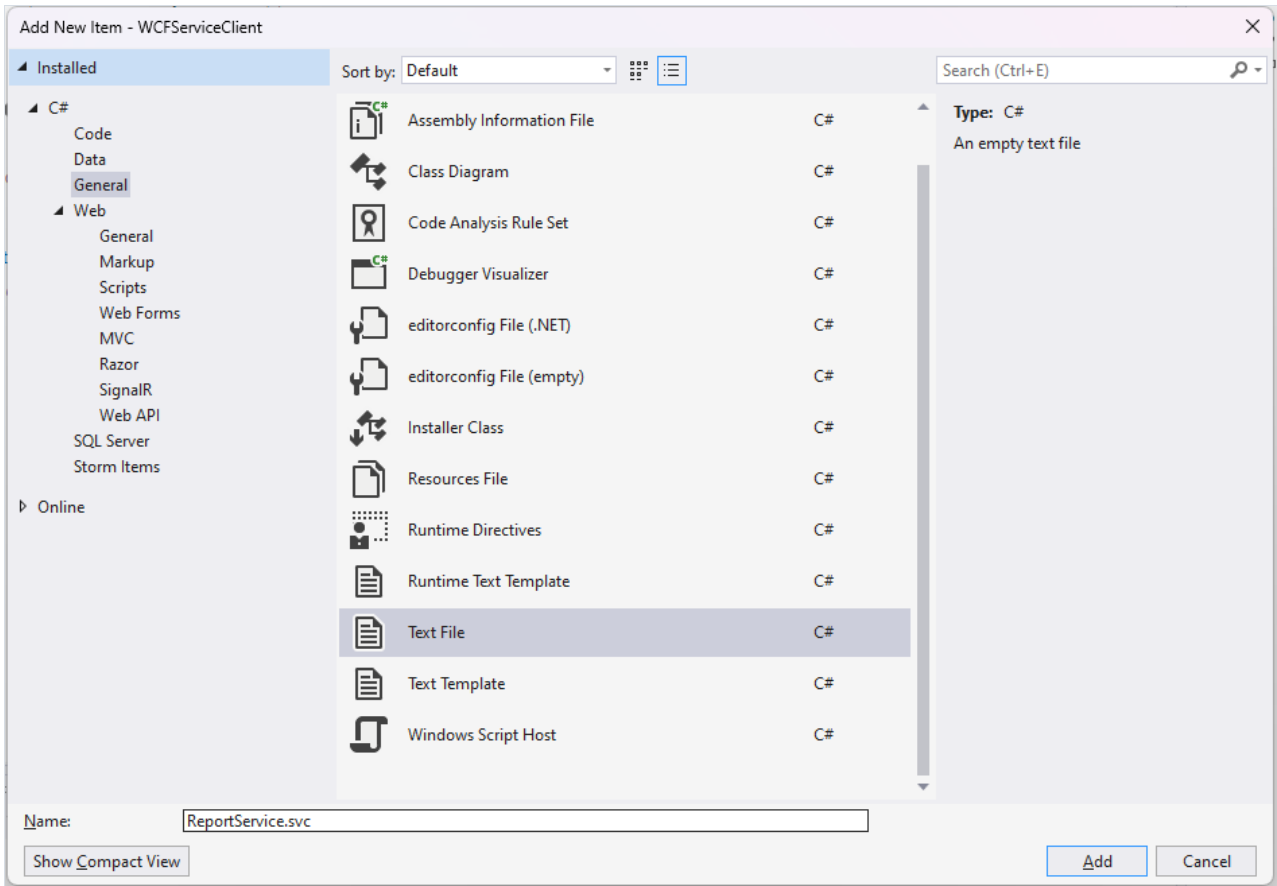
Framework
.NET Framework 4.6.2

Project will be created in "C:\Program Files\FRRep.Net\WCFSERVICEClient\WCFSERVICEClient\"

Back Create

FastReport.dll, FastReport.Bars.dll, FastReport.Service.dll ktphanelerine bařvurular ekleyin.

Site kk dizininde `ReportService.svc` adında yeni bir metin dosyası oluřturun.



ReportService.svc Dosyasının Oluşturulması. Projenizin kök dizinine **ReportService.svc** adında bir dosya ekleyin ve aşağıdaki içeriği ekleyin:

```
<%@ ServiceHost Service="FastReport.Service.ReportService" %>
<%@ Assembly Name="FastReport.Service" %>
```

web.config Ayarları. **web.config** dosyasını açın ve bölümüne aşağıdaki bölümleri ekleyin:

```

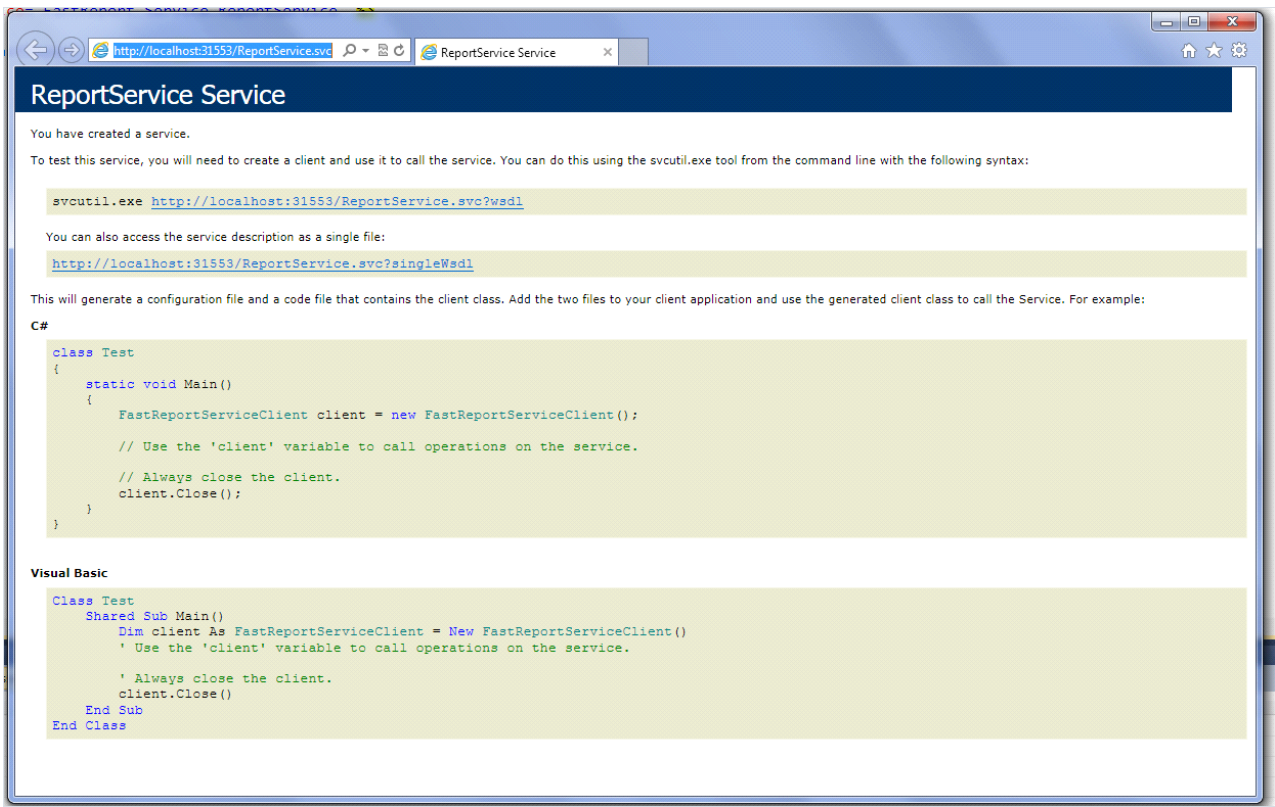
<appSettings>
  <!-- path to folder with reports -->
  <add key="FastReport.ReportsPath" value="C:\Program files\FastReports\FastReport .NET
Professional\Demos\Reports" />
  <!-- name of connection string for reports -->
  <add key="FastReport.ConnectionStringName" value="FastReportDemo" />
  <!-- Comma-separated list of available formats PDF,DOCX,XLSX,PPTX,RTF,ODS,ODT,MHT,CSV,DBF,XML,TXT,FPX.
You can delete any or change order in this list. -->
  <add key="FastReport.Gear" value="PDF,DOCX,XLSX,PPTX,RTF,ODS,ODT,MHT,CSV,DBF,XML,TXT,FPX" />
</appSettings>
<connectionStrings>
  <add name="FastReportDemo" connectionString="XsdFile=;XmlFile=C:\Program Files\FastReports\FastReport
.NET Professional\Demos\Reports\nwind.xml"/>
</connectionStrings>
<system.serviceModel>
  <services>
    <service behaviorConfiguration="FastReportServiceBehavior" name="FastReport.Service.ReportService">
      <endpoint address="" binding="wsHttpBinding" contract="FastReport.Service.IFastReportService">
        <identity>
          <dns value="localhost" />
        </identity>
      </endpoint>
      <endpoint address="mex" binding="mexHttpBinding" contract="IMetadataExchange" />
    </service>
  </services>
  <behaviors>
    <serviceBehaviors>
      <behavior name="FastReportServiceBehavior">
        <serviceMetadata httpGetEnabled="True" />
        <serviceDebug includeExceptionDetailInFaults="True" />
      </behavior>
    </serviceBehaviors>
  </behaviors>
  <bindings>
    <basicHttpBinding>
      <binding messageEncoding="Mtom"
closeTimeout="00:02:00" openTimeout="00:02:00"
receiveTimeout="00:10:00" sendTimeout="00:02:00"
maxReceivedMessageSize="67108864" maxBufferSize="65536"
transferMode="Streamed">
        <security mode="None">
          <transport clientCredentialType="None" />
        </security>
      </binding>
    </basicHttpBinding>
  </bindings>
</system.serviceModel>

```

FastReport.ReportsPath anahtarı, raporların bulunduğu klasöre giden yolu içermelidir. Örneğin, bunu \Demos\Reports klasörüne ayarlayabilirsiniz.

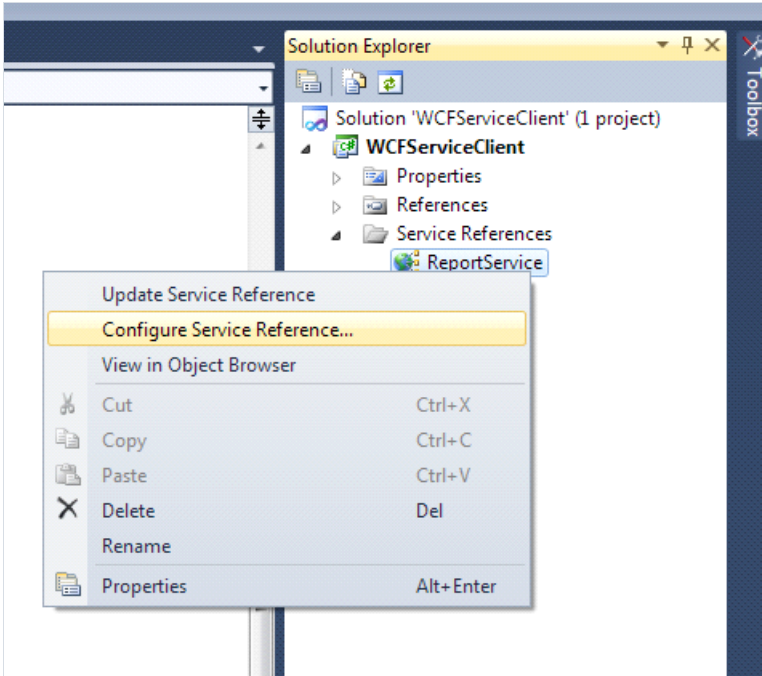
FastReport.ConnectionStringName anahtarı, bağlantı dizesinin adını içermelidir. Bu satır <connectionStrings> bölümünde tanımlanmalıdır.

Siteyi çalıştırılma ve ReportService.svc dosyasına erişerek Web servisinin erişilebilirliğini kontrol edelim. Siteyi çalıştırılma ve ReportService.svc dosyasına erişerek Web servisinin erişilebilirliğini kontrol edelim.



Projeyi sunucuya dağıttığınızda, `FastReport.dll`, `FastReport.Bars.dll`, `FastReport.Service.dll` dosyalarının \bin klasöründe olduğundan emin olun.

İstemci programlarının örnekleri \Demos\C#\Web\WCFWebClient klasörlerinde bulunabilir. Her projeyi Visual Studio'da açın, ReportService üzerine sağ tıklayın ve açılan menüden 'Configure Service Reference' seçeneğini tıklayın.



Yapılandırma penceresinde mevcut bir web servisinin adresini belirtin.

Windows hizmetinde barındırılan WCF hizmetinin oluşturulması

Visual Studio'yu açın ve Windows Service projesi oluşturun.

Configure your new project

Windows Service (.NET Framework) C# Windows Desktop Service

Project name
WCFWindowsService

Location
C:\Program Files\FRRRep.Net

Solution
Create new solution

Solution name
WCFWindowsService

☐ Place solution and project in the same directory

Framework
.NET Framework 4.6.2

Project will be created in "C:\Program Files\FRRRep.Net\WCFWindowsService\WCFWindowsService"

Back Create

Service1.cs tasarımını açın

```
2 using System.Collections.Generic;
3 using System.ComponentModel;
4 using System.Data;
5 using System.Diagnostics;
6 using System.Linq;
7 using System.ServiceProcess;
8 using System.Text;
9 using System.Threading.Tasks;
10
11 namespace WCFWindowsService
12 {
13     3 references
14     public partial class Service1 : ServiceBase
15     {
16         1 reference
17         public Service1()
18         {
19             InitializeComponent();
20         }
21
22         0 references
23         protected override void OnStart(string[] args)
24         {
25         }
26
27         0 references
28         protected override void OnStop()
29         {
30         }
31     }
32 }
```

View Designer Shift+F7

Quick Actions and Refactorings... Ctrl+.

Rename... Ctrl+R, Ctrl+R

Remove and Sort Usings Ctrl+R, Ctrl+G

View Code F7

Peek Definition Alt+F12

Go To Definition F12

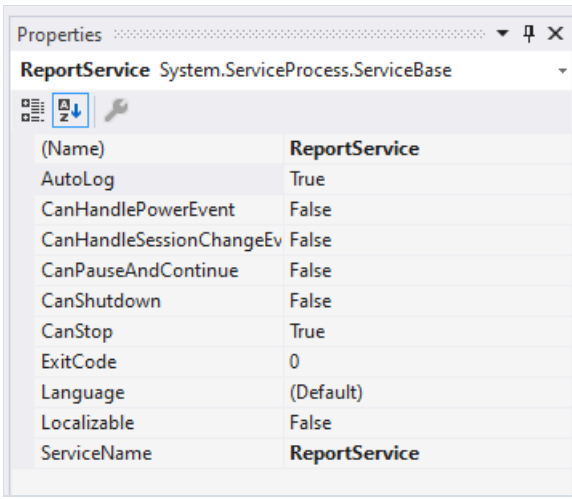
Go To Base Alt+Home

Go To Implementation Ctrl+F12

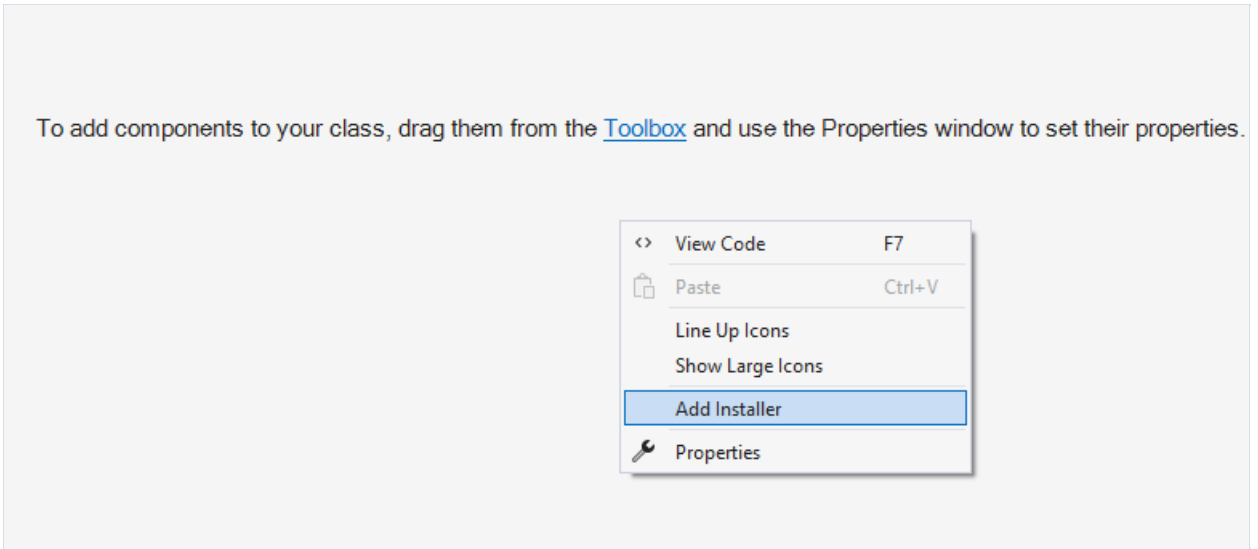
Find All References Shift+F12

View Call Hierarchy Ctrl+K, Ctrl+T

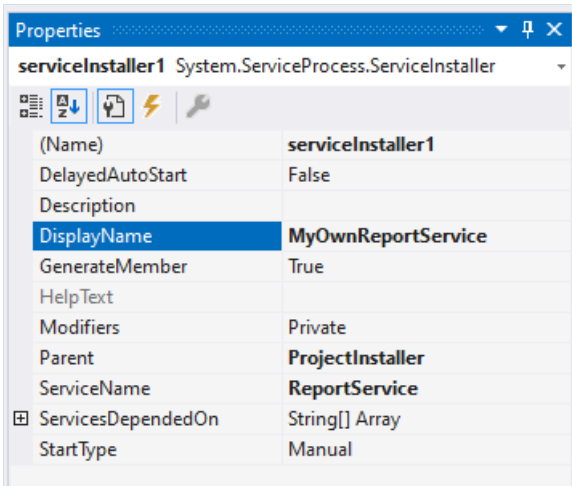
Servisin adını kendi seçiminize göre değiştirin:



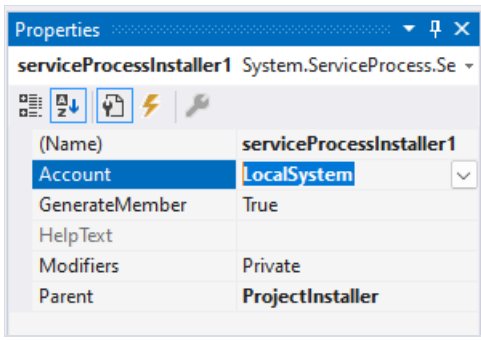
Pencereye sağ tıklayın ve açılan menüden 'Add Installer' (Yükleyici Ekle) seçeneğini seçin:



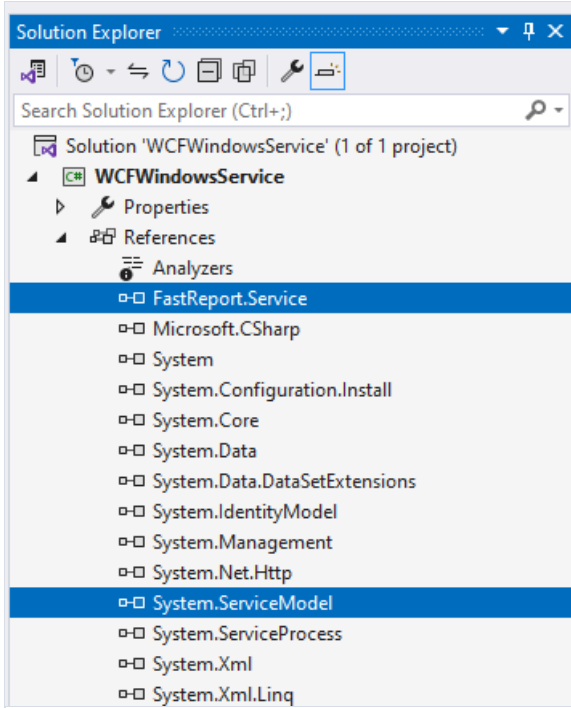
Bileşen `serviceInstaller1` 'in özelliklerini düzenleyin - `DisplayName` (Görünen Ad) ayarlayın.



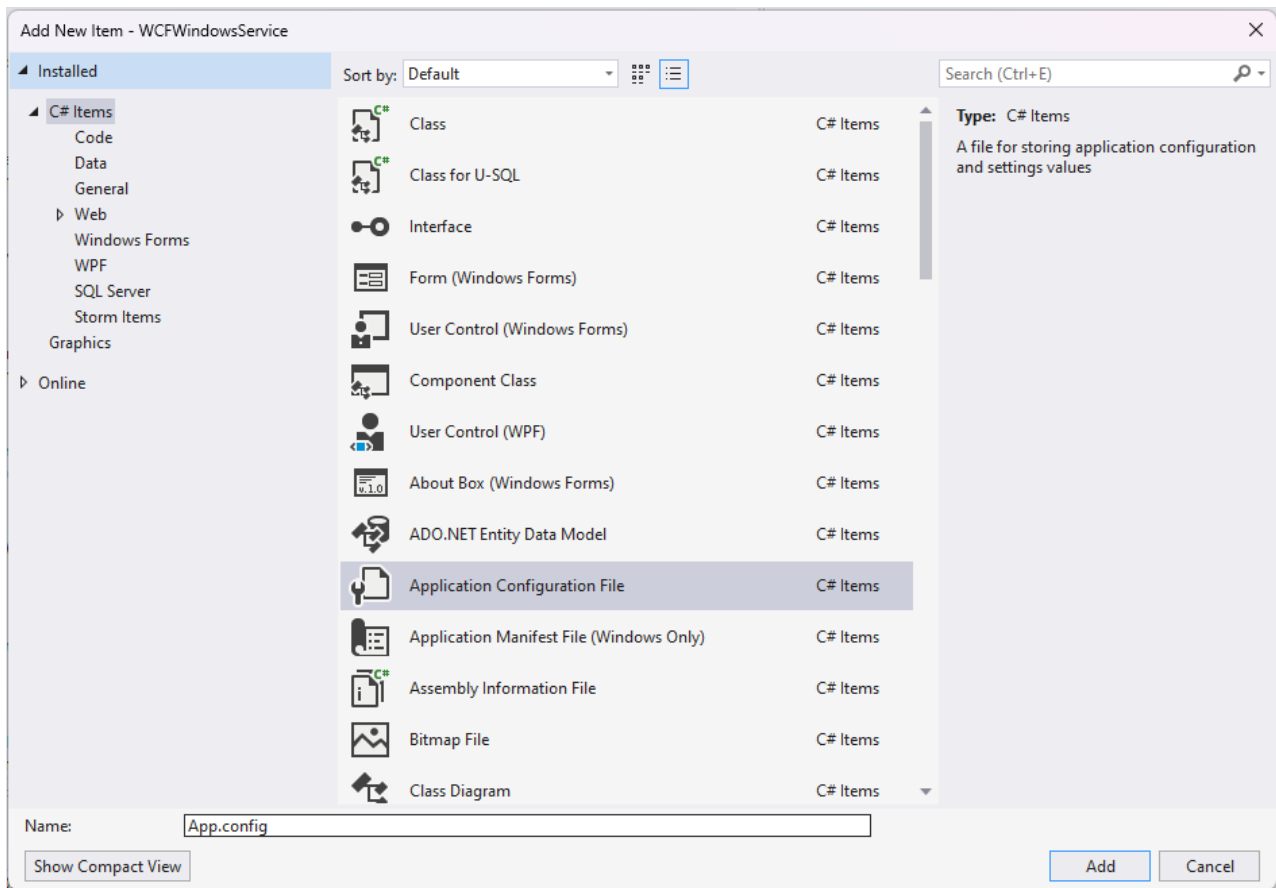
`serviceProcessInstaller1` bileşeninin özelliklerinde, hizmet için hesap türünü `LocalSystem` olarak ayarlayın.



Projeye `System.ServiceModel` ve `FastReport.Service.dll` referansları ekleyin:



Bir uygulama yapılandırma dosyası oluşturun:



Aşağıdaki metni yeni `app.config` dosyasına kopyalayın:

```
<?xml version="1.0"?>
<configuration>
  <appSettings>
    <!-- path to folder with reports -->
    <add key="FastReport.ReportsPath" value="C:\Program files\FastReports\FastReport .NET
Professional\Demos\Reports" />
    <!-- name of connection string for reports -->
    <add key="FastReport.ConnectionStringName" value="FastReportDemo" />
    <!-- Comma-separated list of available formats PDF,DOCX,XLSX,PPTX,RTF,ODS,ODT,MHT,CSV,DBF,XML,TXT,FPX.
You can delete any or change order in this list. -->
    <add key="FastReport.Gear" value="PDF,DOCX,XLSX,PPTX,RTF,ODS,ODT,MHT,CSV,DBF,XML,TXT,FPX" />
  </appSettings>
  <connectionStrings>
    <add name="FastReportDemo" connectionString="XsdFile=;XmlFile=C:\Program Files\FastReports\FastReport
.NET Professional\Demos\Reports\nwind.xml"/>
  </connectionStrings>
  <system.web>
    <compilation debug="true" />
    <membership defaultProvider="ClientAuthenticationMembershipProvider">
      <providers>
        <add name="ClientAuthenticationMembershipProvider"
type="System.Web.ClientServices.Providers.ClientFormsAuthenticationMembershipProvider,
System.Web.Extensions, Version=4.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" serviceUri=""
/>
      </providers>
    </membership>
    <roleManager defaultProvider="ClientRoleProvider" enabled="true">
      <providers>
        <add name="ClientRoleProvider" type="System.Web.ClientServices.Providers.ClientRoleProvider,
System.Web.Extensions, Version=4.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" serviceUri=""
cacheTimeout="86400" />
      </providers>
    </roleManager>
  </system.web>
  <!-- When deploying the service library project, the content of the config file must be added to the
```

```

host's
app.config file. System.Configuration does not support config files for libraries. -->
<system.serviceModel>
<services>
<service behaviorConfiguration="FastReportServiceBehavior" name="FastReport.Service.ReportService">
<endpoint address="" binding="wsHttpBinding" contract="FastReport.Service.IFastReportService">
<identity>
<dns value="localhost" />
</identity>
</endpoint>
<endpoint address="mex" binding="mexHttpBinding" contract="IMetadataExchange" />
<host>
<baseAddresses>
<add baseAddress="http://localhost:8732/FastReportService/" />
</baseAddresses>
</host>
</service>
</services>
<behaviors>
<serviceBehaviors>
<behavior name="FastReportServiceBehavior">
<serviceMetadata httpGetEnabled="True" />
<serviceDebug includeExceptionDetailInFaults="True" />
</behavior>
</serviceBehaviors>
</behaviors>
<bindings>
<basicHttpBinding>
<binding messageEncoding="Mtom"
closeTimeout="00:02:00" openTimeout="00:02:00"
receiveTimeout="00:10:00" sendTimeout="00:02:00"
maxReceivedMessageSize="67108864" maxBufferSize="65536"
transferMode="Streamed">
<security mode="None">
<transport clientCredentialType="None" />
</security>
</binding>
</basicHttpBinding>
</bindings>
</system.serviceModel>
<startup>
<supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.6.2" />
</startup>
</configuration>

```

Service1.cs düzenleyicisine gidin ve şu satırı ekleyin:

```
using System.ServiceModel;
```

Hizmet sınıfını şu şekilde değiştirin:

```

public partial class ReportService : ServiceBase
{
    ServiceHost reportHost;

    public ReportService()
    {
        InitializeComponent();
    }

    protected override void OnStart(string[] args)
    {
        if (reportHost != null)
            reportHost.Close();
        reportHost = new ServiceHost(typeof(FastReport.Service.ReportService));
        reportHost.Open();
    }

    protected override void OnStop()
    {
        reportHost.Close();
        reportHost = null;
    }
}

```

Hizmeti, örneğin .NET Framework ile birlikte gelen komut satırı aracı **InstallUtil.exe** kullanarak kurabilirsiniz:

```

C:\Windows\Microsoft.NET\Framework\v4.0.30319\InstallUtil.exe "C:\Program
Files\FRRRep.Net\WCFWindowsService\WCFWindowsService\bin\Debug\WCFWindowsService.exe"

```

Ve hizmeti şu komutla başlatabilirsiniz:

```

net start ReportService

```

Bir web tarayıcısını açın ve **app.config** dosyasındaki baseAddress bölümünde ayarlanan `http://localhost:8732/FastReportService/` adresini kontrol edin. Klasörü ve portu kendi seçiminize göre değiştirebilirsiniz. Hizmeti durdurmak ve kaldırmak için komutlar:

```

net stop ReportService

```

```

C:\Windows\Microsoft.NET\Framework\v4.0.30319\InstallUtil.exe /u "C:\Program
Files\FRRRep.Net\WCFWindowsService\WCFWindowsService\bin\Debug\WCFWindowsService.exe"

```

Blazor ile Çalışma

Elinizde, Blazor Server web uygulamanıza entegre edilebilecek bileşenleri içeren FastReport.Web paketi bulunmaktadır. 2021.3 sürümünden itibaren, bu kütüphane Blazor Server (sunucu tarafı) uygulamaları için Razor bileşenlerini içermektedir. Bu, tüm işlemlerin sunucu tarafında gerçekleştirileceği ve ardından istemciye hazır görüntü olarak aktarılacağı anlamına gelir.

Mevcut durumda, en geniş uyumluluğu sağlamak amacıyla minimum hedef platform (target framework) .NET 6 olarak belirlenmiştir. Bu, en güncel uzun süreli destek (LTS) sürümü ile maksimum uyumluluğu hedeflemektedir.

Ortak Stiller ve SVG Ayarları

`FastReport.Web` for Core ile karşılaştırıldığında, Toolbar ve Tabs için SVG görüntüleri ile Tabs, Outline ve diğer bazı genel gösterim stilleri, web uygulamanızda (renk, boyut veya görüntü değiştirme gibi ayarlamalar yapılabilmesi amacıyla) ayarlanabilmesi için `staticWebAssets` dizinine taşınmıştır.

Bu kaynaklar, yerel depolamanızda bulunmaktadır. Uygulamanızın geliştirme/derleme aşamasında, bu kaynaklar şu adreste yer alır: `{UserName}/.nuget/packages/fastreport.web/{version}/staticwebassets`

Web uygulamanızın yayınlanması (dotnet publish) sırasında ise, bu kaynaklar `wwwroot/_content/FastReport.Web` dizinine kopyalanır.

Demo Projesi

`FastReport.Web` paketinin çalışmasını göstermek amacıyla hazırlanmış demo projeyi [GitHub](#) hesabımızda bulabilirsiniz. `WebReportContainer` kullanım örneği, kullanıcı bileşeni içerisinde, `Pages/Index.razor` ve `Pages/Index.razor.cs` dosya yollarında yer almaktadır.

FastReport İşlevselliğinin Genişletilmesi

Kendi Bağlantı Türlerinizin Oluşturulması

Kendi Bağlantı Türlerinizi Oluşturma

Bağlantılar (data connection), rapora veri kaynağı eklemek için kullanılır. Bu, uygulamanın sağladığı verileri kullanmak yerine, rapordan doğrudan verilere bağlanmanızı sağlar.

Kendi bağlantınızı oluşturmak için aşağıdaki adımları gerçekleştirmeniz gerekmektedir:

- Bağlantı sınıfını oluşturun: `FastReport.Data.DataConnectionBase` sınıfından türemiş bir sınıf oluşturun ve bu sınıfın bazı metotlarını uygulayın.
- Bağlantı düzenleyicisini oluşturun: `FastReport.Data.ConnectionEditors.ConnectionEditorBase` sınıfından türemiş bir sınıf oluşturun ve kullanıcı arayüzünü, ayrıca `GetConnectionString` ve `SetConnectionString` metotlarını uygulayın.
- Bağlantıyı FastReport'a kaydedin.

Bu adımlar aşağıda detaylı olarak ele alınacaktır.

Temel Bağlantı Sınıfı

Kendi bağlantınızı oluşturmak için, `FastReport.Data.DataConnectionBase` sınıfını kullanın. Bu sınıf, kendi bağlantınızı oluştururken ezmeniz gereken aşağıdaki metotları içerir:

```
public abstract class DataConnectionBase : DataComponentBase
{
    protected virtual string GetConnectionStringWithLoginInfo(string userName, string password);
    public abstract string QuoteIdentifier(string value, DbConnection connection);
    public virtual DbConnection GetConnection();
    public virtual DbDataAdapter GetAdapter(string selectCommand, DbConnection connection,
        CommandParameterCollection parameters);
    public virtual ConnectionEditorBase GetEditor();
    public virtual Type GetParameterType();
    public virtual string GetConnectionId();
    public virtual string[] GetTableNames();
    public virtual void TestConnection();
    public virtual void FillTableSchema(DataTable table, string selectCommand,
        CommandParameterCollection parameters);
    public virtual void FillTableData(DataTable table, string selectCommand,
        CommandParameterCollection parameters);
}
```

Tüm metotları override etmeniz gerekmez; çünkü bazı metotların varsayılan uygulamaları, çoğu durumda yeterli olacaktır. Örneğin, eğer bağlantınız `DbConnection` ve `DbDataAdapter` tipindeki nesneleri kullanıyorsa (ki bu, bağlantıların büyük çoğunluğu için standarttır), aşağıdaki metotları uygulamanız gerekecektir:

```
public abstract string QuoteIdentifier(string value, DbConnection connection);
protected virtual string GetConnectionStringWithLoginInfo(string userName, string password);
public virtual DbConnection GetConnection();
public virtual DbDataAdapter GetAdapter(string selectCommand, DbConnection connection,
    CommandParameterCollection parameters);
public virtual ConnectionEditorBase GetEditor();
public virtual Type GetParameterType();
public virtual string GetConnectionId();
public virtual string[] GetTableNames();
```

Eğer bağlantınız yukarıdaki nesnelerle çalışmıyorsa, örneğin uygulama sunucusu ile rapor arasında bir köprü görevi görüyorsa, aşağıdaki metotları uygulamanız gerekecektir:

```
* public abstract string QuoteIdentifier(string value, DbConnection connection);
* public virtual ConnectionEditorBase GetEditor();
* public virtual Type GetParameterType();
* public virtual string GetConnectionId();
* public virtual string[] GetTableNames();
* public virtual void TestConnection();
* public virtual void FillTableSchema(DataTable table, string selectCommand,
    CommandParameterCollection parameters);
* public virtual void FillTableData(DataTable table, string selectCommand,
    CommandParameterCollection parameters);
```

Aşağıda, her metot için açıklamalar verilmiştir.

ConnectionString Özelliği

Her bağlantı tipi, veritabanı yolu, SQL lehçesi, kullanıcı adı ve şifre gibi kendi parametre setine sahiptir. Bu parametrelerin saklanması için, string tipinde olan `ConnectionString` özelliği kullanılır.

Bu özellik şu şekillerde kullanılır:

- `GetConnection` Metodunda: `DbConnection` tipi nesne oluşturulurken, bağlantı dizesi bu özellik aracılığıyla aktarılır.
- Bağlantı Editöründe: Bağlantı editöründe, bağlantı dizesinden ayrı parametreler (örneğin, veritabanı dosya yolu) ayrıştırılır. Bu işlem için, her bağlantı tipi için özel olarak tasarlanmış `DbConnectionStringBuilder` sınıfı kullanılır. Örneğin, MS SQL için bu sınıf `SqlConnectionStringBuilder` olarak adlandırılır.

GetConnectionStringWithLoginInfo Metodu

Bu metod, mevcut bağlantı dizesini (`ConnectionString` özelliğinden alınan) alır, içine kullanıcı adı ve şifre bilgisini ekler ve modifiye edilmiş dizeyi döndürür. Bağlantı ayarlarında "Bağlanırken şifre sor" seçeneği belirtildiğinde bu metod kullanılır.

Örneğin, `MsSqlDataConnection` için bu metod şu şekilde uygulanır:

```
protected override string GetConnectionStringWithLoginInfo(string userName, string password)
{
    // Mevcut bağlantı dizesini alıyoruz
    SqlConnectionStringBuilder builder = new SqlConnectionStringBuilder(ConnectionString);

    // Kullanıcı adı ve şifre bilgisini dizeye ekliyoruz
    builder.IntegratedSecurity = false;
    builder.UserID = userName;
    builder.Password = password;

    // Modifiye edilmiş dizeyi döndürüyoruz
    return builder.ToString();
}
```

Bu metodu geçersiz kılmanız (override) gerekmektedir.

GetConnection Metodu

`GetConnection` metodu, ilgili bağlantıya özgü yeni bir `DbConnection` nesnesi döndürür. Bu nesne, tabloya veri doldurulması gerektiğinde veritabanına bağlanmak için kullanılır.

Bağlantı parametreleri, `ConnectionString` özelliğinden alınır. Örneğin, `MsSqlDataConnection` için metot aşağıdaki gibi görünür:

```
public override DbConnection GetConnection()
{
    return new SqlConnection(ConnectionString);
}
```

Genellikle, bu metodu geçersiz kılmanız gerekir. Çünkü `GetConnection` metodu, diğer iki metot olan `FillTableSchema` ve `FillTableData` 'da kullanılmaktadır. Eğer bağlantı türünüz, tablo hakkında bilgi almak ve tabloya veri yüklemek için `DbConnection` ve `DbDataAdapter` nesnelerini kullanmıyorsa, bu metodu geçersiz kılmanıza gerek yoktur. Bu durumda, `FillTableSchema` ve `FillTableData` metotlarını geçersiz kılmanız gerekmektedir.

GetAdapter Metodu

Bu metod, belirli bir bağlantıya özgü `DbDataAdapter` tipinde yeni bir nesne döndürür. Dönen bu nesne, tabloya veri doldurulması işlemlerinde kullanılır.

Metoda aşağıdaki parametreler aktarılır:

Parametre	Açıklama
selectCommand	SQL dilinde yazılmış sorgu metni.
connection	<code>GetConnection</code> metodu ile oluşturulmuş olan <code></code> <code>DbConnection</code> nesnesi.
Parameters	Sorgu için tanımlı parametreler.

Aşağıda, `MsSqlConnection` için `GetAdapter` metodunun örnek bir implementasyonu verilmiştir:

```
public override DbDataAdapter GetAdapter(string selectCommand, DbConnection connection,
    CommandParameterCollection parameters)
{
    SqlDataAdapter adapter = new SqlDataAdapter(selectCommand, connection as SqlConnection);
    foreach (CommandParameter p in parameters)
    {
        SqlParameter parameter = adapter.SelectCommand.Parameters.Add(p.Name, (SqlDbType)p.DataType, p.Size);
        parameter.Value = p.Value;
    }
    return adapter;
}
```

Bu örnekte, metodun yaptığı işlemler şu şekildedir: `SqlDataAdapter` oluşturularak, verilen `selectCommand` ve `connection` kullanılır. Parametre koleksiyonundaki her bir `CommandParameter` için, `adapter.SelectCommand.Parameters` listesine uygun tipte bir parametre eklenir. Burada `DataType` özelliği, int tipinde olduğu için, MS SQL için uygun olan `SqlDbType` 'a dönüştürülmektedir. Eklenen parametrenin `Size` ve `Value` özellikleri de ilgili değere göre ayarlanır. Her bir parametre, aşağıdaki özelliklere sahiptir:

Özellik	Açıklama
Name	Parametrenin adı.
DataType	Parametrenin veri tipi. Bu özellik <code>int</code> tipindedir; bağlantı için kullanılan parametre tipine uygun hale getirilmelidir.
Size	Parametrenin veri boyutu.
Value	Parametrenin değeri.

Genellikle, veri erişimi sırasında `FillTableSchema` ve `FillTableData` metotları tarafından kullanılan bu metodu override etmeniz gerekir. Ancak, eğer veri erişimi için `DbConnection` ve `DbDataAdapter` nesnelerini kullanmıyorsanız, bu metodu override etmeye gerek kalmayabilir. Böyle bir durumda, `FillTableSchema` ve `FillTableData` metotlarını override etmeniz gerekecektir.

GetEditor Metodu

Bu metod, belirli bir bağlantı türü için bir editör örneği döndürür. Editörün uygulandığı "Bağlantı Editörü" bölümünde ele alınacaktır.

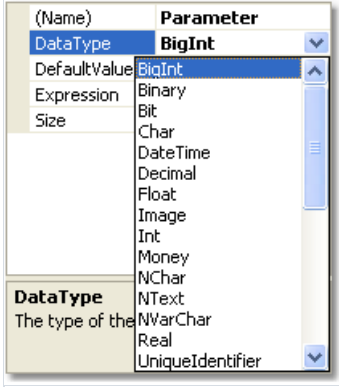
Örneğin, `MsSqlConnection` için bu metod şu şekilde uygulanır:

```
public override ConnectionEditorBase GetEditor()
{
    return new MsSqlConnectionEditor();
}
```

Bu metodu, kendi bağlantı türünüz için uygun şekilde geçersiz kılmanız gerekmektedir.

GetParameterType Metodu

Bu metodun döndürdüğü değer, sorgu parametresi düzenleyicisinde parametrenin veri türünü seçmek için kullanılır:



Örneğin, `MsSqlConnection` için parametre tipi `SqlDbType` olarak belirlenmiştir:

```
public override Type GetParameterType()
{
    return typeof(SqlDbType);
}
```

Bu metodu, bağlantınızın ihtiyaçlarına uygun şekilde geçersiz kılmanız gerekmektedir.

GetConnectionId Metodu

Metodun amacı, bağlantı sihirbazında görüntülenecek kısa bağlantı bilgisini oluşturmaktır.

Bu bilgi genellikle bağlantı türü adı ve veritabanı adını içerir. `MsSqlConnection` sınıfında `GetConnectionId` metodunun implementasyonu şu şekildedir:

```
public override string GetConnectionId()
{
    SqlConnectionStringBuilder builder = new SqlConnectionStringBuilder(ConnectionString);
    string info = builder.InitialCatalog;
    if (String.IsNullOrEmpty(info))
        info = builder.AttachDBFilename;
    return "MS SQL: " + info;
}
```

Dikkat: Bağlantı dizesini (`ConnectionString`) ayrıştırmak için `SqlConnectionStringBuilder` kullanılmıştır.

Bu metodu kendi sınıfınızda override etmeniz gerekmektedir.

FillTableSchema Metodu

Bu metod, bir tablonun şemasını (yani, sütun isimleri ve veri tiplerini) doldurur.

Metoda aşağıdaki parametreler aktarılır:

Parametre	Açıklama
table	Şeması doldurulması gereken <code>DataTable</code> nesnesi.
selectCommand	SQL dilinde yazılmış sorgu metni.
parameters	Tanımlıysa sorgu parametreleri.

Metot, varsayılan olarak `GetConnection` ve `GetAdapter` metodlarının döndürdüğü nesneleri kullanır. Aşağıda bu implementasyonun örneğini görebilirsiniz:

```
public virtual void FillTableSchema(DataTable table, string selectCommand, CommandParameterCollection parameters)
{
    using (DbConnection conn = GetConnection())
    {
        OpenConnection(conn);

        // prepare select command
        selectCommand = PrepareSelectCommand(selectCommand, table.TableName, conn);

        // read the table schema
        using (DbDataAdapter adapter = GetAdapter(selectCommand, conn, parameters))
        {
            adapter.SelectCommand.CommandTimeout = CommandTimeout;
            adapter.FillSchema(table, SchemaType.Source);
        }
    }
}
```

Çoğu durumda bu metodu override etmeniz gerekmez. Ancak, eğer veri erişimi için `DbConnection` ve `DbDataAdapter` nesnelerini kullanmıyor (dolayısıyla `GetConnection` ve `GetAdapter` metodlarını implemente etmiyorsanız) farklı bir yaklaşım uygulamanız gerekebilir.

FillTableData Metodu

Bu metod, bir tabloyu verilerle doldurur.

Metoda aşağıdaki parametreler aktarılır:

Parametre	Açıklama
table	Doldurulması gereken <code>DataTable</code> nesnesi.
selectCommand	SQL dilindeki sorgu metnidir.
parameters	Tanımlıysa sorgu parametreleri.

Metot, varsayılan olarak `GetConnection` ve `GetAdapter` metodlarının döndürdüğü nesneleri kullanan bir implementasyona sahiptir. Aşağıda bu implementasyonun örneğini görebilirsiniz:

```
public virtual void FillTableData(DataTable table, string selectCommand, CommandParameterCollection parameters)
{
    using (DbConnection conn = GetConnection())
    {
        OpenConnection(conn);

        // prepare select command
        selectCommand = PrepareSelectCommand(selectCommand, table.TableName, conn);

        // read the table
        using (DbDataAdapter adapter = GetAdapter(selectCommand, conn, parameters))
        {
            adapter.SelectCommand.CommandTimeout = CommandTimeout;
            table.Clear();
            adapter.Fill(table);
        }
    }
}
```

Çoğu durumda bu metodu override etmeniz gerekmez. Ancak, eğer veri erişimi için `DbConnection` ve `DbDataAdapter` nesnelerini kullanmıyor (dolayısıyla `GetConnection` ve `GetAdapter` metodlarını implemente etmiyorsanız) farklı bir yaklaşım uygulamanız gerekebilir.

Bağlantı Düzenleyicisi

Bağlantıyı düzenlemek için, bağlantı seçim penceresinde görüntülenen (resimde kırmızı çerçeve ile vurgulanan) bir UserControl türünde kontrol elemanı kullanılır.

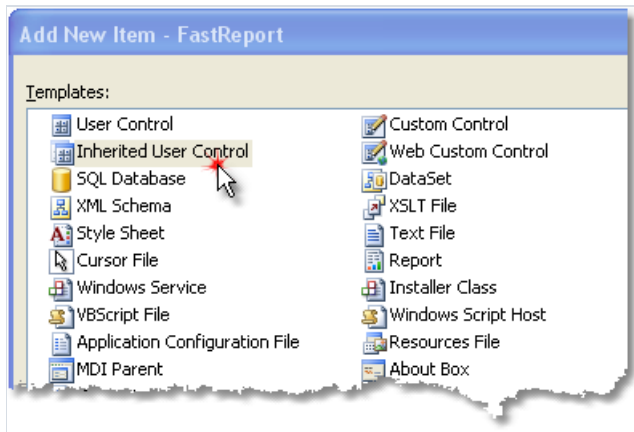
Her bağlantı türü için kendine özgü bir editör mevcuttur. Tüm editörler, temel sınıf olan

`FastReport.Data.ConnectionEditors.ConnectionEditorBase` den türetilmiştir:

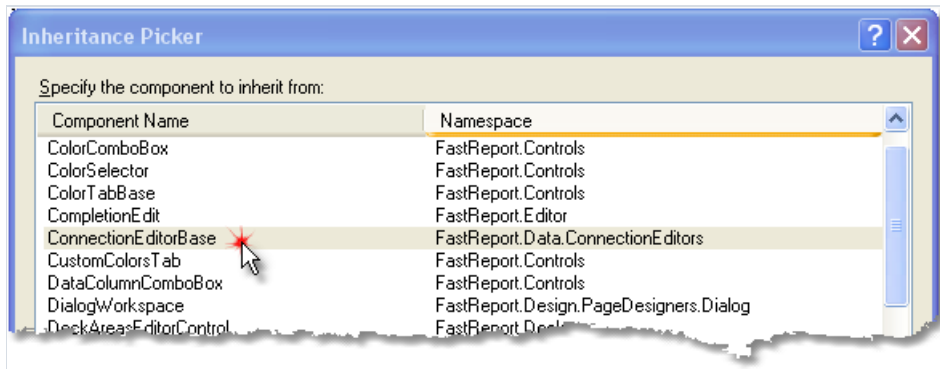
```
public class ConnectionEditorBase : UserControl
{
    protected virtual string GetConnectionString();
    protected virtual void SetConnectionString(string value);
}
```

Kendi bağlantı editörünüzü oluşturmak için şunları yapın:

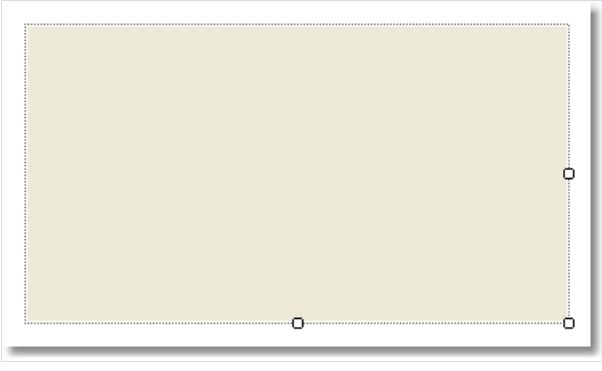
- Projeye yeni bir kontrol elemanı ekleyin (Add -> User Control...) ve "Add New Item" penceresinde Inherited User Control ögesini seçin.



- Temel sınıf türü olarak ConnectionEditorBase'i seçin.



Projeye, aşağıdaki gibi görünen bir kontrol elemanı eklenmiş olacaktır:



- Editörün yüzeyine ihtiyacınız olan kontrol elemanlarını yerleştirin. Tüm elemanları sığdırmak için editörün yüksekliğini ayarlayın. Editörün genişliği sabittir ve değiştirilemez.

Temel sınıfın metotları, kontrol elemanlarını bağlantıdaki değerlerle doldurmanıza ve tersine, kontrol elemanlarından alınan değerleri bağlantıya aktarmanıza olanak tanır.

GetConnectionString metodu

Bu metod, editör penceresinin "Tamam" butonuna tıklanarak kapatıldığında çağrılır. Editörde girilen değerleri içeren bağlantı dizesini döndürmelidir.

SetConnectionString metodu

Bu metod, editör ilk açıldığında çağrılır. Bağlantı dizesini bileşenlerine ayrıştırmalı ve editörde bu bileşenlerin değerlerini göstermelidir. Bağlantı dizelerini ayrıştırmak için her bağlantı tipinde bulunan

`DbConnectionStringBuilder` sınıfı kullanılabilir. Örneğin, MS SQL için bu `SqlConnectionStringBuilder` 'dır.

FastReport'ta Bağlantı Kaydı

Bağlantınızın FastReport tasarımcısında kullanılabilmesi için, öncelikle bağlantıyı kaydetmeniz gerekmektedir. Bu işlem iki farklı yöntemle gerçekleştirilebilir.

Yöntem 1: Proje İçerisinde Yer Alan Bağlantı. Bağlantınız projenizin bir parçası ise (yani ayrı bir .dll modülüne ayrılmamışsa), bağlantıyı aşağıdaki kod ile kaydedebilirsiniz:

```
FastReport.Utls.RegisteredObjects.AddConnection(typeof(MyDataConnection));
```

Bu kod, tasarımcı çalıştırılmadan önce, uygulamanın çalışma süresi boyunca **sadece bir kez** çalıştırılmalıdır.

Yöntem 2: Ayrı Bir .dll Modülünde Bulunan Bağlantı. Bağlantınız ayrı bir .dll modülünde yer alıyorsa, bu modülü FastReport'a ekleyerek her tasarımcı çalıştırıldığında otomatik olarak yüklenmesini sağlayabilirsiniz. Bunun için modül içerisinde, `FastReport.Utls.AssemblyInitializerBase` türünde bir public sınıf tanımlamalı ve bu sınıfın yapıcısında bağlantınızı kaydetmelisiniz:

```
public class AssemblyInitializer : FastReport.Utls.AssemblyInitializerBase
{
    public AssemblyInitializer()
    {
        FastReport.Utls.RegisteredObjects.AddConnection(typeof(MyDataConnection));
    }
}
```

Modülünüz yüklendiğinde, FastReport `AssemblyInitializerBase` türündeki sınıfları bulur ve başlatır.

Modülü FastReport'a Bağlama. Modülünüzü FastReport'a eklemenin iki yolu vardır:

Tasarımcı Arayüzü Üzerinden:

- Tasarımcıyı çalıştırın.
- Menüden "Görünüm" (View) seçeneğine tıklayın ve "Ayarlar..." (Settings...) seçeneğini seçin.
- Açılan pencerede "Modüller" (Modules) sekmesini seçin.
- .dll modülünüzü ekleyin.

Yapılandırma Dosyası Üzerinden: Modül adını FastReport yapılandırma dosyasına ekleyebilirsiniz. Bu dosya varsayılan olarak kullanıcı dizininde oluşturulur: mathematica.

```
C:\Documents and Settings\Kullanıcı_Adı\Local Settings\Application Data\FastReport\FastReport.config
```

Yapılandırma dosyasına, modülü `Plugins` etiketi içerisine ekleyin:

```
<?xml version="1.0" encoding="utf-8"?>
<Config>
  ...
  <Plugins>
    <Plugin Name="c:\Program Files\MyProgram\MyPlugin.dll"/>
  </Plugins>
</Config>
```

Sürüm Bilgileri

[2025.1 Sürümü](#)

[2024.2 Sürümü](#)

[2024.1 Sürümü](#)

[2023.3 Sürümü](#)

[2023.2 Sürümü](#)

[2023.1 Sürümü](#)

[2022.3 Sürümü](#)

[2022.2 Sürümü](#)

[2022.1 Sürümü](#)

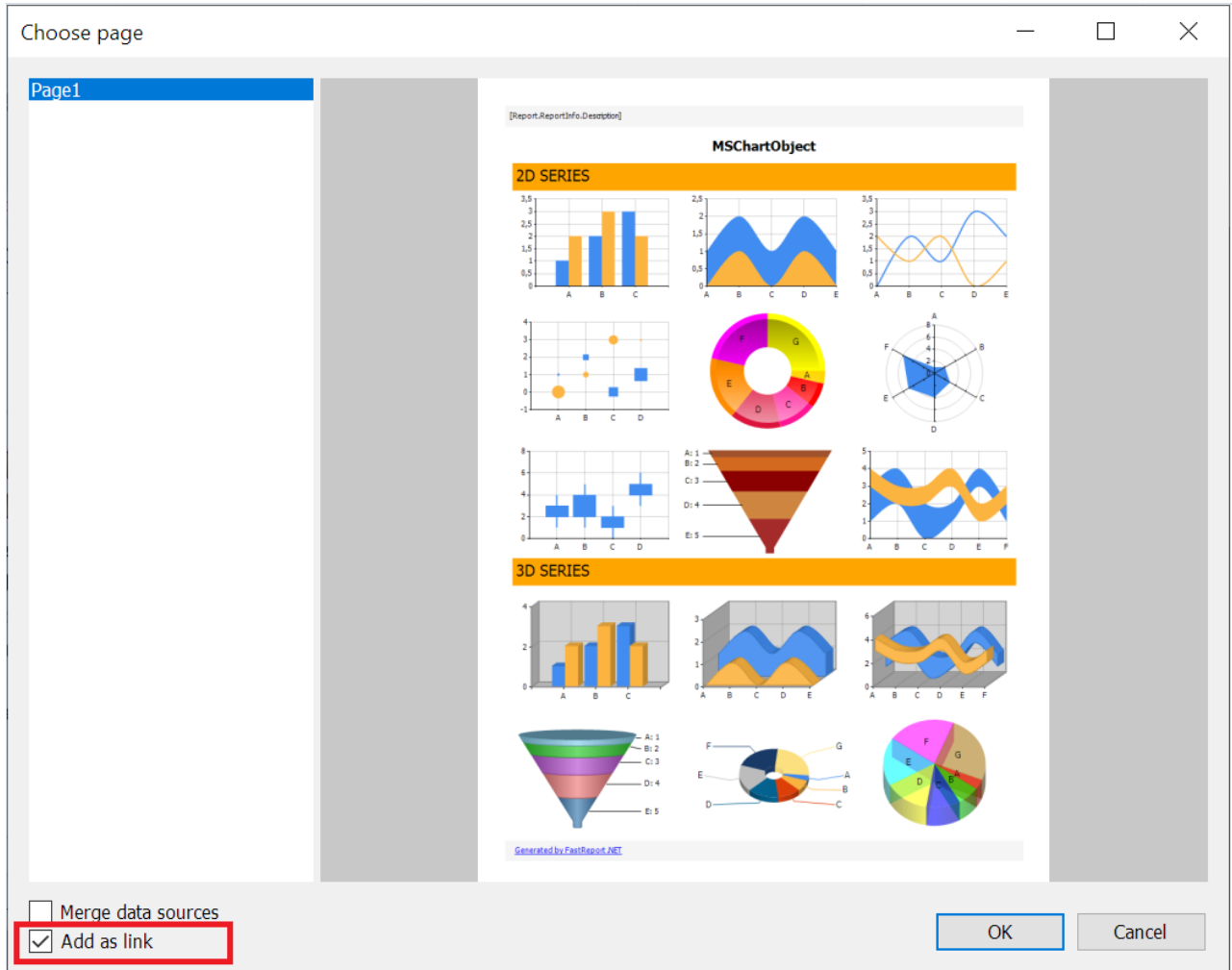
[2021.4 Sürümü](#)

Sürüm 2025.1

Yeni Olanaklar

Bir Rapor Sayfasını Bağlantı Olarak Ekleme

Önceki sürümlerde, başka bir raporun sayfasını bir rapora eklemek mümkün idi. Bu seçenek "**Dosya -> Sayfa Aç..**" menüsünde bulunabiliyordu. Varsayılan olarak, sayfanın bir kopyası rapora ekleniyordu.



Artık "**Bağlantı Olarak Ekle**" seçeneğini etkinleştirebilirsiniz; bu seçenek sayfanın kopyası yerine sayfaya bir bağlantı ekler. Bu, orijinal raporda bir sayfada değişiklik yaptığınızda, o sayfanın bağlantı olarak eklendiği tüm raporlarda değişikliklerin yansıtılacağı anlamına gelir. Ve tersi de geçerlidir; bir raporda bağlantısı bulunan bir sayfada değişiklik yapıldığında, orijinal raporda da güncelleme gerçekleşir.

Eşzamansız Rapor Hazırlama

`report.PrepareAsync()` yöntemi eklendi; bu, mevcut senkron `report.Prepare()` yöntemine ek olarak eşzamansız rapor hazırlamayı mümkün kılıyor. Bu yöntem aynı zamanda `CancellationToken` desteği sunarak, kullanıcıların ihtiyaç duyduklarında rapor hazırlama sürecini iptal etmelerine olanak tanır. Böylece, engellemesiz ortamlarda büyük raporlar için kontrol ve performans iyileştirilmiş olur. Gelecekte, ek asenkron erişim sağlayacak yeni yöntemlerle bu işlevsellik daha da geliştirilebilir.

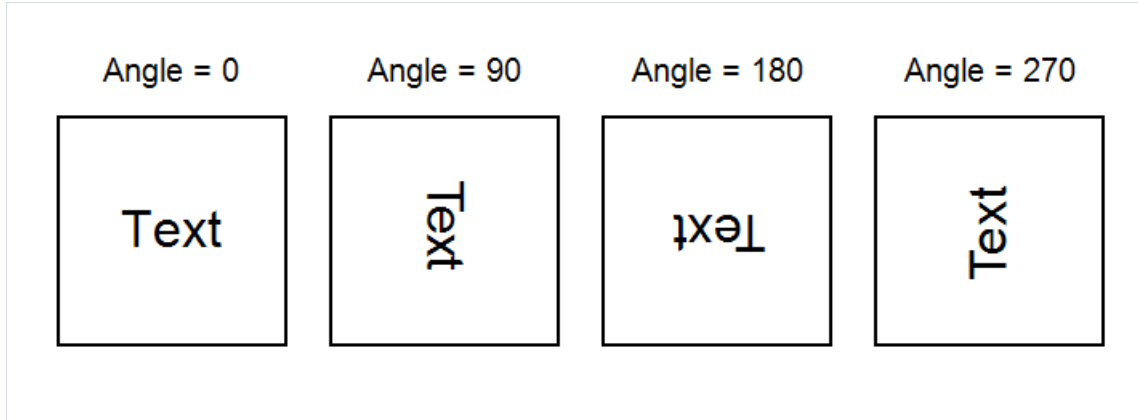
IfNull Fonksiyonu

```
object IfNull(object expression, object defaultValue)
```

İfadeleri değerlendirirken System.NullReferenceException hatasından kaçınmayı sağlayan yeni bir fonksiyon eklenmiştir. Fonksiyon iki parametre alır: ilki değerlendirilecek ifade, ikincisi ise varsayılan değerdir. Eğer ifade değerlendirilebiliyorsa, fonksiyon ifadenin sonucunu döndürür; aksi halde belirtilen varsayılan değeri döndürür.

TextRenderType.HtmlParagraph ile Metin Döndürme

TextRenderType.HtmlParagraph kullanılarak metin döndürme desteği eklendi. Daha önce, metin döndürme yalnızca diğer metin işleyici türleriyle mümkün olabiliyordu. Aşağıda metin döndürme örneklerini görebilirsiniz.

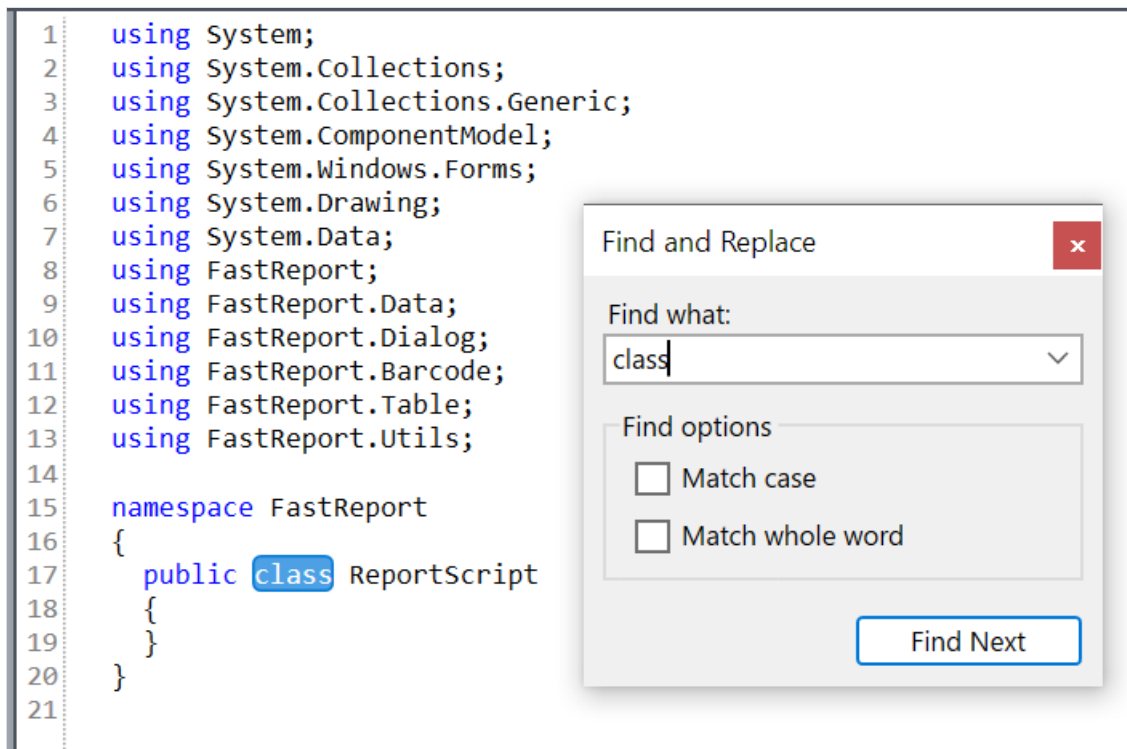


Ayrıca, bu tür metinler artık PDF'ye doğru şekilde aktarılmaktadır.

FastReport WPF ve FastReport Mono Kod Düzenleyicilerinde Metin Arama

Artık metni sadece FastReport .NET kod düzenleyicisinde değil, FastReport WPF ve FastReport Mono düzenleyicilerinde de arayabilirsiniz.

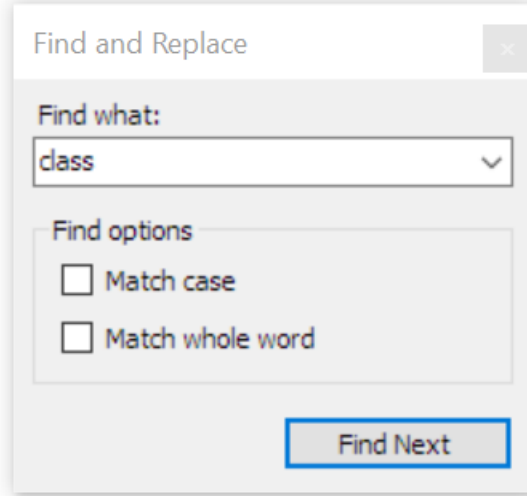
FastReport WPF kodunda metin aramanın bir örneği:



Ve FastReport Mono kod düzenleyicisinde:

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.ComponentModel;
using System.Windows.Forms;
using System.Drawing;
using System.Data;
using FastReport;
using FastReport.Data;
using FastReport.Dialog;
using FastReport.Barcode;
using FastReport.Table;
using FastReport.Utils;

namespace FastReport
{
    public class ReportScript
    {
    }
}
```



WebReport'te Yapılan Değişiklikler

Blazor WASM WebReport için Yerelleştirme Desteği

FastReport Blazor WebAssembly'deki WebReport arayüzü için yerelleştirme desteği tanıtıldı. Daha önce, yerelleştirme dosya tabanlı yöntemlerle yönetiliyordu, bu da WASM ortamıyla uyumlu değildi. Yeni yöntem olan `webReport.SetLocalization(Stream)`, yerelleştirmeyi bir akıştan yüklemeye olanak tanıyarak Blazor WASM uygulamalarıyla uyumlu hale getirir.

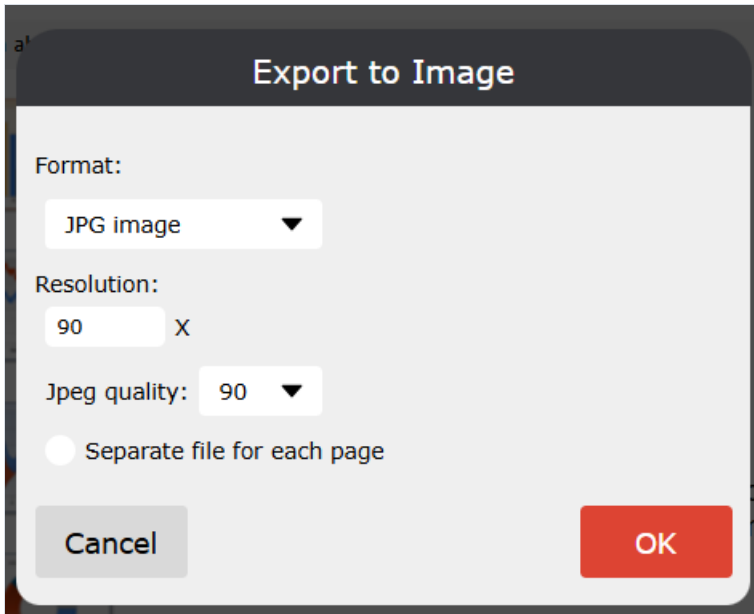
WebReport'e Görüntü Aktarımı

Raporun görüntülere aktarılması eklendi. Aktarımlar listesinde görüntülenmesi için aşağıdaki kodu ekleyin:

```
WebReport.Toolbar.Exports.ShowImageExport = true;
```



Gerekirse, görüntülere aktarım yapılandırmasını ayarlamak için WebReport seçeneğini etkinleştirmeniz gerekir: WebReport.Toolbar.Exports.Enable Settings. Etkinleştirdikten sonra, "dişli" simgesine tıklayarak açılan modal pencerede ayarları değiştirebilirsiniz.



Değişikliklerin Tam Listesi

```
[Engine]
+ RichObject'e PicturesInParagraph özelliği eklendi;
+ Asenkron rapor hazırlama için PrepareAsync() yöntemi eklendi;
+ Dizelerin dbtype ile uyumlu hale çevrilmesi eklendi;
+ Yazdırma ölçeği eklendi;
+ ToWords fonksiyonlarında ondalık sayıların kelimelere dönüştürülmesi eklendi;
+ İspanyolca için yerel tanımlayıcı eklendi: 22538 (Latin Amerika İspanyolcası) ve 3082 (İspanya İspanyolcası (Modern Sıralama));
+ İfadelerle çalışmak için yeni bir IfNull fonksiyonu eklendi. Fonksiyon, hesaplanan ifade null değilse sonucunu, aksi halde belirtilen varsayılan değeri döndürür;
+ RichObject içindeki resimlerin yatay konumunun hesaplanması uygulandı;
```

- + Sanal ana bilgisayar tarzında istek gönderebilme yeteneği eklendi;
- + TextRenderType = HtmlParagraph ile metin döndürme desteği eklendi;
- + Totals'ın "PrintOn" özelliği için header band kullanabilme yeteneği eklendi;
- * FastReport.Data.OracleODPCore'da Oracle.ManagedDataAccess.Core güncellendi;
- * GetConnection, OpenConnection ve Dispose yöntemleri virtual olarak işaretlendi;
- * Hyperlink.Value özelliği için gelen değerin null kontrolü eklendi;
- QRCodes sınıflarına TryParse isimli statik doğrulama yöntemleri eklendi;
- Metin kırılması sorunları düzeltildi;
- PageStart olayından sonra sayfa görünürlüğü değişikliği düzeltildi;
- Parametre tipine dönüşüm sorunu düzeltildi;
- Rapor betiğinde, değişken adında bulunan stop-word'lerin kontrolü düzeltildi;
- GrowToBottom etkin olan bir metin nesnesinin alt kenar çizgisi görünürlüğü düzeltildi;
- Grup DataBand'inde GrowToBottom seçeneği aktifken kenar çizgilerinin çiftlenme sorunu düzeltildi;
- SVGPictureObject içindeki child clipPath etiketlerinin render edilmesi kaldırıldı;
- FinishReport olayındaki hata düzeltildi;
- ContainerObject'e SubreportObject ekleme yeteneği kaldırıldı;
- GetAdapter'da ayarlanmıssa isteğin CommandType değişimi düzeltildi;

[Designer]

- + Başka bir rapordan sayfayı bağlantı olarak açabilme yeteneği eklendi;
- + span etiketine italic, bold, underline ve strikethrough font stilleri eklendi;
- + TreeView'de klavyeden girilen karakterle arama eklendi;
- + WPF ve Mono kod düzenleyicilerine arama fonksiyonu eklendi;
- * İndirilen fontların tekrarı kontrolü eklendi;
- * CurrencyFormat, NumberFormat ve PercentFormat sınıflarının yapıcılarındaki varsayılan özellik değerleri, sabit değerlerden CultureInfo.CurrentCulture değerlerine değiştirildi;
- Font seçimi açılır listesindeki Amiri, Cambria Math, DejaVu Math TeX Gyre fontlarının yanlış konumu düzeltildi;
- Border Editor üzerinden kenarların kaydedilmesinde System.NullReferenceException'a yol açan hata düzeltildi;
- Tasarımcıda SVG-görüntülerin yanlış görüntülenmesi düzeltildi;
- Code sekmesinde tek satırda bildirilen değişkenlerin tooltip'lerde görüntülenmesi düzeltildi;
- ExtraDesignWidth modunda sayfa kenar boşlukları uzunluğu düzeltildi;
- Uzun raporlar için tasarımcıdaki rehberlerin uzunluğu düzeltildi;
- Seçilen fontun açılır listede görüntülenmemesi hatası düzeltildi;
- Veri formatlarının yanlış uygulanması düzeltildi;
- Subreport nesnesi içeren bir band silinirken System.NullReferenceException hatasına yol açan hata düzeltildi;

[Preview]

- + PreviewControl'de Outline.Expand ve Outline.Width özellikleri eklendi;
- Boş SvgObject önizlemesi sırasında index out of range hatası düzeltildi;
- "Next" butonuna tıkladıktan sonra PreviewSearchForm'un kapanması düzeltildi;

[Exports]

- + Excel'e aktarılırken tüm rapor sayfalarını tek bir sayfada birleştirme yeteneği eklendi;
- + Excel aktarımında genel yerine özel bir format kullanma seçeneği eklendi;
- + Word aktarımında strikethrough metin formatlaması eklendi;
- + Word aktarımı için MemoryOptimized seçeneği eklendi; bu seçenek, MemoryStream yerine FileStream kullanımını sağlar;
- + PDF'ye aktarılırken TextRenderType = HtmlParagraph ile metin döndürme desteği eklendi;
- * Format görüntü ayarlamaları yapıldı - 'D' ve 'MMMM yyyy' formatları tarih olarak (mümkünse 'MM yyyy' formatında) görüntüleniyor, negatif desen '-' olan sayısal format standart Excel sayısal formatında sunuluyor;
- * Word'de PictureObject kenarının görüntü olarak aktarılması değiştirildi;
- * PDF'ye aktarım sırasında bellek kullanımı optimize edildi;
- * Tablo aktarım düzeni sabit olarak değiştirildi;
- HTML aktarımında HTML etiketlerinin render edilme sorunu düzeltildi;
- Negatif PDF özellik değerlerinin aktarım sorunu giderildi;
- Excel'e aktarım sonrasında tarayıcıda hücre kenarlarının rengi düzeltildi;
- Word ve PowerPoint'te hücre kenar çizgi stili düzeltildi;
- Word'e başlık ve altlıkta bulunan resimlerin aktarım sorunu düzeltildi;
- Geçici dosya silme hatası giderildi;
- HTML aktarımında satır yüksekliği hesaplama hatası düzeltildi;
- PDF'ye çift çizgi stilindeki kenarların yanlış aktarılması düzeltildi;
- HTML aktarımında şeffaflık hatası giderildi;
- HTML aktarımında <p> etiketinin yanlış görüntülenmesi sorunu düzeltildi;
- Word aktarımında "UseHeaderAndFooter" seçeneğinin varsayılan değeri düzeltildi;
- Word'e tablo şeklinde aktarımda resimlerin yanlış konumu düzeltildi;

- Excel'e aktarım sırasında TableObject sonrası nesne setlerinin satır yüksekliği düzeltildi;
- Yerine koyma için alternatif arama ile PDF'ye font aktarımında NullReferenceException hatası düzeltildi;

[WebReport]

- + Sekme içerisinde parametreler yerine rapor adının görüntülenmesi yeteneği eklendi;
- + WebReport yerelleştirmesini bir akıştan yüklemek için SetLocalization yöntemi eklendi;
- + WebReport'te raporu görüntü formatına aktarma yeteneği eklendi;
- WebReport'te özel uygulama stillerinden "box-sizing" kalıtım sorunu giderildi;
- WebReport önizlemesi sırasında IndexOutOfRangeException hatası düzeltildi;
- WebReport.Debug özelliği etkinleştirildiğinde raporda hata bilgisini görüntülemeyen hata giderildi;
- WebReport'te bir sekmeye tıklanırken NullReferenceException hatası oluşma sorunu giderildi;
- WebReport'te AdditionalFilter sıfırlama sorunu düzeltildi;
- Yatay oryantasyonda sayfalarla WebReport yazdırma sorunu düzeltildi;

[Online Designer]

- + Tablo güncelleme yöntemi eklendi;
- Online Designer'da boş SVG nesnesinin önizlemesi düzeltildi;

[.Net Core]

- + FastReport Core'da MS SQL stored prosedürleri için yöntemler eklendi;

[Common]

- + Kod aracılığıyla parametre ifadesi ayarlamak için yeni bir yöntem eklendi;
- + Kurulum imzalanırken zaman damgası eklendi;

[Extras]

- + Oracle'da stored prosedürlere bağlantı yeteneği eklendi;
- * Firebird.Client sürümü 10.0.0'a güncellendi;
- * Npgsql (Postgres) ve System.Data.SqlClient gibi güvenlik açığı bulunan paketler güncellendi;
- * Linter bağlantısında "Advanced" butonuna basıldığında hata mesajı metni değiştirildi;
- Rapor nesnesi formlarının tasarımcısında eksik menü hatası giderildi;
- Postgres'in "character varying" tipi ile ilgili hata düzeltildi;

[Demos]

- demo-rapor Barcode.frx hatası giderildi.

Sürüm 2024.2

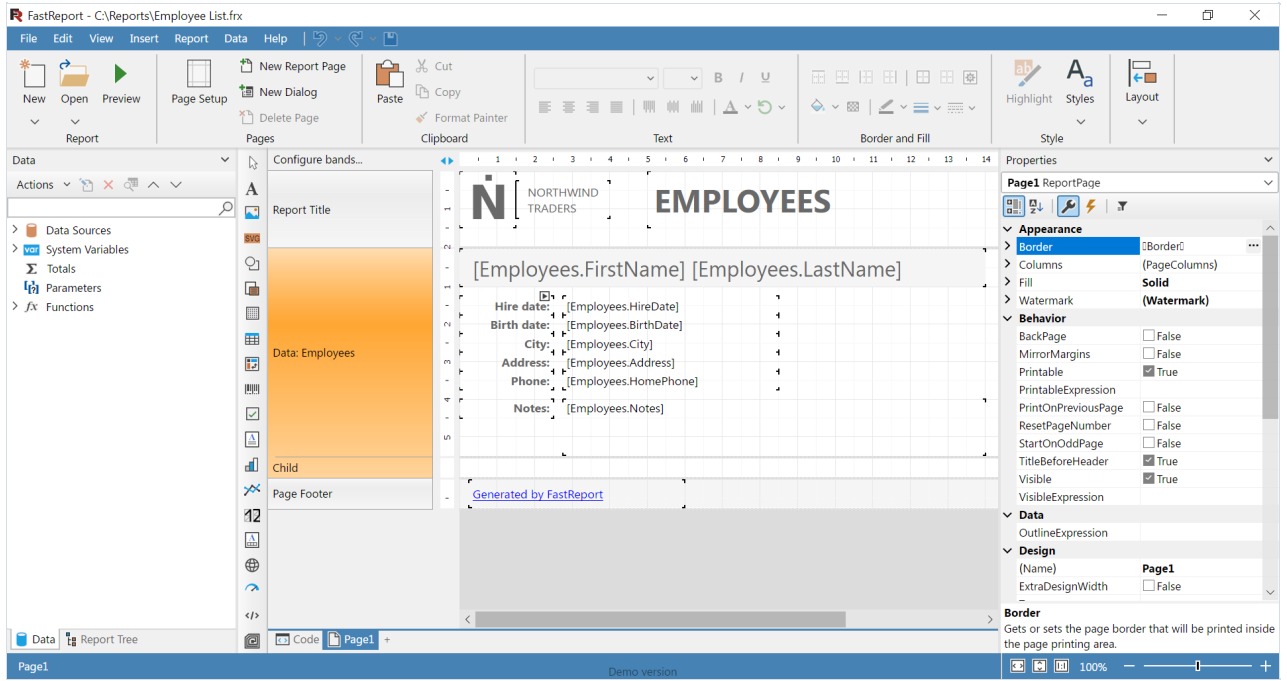
Yeni FastReport Avalonia Bileşeni

FastReport .NET bileşen serisine yeni bir ek geldi: FastReport Avalonia. Bu, Avalonia UI kullanarak çapraz platform uygulamaları geliştirirken rapor oluşturmaya mümkün kılan çok yönlü bir kütüphanedir. Bu bileşen, macOS, Linux ve Windows için birleşik bir kullanıcı arayüzü ile uygulama geliştirmeye olanak tanır. FastReport Avalonia, x64, x86 ve arm64 işlemci mimarileriyle uyumludur. Ayrıca, Avalonia UI ve .NET 6 ve üzeri sürümleri destekler.

FastReport Avalonia, C#'daki birleşik FastReport ekosisteminin bir parçasıdır. Diğer ürünlerde oluşturulan raporlar FastReport Avalonia'da çalışacak, aynı şekilde FastReport Avalonia'da oluşturulan raporlar diğer ürünlerde de kullanılabilir.

Ürün, rapor geliştirme çekirdeğini, tasarımcıyı ve tanıdık bir arayüze sahip görüntüleyiciyi içermektedir.

Rapor tasarımcısı şu şekilde görünüyor:



Görüntüleyici ise şu şekilde:



Yeni bileşen hakkında daha fazla bilgi için lütfen makaleleri okuyun.

Yeni İmkanlar

ReportPage Nesnesinin CreatePage Olayı

ReportPage nesnesinde, sayfa render edilmeden önce tetiklenen StartPage olayı bulunmaktadır. Bu olay, rapordaki her şablon sayfası için bir kez çağrılır. Artık, hazırlanan raporda bir sayfa oluşturulduğunda tetiklenen yeni bir olay olan CreatePage de bulunmaktadır. StartPage olayının aksine, CreatePage, şablon sayfasına karşılık gelen her hazırlanan sayfa için daha sık çağrılır. Her iki olaya da, standart işlemlere ek olarak ek işlemler gerçekleştirmek amacıyla handler atanabilir.

CreatePage hakkında daha fazla bilgi için ilgili makaleye bakın.

Word 2007'de İhracat İyileştirmeleri

Word ihracatında önemli değişiklikler yapıldı. Performansı artırılarak daha hızlı çalışması sağlandı. "Satır Yüksekliğini Korum" ve "Word Sayfalarının Üstbilgi ve Altbilgilerini Kullan" gibi yeni seçenekler eklendi. Ayrıca, ihracat işlevselliğini önemli ölçüde etkileyen çeşitli hatalar giderildi.

SVG'ye Hiperlink ve Yer İmi İhracatı

Bu güncelleme ile rapor nesnelerinin hiperlinkleri ve yer imleri, SVG formatındaki görsellere de aktarılmaktadır.

PostgreSQL Fonksiyonları ve Görünümleri Desteği

Postgres veritabanlarına bağlanırken, raporlarınızda fonksiyon ve görünüm verilerini kullanma imkanı sunulmaktadır. Bunun için FastReport.Data.Postgres eklentisini kullanın.

FastReport.Core.Skia için HTML Eklentisi Desteği

FastReport.Core.Skia'ya, HTML Eklentisi desteği eklendi. Artık raporlarınıza kolayca HTML içeriği ekleyerek, rapor oluşturma ve sunumunu geliştirebilirsiniz.

Bu işlevi kullanmak için NuGet üzerinden FastReport.Plugins.HtmlObject paketini yükleyin.

Blazor WebAssembly'de Değişiklikler

.NET 8 için Webcil Desteği

.NET 8’den itibaren, varsayılan olarak gerekli tüm kütüphaneler, .dll yerine Webcil formatında (.wasm) paketlenmektedir. Bu, daha güvenli ve web dostu bir formattır (daha fazla bilgi için bkz.).

Ancak, rapor betiğini derlemek için bu kütüphaneleri kullanmamız gerekmektedir. Daha önce kullanıcılarımıza .wasm paketlemesini devre dışı bırakmalarını öneriyorduk. Mevcut sürümden itibaren, FastReport.Blazor.Wasm, Webcil kaynakları ile otomatik olarak çalışabilir ve rapor betiğini derlemek için ön ayar yapmaya gerek kalmadan işlemi gerçekleştirebilir.

Gerekli SkiaSharp Kaynaklarının Otomatik Eklenmesi

Daha önce, FastReport.Blazor.Wasm’i kullanırken, kullanıcıların Blazor WebAssembly projelerine gerekli SkiaSharp ve HarfBuzzSharp kaynaklarını manuel olarak eklemesi gerekiyordu. Sorun, bu kaynakların .NET sürümüne ve WASM’de çoklu iş parçacığı kullanımına bağlı olarak değişmesiydi. Kullanıcıların ihtiyaç duydukları kütüphaneleri manuel olarak seçmeleri büyük bir karmaşıklığa neden oluyordu.

Mevcut sürümden itibaren, FastReport.Blazor.Wasm, ortamınıza bağlı olarak gerekli kaynakları otomatik olarak belirleyip uygulamanıza ekleyebilecektir. Ancak, hâlâ manuel seçim yapmak isterseniz, .csproj projenize aşağıdaki kodu ekleyerek otomatik davranışı devre dışı bırakabilirsiniz:

```
<PropertyGroup>
  <ShouldIncludeNativeSkiaSharp>False</ShouldIncludeNativeSkiaSharp>
</PropertyGroup>
```

.NET Core 2.0, 2.1, 2.2, 3.0, 3.1, .NET 5 Desteğinin Kaldırılması

.NET dünyasına sürekli eklenen yeni teknolojilerle uyum sağlamak amacıyla, artık kütüphanelerimizde eski .NET Standard 2.0–3.1 ve .NET 5 uyumluluk katmanlarına destek verilmeyecektir. Minimum desteklenen sürüm artık .NET 6, ve minimum desteklenen framework ise .NET Framework 4.6.2’dir.

Değişikliklerle ilgili daha fazla bilgi için haberleri inceleyin.

Tüm Değişiklikler

```
[Engine]
+ ReportPage nesnesine CreatePage olayı eklendi;
* RTF dönüştürülürken, sekmelerden sonra gelen önemsiz boşluklar atılmaktadır;
- RichObject içeriği hatalı olduğunda istisna yakalama eklendi;
- TextObject’in sayfa bölünmesinde üst ve alt dolgu kaldırıldı;
- RichObject’deki dikey girintiler düzeltildi;
- Birleştirilmiş sütunlu bir satırı takip eden satırın görüntülenmesindeki hata giderildi;
- MSChartObject içeren TableObject ile rapor hazırlanırken oluşan istisna düzeltildi;
- RTF ayrıştırıcısında boşlukların kaybolması sorunu giderildi;
- GrowToBottom kullanıldığında alt sınır çizgisinin görüntülenmesindeki hata düzeltildi;
- RTF ayrıştırıcısında Portekizce dil desteği düzeltildi;
- Rapor asenkron hazırlanırken Report.IsPrepared parametresinin yanlış değer döndürme hatası giderildi;
- RichObject’ten TableObject’e dönüşümdeki çeviri hatası düzeltildi;
- Farklı sayfalar seçildiğinde yazdırma sırasında oluşan hata giderildi;
- Graphics.Path.AddBeziers çağrılırken oluşan IndexOutOfRangeException hatası düzeltildi;
- Bir hiperlinkten JS kodu çağırma yeteneği ile ilgili güvenlik açığı giderildi;
- RTF dönüştürürken varsayılan sekme ayarındaki hata düzeltildi;
```

```
[Designer]
+ Contains (string, string) fonksiyonu eklendi; bu fonksiyon, bir metnin başka bir alt metni içerip içermediğini belirler;
* Sorgu sihirbazındaki yinelenen parametre isimlerine ilişkin uyarı mesajı metni değiştirildi;
* SQL sorgusundaki aynı isimli parametreler konusunda hata yerine uyarı formu gösterildi;
* SwissQR’de değişiklikler yapıldı: “Amount” alanına islenmesi değiştirildi; “Currency” alanı artık
```

SwissSign de değişiklikler yapıldı. Amount alanının işlenmesi değiştirildi, Currency alanı artık metin olarak işleniyor; Alıcı, Ödeyen ve "Link" alanındaki veriler veritabanından işlenmeye eklendi;

- Datamatrix fırça rengindeki hata düzeltildi;
- PictureBox editöründeki hatalar giderildi;
- Format editöründeki hata düzeltildi;
- Diyalog formunun yeniden boyutlandırılmasında yaşanan hata giderildi;
- Döndürülmüş ITF14 barkodunun çizimindeki hata düzeltildi;
- "Tümünü Seç" kısayol tuşunun çalışmasındaki hata giderildi;
- İfade değiştirildiğinde formatın sıfırlanmasıyla ilgili hata düzeltildi;
- Kaynak verilerin satırlara ayrılmasıyla ilgili hatalar giderildi; her iki ayırıcı seçeneği (\r\n ve \n) artık desteklenmektedir;
- QR kodu oluşturma sürecinde, satır sonundaki ekstra \r\n karakterleri kaldırılmaktadır;
- Veri kaynağı tablosunun SelectCommand'ini düzenlerken oluşan NullReferenceException hatası giderildi;
- AdvMatrix'te metin kaydırması sorunu giderildi;
- RFIDLabel nesnesinin sağ tıklama menüsündeki hata düzeltildi;
- Klasik bant görünümünde bant başlığının görüntülenmesindeki hata giderildi;
- hh:mm zaman formatı kullanılırken oluşan hata düzeltildi;
- JSON'a bağlanırken meydana gelen NullReferenceException hatası giderildi;
- Shift tuşuna basılıyken nesnelerin yeniden boyutlandırılması sırasında oluşan hata düzeltildi;
- DontEditCode kullanıldığında ortaya çıkan istisna giderildi;
- Diagonal özelliği False olduğunda, yatay veya dikey çizginin Height ya da Width özelliği değiştirildiğinde oluşan hatalı çizgi davranışı düzeltildi;

[Preview]

- Ön izleme penceresindeki arama diyalogundaki hata giderildi;
- Rapor nesnelerinin sayfa dışına taşarak render edilme sorunu düzeltildi;
- RichObject'te satır başında bulunan üst simge veya alt simge metninin hatalı görüntülenmesi düzeltildi;
- LineHeight özelliği kullanıldığında ön izlemeye dikey dolguların görüntülenmesindeki hata giderildi;

[Exports]

- + Word'e ihracat sırasında "Word Sayfalarının Üstbilgi ve Altbilgilerini Kullan" seçeneği eklendi;
- + docx'e ihracat hızı artırıldı;
- + Word 2007'e ihracat için "Satır Yüksekliğini Koru" seçeneği eklendi;
- + SVG'ye hiperlink ve yer imlerinin ihracatı eklendi;
- Döndürülmüş svg görüntüsünün pdf'e ihracatında oluşan hata giderildi;
- FastReport.Skia'da PdfA_1a standardına uygun PDF'e, şeffaf arka plana sahip resimlerin hatalı ihracat hatası giderildi;
- Boş hücrede yazı tipi sıfırlanması hatası, Word'e rapor ihracatı sonrasında düzeltildi;
- JSON dosyasına ihracat sırasında oluşan hata giderildi;
- Word 2007'e ihracat sırasında, watermark ve "Page Title" bandı kullanılırken yaşanan hata düzeltildi;
- SVG ihracatında (metin içerisinde hatalı cr/lf dizileri bulunması durumunda asılı kalma) oluşan hata giderildi;
- Excel'e ihracat sırasında tablo satır yüksekliğinin yanlış hesaplanması düzeltildi;
- Vektörel grafiklerin PDF'e ihracatında oluşan hata giderildi;
- Birleştirilmiş hücreler içeren tablolar için SVG ihracatındaki hata düzeltildi;
- Word 2007'de ihracat edilen raporların açılmasındaki hata giderildi;
- Katmanlı HTML ihracatında RichObject'te bulunan resmin yanlış siyah arka planı sorunu giderildi;
- ImageExport (OpenSource) içinde gereksiz dolgu giderilerek paddingNonSeparatePages değişkeninin değeri düzeltildi;

[WebReport]

- + FastReport.Blazor.Wasm'da Webcil format desteği eklendi;
- + WebReport'e diyalog form başlığı eklendi;
- + WebReport ön izlemesinde rapor derleme hatası hakkında detaylı açıklama eklendi;
- * WebReport'te standart resim mantığı yeniden düzenlendi. Artık resimler, sunucudan çok sayıda istekle yüklenmek yerine doğrudan rapora dahil edilmektedir;
- Word 2007 üzerinden rapor ihracatı yapılırken, belgedeki tabloların düzenlenmesinde oluşan hata giderildi;

[Extras]

- + FastReport.Core.Skia için FastReport.Plugins.HtmlObject desteği eklendi;
- + PostgreSQL konektöründe, görünüm ve fonksiyonlar desteği eklendi (Extras/Core/FastReport.Data/FastReport.Data.Postgres);
- + SQLite'ye bağlanırken, .db ve .db3 dosya uzantılarını kapsayan yeni bir dosya filtresi eklendi; bu filtre varsayılan olarak seçilmektedir;
- TargetFramework net6.0 ve üzeri kullanılan FastReport .NET eklentilerinin importunda yaşanan hata giderildi;
- HTMLObject eklentisi içeren raporlarda nesnelerin hizalanmasında yaşanan sorun giderildi;

[Mono]

- * RichObject üzerindeki araç ipucu metni Mono olarak değiştirildi;
- Tasarım modunda kod sayfalarında maksimum metin uzunluğuyla ilgili hata giderildi.

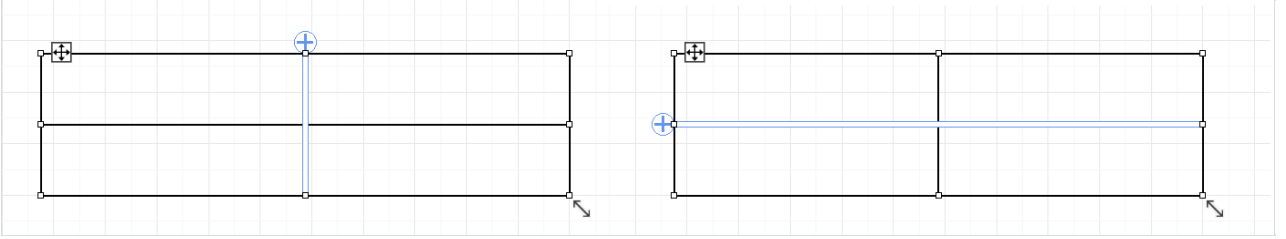
Sürüm 2024.1

Yeni Olanaklar

Tablo Nesnesi ile Geliştirilmiş Çalışma

Rapor tasarımcısı ile çalışma artık daha kolay ve kullanışlı hale geldi. "Tablo" nesnesiyle ilgili yeni yetenekler eklenmiştir.

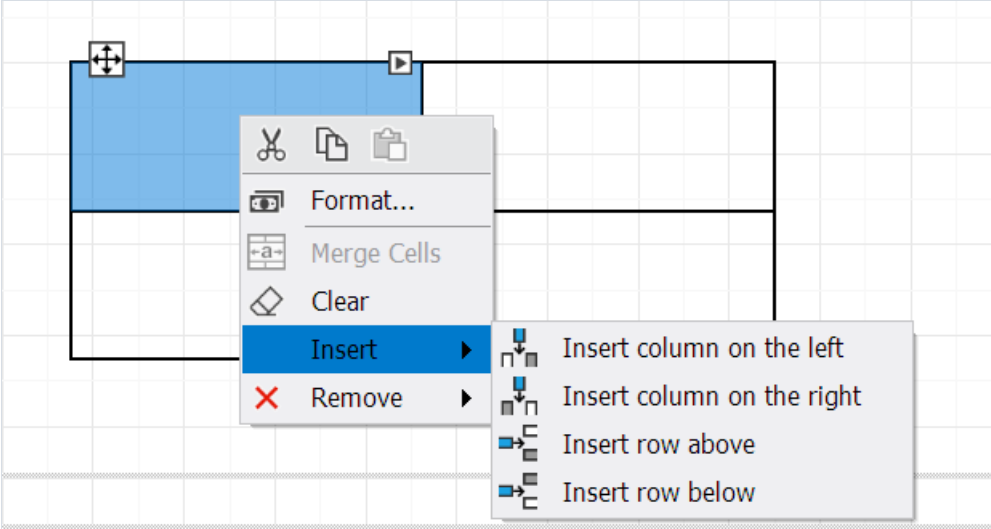
1. Sütun ve Satır Ekleme Hızlıca sütun ve satır ekleyin. Tabloyun solundaki bir satır sınırına ya da üstündeki bir sütun sınırına işaret ederseniz, yeni bir satır veya sütunun ekleneceği yeri gösteren koşullu bir görüntü belirir. Ayrıca, tıkladığınızda tabloya yeni bir satır veya sütun ekleyen bir düğme bulunmaktadır.



Not: Tablo etkin (seçilmiş) olmalıdır; aksi halde yeni kontroller görünmeyecektir.

2. Satır Yüksekliği ve Sütun Genişliğini Değiştirme Artık fare ile sütun veya satır sınırını sürükleyerek ilgili sütun genişliği veya satır yüksekliğini değiştirebilirsiniz.

3. "Ekle" Açılır Menüsü Bir hücreye sağ tıkladığınızda, hücrenin yanına yeni bir sütun veya satır eklemenizi sağlayan bir açılır liste görünür.



4. Kısayol Tuşları Hücre metnini kopyalamak ve bir hücreye metin yapıştırmak için Ctrl+C ve Ctrl+V kısayol tuşlarını kullanabilirsiniz.

Yeni tablo yetenekleri hakkında daha fazla bilgiyi makalede okuyabilirsiniz.

Metin Nesnelerinin Birleştirilmesi

Artık aynı metne sahip metin nesnelerini birleştirmek için bir mekanizma mevcut. Bu amaçla, "Text" nesnesine yeni bir MergeMode özelliği eklenmiştir; bu özellik, birleştirme modunu yapılandırmanıza olanak tanır. Yeni özellik, Merge modunda kullanılan Duplicates özelliğine çok benzer çalışır, ancak önemli farklar vardır:

- Duplicates: Duplicates yalnızca "Data" bandında yer alan tek bir nesne ile çalışır. Örneğin, Data1 bandında Text1 adlı bir metin nesnesi varsa ve Duplicates özelliği Merge olarak ayarlanmışsa, rapor oluşturulurken Data1'in ilk yinelenmesinde ilk kayıttaki Text1'de "10" görüntülenir; ikinci yinelemede de aynı metin tekrar gösterilir. Sonuç olarak, iki Text1 örneği birleştirilir ve "10" metni yalnızca bir kez görüntülenir.

- MergeMode: MergeMode, Duplicates'ten farklı olarak, farklı metin nesnelerinin örneklerini hem dikey hem yatay olarak birleştirebilir. Örneğin, rapor oluşturulurken "10" metni, yan yana bulunan Text1 ve Text2'de gösterilecektir. Bu durumda, Text1'in MergeMode özelliği Horizontal olarak ayarlanmıştır; böylece metin nesneleri birleştirilir ve "10" metni tek sefer görüntülenir.

Yeni özellik hakkında daha fazla bilgiyi makaledeki bağlantıdan edinebilirsiniz.

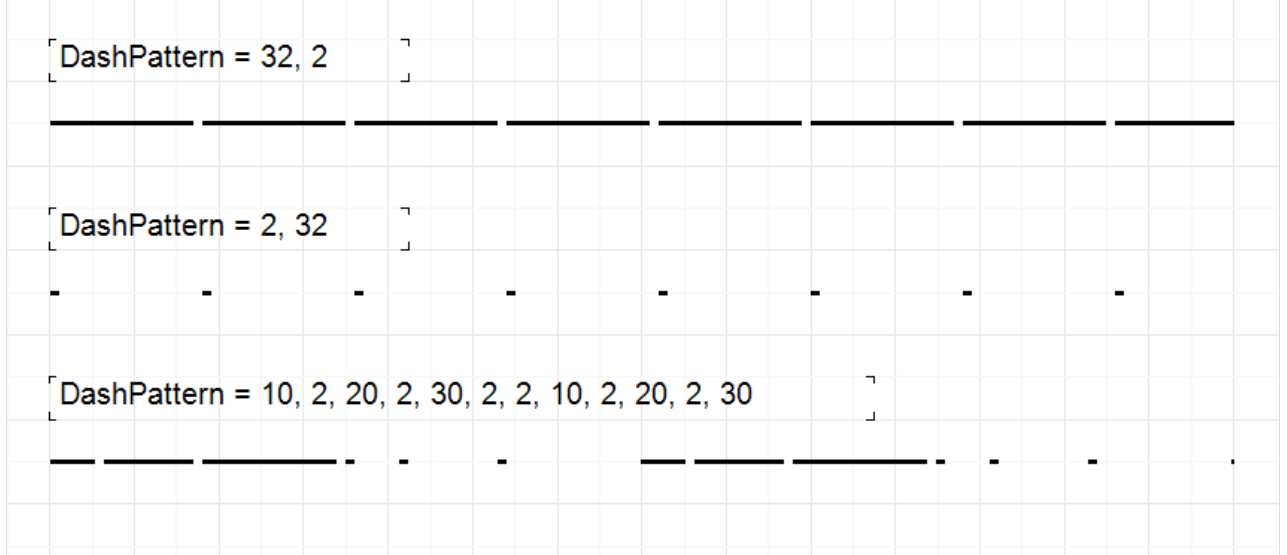
Özel Çizgi Stilleri

LineObject, ShapeObject, PolyLineObject ve PolygonObject nesneleri için yeni bir DashPattern özelliği eklenmiştir; bu özellik, çizgi stillerinizi oluşturmanıza olanak tanır. Daha önce, bu nesnelerin çizgi stili Border.LineStyle özelliğiyle ayarlanıyordu ve yalnızca şu altı stil mevcuttu: Solid, Dash, Dot, DashDot, DashDotDot ve Double. Yeni özellik sayesinde, çizgilerin ve boşlukların uzunluklarını sırasıyla belirleyecek değerler koleksiyonu tanımlayabilirsiniz.

Örneğin, 5, 4, 3 ve 2 değerleri ile; 5 birim uzunlukta bir çizgi, 4 birim boşluk, 3 birim çizgi ve 2 birim boşluktan oluşan bir desen ayarlanır. Ardından, değerler 5'ten başlayarak dairesel olarak tekrarlanır. Burada ölçü birimi Border.Width'dir.

DashPattern koleksiyonunda en az bir değer varsa, bu yeni mekanizma çalışır ve Border.LineStyle özelliği göz ardı edilir. Eğer DashPattern koleksiyonu boşsa, Border.LineStyle mekanizması devreye girer.

Aşağıda bazı örnekleri görebilirsiniz:



Çizgilerin ayarlanmasına ilişkin talimatları aşağıdaki bağlantıda bulabilirsiniz.

"Picture" Nesnesinin Şeklini Değiştirme

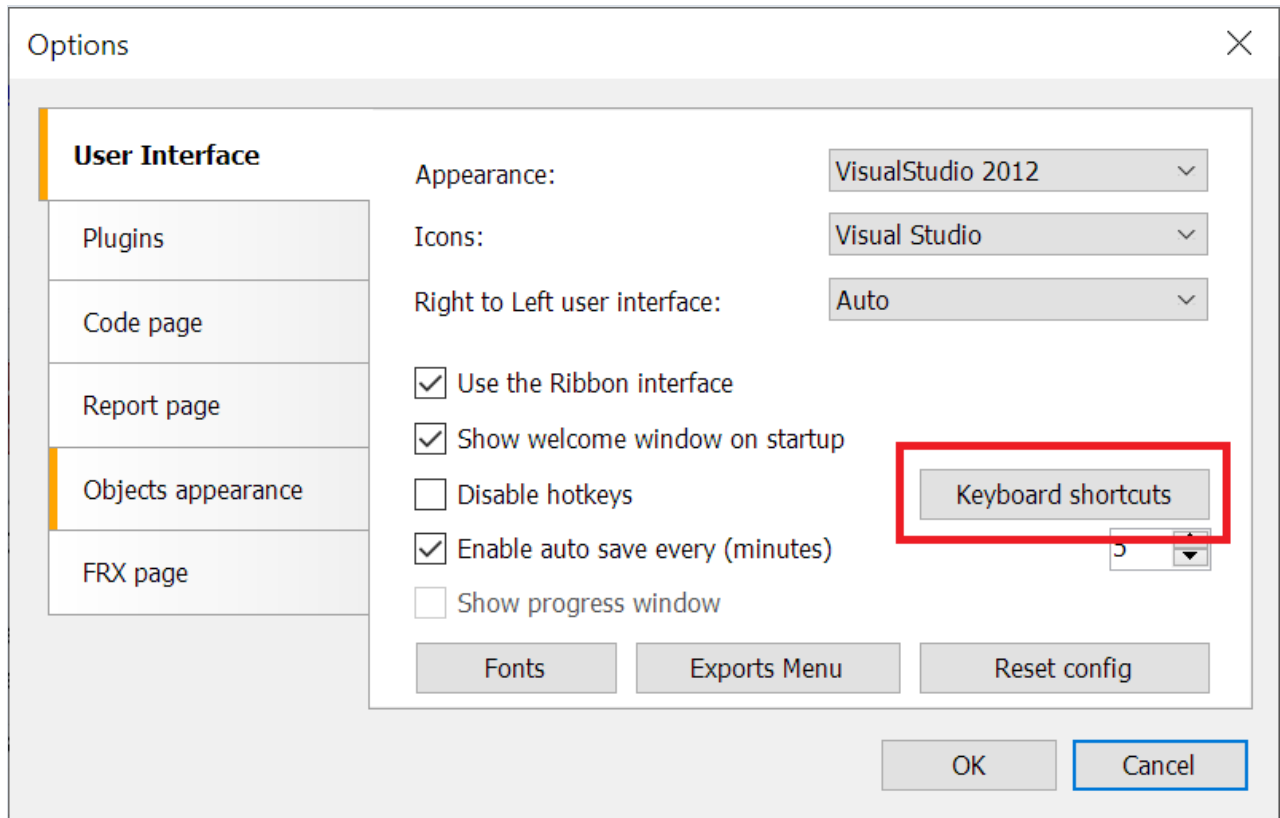
Artık "Picture" nesnesinin şeklini değiştirmek mümkün. PictureObject'e eklenen yeni Shape özelliği ile şu şekillerden birini belirtebilirsiniz: dikdörtgen (varsayılan), yuvarlak dikdörtgen, elips, üçgen ve elmas.



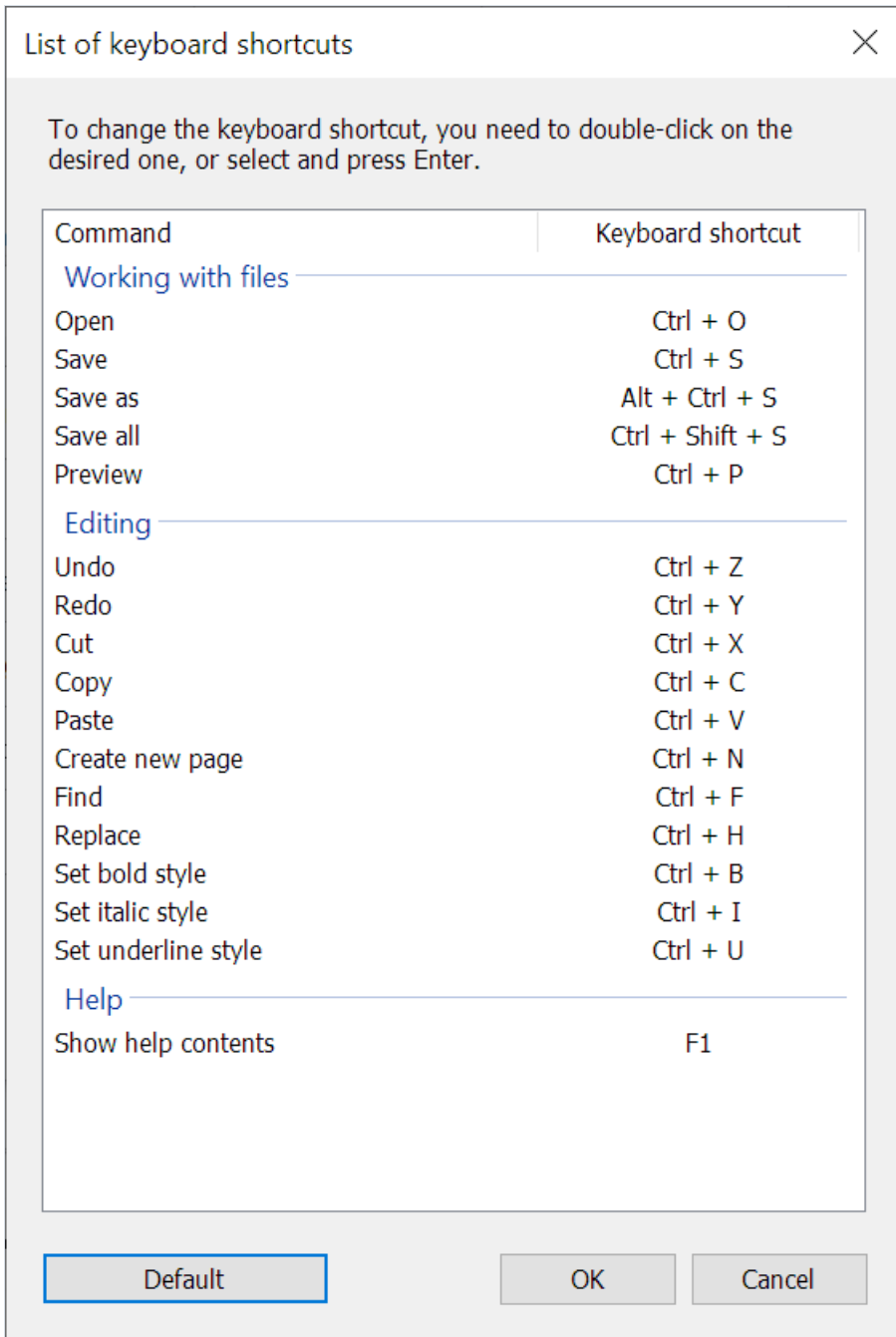
Detaylı bilgi için makaleye bakabilirsiniz.

Kısayol Tuşu Kombinasyonlarının Ayarlanması

Artık kısayol tuşu kombinasyonlarını dilediğiniz gibi özelleştirebilirsiniz. "Dosya aç", "Dosyayı kaydet", "Rapor hazırla" gibi işlemler için komutları yapılandırabilirsiniz. Bunun için tasarımcı ayarlarının "Interface" sekmesine yeni bir düğme eklenmiştir.



Bu düğmeye basıldığında, kısayol tuşu kombinasyonlarını ayarlamanıza olanak tanıyan bir pencere açılır.



Aşağıdaki tabloda, işlemler ve atanan klavye kısayolları yer almaktadır. İstediğiniz satıra çift tıklayarak kombinasyonu değiştirebilir; ayrıca yukarı ve aşağı tuşlarıyla tablo içinde gezinebilir ve Enter tuşuna basarak değişiklik yapabilirsiniz. Tüm kombinasyonları varsayılan değerlere döndürebilirsiniz.

Kısayol tuşlarının ayarlanması hakkında daha fazla bilgiyi makalede bulabilirsiniz.

.NET 8 Desteği

FastReport .NET, FastReport.Core, FastReport.Core.Skia ve FastReport.WPF için .NET 8 desteği eklendi. Bu platform, uygulama performansını artırır ve projelerinize birçok yeni özellik ekler.

FastReport.Web'de .NET Standard 2.0 Desteğinin Kaldırılması

.NET dünyasına sürekli eklenen daha fazla teknolojiyi kapsayabilmek amacıyla, Web entegrasyon kütüphanemiz FastReport.Web'de (WebReport Core/Skia) eski .NET Standard 2.0 uyumluluk katmanından vazgeçmeye karar verdik. Bu ürün için artık desteklenen minimum TargetFramework sürümü .NET Core 3.1 ve üstü (yani .NET 5, 6, 7 ve 8 dahil) olacaktır. FastReport.Core ve FastReport.Core.Skia ise herhangi bir değişiklik olmaksızın .NET Standard 2.0 desteğine devam edecektir.

FastReport.Core için ODBC Konektör Desteği

Kullanıcılarımız uzun süredir, çapraz platform ürünlerimizde ODBC protokolü aracılığıyla veritabanlarına bağlanma yeteneğinin eklenmesini talep ediyordu. Bu özelliği daha önce sadece FastReport .NET ve FastReport WPF’de mevcuttu. Bu güncelleme ile FastReport.Core ve FastReport.Core.Skia’da da kullanılabilir hale gelmiştir. Kullanmak için, projenize FastReport.Data.Odbc eklentisini ekleyip aşağıdaki kodla kaydedin:

```
FastReport.Utils.RegisteredObjects.AddConnection(typeof(OdbcDataConnection));
```

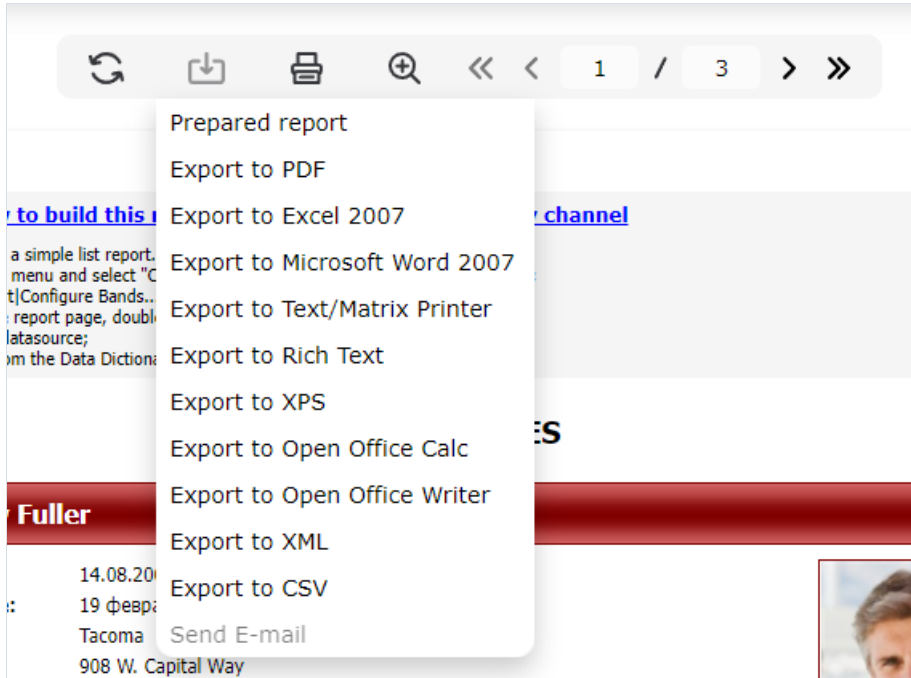
WebReport'deki Değişiklikler

WebReport'e E-posta İle Dışa Aktarma

Artık WebReport, raporları e-posta ile gönderme işlevine sahiptir. Bu özelliği etkinleştirmek için FastReport servislerini kaydederken SMTP sunucu parametrelerini yapılandırmanız gerekmektedir. Aşağıdaki kodu ekleyin:

```
services.AddFastReport(options => options.EmailExportOptions = new FastReport.Web.EmailExportOptions
{
    Address = "SomeAddress@example.com",
    EnableSSL = true,
    Host = "Host",
    MessageTemplate = "Message template here",
    Name = "John",
    Password = "password",
    Port = 25,
    Username = "Username"
});
```

Bunun ardından, WebReport.Toolbar.Exports.ShowEmailExport seçeneğini etkinleştirin; böylece kullanıcılar raporları e-posta ile gönderebilecektir:



"Send by mail" düğmesine tıkladığınızda, kullanıcıdan uygun bir modal pencere aracılığıyla mesajı yapılandırması istenecektir.

Send E-mail

E-mail

Address:*

1 E-mail; 2 E-mail; 3 E-mail...

Subject:

Message:

Attachment:

FPX

Name attachment file:

Simple List

Cancel

OK

Blazor WebAssembly'de Yazdırma

WebReport artık Blazor WebAssembly'de rapor yazdırmanıza olanak tanır. Bu özellik varsayılan olarak etkindir; devre dışı bırakmak isterseniz aşağıdaki kodu kullanın:

```
webReport.Toolbar.ShowPrint = false;
```

Raporlarınızı artık doğrudan Blazor WebAssembly üzerinden yazdırabilirsiniz.

BlazorWasm .Net 7

About Microsoft Chart

Advanced Matrix

AdvMatrix - Collapse + Sort

AdvMatrix - Items comparison

AdvMatrix - Order details

AdvMatrix - Sort Year by total

AdvMatrix - Sparkline, Gauge

AdvMatrix - Stepped layout

AdvMatrix - Two data cells

AdvMatrix - User function

Alternate Color Each Row

Badges

Barcode

Box

Business Objects

Cascaded Data Filtering

Print from browser

Print from Adobe Reader

[Learn how to build this report on the Fast Reports Aca...](#)

Demonstrates a simple list report. To create it:

- go to "Data" menu and select "Choose Report Data..." item to select a datasource;
- go to "Report/Configure Bands..." menu to create the band structure;
- return to the report page, doubleclick the data band to show its editor;
- choose the datasource;
- drag data from the Data Dictionary window to the band.

EMPLOYEES

Andrew Fuller

Hire date: 8/14/2009

Birth date: Saturday, February 19, 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: 11/15/2011

Birth date: Monday, January 27, 1986

City: London

Address: 7 Houndstooth Rd.

Phone: (71) 555-4444

Notes:

Anne has a BA degree in English from St. Lawrence College. She is fluent in French and German.



Janet Leverling

Hire date: 4/1/2009

Birth date: Tuesday, August 30, 1983

City: Kirkland

Address: 722 Moss Bay Blvd.



Değişikliklerin Tam Listesi

www.fast-report.com

101 / 158

[Engine]

- + Metin nesnelerinin birleştirilmesi eklendi;
- + PictureBox'ın şeklini değiştirme yeteneği eklendi;
- + Özel çizgi stilleri oluşturma yeteneği eklendi;
- * Artık fontlarla çalışma engelleme olmadan yapılıyor;
- TextRenderer = HTMLParagraph olduğunda TextObject'ın sınırlarının dışına çıkan metin hatası giderildi;
- PrivateFontCollection'dan font oluşturma hatası düzeltildi;
- RichObject'te hatalı metin rengi düzeltildi;
- RichObject ile resim arasındaki kesinti hatası giderildi;
- DetailedControl özelliği belirtilmiş DateTimePicker nesnesinde odak kaybı hatası düzeltildi;
- Barkodlarda (HiDPI'de görüntüleme, PDF'ye dışa aktarma) hata giderildi;
- HTMLTextRenderer'deki girinti hatası düzeltildi;
- Yanlış RichObject kırılmaları giderildi;
- HTMLTextRenderer ile render yapılırken TextObject'ın kesilme sorunu düzeltildi;

[Designer]

- + SberQR üzerindeki işaretin kaldırılması yeteneği eklendi;
- + Tasarımcı ayarlarına "İlerleme penceresini göster" özelliği eklendi;
- + Kısayol tuşu kombinasyonlarını yapılandırma yeteneği eklendi;
- + Tasarımcıda "Tablo" nesnesiyle çalışma kullanılabilirliği iyileştirildi;
- * Bağlantılar için kontroller güncellendi; boşluk içeren bağlantılar artık doğru şekilde işleniyor;
- * Arama alanı temizlendikten sonra ağaçların davranışı değiştirildi: veri ağacı daralırken, rapor ağacı genişliyor;
- Küçük boyutlu RoundedRectangle ölçeklendirilirken fazladan çizgilerin görünümü düzeltildi;
- Barcode 93 Extended'de eğik çizgi kodlaması düzeltildi;
- Sözlükler birleştirilirken bir bağlantının silinmesi hatası giderildi;
- Macar yerelleştirmesinde tarih veya saat biçimlendirme seçimindeki hata düzeltildi;
- DataTree ve ReportTree'de arama sıfırlama hatası giderildi;
- FloatCollection'a yanlış veri girildiğinde oluşan System.NullReferenceException ve System.FormatException hataları giderildi;

[Preview]

- Sayfa yüksekliği veya genişliği sonsuz olduğunda sayfa kenarlığının yanlış boyutu düzeltildi;

[Exports]

- + Her görüntünün ayrı iş parçasığında kaydedilmesi uygulandı;
- + Excel 2007, Word 2007 ve RTF'ye yapılan dışa aktarımlarda eksik olay işleyici bağlantıları eklendi;
- + ZPL'ye dışa aktarım sırasında barkodların ölçeklendirilmesi için yeni özellik eklendi;
- + Excel 2007'de raporun sayfalara bölüneceği grup seçimi eklendi;
- + Excel 2007'ye dışa aktarım sırasında sayfa gruplandırmasının devre dışı bırakılması yeteneği eklendi;
- + SVG'ye dışa aktarım sırasında doku doldurması için sarma modunun kullanılması eklendi;
- * Bulut depolamaya dışa aktarım yapıldığında, yetkilendirme kodu alındıktan sonra pencere otomatik kapanır;
- Özel font koleksiyonları düzeltildi;
- GSUB tablosunun ayrıştırılmasındaki hata giderildi;
- PDF'ye dışa aktarımda DashDot, DashDotDot ve Double nesne kenarlık stillerinin yanlış dışa aktarımı düzeltildi;
- HTML dışa aktarım sırasında Gauge içindeki sayıların bulanık görüntülenmesi hatası giderildi;
- S3 dışa aktarımında ContentMD5 başlığının hesaplanması düzeltildi;
- ZPL'ye dışa aktarım sırasında metnin yanlış konumlandırılması giderildi;
- GaugeObject'in PowerPoint 2007'ye dışa aktarımındaki hata düzeltildi;
- Word 2007'de katmanlı dışa aktarımda dolgu ile RadialGauge'in yanlış dışa aktarımı giderildi;
- Katmansız HTML'de dolgu ile RadialGauge'in dışa aktarımındaki hata düzeltildi;
- SVG'ye dışa aktarım sırasında "Clamp" transfer modunda doku doldurmasının görüntülenmesi hatası giderildi;
- Excel 2007 dışa aktarımında HTML etiketleri kullanılırken metin boyutundaki değişiklik giderildi;
- Excel 2007'ye dışa aktarımda HTML etiketlerinin sekmelerle olan yanlış davranışı düzeltildi;
- PDF'ye dışa aktarım sırasında filigranın kalitesinin düşürülmesi sorunu giderildi;
- Word 2007'de tablolara dışa aktarım sırasında yanlış girinti hatası düzeltildi;
- Word 2007'ye dışa aktarımda CheckBox içindeki görüntü konumlandırması giderildi;

[WebReport]

- + WebReport'den raporla birlikte e-posta gönderme yeteneği eklendi;
- + FastReport.Blazor.WASM'ye yazdırma yeteneği eklendi;
- FastReport.Web'de .NET Standard 2.0 desteği kaldırıldı;
- Blazor uygulamasında dışa aktarım hatası giderildi;
- WebReport'de PrintHtml ile yazdırılırken Margin'in göz ardı edilmesi hatası düzeltildi;

[.Net Core]

- PDF'ye dışa aktarım sırasında metin genişliğinin yanlış hesaplanması hatası giderildi;

[Demos]

- Demo New küçültüldükten sonra navigasyon menüsünün görüntülenmesindeki hata giderildi;

[Extras]

- + MySqlConnection'da CLR türlerine Variant dönüşümü eklendi;
- + FastReport.Data.Odbc eklentisi eklendi;
- + FastReport.Data konektör eklentileri için FastReport.WPF desteği eklendi;
- * Bir istekteki yinelenen isimlerle ilgili mesajın davranışında değişiklik yapıldı;
- Bir istekte parametrelerin otomatik oluşturulması hatası giderildi.

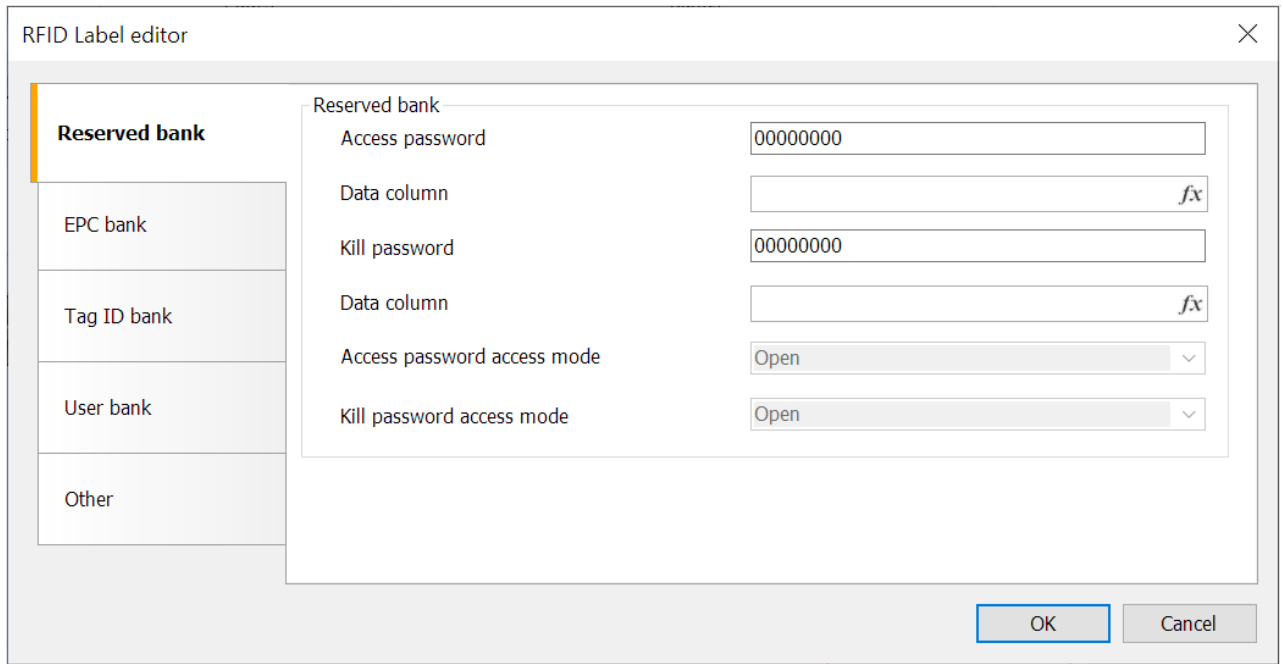
Sürüm 2023.3

Yeni Özellikler

Yeni RFIDLabel Nesnesi

Yeni sürüm, RFID etiketi adında yeni bir nesne içerir. Bu nesne, ürünlerin tanımlanmasını sağlar ve barkoda çok benzer, ancak barkodun aksine radyo sinyalleri kullanır. Bu sayede kısa zaman aralıklarında çok sayıda ürün taranabilir.

Etiket, erişim ve imha şifrelerini depolamak için ayrılmış bir rezerve bankası, elektronik ürün kodu bankası, etiket tanımlayıcı bankası ve kullanıcı veri bankası olmak üzere 4 veri bankası içerir. FastReport .NET ürün gamında, RFID etiketi bir rapor nesnesi olarak temsil edilir. Etiket, çift tıklayarak erişilebilen kullanıcı dostu bir düzenleyici kullanılarak özelleştirilebilir.



The image shows a screenshot of the 'RFID Label editor' dialog box. It has a title bar with a close button (X). On the left, there is a sidebar with five options: 'Reserved bank' (selected), 'EPC bank', 'Tag ID bank', 'User bank', and 'Other'. The main area is titled 'Reserved bank' and contains several fields: 'Access password' (00000000), 'Data column' (empty), 'Kill password' (00000000), 'Data column' (empty), 'Access password access mode' (Open), and 'Kill password access mode' (Open). Each 'Data column' field has a small 'fx' icon to its right. At the bottom right, there are 'OK' and 'Cancel' buttons.

RFID etiketleri bazı Zebra yazıcıları tarafından oluşturulabilir, bu nedenle etiket nesnesinin kendisinin yanı sıra, ZPL'ye aktarım desteği de eklenmiştir. Doğru aktarım için, RFID etiketinin sayfada tek bir kopya halinde bulunması gerekmektedir.

Daha fazla bilgi için ilgili makaleye göz atın.

WebP Formatındaki Görsellerin Desteği

Artık, rapor tasarımcısındaki editör veya kod üzerinden PictureObject'e yüklenebilen WebP formatındaki görselleri destekleyen bir eklenti bulunmaktadır. FastReport.Skia, WebP görsellerini eklenti olmaksızın destekler; ancak yükleme sırasında PNG formatına dönüştürülür.

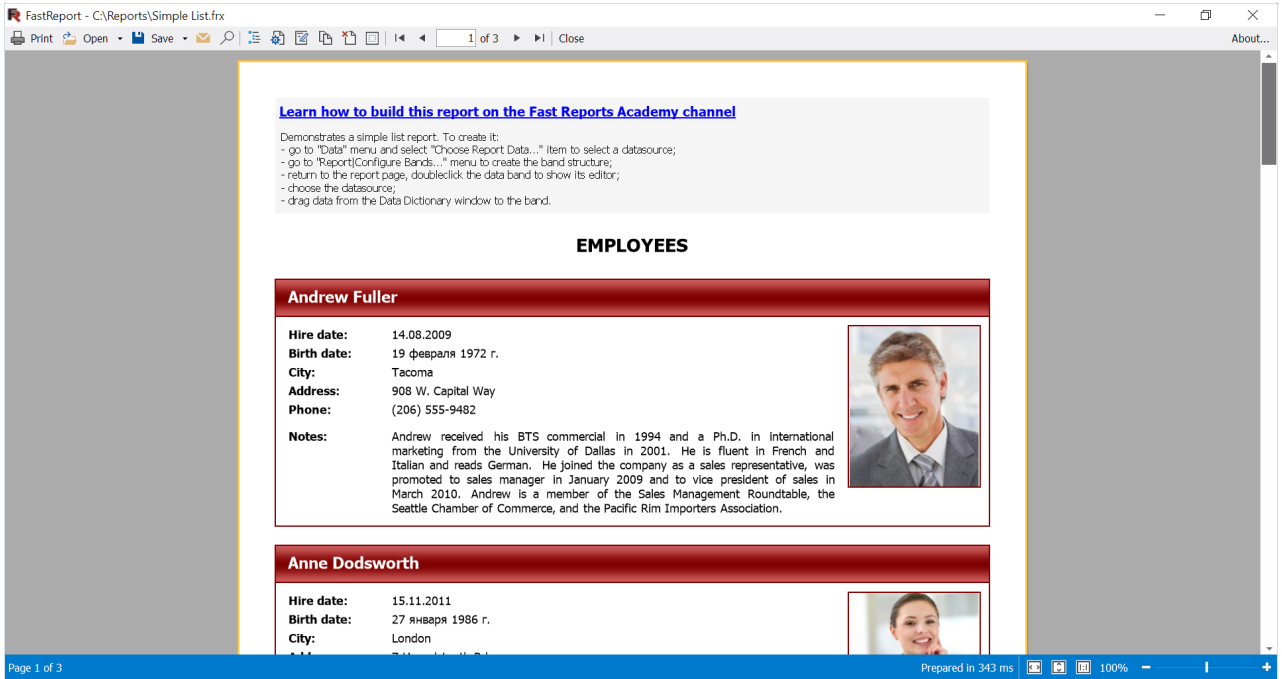
Format hakkında detaylar ve eklentinin nasıl kullanılacağına dair talimatlar bu makalede yer almaktadır.

Tasarımcı Penceresinde Önizleme ve Asenkron Rapor Görüntüleme

Artık, uygulamanızdaki tasarımcı kullanılırken rapor önizlemesini tasarımcı penceresinde başlatabilirsiniz. Önceki sürümlerde önizleme her zaman ayrı bir pencerede açılıyordu. Bunu yapmak için kodunuza aşağıdaki satırı ekleyin:

```
Config.DesignerSettings.EmbeddedPreview = true;
```

Bu, bazı durumlar



da daha kullanışlı olabilir:

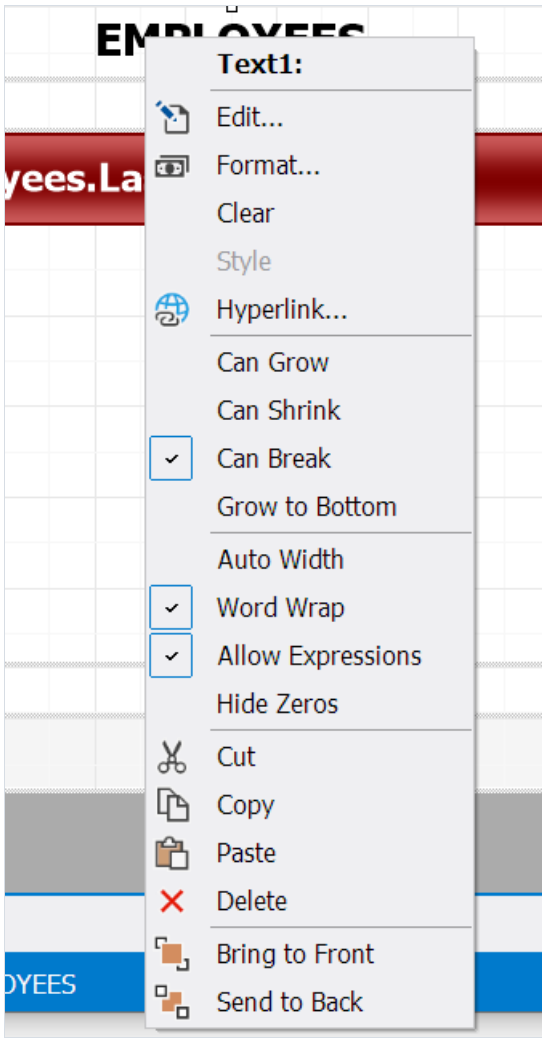
Ayrıca, büyük raporlar işlenirken kullanılabilen asenkron rapor hazırlama ve görüntüleme yöntemleri tanıtılmıştır:

`Report.PrepareAsync()` ve `Report.ShowAsync()`. Bu sayede, rapor hazırlanırken önizleme penceresi kullanılabilir ve kullanıcı uygulamanın donmuş veya yanıt vermiyor izlenimini yaşamaz.

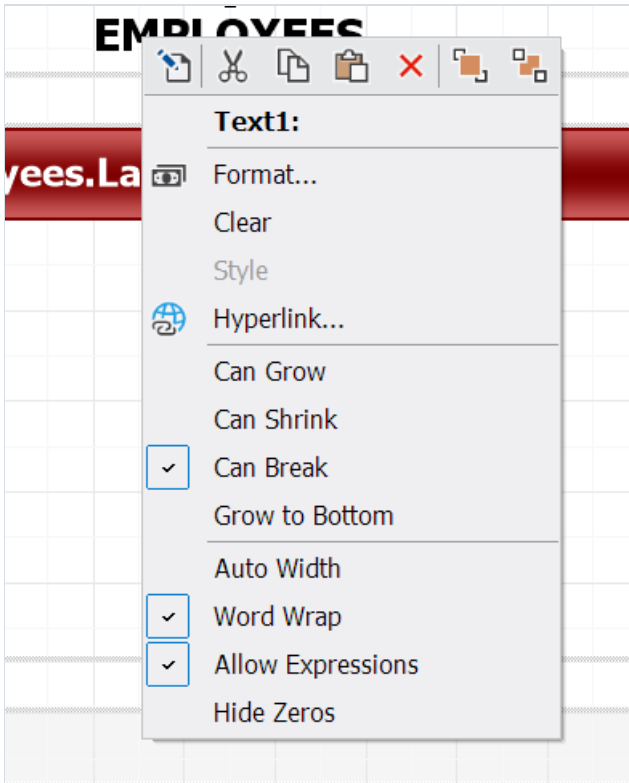
Sağ Tıklama Menüsünde Araç Çubuğu

Bir nesneye sağ tıklanıldığında, bağlam menüsü geliştirilmiştir. Üst kısımda, düzenle, kes, kopyala, yapıştır gibi sık kullanılan öğeleri içeren bir araç çubuğu görünür hale gelmiştir.

Önceki menü şu şekilde görünüyordu:



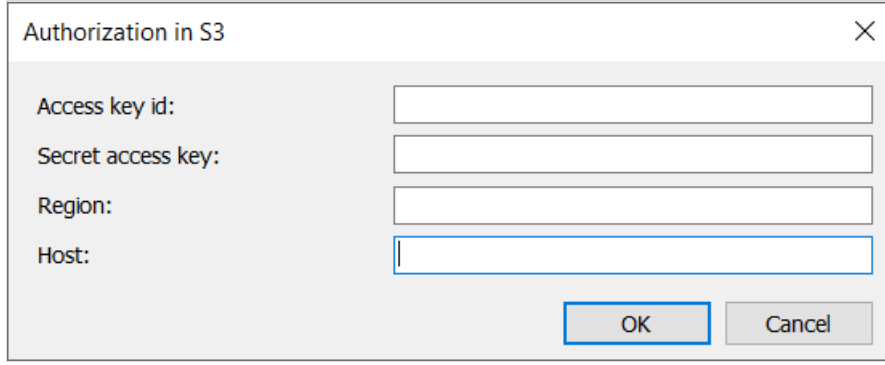
Yeni menü ise daha kompakt ve ergonomik hale gelmiştir:



S3'e Aktarım

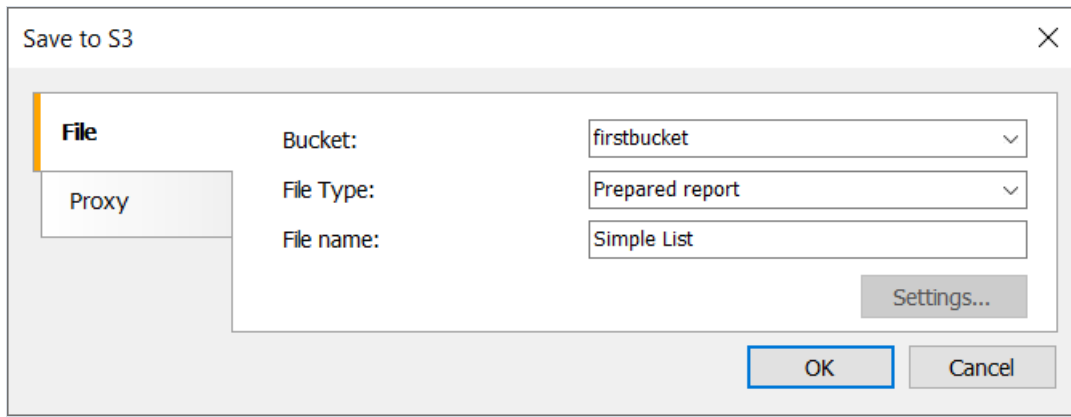
Hazırlanan ve dışa aktarılan raporların Simple Storage Service'e (kısaca S3) yüklenebilme yeteneği eklenmiştir. Yeni aktarım, hazırlanan raporun kaydedilme menüsündeki "Storage" sekmesi altında yer almaktadır.

İlk aktarım sırasında, yetkilendirme penceresinde kayıt verilerini girmeniz gerekecektir.

A dialog box titled "Authorization in S3" with a close button (X) in the top right corner. It contains four input fields: "Access key id:", "Secret access key:", "Region:", and "Host:". At the bottom, there are "OK" and "Cancel" buttons.

Gerekli anahtarları S3 hesap ayarlarınızdan edinebilirsiniz. Hizmet dokümantasyonunda daha fazla ayrıntıyı bulabilirsiniz.

Yetkilendirme başarılı olduktan sonra, bir aktarım penceresi göreceksiniz.

A dialog box titled "Save to S3" with a close button (X) in the top right corner. It has a sidebar with "File" and "Proxy" options. The "File" section is active, showing "Bucket:" with a dropdown menu set to "firstbucket", "File Type:" with a dropdown menu set to "Prepared report", and "File name:" with a text field containing "Simple List". There is a "Settings..." button and "OK" and "Cancel" buttons at the bottom.

Burada, kaydedilecek bucket'ı seçebilir, dosya türü ve dosya adını belirleyebilirsiniz. "Ready Report" dışındaki bir dosya türü seçerseniz, ilgili aktarım için ayarlar aktif hale gelecektir.

Detaylar için ilgili makaleye bakın.

Barkod Yazı Tipi Ayarlarının Özelleştirilebilme Yeteneği

Artık "Barcode" nesneleri için "Font" özelliği kullanılabilir. Bu özellik, barkod metinlerinin görüntülenmesinde kullanılan yazı tipi parametrelerini ayarlamanıza olanak tanır. Varsayılan yazı tipi, önceki sürümlerde kullanılan Arial'dır. Artık farklı bir yazı tipi seçebilir, boyutunu, stilini vb. ayarlayabilirsiniz. Örneğin, aşağıdaki gibi barkodlar oluşturabilirsiniz:



Ancak, yazı tipi ayarlarına dikkat etmelisiniz. Tüm tarayıcılar bu tür barkodları okuyamayabilir.

Excel 2007'e Aktarımda "Genel Formatı Metin Formatına Dönüştür" Seçeneği

Excel 2007, genel ve metin olmak üzere birbirine çok benzer iki veri formatına sahiptir.

Genel format varsayılandır. Çoğu durumda, bu formatta sayılar girildiği gibi görünür. Ancak hücre genişliği yetersizse, sayı yuvarlanır.

Metin formatı ise her zaman veriyi girildiği gibi görüntüler.

FastReport .NET'in ayrıca genel, sayısal, tarih ve daha birçok farklı biçimi vardır. Dışa aktarma sırasında uygun biçim seçilir, sayısal veriler sayısal verilere dönüştürülür ve tarih, tarih olarak kalır.

FastReport .NET'te genel format System.String olarak kullanılır. Öte yandan, FastReport .NET'te ayrı bir metin formatı bulunmamaktadır.

Excel 2007 aktarımında, FastReport .NET genel formatının Excel metin formatına dönüştürülmesine olanak tanıyan yeni bir seçenek eklenmiştir (varsayılan olarak genel format dışa aktarılır).

Export to Excel 2007

Basic

Other

Pinned cells

☒ Without pinning

☐ First line

☐ First column

☐ Other Pinning

Columns: 0 Rows: 0

Print Scaling

☒ No Scaling

☐ Set scale in percent: 10

Pictures properties

☒ Move and modify along with the cell

☐ Move but not resize

☐ Do not move or resize

Options

☐ Use locale formatting of data

☐ Convert currency to accounting format

☒ Convert general to text format

☐ Export all tabs

☐ Open after export

OK Cancel

Önceki FastReport .NET sürümleriyle oluşturulan raporlar, bu seçenek varsayılan olarak devre dışı bırakıldığı için aynı şekilde aktarılacaktır.

Kısmi Rapor Derlemesi Desteği

FastReport.Core, FastReport.Core.Skia ve FastReport.CoreWin artık, rapor betiğinde değişiklik yoksa ve kısmi derlemeyi desteklemeyen nesneler bulunmuyorsa, raporun hazırlanma hızını artırmak amacıyla kısmi derlemeye izin verir. Aşağıdaki komut ile etkinleştirilebilir:

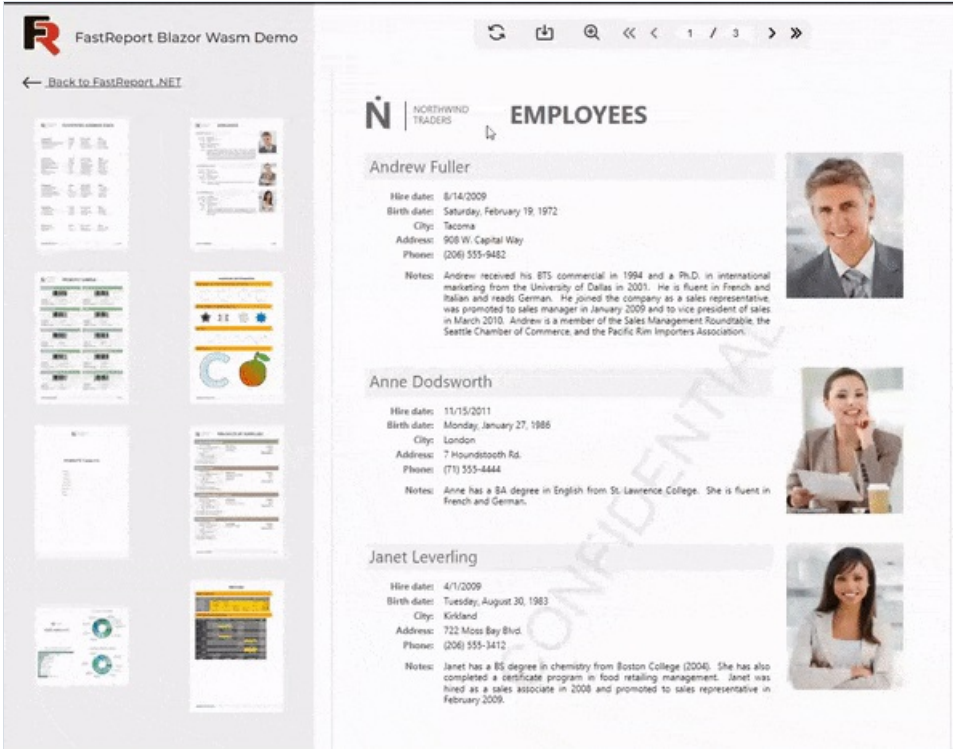
```
FastReport.Utils.Config.CompilerSettings.ReflectionEmitCompiler = true;
```

Reflection.Emit Compiler etkinleştirildiğinde herhangi bir performans düşüşü olmaz. Yeni raporda bu derleyici kullanılamazsa, rapor zarar görmeden standart prosedür kullanılır.

Yeni WebReport Özellikleri

WASM'de iyileştirmeler

Önceden, sadece FastReport.Blazor.Wasm kütüphanesi ile açılan raporlar tarayıcıda görüntülenebiliyordu. Bu güncelleme ile dışa aktarımlar desteklenmeye başlandı. Artık kullanıcılar, oluşturulan raporları çeşitli formatlara, normal WebReport'te olduğu gibi, dışa aktarabiliyor.



Ayrıca, betik içermeyen raporların yüklenmesi ve hazırlanması Reflection.Emit derlemesi sayesinde hızlandırıldı.

Araç Çubuğu Kişiselleştirme

Artık araç çubuğuna, buton, açılır liste ve giriş alanı gibi öğeler ekleyebilirsiniz. Bu öğeler, resimler, başlıklar ve stiller gibi çeşitli seçeneklere sahip olabilir. JavaScript ve C# kullanarak mantık da ekleyebilirsiniz.

Araç çubuğuna özel öğe eklemeye dair örnek:

```

var button = new ToolbarButton()
{
    Title = "MyCustomButton",
    OnClickAction = new ElementClickAction()
    {
        OnClickAction = async (webreport) =>
        {
            webreport.LocalizationFile = "MyLocalizationFile";
        }
    },
};

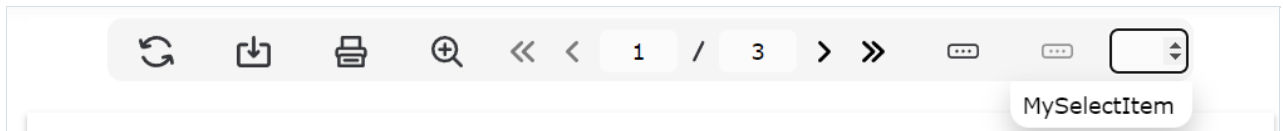
var select = new ToolbarSelect()
{
    Title = "MyCustomSelect",
    Items = new List<ToolbarSelectItem>
    {
        new ToolbarSelectItem()
        {
            Title = "MySelectItem",
            OnClickAction = new ElementScript()
            {
                Script = "console.log('My item is working')"
            }
        }
    }
};

var input = new ToolbarInput()
{
    InputType = "number",
    OnChangeAction = new ElementChangeAction()
    {
        OnChangeAction = async (webreport, inputValue) =>
        {
            webreport.Report.Prepare();
            webreport.Toolbar.Height = int.Parse(inputValue);
        }
    }
};

webReport.Toolbar.InsertToolbarElement(button);
webReport.Toolbar.InsertToolbarElement(select);
webReport.Toolbar.InsertToolbarElement(input);

```

Bu özel öğeler, yapılan değişikliklerin ardından araç çubuğunuzda görünecektir.



Değişikliklerin Tam Listesi

```

[Engine]
+ Yeni RFIDLabel nesnesi eklendi;
+ GS1-128 barkodu için GS1 otomatik biçimlendirmesi eklendi;
+ RDL şablonları dönüştürülürken diğer tabloların hücrelerine tablolar yüklenmesi eklendi;
+ Rapor betiğinde değişiklik yoksa rapor hazırlığını hızlandırılan
Config.CompilerSettings.ReflectionEmitCompiler özelliği eklendi (sadece .NET Core/.NET üzerinde çalışır);
+ Yeni "Font" özelliği kullanılarak barkod yazı tipinin yapılandırılabilmesi eklendi;
* Özel yazı tipi koleksiyonlarıyla çalışma geliştirildi;
* Demo sürümünde 5 sayfa limiti kaldırıldı; metin rastgele "Demo version" ile değiştiriliyor;

```

- Parametre ifadesinin bu parametreye eşit olması durumunda oluşan sonsuz döngü düzeltilti;
- Mobil tarayıcı tarafından DataMatrix barkodunun okunması sorunu düzeltilti;
- Manuel transferler sırasında çizgili üstü çizilmiş metinlerin yanlış görüntülenmesi hatası düzeltilti;
- RichObject nesnelerinin çevrilmesi sırasında kayma hesaplaması düzeltilti;
- Boş Variant tipinin diğer tiplere dönüştürülmesi sorunu düzeltilti;
- Sütun silindikten sonra sütun verilerinin raporda kalması hatası düzeltilti;
- Matris ve tablo satır ve sütunları için VisibleExpression özelliğinin çalışması düzeltilti;
- Font_hash sözlüğünden artık bulunmayan yazı tiplerinin silinmesi hatası düzeltilti;
- RichObject içindeki sıralanmamış tab durakları hatası düzeltilti;
- GSUB tablosunun ayrıştırılmasında istisnaya yol açan hata düzeltilti;
- "Text in curves" seçeneğiyle PDF'ye aktarım sırasında System.StackOverflowException hatasına yol açan akış sonu kaybı düzeltilti;
- RDL şablonları dönüştürülürken nesne kenarlarının yüklenmesi hatası düzeltilti;
- GS1-128 barkodundaki ilk üç karakterin silinmesi hatası düzeltilti;
- Code93 Extended barkodu için kodlama tablosu düzeltilti;
- DataMatrix barkodu için metin kodlaması düzeltilti;
- Boşluk yetersizliği nedeniyle kelime bölünmesi sırasında metin renderlama hatası düzeltilti;
- ConvertRichText seçeneği etkinleştirildiğinde RightToLeft metin dönüşümü düzeltilti;
- HtmlTextRenderer'da satır sonu hatası düzeltilti;
- Sayfa sütunlarının bant sütunlarının üzerine yazdırılması hatası düzeltilti;
- RichObject içerisinde dolgu kullanıldığında metin paragrafları arasındaki boş satırların beyaz vurgulanması düzeltilti;
- ConvertRichText = true kullanıldığında RichObject içindeki beyaz renkli metin parçalarının seçimi düzeltilti;
- ConnectionStringExpression null döndüğünde ConnectionString'in göz ardı edilmesi hatası düzeltilti;
- RichObject'den çevrilen metin nesnelerinin girintileri düzeltilti;
- RichObject çevrilirken nesne konumlandırması düzeltilti;
- JasperReports'tan tabloların içe aktarılması düzeltilti;
- TableObject temizlenirken oluşan System.NullReferenceException hatası düzeltilti;
- ConvertRichText = true kullanıldığında RichObject'de yatay görüntü hizalaması düzeltilti;
- TextObject tab durak değeri negatif olduğunda oluşan System.NotImplementedException hatası düzeltilti;
- İfade içerisinde fonksiyon varsa null dönüş hatası düzeltilti;
- JSON veri kaynağı sunucusunun boş CharacterSet içermesi durumunda System.ArgumentException hatası düzeltilti;
- RichObject çevrilirken nesne konumlandırması düzeltilti;

[Designer]

- + Excel bağlantısında sütun adlarını ilk satırdan alma yeteneği eklendi;
- + "Barcode" nesneleri için kategoriler eklendi;
- + Rapor önizlemesini tasarımcı penceresinde açmak için Config.DesignerSettings.EmbeddedPreview özelliği eklendi;
- + "Nesneler" panelinde "Diğer" kategorisi eklendi;
- + Çevrilen nesnenin Online Tasarımcı'da görüntülenebilmesi eklendi;
- + Veri bağlantısı sihirbazı şeklinde prosedür seçimi sayfası eklendi;
- + Bağlam menüsüne araç çubuğu eklendi;
- + SberbankQr düzenleyicisindeki "Ödeme Tutarı" alanında ifadelerin kullanılabilmesi eklendi;
- + SQL sorgusundan parametrelerin ayrıştırılması eklendi;
- + CsvFile özelliği aracılığıyla CSV bağlantısında dosya değiştirildiğinde dosyanın varlığı kontrol edildi;
- + "Query Builder" arayüzünde değişiklikler yapıldı;
- * "Data Connection Wizard" güncellendi. Arayüz geliştirildi, hatalar düzeltilti ve hız artırıldı;
- * Tasarımcıdaki koordinatlar/ölçülerle ilgili ipuçlarının render edilme biçimi değiştirildi;
- * URL üzerinden CSV'ye bağlanma sorunu düzeltilti;
- Bulut üzerinden açılan dosya için "Farklı Kaydet..." işleminde hata düzeltilti;
- NET 6.0'da "Map" nesnesinde (boş çokgen etiketleri) hata düzeltilti;
- Tasarımcı yapılandırma dosyasından değer okunurken oluşan hata düzeltilti;
- "Code" sekmesinde çift sağ tıklama sonrası yeni rapor sayfası oluşturulurken oluşan hata düzeltilti;
- Sayısal parametrelerin boş değerleri ile önizleme penceresi kapatıldığında oluşan hata düzeltilti;
- Yetkilendirme sürecinde tasarımcının yanıt vermemesi hatası düzeltilti;
- Gauge nesnesi düzenleyicilerindeki hatalar düzeltilti;
- Parametre bağlantılarını içeren sözlüklerin birleştirilmesi sırasında oluşan System.NullReferenceException hatası düzeltilti;
- ConvertRichText = true kullanıldığında RichObject'de metin vurgulama hatası düzeltilti;
- Bir metin nesnesinde birden fazla ifade olduğunda format sırası hatası düzeltilti;
- "Plugins" sekmesindeki tasarımcı ayarları penceresindeki ölçeklendirme hatası düzeltilti;
- Visual Studio'da veri kaynağı seçme formunun ölçeklendirmesi hatası düzeltilti;
- Sonsuz genişlikteki sayfaların önizleme eklenmesi sırasında eksik görüntülenmesi hatası düzeltilti;
- Şifre korumalı rapor yüklenirken oluşan hata düzeltilti;
- Bazı kontrollerin ölçeklendirilmesiyle ilgili sorunlar düzeltilti.

- Bağlantı düzenlemesi sırasında seçilmeyen tablolar için alanlar seçildiğinde oluşan hata düzeltildi;
- Aslında sadece bazı tablolar seçilmiş olmasına rağmen, tüm tabloların seçildiği hatası düzeltildi;
- FRX sayfasında yanlış XML rapor kullanımı durumunda oluşan System.IO.FileFormatException hatası düzeltildi;
- Ölçek %100'den büyük olduğunda MSChartObject'te yazı tipi ayarlarının yanlış çalışması düzeltildi;
- URI aracılığıyla CSV veri tabanına bağlanırken oluşan hata düzeltildi;
- DataBand üzerinde CanBreak özelliği etkin olduğunda MSChartObject ve SparklineObject içeren rapor çalıştırılırken oluşan hata düzeltildi;
- Tasarımcıda SVG görüntülenmesi ile ilgili sorunlar düzeltildi;
- "Report Tree" penceresinde yazı tipi boyutu hatası düzeltildi;
- Ölçek değiştirildiğinde "About" penceresinin davranışı düzeltildi;
- Title eksik olduğunda MSChartObject render edilme hatası düzeltildi.

[Preview]

- AutoWidth = true kullanıldığında metin nesnesinin yatay hizalaması düzeltildi;
- Önizlemede SVG görüntüleme ile ilgili sorunlar düzeltildi.

[Exports]

- + S3'e aktarım eklendi;
- + Görsel dışa aktarımında sayfa kenarlarının aktarılması eklendi;
- + HTML aktarımında "Sayfa Sonları Kullan" seçeneği eklendi;
- + Word 2007 aktarımında her sayfaya yer imlerinin eklenmesini etkinleştirme/devre dışı bırakma seçeneği eklendi;
- + Excel 2007 sayfasında izin verilen maksimum satır sayısına yaklaşılsa yeni bir sayfa oluşturulması eklendi;
- + Excel 2007 aktarımında "Genel formatı metin formatına dönüştür" seçeneği eklendi;
- + Yazı tipi isimlerinin genişletilmesi eklendi;
- + PDF aktarımı için yazı tipi paketleme alt sistemi geliştirildi;
- * PDF'ye aktarım hızı artırıldı;
- * PDF'ye etkileşimli formların aktarımı optimize edildi;
- Skia kullanılarak yapılan PDF aktarımında LineHeight göz ardı edilme hatası düzeltildi;
- Çoklu iş parçacıklı PDF aktarımı ve özel yazı tipi koleksiyonları hatası düzeltildi;
- Geleneksel Çince karakterler için yazı tipi yüklenmesi hatası düzeltildi;
- PDF aktarımında sağdan sola yazı tipi kerning hatası düzeltildi;
- ConvertRichText özelliği etkinleştirildiğinde 10'dan küçük yazı tiplerinin yanlış görüntülenmesi hatası düzeltildi (RTF aktarımında);
- PDF aktarımında kerning hataları düzeltildi;
- Yüksek monitör çözünürlüğünde "Text in curves" modunda PDF aktarım hatası düzeltildi;
- Karanlık çerçeve çizimiyle ilgili bazı nesneler PDF aktarımında hatalı görüntülenmesi düzeltildi;
- FastReport.Utils.FRPPrivateFontCollection'da kayıtlı yazı tiplerinin aktarım hatası düzeltildi;
- HTML , <sub> ve <sup> etiketlerinin RTF aktarımında görüntülenmesi düzeltildi;
- Skia ile yapılan rapor dışa aktarımında resim bulunan raporun hata vermesi düzeltildi;
- RTF ve DOCX'ye alt nesnelerin (footer) aktarım hatası düzeltildi;
- Tek baytlı boşluklar, Excel 2007'ye aktarım sonrası kaybolmaması sağlandı;
- CSV aktarımında ekstra metin kesmeleri eklendi;
- CSV aktarımında ekstra ayraçlar hatası düzeltildi;
- Çoklu iş parçacıklı PDF aktarımında yazı tiplerinin zarar görmesi hatası düzeltildi;
- HTML aktarımında tire karakterlerinin işlenmemesi hatası düzeltildi;
- RichObject'deki bağlantıların PDF'ye aktarımında yanlış çalışması hatası düzeltildi;
- RTF aktarımında satır yüksekliği çarpanı hatası düzeltildi;
- Google Drive'da raporun çift kaydedilmesi hatası düzeltildi;
- OneDrive'da rapor kaydetme API çağrısı düzeltildi;
- PDF'ye aktarımda SVG görüntüleme sorunları düzeltildi;
- Aktarım ağacındaki hatalar düzeltildi;
- HTML etiketleri içeren metinlerin Word 2007'ye aktarımında hatası düzeltildi.

[WebReport]

- + WebReport'e rapor gölgesi eklendi;
- + Wasm'ye rapor aktarım desteği eklendi;
- * Tek sayfalı raporlar için araç çubuğu davranışında değişiklik yapıldı;
- * WebReport'te tarayıcıdan rapor yazdırma davranışı değiştirildi. Artık bir yazdırma sayfası otomatik olarak kapanıyor;
- WebReport'te tıklama olaylarının çalışmaması hatası düzeltildi;
- Web raporlarında Word 2007'ye aktarımda hatalı dışa aktarım düzeltildi;
- Web tasarımcısında bazı rapor nesnelerinin (örneğin, RichObject) görüntülenmemesi hatası düzeltildi;
- Ayarlar kullanıldığında tek sayfalı raporun dışa aktarılmaması hatası düzeltildi;
- Parametre değiştirildiğinde raporun güncellenmemesi hatası düzeltildi.

[NET Core]

[.NET Core]

- InvariantGlobalization seçeneği etkinleştirildiğinde oluşan hata düzeltildi.

[Demos]

- * "Sort Group By Total" şablonundaki script, "Group Footer" bandının "Can grow" ve "Can shrink" özellikleri kullanıldığında raporun doğru çalışması ve toplamaların görüntülenmesi için güncellendi.

[Extras]

- + PDFSimpleExport kullanılarak yapılan dışa aktarımda sayfa kenarlarının aktarılması eklendi;
- + MySqlConnection eklentisi kullanılarak MariaDB'ye bağlanabilme yeteneği eklendi;
- + SQLite'ye bağlanırken dosya filtresine .db formatı eklendi;
- + WebP formatındaki görselleri destekleyen eklenti eklendi;
- * RPTImportPlugin, .NET Framework 4.7.2'ye güncellendi;
- ClickHouse ve MongoDB'ye bağlanırken oluşan System.IO.FileLoadException hatası düzeltildi;
- CSV bağlantısında dosya seçimi formunun öne çıkmaması hatası düzeltildi.

Sürüm 2023.2

Yeni Fırsatlar

Blazor WebAssembly Desteği

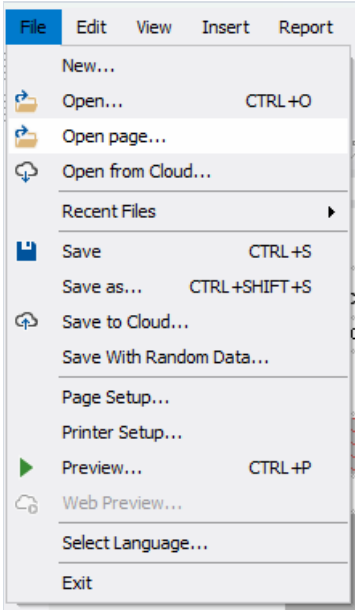
FastReport .NET Enterprise ve üzeri sürümlere sahip kullanıcılar için Blazor WebAssembly desteği sunan **FastReport.Blazor.Wasm** paketi eklendi. Artık WebAssembly uygulamanızda bir raporu görüntülemek için Razor bileşenlerini kullanabilirsiniz. **Dikkat!** Blazor WebAssembly desteği şu anda beta aşamasındadır.

```
<WebReportContainer WebReport="WebReport" />
```

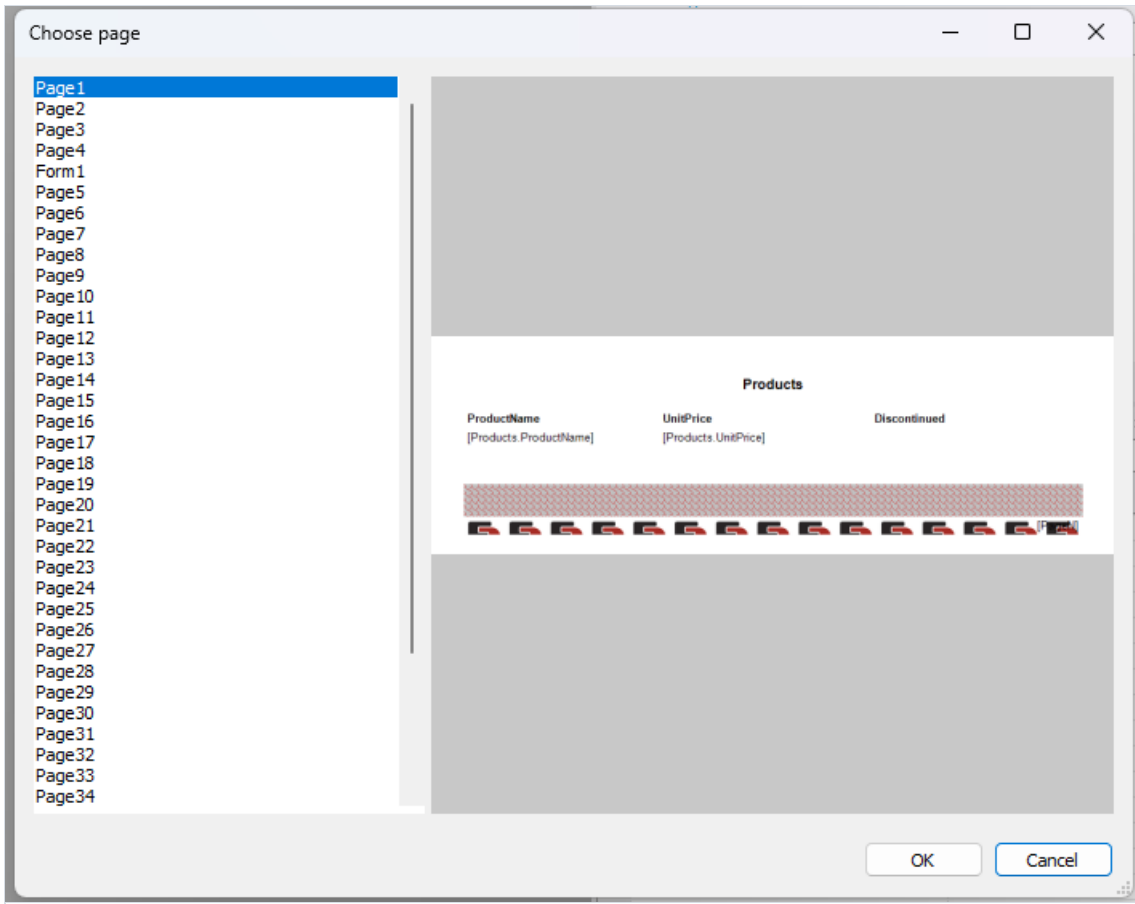
Bu konuda daha fazla bilgi için bu makaleyi okuyun.

Başka Bir Rapor Sayfasını Açabilme

Tasarımcı, geliştirilen rapora başka bir raporun sayfalarını ve diyalog formlarını açıp eklemenize olanak tanır. Bunun için "Dosya" menüsüne gidip "**Sayfa Aç...**" seçeneğini seçmeniz yeterlidir.



Sonrasında, rapor seçebileceğiniz standart dosya seçim iletişim kutusu açılır. Ardından, sayfa listesinin ve seçilen sayfanın önizlemesinin yer aldığı bir pencere görüntülenecektir.

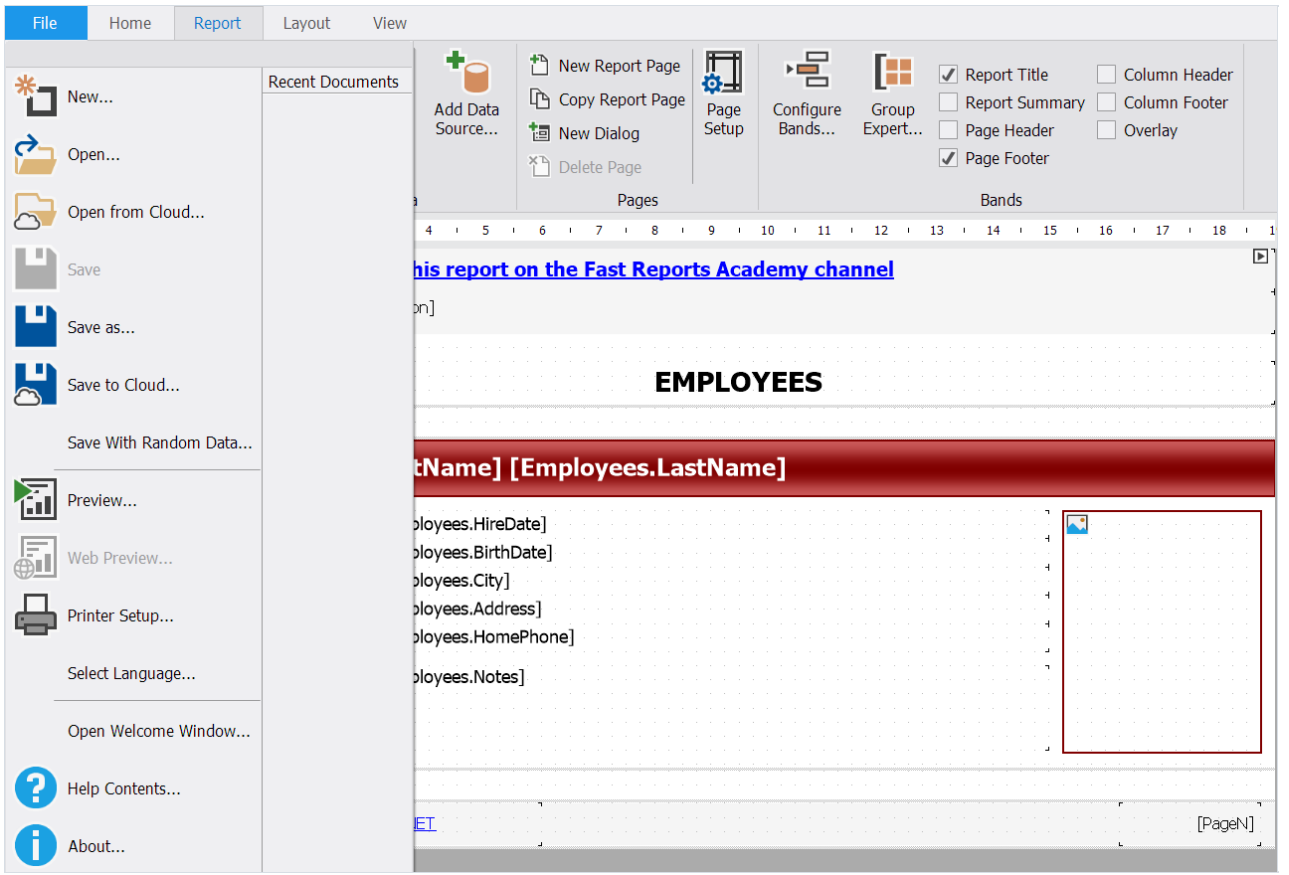


Buradan, mevcut rapora eklemek üzere bir veya birden fazla sayfa seçebilirsiniz. Eğer raporda aynı isimde sayfalar veya nesneler varsa, bu nesnelerin isimleri benzersiz olacak şekilde değiştirilecektir. Bu durum, aynı isimlerin kullanılmayacağı için hataların önüne geçmek amacıyla yapılmaktadır.

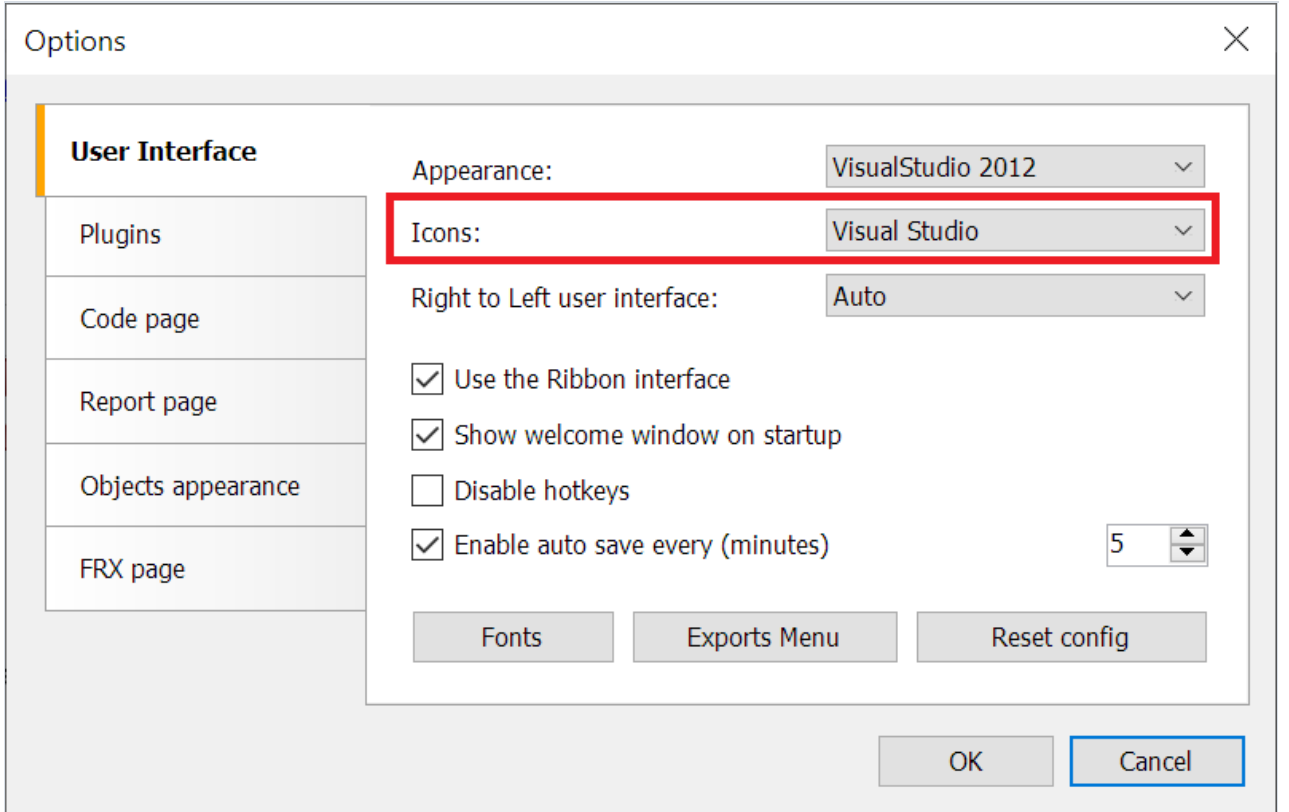
Bu konuda daha fazla bilgi için bu makaleyi okuyun.

Şerit Arayüzü İçin Yeni Simgeler

Tasarımcıdaki Şerit (Ribbon) arayüzüne Visual Studio tarzı yeni simgeler eklendi.



Bunları kullanıcı arayüzü seçeneklerinden seçebilirsiniz.



Değişikliklerin etkili olması için tasarımcıyı yeniden başlatmanız gerekmektedir.

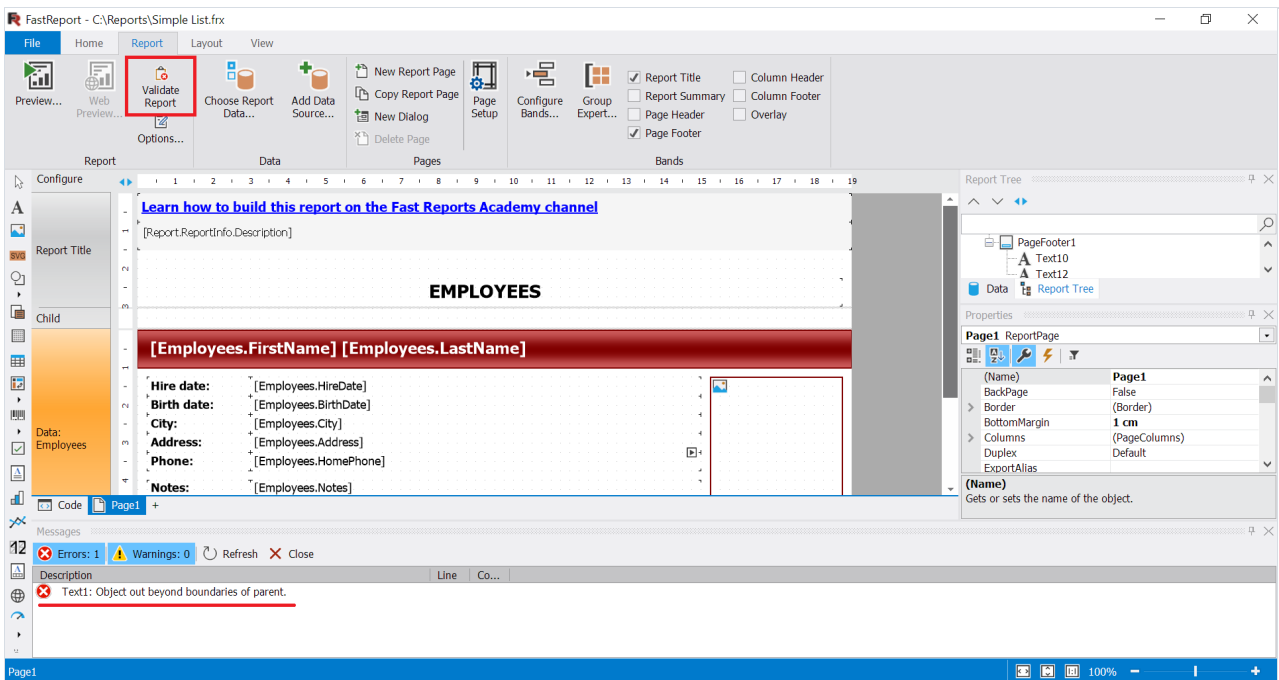
Özellikler Penceresinde Filtre

Özellikler penceresine, nesneye özgü özelliklerin (örneğin; metin nesnesi için Metin ve Yazı Tipi özellikleri) gösterilmesini etkinleştiren yeni bir düğme eklendi. (Bununla birlikte, Top, Sol, Yükseklik ve Genişlik gibi genel nesne özellikleri görüntülenmeyecektir.)

Properties		
Text10 TextObject		
		
RightToLeft		False
WordWrap		True
Trimming		None
Clip		True
Wysiwyg		False
TextRenderType		Default
HideZeros		False
HideValue		
NullValue		
ProcessAt		Default
Duplicates		Show
Editable		False
CanBreak		True

Rapor Doğrulayıcı Değişiklikleri

Rapor doğrulayıcı artık arka planda çalışmıyor; bunun yerine "Rapor" menüsünde bulunan ayrı "**Raporu Doğrula**" düğmesi aracılığıyla çalıştırılıyor. Ayrıca, doğrulayıcı penceresi kaldırıldı ve doğrulama mesajları "Mesajlar" penceresinde görüntülenmektedir.



Bağlantı Dizesini Gizleyebilme

Tasarımcıda, veri kaynaklarının bağlantı dizelerinin görüntülenip görüntülenmeyeceğini belirleyen yeni bir özellik **Config.ConnectionStringVisible** eklendi. Bu özellik, uygulama geliştiricisi ile rapor kullanıcısı arasındaki izin farklarını belirlemede kullanılabilir. False olarak ayarlandığında, kullanıcı tasarımcıda bağlantı dizelerini göremeyecek ve düzenleyemeyecektir.

WebReport Değişiklikleri

MemoryCache Desteği: **Varsayılan olarak mevcut WebReportCache kullanılırken, FastReport servisleri kayıt edilirken aşağıdaki gibi MemoryCache etkinleştirilebilir:

```
services.AddFastReport(options =>
{
    options.CacheOptions.UseLegacyWebReportCache = false;
});
```

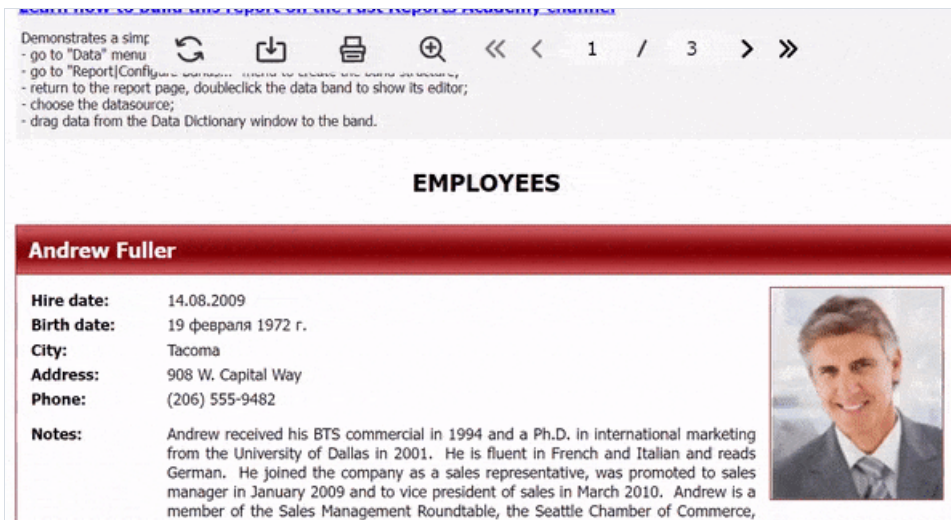
WebReport'in yerleşik önbelleğinden farklı olarak, MemoryCache, belirli bir süre boyunca (CacheOptions.CacheDuration) etkin olmayan WebReport örneklerini daha agresif bir şekilde boşaltır. Bu, eski önbelleğin (herhangi bir nedenle) belleği temizlememesi durumunda faydalı olabilir.

Araç Çubuğunun Sabitlelmesi:

Araç çubuğunu sayfada kaydırma yapılsa bile her zaman görünür olacak şekilde sabitleme yeteneği eklendi. Bu özellik, özellikle büyük raporlarla çalışırken çok kullanışlıdır. Araç çubuğunu ekranda sabitlemek için aşağıdaki özelliği ayarlamanız gerekir:

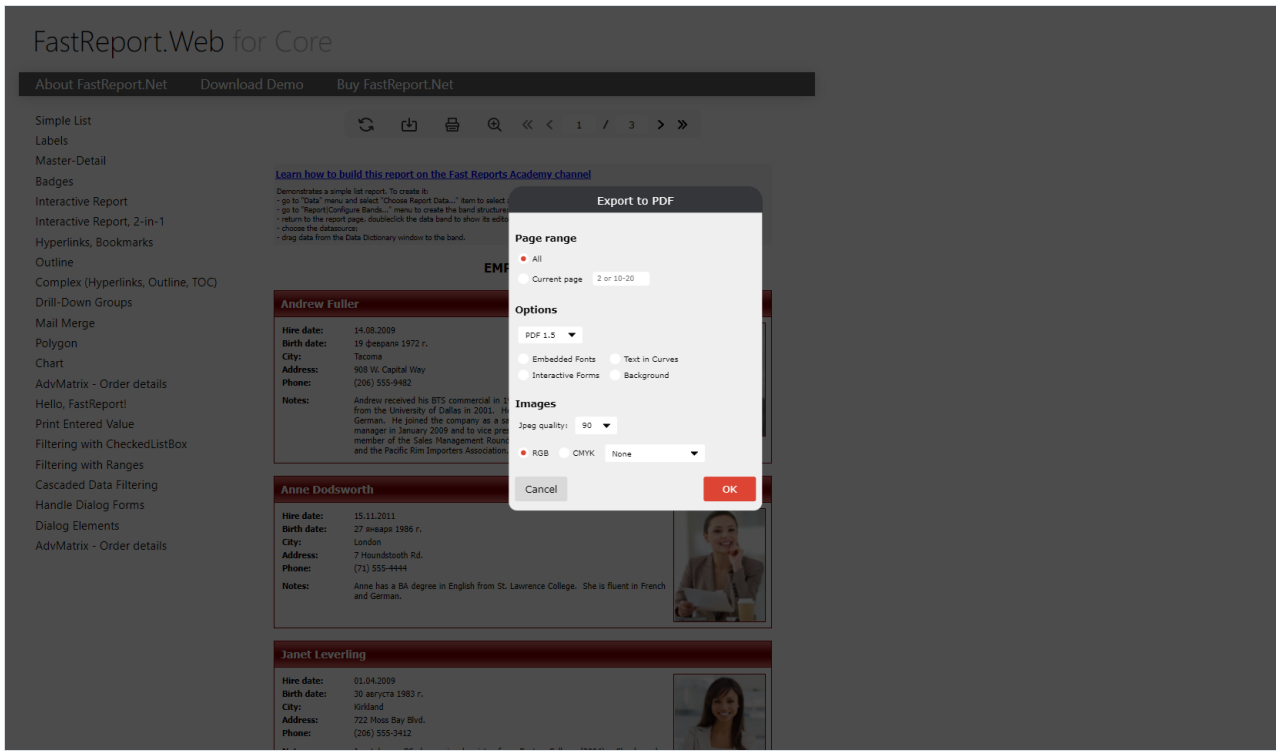
```
webReport.Toolbar.Sticky = true;
```

Artık araç çubuğu her zaman görünür olacaktır.

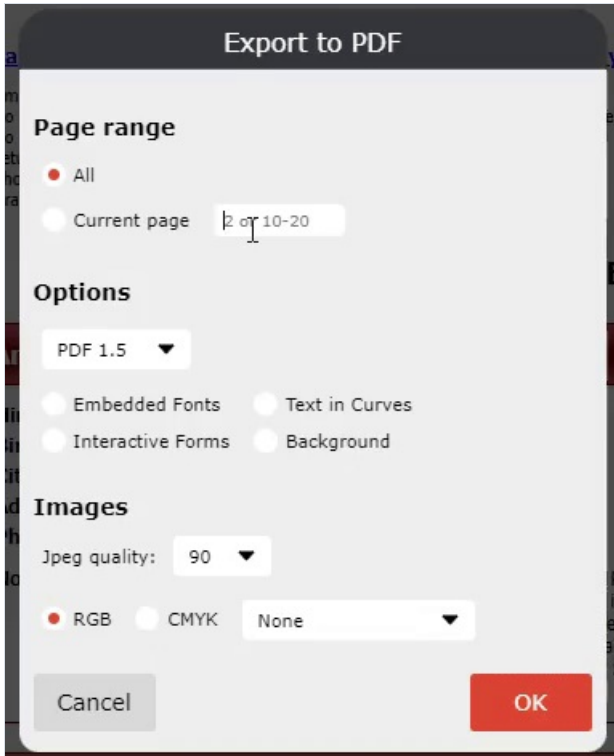


Dışa Aktarma Ayarları Penceresinin Özelleştirilmesi: Dışa aktarma ayarları penceresinin ekranda sabitlenip ön planda görüntülenebilmesi için aşağıdaki özelliği ayarlayabilirsiniz:

```
webReport.Toolbar.Exports.PinnedSettingsPosition = true;
```



Sayfa Aralığı Doğrulaması: Dışa aktarma ayarları penceresine, girilen sayfa aralığı için doğrulama eklenmiştir. Yanlış giriş yapıldığında, alan aşağıdaki gibi görünecektir.



FastReport.Core.Skia İyileştirmeleri

FastReport.Core.Skia paketinin performansı artırıldı ve dışa aktarma sırasında oluşan bazı hatalar giderildi.
Örnekler:

CanShrink = true olan nesnelerin renderlanması düzeltildi.

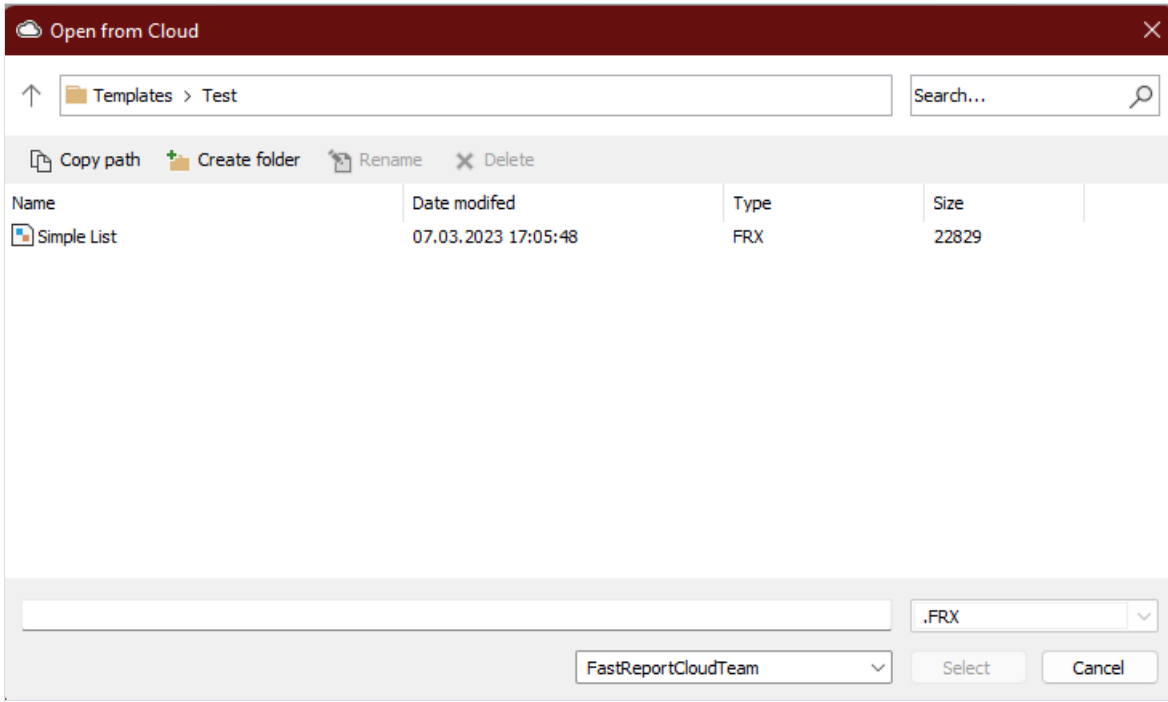
[illegible][illegible]

Before

TEST LOGO

TEST LOGO

Dosya yöneticisi penceresinin tasarımı, servisin daha kullanışlı ve sezgisel hale gelmesi amacıyla güncellendi. Arayüz öğelerinin düzeni ve renk şeması üzerinde yapılan değişiklikler, kullanıcıların genel görsel deneyimini iyileştirecektir.



Tam Değişiklik Listesi

[Engine]

- + Config.ConnectionStringVisible özelliği eklendi; veri kaynaklarının bağlantı dizelerinin tasarımcıda görüntülenip görüntülenmeyeceğini belirler.
- Prosedür içermeyen bağlantıdaki prosedürlerin çıkarılmasıyla ilgili hata düzeltildi.
- Sayfanın ilk sütununun her zaman en sol konumda görüntülenmesi hatası giderildi.
- GaugeObject.Value özelliği, yeni değer GaugeObject.Maximum'tan büyük olduğunda GaugeObject.Minimum'a eşitleniyordu; artık GaugeObject.Value, GaugeObject.Maximum'a eşitlenecektir.

[Designer]

- + FastReport Cloud'dan, son dosyalar listesini kullanarak rapor açma yeteneği eklendi.
- + Sayfa paneli öğelerine sağ tıklama menüsü eklendi.
- + Rapor sayfaları panelinde, yeni sayfalar ve diyalog formları oluşturmak için sağ tıklama menüsü eklendi.
- + Şerit arayüzü için Visual Studio tarzı yeni simgeler eklendi.
- + "Rapor Ağacı" penceresine **Sync** düğmesi eklendi.
- + Özellikler penceresine Filtre düğmesi eklendi.
- + Şerit arayüzü için HiDPI simgeler eklendi.
- + Parametreler için DBNull ve Guid tiplerinin desteği eklendi.
- * Mail'e dışa aktarımda ekli dosyanın adı, dışa aktarma formu oluşturulurken kod üzerinden ayarlanabilir hale getirildi.
- * Rapor doğrulayıcı artık **"Rapor|Raporu Doğrula"** menüsünden çalıştırılıyor; doğrulama mesajları "Mesajlar" penceresinde gösteriliyor.
- * QR kod düzenleyicisinin arayüzü değiştirildi.
- Veri Kaynakları menü öğesine sağ tıklamada oluşan hata giderildi.
- Data sihirbazında "Tümünü Seç" onay kutusunun görünmemesi hatası düzeltildi.
- Diyalog formu silinirken oluşan System.NullReferenceException hatası giderildi.
- Standart sunucu öğesinde girildiği takdirde, Account → Server penceresi yeniden açıldığında API anahtarının görünmemesi sorunu çözüldü.
- Özel sunucu için webview önizlemesi yapılırken yanlış web adresi sorunu düzeltildi.
- Şerit arayüzünde yerleştirme değiştirildiğinde, sekmelerin ve çubukların dilinin yanlış değiştirilmesi ile panellerin çökmesi sorunu giderildi.
- Bağlantı sihirbazında seçilmeyen tabloların eklenmesiyle ilgili hata düzeltildi.
- Saklı prosedüre bağlantı oluşturulurken oluşan System.NullReferenceException hatası giderildi.
- Geçersiz bir parametre tipi manuel girildiğinde oluşan istisna hatası düzeltildi.
- Bir nesnenin şeffaf renge ayarlanamaması hatası çözüldü.
- Sorgu sihirbazının yeniden açılması hatası giderildi.

[Preview]

- + Durum çubuğuna, raporun mail ile gönderilmesi hakkında bilgi veren mesaj eklendi.

[Exports]

- + Excel 2007'ye dışa aktarım sırasında hücrelerde kelime kaydırması (word wrapping) eklendi.
- HTML dışa aktarımından sonra MSChart metninin bulanıklaşmasına neden olan hata düzeltildi.
- Rapor HTML'ye aktarılırken yanlış kenar boşlukları problemi giderildi.
- Skia ile dışa aktarımda şeffaf arka planın beyaz hale gelmesine neden olan hata düzeltildi.
- Exportable özelliği false olan bantlar varsa, dışa aktarım sırasında ekstra boş sayfa oluşmasına neden olan hata giderildi.
- Katmanlı HTML dışa aktarımında üst boşluk (padding top) dikkate alınmadığında oluşan hata düzeltildi.
- Sayfa HTML dışa aktarımında küçültüldüğünde metnin tablonun ötesine geçmesine neden olan hata giderildi.

[WebReport]

- + Blazor WebAssembly desteği eklendi.
- + WebReport.Core/Blazor'da DI (bağımlılık enjeksiyonu) desteği eklendi. Kullanmak için:
`services.AddFastReport();` çağrısını yapın.
- + Standart WebReportLegacyCache yerine Microsoft.Extensions.Caching.Memory.MemoryCache desteği eklendi. DI konteyneri kaydederken, `services.AddFastReport(options => options.CacheOptions.UseLegacyWebReportCache = false);` kullanın.
- + CheckedListBox'ta ****ItemCheck**** olayı uygulandı.
- + ****webReport.Toolbar.Sticky**** özelliği kullanılarak, araç çubuğunun ekran konumundan bağımsız olarak görüntülenmesini sağlayan seçenek eklendi.
- + WebReport.Designer.SaveMethod metodunun asenkron sürümü (WebReport.Designer.SaveMethodAsync) eklendi.
- + WebReport dışa aktarma ayarları penceresinde sayfa aralığı doğrulaması eklendi.
- + WebReport.Toolbar.Exports.PinnedSettingsPosition özelliği eklendi. Etkinleştirilirse, dışa aktarma ayarları konteyneri ekranda sabitlenip ön planda gösterilir.
- ListBox'ta seçim modu çoklu iken yalnızca tek öge seçilebilmesine neden olan hata düzeltildi.
- Olayı başlatan CheckedException nedeniyle dialog pencerenin yenilenmemesi sorunu giderildi. (Bu durumda, CheckedException'un DetailControl özelliğine en az bir bağımlı nesne ekleyin.)
- .NET Framework MVC uygulamasında, "OK" butonuna tıklanarak açılan diyalog formunun gizlenmemesi ve raporun yükleme durumunun gösterilmemesi sorunu düzeltildi.
- Yazdırma sırasında ekstra sayfaların görünmesine neden olan hata giderildi.
- WebReport.Core üzerinde "Etkileşimli Rapor"un yanlış çalışması sorunu düzeltildi.
- WebReportLegacyCache'te nadir görülen NullReferenceException hatası giderildi.

[Online Designer]

- ReportPage değiştirildiğinde ****First Page Source****, ****Other Page Source****, ****Last Page Source**** ve ****Duplex**** özelliklerinin kaydedilmemesi hatası düzeltildi.
- "Yenile" düğmesine basılmadan önce rapor önizlemesinin yenilenmemesine neden olan hata giderildi.

[.Net Core]

- + Script derleyicisi, FastReport.Utills.Res.LoadLocale() veya FastReport.Utills.Config.CompilerSettings.CultureInfo ile ayarlanan seçili yerel ayara bağlı olarak artık hataları gösterecektir.
- CanShrink = True olan metnin Skia ile dışa aktarımdan sonra yanlış renderlanması sorunu giderildi.
- Skia'da ****TextRenderType = HtmlTags**** olan karakterler arasındaki girinti genişliğinin yanlış hesaplanması hatası düzeltildi.
- Skia ile dışa aktarımda, şeffaflığa sahip filigranın gri arka plana sahip olmasına neden olan hata giderildi.
- Tablo satır yüksekliğinin yanlış hesaplanmasına neden olan hata düzeltildi.

[CoreWin]

- Yeni veri bağlantısı eklenmeye çalışılırken oluşan hata giderildi.

[Mono]

- + Önizleme ve tasarımcı pencerelerinde yakınlaştırma (zoom) kontrolü eklendi.
- PreviewControl ölçeklendirme sorunu düzeltildi.

[Demos]

- + .NET 6.0 altında ASP.NET Core (Razor sayfaları) demo uygulaması eklendi.
- * FastReport Core için demo uygulamaları güncellendi.

[Extras]

- Çıkış yaparken ev sahibinin (host) yetkilendirme sırasında olanla eşleşememesi durumu giderildi.
- Account penceresinde süresi dolmuş oturum güncellenirken, tarayıcı açılarak yeniden yetkilendirme istenmesi hatası düzeltildi.

Sürüm 2023.1

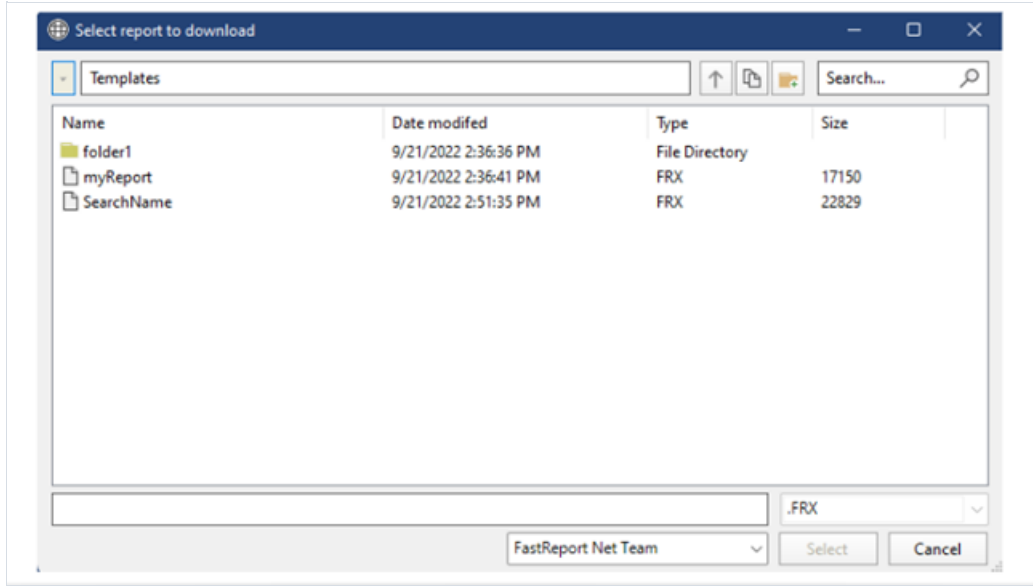
Yeni Fırsatlar

FastReport Cloud ile Entegrasyon

FastReport .NET, FastReport CoreWin ve FastReport Mono artık FastReport Cloud ile bazı etkileşim deneyimlerini destekliyor.

Raporların İndirilmesi ve Yüklenmesi

Artık Cloud'dan raporu indirip tasarımcıda üzerinde çalışabilir ya da tersine dosyalarınızı Cloud'a yükleyebilirsiniz.



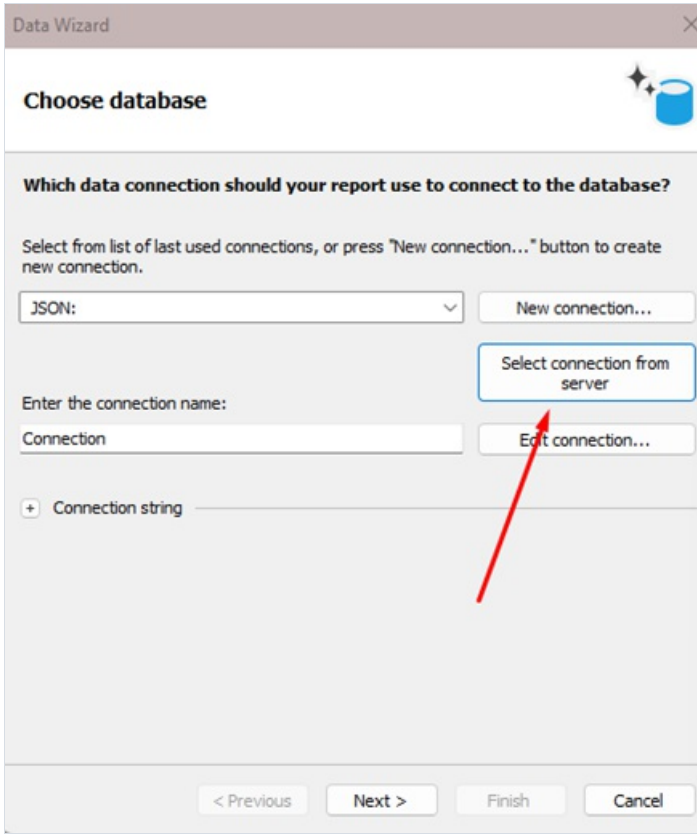
Web Önizlemesi

Standart önizlemenin yanı sıra bir web önizleme işlevi de eklendi. Rapor, yalnızca Cloud'dan açıldıysa bu şekilde görüntülenebilir.

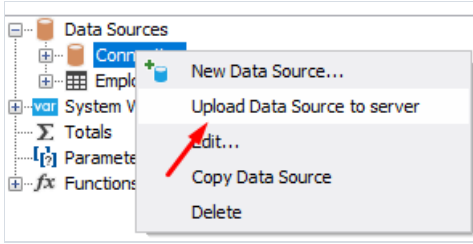


FastReport Cloud Veri Kaynaklarına Bağlantı

FastReport Cloud, veri kaynaklarına bağlantıları depolayabiliyor. Artık bu veri kaynaklarını raporunuza ekleme seçeneğiniz bulunuyor.



Ayrıca, bağlantıyı Cloud'a yüklemek mümkün hale geldi.



Yeni özellikler hakkında daha fazla bilgi edinmek için makaleyi okuyun.

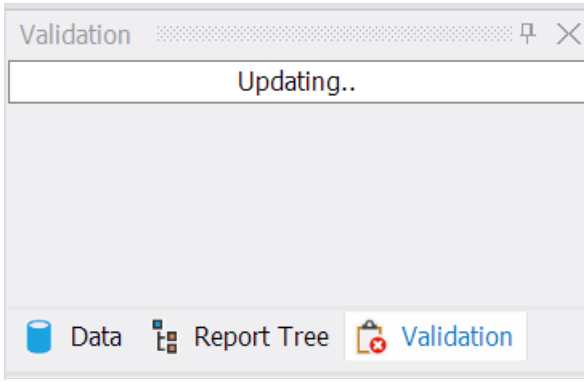
.NET 7 Desteği

FastReport.Core ve FastReport.CoreWin için .NET 7 desteği ekledik. Bu platform, uygulama performansını artırır ve projelerinize birçok yeni özellik ekler.

Rapor Doğrulayıcı İyileştirmeleri

Artan Çalışma Hızı

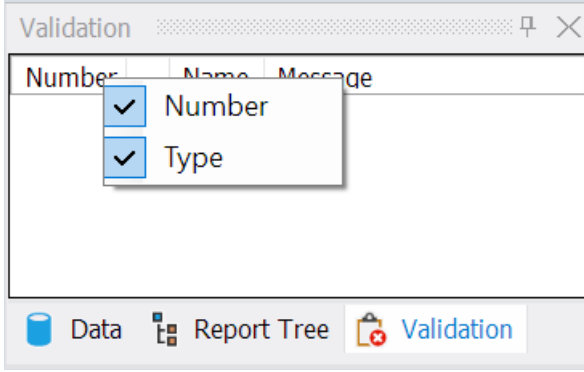
Artık rapor doğrulayıcı tek iş parçasığında çalışıyor. Çalışma hızı önemli ölçüde optimize edildi. Çok sayıda hata içeren raporların işlenmesinde fark edilebilir değişiklikler göreceksiniz. Doğrulayıcı raporu kontrol ederken, kontrol penceresinde ilgili bir mesaj görüntüleniyor.



Bu durumda raporu düzenleyebilirsiniz. Doğrulayıcı tamamlandığında, hataların bulunduğu bir tablo görünecektir.

Doğrulayıcı Tablosu Ayarları

Kolaylık olması için, hata numaralarını içeren yeni bir sütun ekledik. Bu sütunun görüntülenmesi, tablo bağlam menüsü aracılığıyla etkinleştirilebilir veya devre dışı bırakılabilir. Aynı şekilde, hata türü sütununun görüntülenmesi de özelleştirilebilir.



JasperReports Şablon Dönüştürücü

JasperReports şablonlarını FastReport .NET şablonlarına dönüştürme seçeneğini ekledik. JasperReports raporları, FastReport tasarımcısı tarafından desteklenmeyen nesneler içerebilir. Bu nesneler dönüştürülmeyecek ya da JasperReports'ta oluşturulan rapora mümkün olduğunca benzer olması için değiştirilecektir.

Detaylar için makaleyi okuyun.

MSChartObject İyileştirmeleri ve Düzeltmeleri

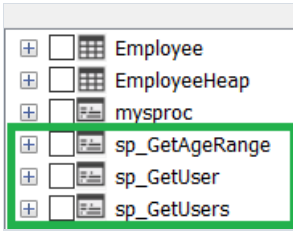
MSChartObject nesnesi birçok özellik ve ayara sahiptir. En çok kullanılanlar nesne düzenleyiciye taşındı. Düzenleyicide bulunmayan özellikler, Object Inspector kullanılarak değiştirilebilir. Ancak, bu özelliklerde, değiştirildiklerinde raporun değiştiği algılanmama sorunu vardı; bu nedenle kaydetme işlemi mümkün olmuyordu.

Raporu kaydetmek için başka bir özelliğin ya da nesnenin değiştirilmesi gerekiyordu. Ayrıca, belirtilen özelliklerin değerleri rapor hazırlanırken ve önizleme penceresi kapandıktan sonra varsayılan değerlere sıfırlanıyordu.

Bu hata yeni sürümde giderildi.

MsSQL'deki Saklı Yordamlar ile Bağlantı

MsSQL'de saklanan prosedürlere bağlanma seçeneğini ekledik. Bu özellik daha önce bir veritabanı sorgusu aracılığıyla sağlanıyordu. Artık veritabanı tablo bağlantısı arayüzü sayesinde prosedürlere çok daha kolay bağlanabilirsiniz. Seçim penceresinde, tablolarla birlikte prosedürler de görüntülenecektir.



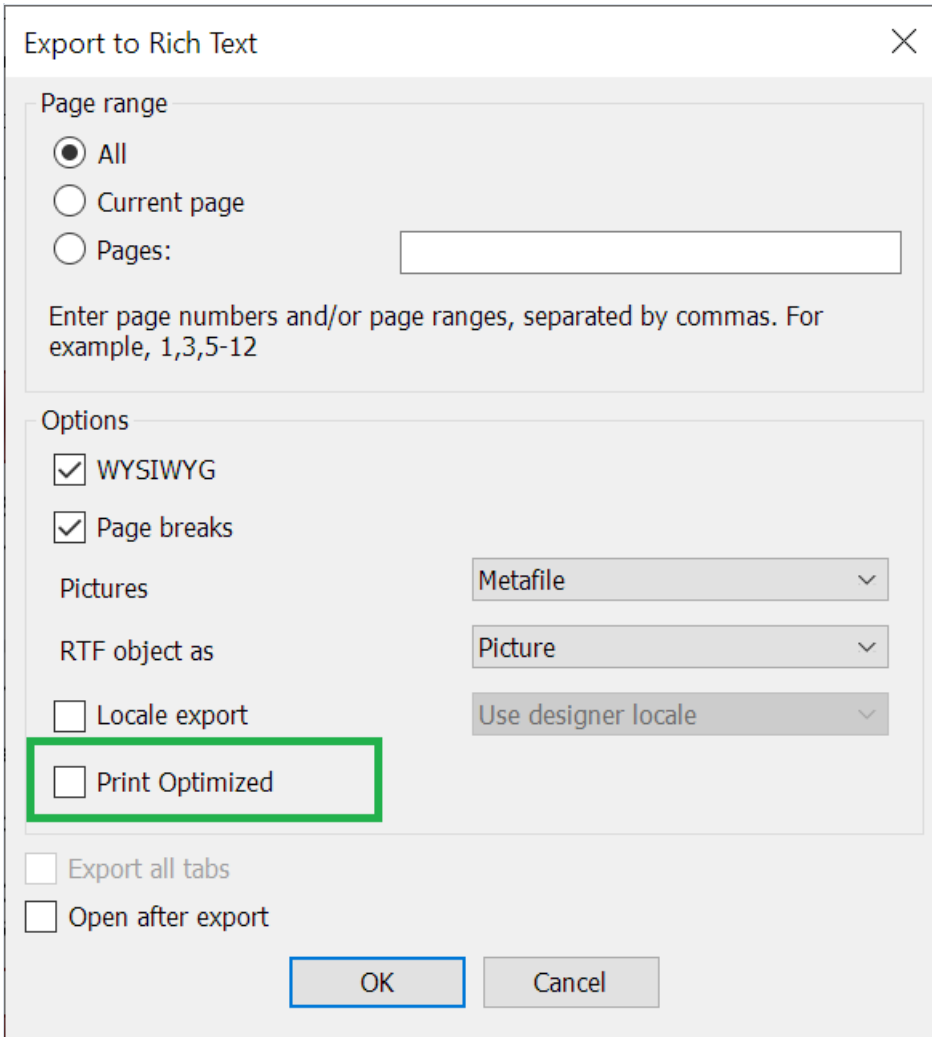
Bir prosedür seçtiğinizde, varsa parametre ayarlarını içeren bir pencere açılacaktır.

Detaylar için makaleyi okuyun.

Dışa Aktarma İyileştirmeleri

RTF Dışa Aktarımında "Baskı Optimize Edildi" Seçeneği

Yeni bir PrintOptimized özelliği ve dışa aktarma penceresinde buna karşılık gelen bir seçenek ekledik. Bu seçeneğin etkinleştirilmesi, dışa aktarılan görüntünün kalitesini büyük ölçüde artırır; ancak çıktı dosyasının boyutu artacaktır.



Excel 2007 Dışa Aktarımında UseFileStream Özelliği

Excel 2007'e dışa aktarım için UseFileStream adlı yeni bir seçenek ekledik. Bu seçenek yalnızca koddaki raporu bir dosyaya dışa aktarırken kullanılabilir. Çok sayıda sayfaya (on binlerce) sahip raporların birden fazla iş parçasığında dışa aktarımında bellek yetersizliği hatalarını önlemede faydalıdır. Diğer durumlarda kullanılması pek mantıklı değildir ve önerilmez. Örnek:

```
Report report = new Report();
Excel2007Export export = new Excel2007Export();
export.UseFileStream = true;
report.Export(export, "report.xlsx");
```

Excel 2007 Dışa Aktarımında Muhasebe Formatı

Artık para birimi veri formatını muhasebe formatı olarak dışa aktarabilirsiniz. Bunun için dışa aktarma penceresinde ilgili bir seçenek ve CurrencyToAccounting özelliği eklenmiştir.

Export to Excel 2007

Basic

Other

Pinned cells

☒ Without pinning

☐ First line

☐ First column

☐ Other Pinning

Columns: 0

Rows: 0

Print Scaling

☒ No Scaling

☐ Set scale in percent: 10

Pictures properties

☒ Move and modify along with the cell

☐ Move but not resize

☐ Do not move or resize

Options

☐ Use locale formatting of data

☒ Convert currency to accounting format

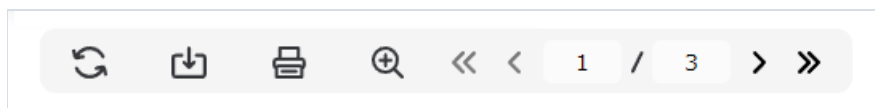
☐ Export all tabs

☒ Open after export

OK Cancel

WebReport Core/Blazor Tasarım Güncellemesi

Araç çubuğunun görünümü değiştirildi; renk paleti daha açık tonlara çevrildi ve araç çubuğu varsayılan olarak ortalandı.



Ayrıca, dışa aktarma ayarlarının formları da değiştirildi. Önceden bazı özellikler kaydırıcı kullanılarak ayarlanıyordu; artık bu özellikler açılır liste olarak ayarlanıyor.

Export to PDF

Page range

☒ All

☐ Current page 2 or 10-20

Options

PDF 1.5

☐ Embedded Fonts

☐ Text in Curves

☐ Interactive Forms

☐ Background

Images

Jpeg quality: 90

☒ RGB

☐ CMYK

None

Cancel

OK

Araç çubuğu tasarımını değiştirdik, ancak özelleştirme imkânını koruduk; böylece önceki tasarımı kısmen geri yüklemek mümkün. Bunu yapmak için aşağıdaki ayarları kullanın:

```
var toolbarSettings = new ToolbarSettings
{
    Color = Color.LightGray,
    Roundness = RoundnessEnum.None,
    ContentPosition = ContentPositions.Left,
    IconTransperency = IconTransparencyEnum.None
};

webReport.Toolbar = toolbarSettings;
```

.NET Framework Minimum Sürümünün 4.0'dan 4.6.2'ye Yükseltilmesi

Aşağıdaki nedenlerden dolayı FastReport .NET'in desteklenen minimum sürümünü .NET Framework 4.6.2'ye yükseltiyoruz:

- .NET Framework 4.0 desteği uzun zamandır sona erdi;
- Microsoft Visual Studio'nun en son sürümlerinde FastReport kaynak kodunun derlenmesinde sorunlar var;
- Yeni API'lerin uygulanması gerekliliği.

Değişikliklerin Tam Listesi

```
[Engine]
+ Report.IsPrepared özelliği eklendi;
+ TextRenderType.Inline eklendi;
+ JasperReports şablon dönüştürücüsü uygulandı;
+ MsSQL'deki saklı yordamlarla bağlantı uygulandı;
* .NET Framework minimum sürümü 4.0'dan 4.6.2'ye yükseltildi;
* Veri kaynağında JSON alınması arayüz kısmına açıldı;
- TextObject.FontWidthRatio özelliği sıfır olduğunda System.ArgumentException'a neden olan hata düzeltildi;
- RTF ayrıştırıcısında metin vurgulaması düzeltildi;
- ImageLocation'da URL kullanıldığında görüntü almak için yapılan çoklu istekler düzeltildi;
- IsNull fonksiyonu düzeltildi;
```

- Hizalanmış resimlerle RichObject çizimindeki hata düzeltildi;
- RichObject'in metne dönüştürülmesi sırasında dikey mesafe hesaplaması hatası düzeltildi;
- Rapor yenileme sırasında AdvMatrix nesnesi hatası düzeltildi;
- JsonTableDataSource'in JSON satırı alınırken oluşan hata düzeltildi;
- Sayfada bir satır için yeterli yer olmadığında tablo oluşturulurken sonsuz döngüye neden olan hata düzeltildi;

[Designer]

- + IEnumerable veri kaynakları için hesaplanmış sütun oluşturma yeteneği eklendi;
- + Dosya açılırken rapor yüklemesi hakkında mesaj içeren pencere eklendi;
- + Rapor doğrulama tablosuna hata numaralarını içeren sütun eklendi;
- + Rapor doğrulama tablosundaki hata numarası ve türü sütunlarını gizleme/gösterme yeteneği eklendi;
- + Zaten değiştirilmiş raporu yeniden kaydetmeye çalışırken bildirim formu eklendi;
- + FastReport Cloud'dan açılan raporun web önizlemesini gösterme yeteneği eklendi;
- + Cloud'daki veri kaynağı ile etkileşime girme (indirme, yükleme, güncelleme) yeteneği eklendi;
- * Rapor doğrulayıcısının hızı artırıldı;
- * Bantın silinemediği durumlarda silme düğmesi devre dışı bırakıldı;
- * FastReport Cloud formundaki kök klasör adı, artık yerelleştirmeye bağlı olarak değiştirildi;
- * Aynı isimde başka bir tablo varsa, sorgu sihirbazında tablo oluşturmak engellendi;
- IEnumerable veri kaynağı kullanıldığında, değerden oluşan sütunun eklenmemesi sorunu düzeltildi;
- "Script" sekmesindeki rapor özelliklerinde "Kaldır" düğmesinin yerelleştirilmesi hatası giderildi;
- "Doğrulama" penceresinde satıra tıklandığında seçim nesnesi hatası düzeltildi;
- Fare ile bant yüksekliği değiştirilirken seçili nesnenin değişmemesi sorunu giderildi;
- Düzenleyici olmadan metin nesnesi düzenlenirken System.OverflowException hatası düzeltildi;
- Biçim kopyalama sırasında System.StackOverflowException hatası giderildi;
- "Doğrulama" penceresinde, aktif olmayan sayfadaki nesnenin seçilme hatası düzeltildi;
- Rapor değiştirildiğinde "Doğrulama" penceresinde hata listesi güncelleme ilerlemesi sorunu düzeltildi;
- MSChartObject düzenleyicisinde çizgi rengi değiştirilirken geçersiz değer hatası giderildi;
- Bağlantı formlarında "Tab" tuşu ile geçiş sırası düzeltildi;
- Düzenleyicide değiştirildiğinde grafik eksenlerinin kenar özellikleri kaydedilmediği sorunu düzeltildi;
- MSChartObject'de eksen şerit aralığı değiştirilirken yanlış değer hatası giderildi;
- Bantların klasik modda görüntülenmesi seçiliyken bant yapılandırıcısı aracılığıyla bant silinmesi sırasında oluşan hata düzeltildi;
- Rapor içinde bant yoksa bant yapılandırma formunda "Sil" düğmesine tıklanması hatası giderildi;
- Sol fare düğmesine basılı tutarak çalışma alanından bant kaldırılırken oluşan hata düzeltildi;
- Sütun düzenleyici form kapatıldığında GridControl durumunun geri yüklenmesi hatası düzeltildi;
- Grid nesnesi sütun düzenleyicisinde "İptal" düğmesine tıklanırken oluşan hata giderildi;
- MSChartObject değiştirildiğinde rapor değişikliği etiketinin gösterilmesi sorunu düzeltildi;
- GridControl sütunlarının sütun düzenleyici formda taşınması hatası giderildi;
- Rapor ağacından sayfalara ve "Kod" sekmesine nesne sürüklerken oluşan hatalar düzeltildi;
- Çalışma alanına tablo eklenirken ve tablolar arası ilişki kurulurken sorgu yapıcı penceresindeki hatalar giderildi;

[Preview]

- + "E-posta ile Gönder" formundaki "Kopyala" alanı için araç ipucu eklendi;
- Görüntü ölçeklendirmesi artırıldığında yazdırma formu görüntüleme hatası düzeltildi;
- Yeni dışa aktarımların menüde görünmemesi sorunu giderildi;
- Dışa aktarma formlarında "Tab" tuşuyla geçiş sırası düzeltildi;
- RichObject'in sol girintisi düzeltildi;

[Exports]

- + RTF dışa aktarımında "Baskı optimize edildi" seçeneği eklendi;
- + Excel 2007 dışa aktarımında para birimi veri formatını muhasebe formatı olarak dışa aktarma yeteneği eklendi;
- + Excel 2007 dışa aktarımı için UseFileStream seçeneği eklendi;
- * Farklı yerelleştirmelerde yazıların doğru görüntülenebilmesi için dışa aktarma formları geliştirildi;
- Katmanlı HTML dışa aktarımında sağdan sola veya alttan üste çizilen çizgilerin dışa aktarım hatası giderildi;
- Tahoma italik yazı tipinin PDF'e dışa aktarım hatası düzeltildi;
- ImageExport.SeparateFiles özelliği etkinleştirilmişken akışa dışa aktarım sırasında System.ArgumentException hatası giderildi;
- SVG dışa aktarımında bazı şekillerin iki kez çizilmesi sorunu düzeltildi;
- Excel 2007'de muhasebe formatı dışa aktarımında ondalık basamak sayısını hesaba katmama hatası giderildi;
- Tablo tipi dışa aktarımlarda bellek sızıntıları giderildi;
- PDF dışa aktarımı sırasında acil kapanma durumunda geçici dosya silme hatası düzeltildi;
- PDF'e italik ve kalın yazı tiplerinin dışa aktarım hatası giderildi;
- Tarayıcıdan baskıya gönderilen Solid dolguya sahip nesnelerin arka planının yazdırılmama sorunu düzeltildi.

- HtmlTemplates.IndexTemplate özelliğinin set metoduyla ilgili hata giderildi;
- 4 baytlık sembollerin PDF'e dışa aktarım hatası düzeltildi;
- RTF dışa aktarımında satır yüksekliği çarpanı hatası giderildi;
- Word 2007'e tablo dışa aktarımında satır yüksekliği çarpanı hatası düzeltildi;
- Word 2007'e dışa aktarımda, sıfır olmayan değere sahip sayfadaki ilk nesnenin pozisyonu düzeltildi;
- UseFileStream ve SplitPages özelliklerini kullanarak Excel 2007 dışa aktarımında geçici dosya erişim hatası giderildi;
- Excel 2007 dışa aktarımında CurrencyToAccounting özelliğinin yerelleştirilmesi hatası düzeltildi;
- HTML dışa aktarımında gezinme düğmeleri ve sayfa numaralandırma görüntüleme hatası giderildi;
- PDF dışa aktarımında yazı tipi yükselme ve alçalma sorunları düzeltildi;

[WebReport]

* WebReport.ReportPrepared özelliği yeniden düzenlendi; artık bu özellik aynı raporun özelliğine bağlı;
* FastReport.Core.Web ve FastReport.Web.Blazor için WebReport tasarımı güncellendi;
- WebReport'a boş bir veri kaynağı eklenmeye çalışıldığında nadir görülen çökme hatası giderildi;
- WebReport.LoadPrepared() çağrısında Outline'ın çalışmama sorunu düzeltildi;
- Hazırlanmış rapor (.fpx) yüklenirken yenileme düğmesi kaldırıldı;
- WebReport'ta RichObject sekmelerinin yanlış hesaplanması sorunu giderildi;
- Tek sayfalı raporlar için dışa aktarma ayarlarında sayfa seçimi kaldırıldı;

[.Net Core]

+ .NET 7 desteği eklendi;
+ Stimulsoft Import için dosya adı yerine akış kullanarak LoadReport yöntemi eklendi;

[CoreWin]

- Visual Studio araç çubuğundan ekstra bileşenler kaldırıldı;

[Demos]

* Demo raporlarının tasarımı güncellendi;
- Yeni demo uygulamasında alt pencerelerin ölçeklendirilmesi hatası düzeltildi;

[Extras]

+ FastReportBGObjects eklentisi güncellendi, Bubble chart desteği eklendi;
* ElasticSearch bağlantısı ayrı bir eklentiye taşındı.

Sürüm 2022.3

Yeni Özellikler

Skia Desteği

Artık FastReport.Core, Linux sistemlerinde System.Drawing.Common + libgdiplus yerine kullanılan SkiaSharp kütüphanesiyle grafik ve metin çizimini destekliyor (diğer işletim sistemlerinde de çalışıyor).

Bu amaçla .Skia son ekine sahip paketler kullanılıyor:

- FastReport.Core.Skia
- FastReport.Web.Skia

Bu sürüm, .NET Framework'ü sınırlı olarak desteklemekte olup, esas olarak .NET Core/.NET projeleri için tasarlanmıştır. Uygulamanızda kullanmak için tek yapmanız gereken, FastReport.Core paketinin adını FastReport.Core.Skia olarak değiştirmek ve Linux'ta çalışabilmesi için aşağıdaki paketleri eklemek (Windows ve macOS'ta gerekli paketler otomatik olarak eklenir):

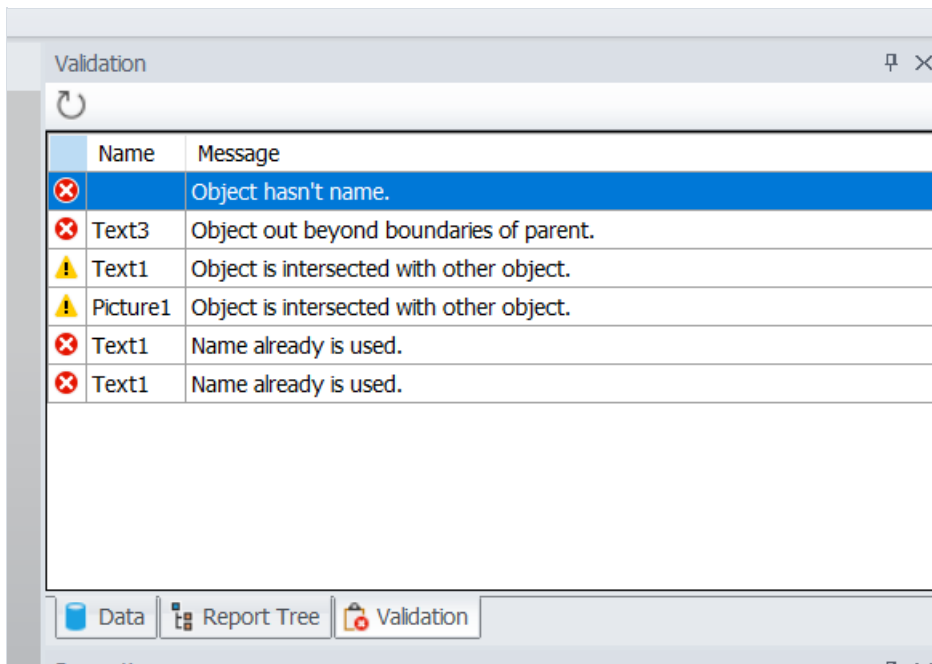
- SkiaSharp.NativeAssets.Linux
- HarfBuzzSharp.NativeAssets.Linux

Detaylar için makaleye bakınız.

Rapor Doğrulayıcı

Rapor tasarımcısına, "Veriler" ve "Rapor Ağacı" sekmelerinin yanında yer alan "Doğrulama" sekmesi eklendi. Bu sekmeden rapor şablonunuzu kontrol edebilir, hata ve uyarı listesini görebilirsiniz.

Liste, nesne adı (varsa) ve hata açıklaması şeklinde tablo formatında sunulur. Tablo üzerinde bir satır seçildiğinde, tasarımcıda ilgili nesne vurgulanır.



Doğrulayıcı aşağıdaki durumları tespit eder:

- Adı olmayan nesneler
- Aynı ada sahip nesneler (kritik hata)

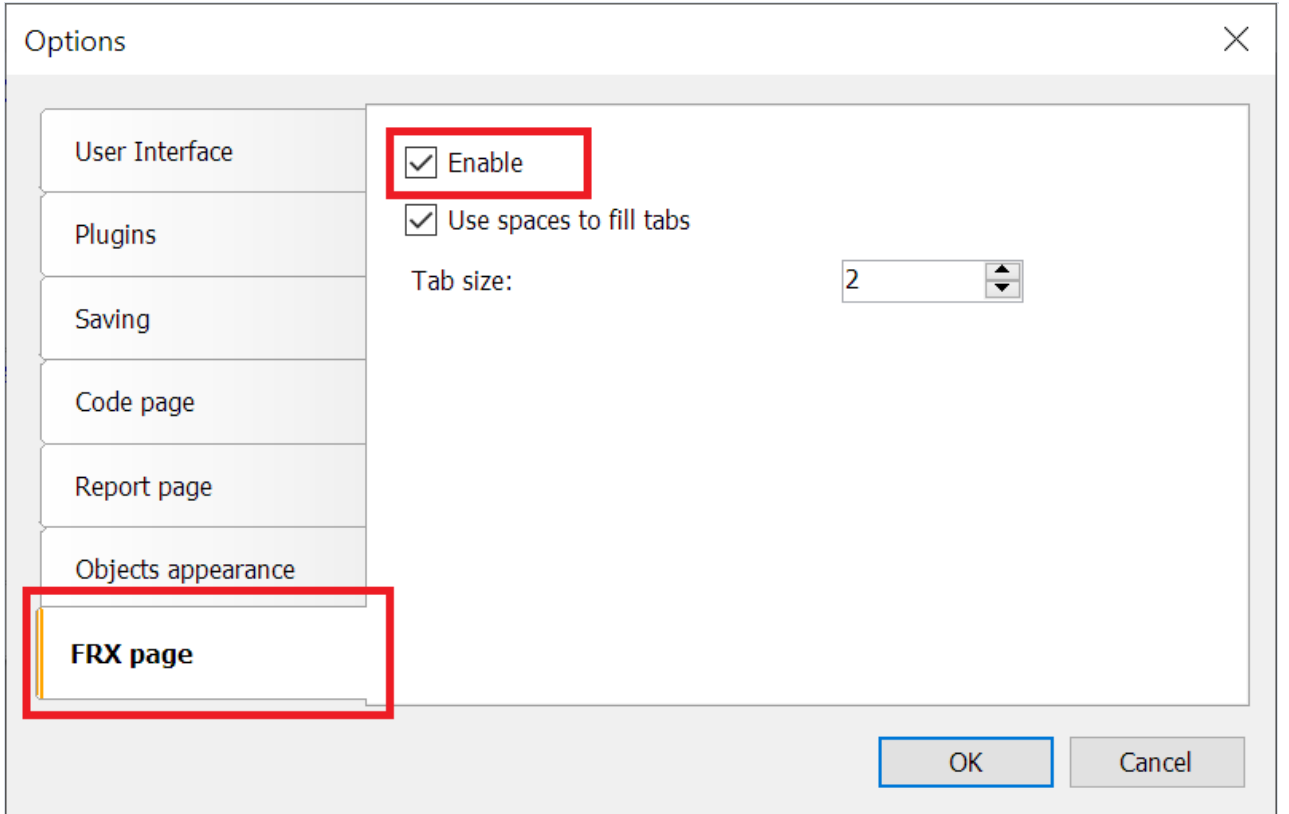
- Üst üste binen nesneler (bazı durumlarda kasıtlı kullanım için uygun, ancak metin nesnelerinde dışa aktarma sorunlarına yol açabilir)
- Sıfır yükseklik veya genişliğe sahip nesneler
- Ebeveyn nesnenin sınırlarının kısmen veya tamamen dışında kalan nesneler (tamamen dışarıda olanlar ciddi hatadır)

Raporunuzun beklediğiniz gibi çalışmadığı veya görünmediği durumlarda doğrulayıcıdan yararlanabilirsiniz.

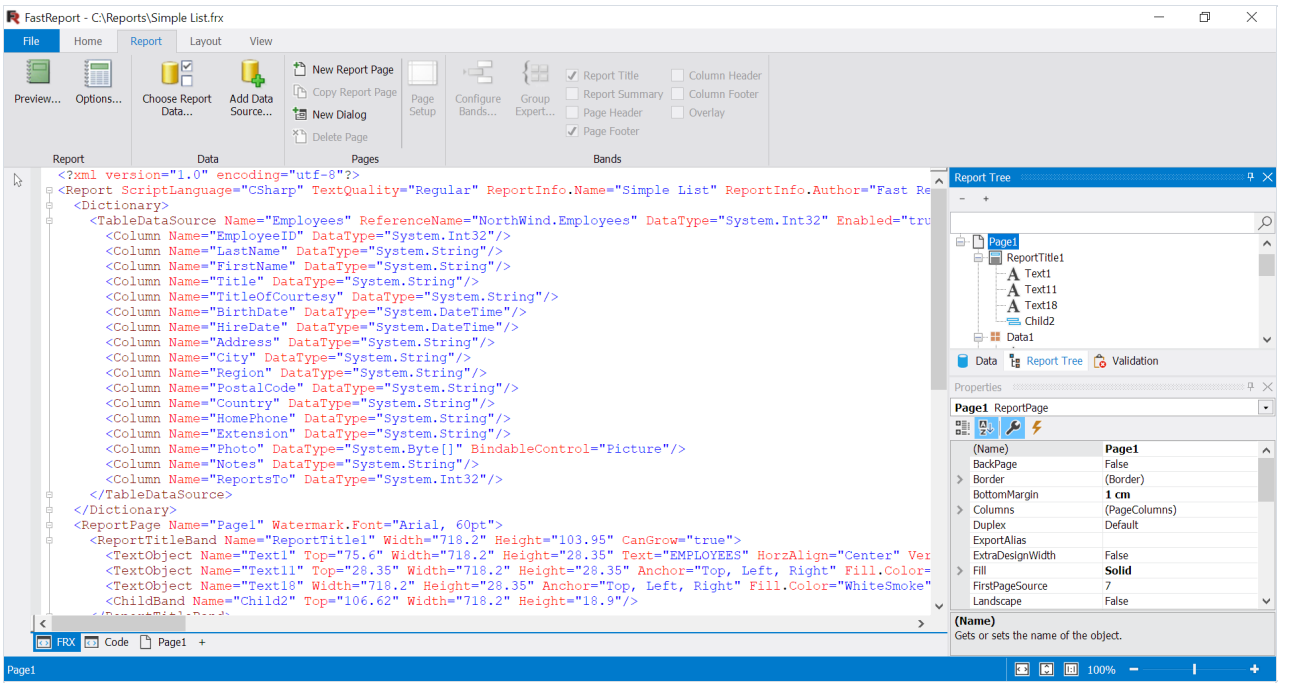
Detaylar için makaleye bakınız.

FRX Editörü

Bazen FRX dosyasının içeriğini üçüncü taraf metin editörleriyle düzenleme ihtiyacı doğabiliyor. Artık rapor tasarımcısı içinde FRX dosya içeriğini düzenlemek mümkün. Bu amaçla FRX formatı için bir editör eklendi. Varsayılan olarak kapalıdır; etkinleştirmek için tasarımcı ayarlarına gidin.



Tasarımcıda, Code sekmesinin solunda FRX sekmesi görünecektir.



Yapılan düzenlemeler anında rapora uygulanır ve sayfalarda görüntülenir. Detaylar için makaleye bakınız.

StimulSoft Raporlarından Dönüştürücü

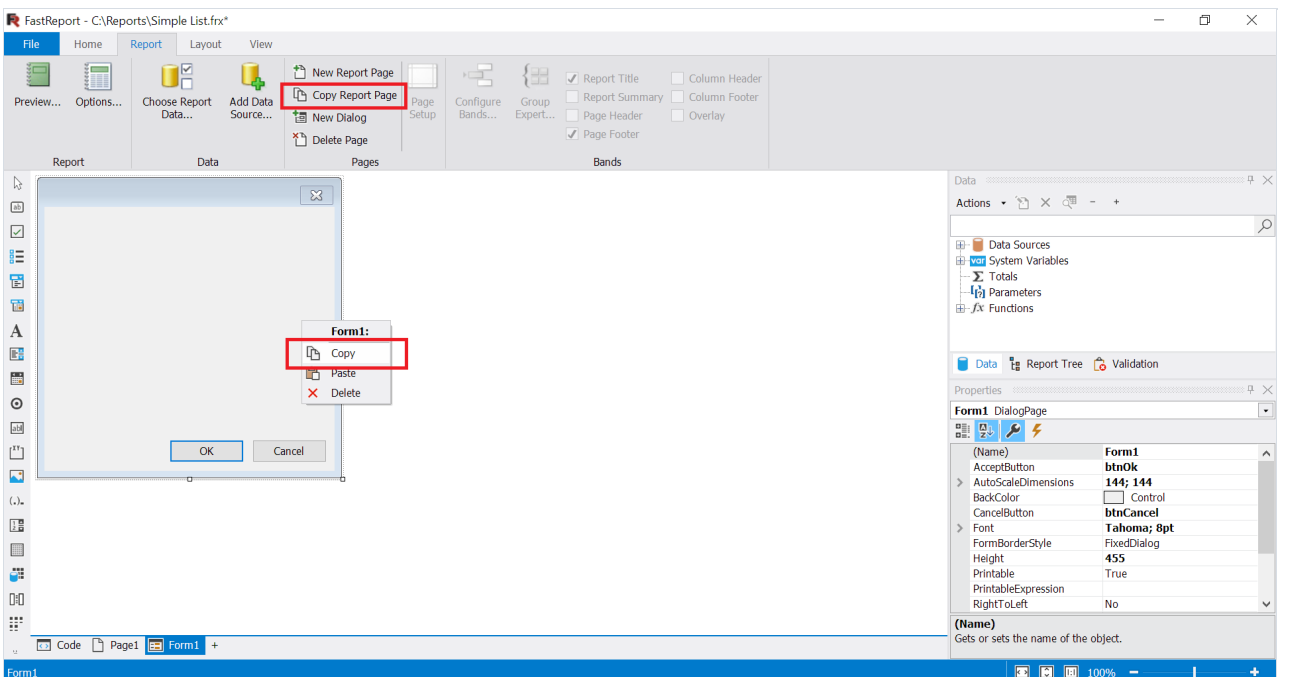
StimulSoft rapor şablonlarını FastReport .NET şablonlarına dönüştürme imkanı eklendi.

StimulSoft raporlarında, FastReport tasarımcısının desteklemediği bazı nesneler bulunabilir. Bu nesneler dışa aktarılmayacak ya da, raporun orijinaline olabildiğince benzemesi için farklı nesnelerle değiştirilecektir. Özellikle çapraz bantların dönüşümünde, içerikleri ebeveyn banda taşınarak uygulanır.

Detaylar için makaleye bakınız.

Diyalog Sayfalarının Kopyalanması

Diyalog sayfalarını kopyalama imkanı eklendi. Bu işlem, hem diyalog sayfası üzerindeki bağlam menüsü hem de "Rapor -> Rapor Sayfasını Kopyala" seçeneği ile gerçekleştirilebilir.



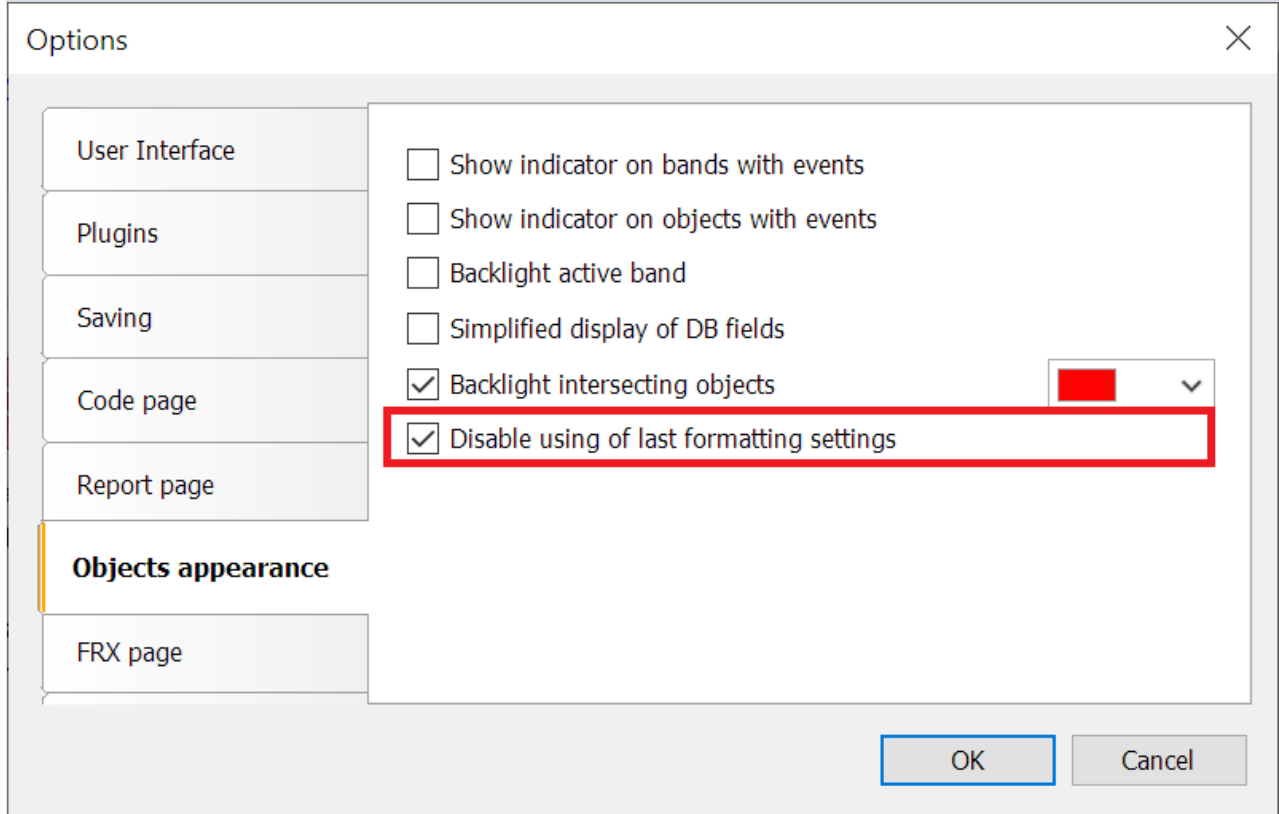
Kopyalama işlemi sırasında, diyalog sayfasının benzersiz bir adıyla kopyası oluşturulur. Alt nesneler de benzersiz adlar alır; ancak, olay işleyicileri orijinal sayfayla aynı kalır (gerekirse yeni işleyiciler ekleyebilirsiniz). Ayrıca, diyalog

sayfalarını artık form editörü veya rapor ağacı üzerinden bağlam menüsü ile silebilirsiniz.

Son Formatlama Ayarlarının Devre Dışı Bırakılması

Tasarımcıda oluşturulan nesnelerin ayarları, aynı türdeki bir sonraki nesneye otomatik olarak uygulanır. Örneğin, bir metin nesnesinde yazı tipi boyutu, kenarlık veya dolgu rengi ayarlandıysa, sonraki metin nesnesi bu ayarlarla oluşturulur.

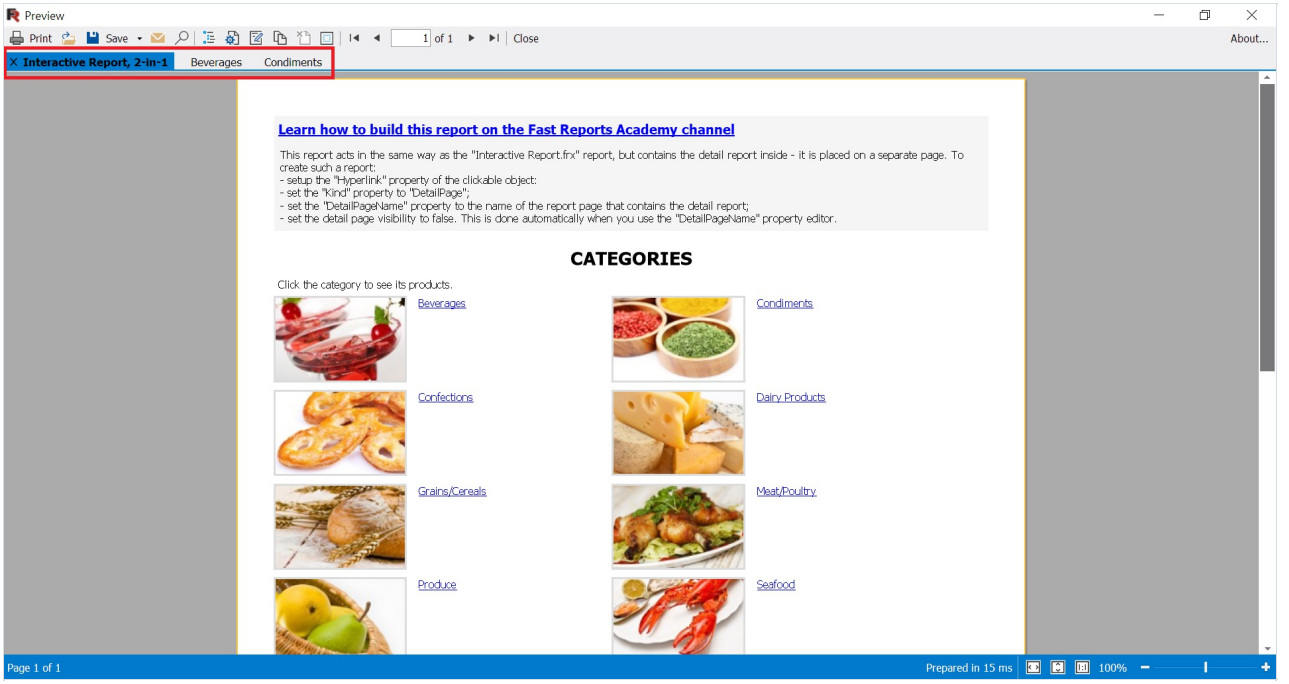
Bu, benzer ayarlarla birden fazla nesne oluşturmak için kullanışlıdır. Ancak, bu davranışı istemiyorsanız tasarımcı ayarlarından devre dışı bırakabilirsiniz.



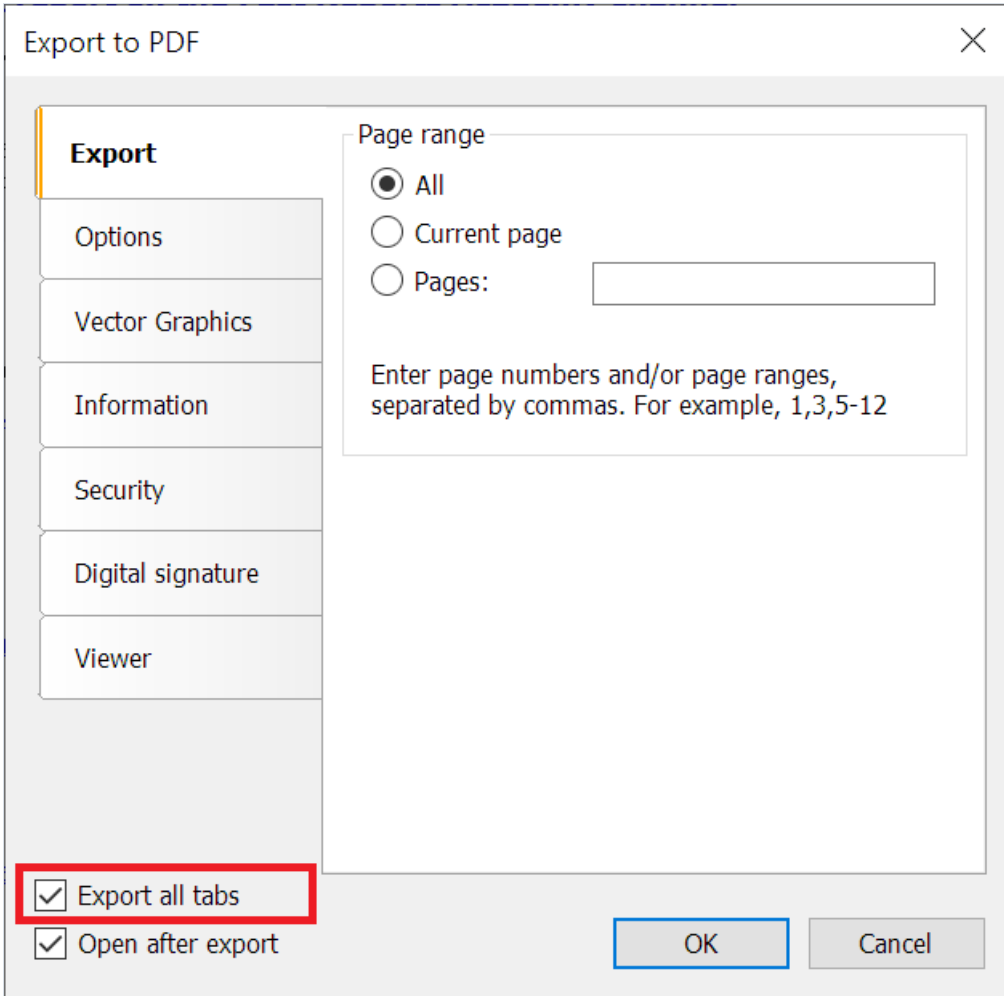
Bu durumda, nesneler varsayılan ayarlarla oluşturulur.

Tüm Sekmelerin Dışa Aktarılması

Etkileşimli raporlar incelenirken detay raporlar yeni sekmelerde açılabilir.



Önceki sürümlerde sadece aktif sekme dışa aktarılırken, artık "Tüm Sekmeleri Dışa Aktar" seçeneği ile tüm sekmeler tek bir dosyada dışa aktarılabilir.



İlgili Derlemelerin ve Yüklenmiş Eklentilerin Ayrıntılı Açıklaması

Ayarlar -> Eklentiler menüsündeki dll'lerin üzerine fare ile geldiğinizde ve Rapor -> Özellikler -> Script bölümündeki derleme bağlantılarında, açıklama, sürüm, boyut, oluşturulma tarihi vb. detaylı bilgiler görüntülenmektedir.

Dışa Aktarmalarda İyileştirmeler

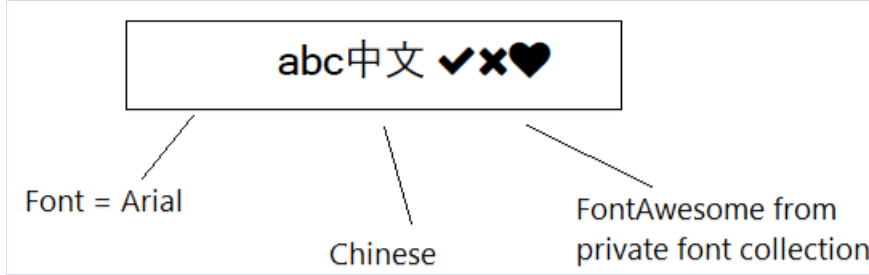
PDF Dışa Aktarma İyileştirmeleri

Linux Sürümü:

- Skia versiyonunda, karmaşık diller (Arapça, İbranice vb.) desteği eklenmiştir.

Tüm Sürümler:

- Font Fallback desteği (mevcut yazı tipinin desteklemediği karakterler için otomatik yazı tipi seçimi)



- Özel sembollerin (vokal işaretleri, aksanlar vb.) hassas konumlandırılması sağlanmıştır.

PDF Dışa Aktarımında UseFileStream Özelliği

PDF dışa aktarımına, yalnızca kod aracılığıyla dosyaya dışa aktarım yapıldığında kullanılabilen yeni bir UseFileStream seçeneği eklenmiştir. Bu özellik, çok sayıda sayfa (on binlerce) içeren raporların çoklu iş parçacıklarıyla dışa aktarımında bellek yetersizliği hatalarını önlemeye yardımcı olur. Diğer durumlarda özel bir anlamı yoktur. Örnek:

```
Report report = new Report();
PDFExport export = new PDFExport();
export.UseFileStream = true;
report.Export(export, "report.pdf");
```

Word, PowerPoint, Rich Text, OpenOffice Write ve OpenOffice Calc Dışa Aktarımında Dil Seçimi

Artık bu dışa aktarımlarda belge dili seçilebilir; varsayılan olarak tasarımcıda seçilen dil kullanılır.

Export to Microsoft Word 2007

Page range

☒ All

☐ Current page

☐ Pages:

Enter page numbers and/or page ranges, separated by commas. For example, 1,3,5-12

Options

☒ Table

☐ Layers

☐ Paragraphs

Row height is

☒ Wysiwyg

☐ Print Optimized

☐ Do not expand shift return

☐ Do not add section breaks on page breaks

☒ Locale export

☐ Export all tabs

☒ Open after export

OK Cancel

Ayrıca, Excel 2007 dışı aktarımında "Kılavuz Çizgilerini Göster" seçeneği eklenmiştir.

Tam Değişiklikler Listesi

[Engine]

- + StimulSoft rapor dönüştürücüsü eklendi;
- + İfadelerde JSON veri kaynağının adının değiştirilmesi durumunda ad güncellemesi yapıldı;
- + StimulSoft raporlarından dönüştürme sırasında PaperSize özelliğinin çevrilmesi eklendi;
- + Dönüştürme sırasında ilgili derlemelerin varlığı kontrol edildi;
- + Table ve Matrix nesnelere PrintOnParent özelliği eklendi;
- + RDL raporlarından dönüştürmede rapor parametrelerinin yüklenmesi sağlandı;
- + RDL raporlarından dönüştürmede iç raporların yüklenmesi desteklendi;
- + JSON bağlantı verilerinin StoreData özelliği ile saklanabilmesi eklendi;
- + Çok sayıda nesne içeren raporlarda performans iyileştirildi;
- * Diğer hata düzeltmeleri ve iyileştirmeler yapıldı;

[Designer]

- + Tekrarlanan isimler, negatif boyutlar vb. hataları tespit eden rapor doğrulayıcı eklendi;
- + RichObject.Text için editör eklendi;
- + Tasarımcıya FRX editörü eklendi;
- + İlgili derlemelerin ve yüklenmiş eklentilerin detaylı açıklaması sunuldu;
- + Diyalog sayfalarını kopyalama ve bağlam menüsü ile silme imkanı eklendi;
- + Nesne oluşturulurken önceki formatlama ayarlarının kullanılması devre dışı bırakılabilir hale getirildi;
- + FastReport.Id entegrasyonu, çevrimiçi dokümantasyon, kontrol kimlik işaretleri görselleştirme sihirbazı ve diğer geliştirmeler eklendi;
- + Bağlantı editörü ve diğer hata düzeltmeleri gerçekleştirildi;

[Preview]

- + Açık tüm sekmelerin dışa aktarılması sağlandı;
- Yazıcısız sistemlerde RichObject bulunan raporların hazırlanmasında oluşan hatalar giderildi;
- HiDPI modunda MSChart nesnesiyle ilgili sorunlar düzeltildi;

[Exports]

- + Word, PowerPoint, Rich Text, OpenOffice Write ve OpenOffice Calc dışa aktarımlarında yerel ayar desteği eklendi;
- + PDF dışa aktarımında dijital imza sertifikasının şifre parolasının şifrelenmesi sağlandı;
- + Excel 2007 dışa aktarımında "Kılavuz Çizgilerini Göster" seçeneği eklendi;
- + DBF dışa aktarımında veri türlerinin aktarımı desteklendi;
- + SVG dışa aktarımında tüm stiller için ön ek tanımlanabilmesi amacıyla SVG PrefixStyle özelliği eklendi;
- + Excel 2007 dışa aktarımında "Sistem Veri Biçimlendirmesini Kullan" seçeneği eklendi;
- + PDF dışa aktarımında büyük raporlar için PDFExport.UseFileStream özelliği eklendi;
- * DBF dışa aktarımında varsayılan kodlama UTF-8 olarak ayarlandı;
- Katmanlı HTML, PDF, Excel ve diğer formatlardaki çeşitli hata düzeltmeleri yapıldı;

[WebReport]

- * OnlineDesigner özellikleri, geriye dönük uyumluluk korunarak WebReport.Designer'a taşındı;
- RadioButton CheckedChanged olayının çalışmama hatası giderildi;
- Blazor'da diyalog bileşenlerinin ölçeklendirilmesinde ve Excel dışa aktarımında yazı tipi boyutu hataları düzeltildi;

[.NET Core]

- Rapor betiğinde public metodların hatalı aranması düzeltildi;
- Azure ortamında fontlist dosyası oluşturulması sorunu giderildi;

[CoreWin]

- Visual Studio için WinForms bileşenlerinin Toolbox'daki davranışları düzeltildi;
- CoreWin'de bağlantıya tıklandığında tarayıcının hatalı başlatılması sorunu giderildi;
- FastReport.CoreWin için, WinForms API'si kullanan betik içeren raporlarda hata düzeltildi;

[Demos]

- + Yeni demo uygulamasının yerelleştirmesinin, uygulama yeniden başlatılmadan değiştirilmesi imkanı eklendi;
- + FastReport.Core ile React tabanlı demo uygulaması sunuldu;
- Chart.frx içindeki bir grafiğin konumlandırılması hatası giderildi;

[Plugins]

- + LINTER entegrasyonu eklendi;
- + Cassandra bağlantısı eklendi;
- * RPTImportPlugin güncellendi;

[Extras]

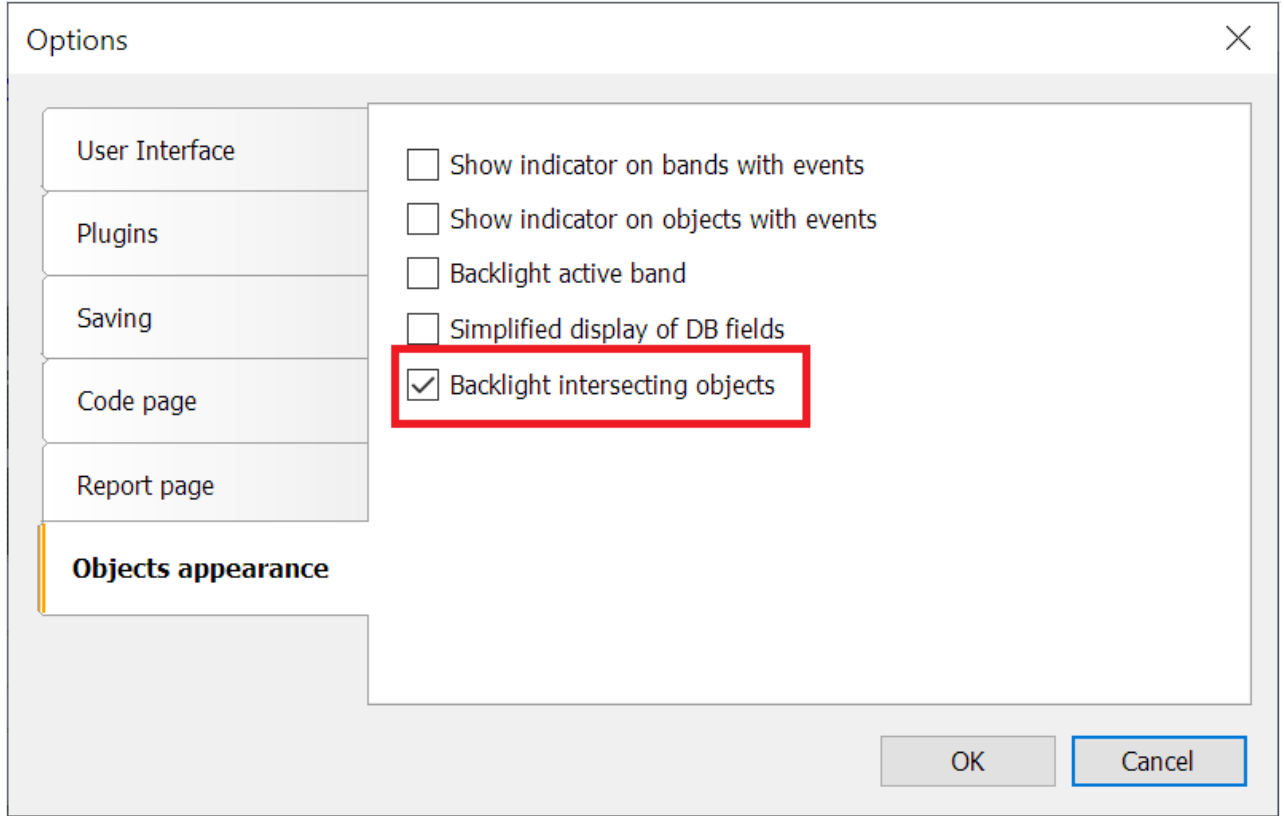
- FastReport.Net paketlerine, .NET Framework için FastReport.Web ve FastReport.VSDesign kütüphaneleri eklendi;
- Raporların akışlar kullanılarak içe aktarılması imkanı sunuldu.

[Service]

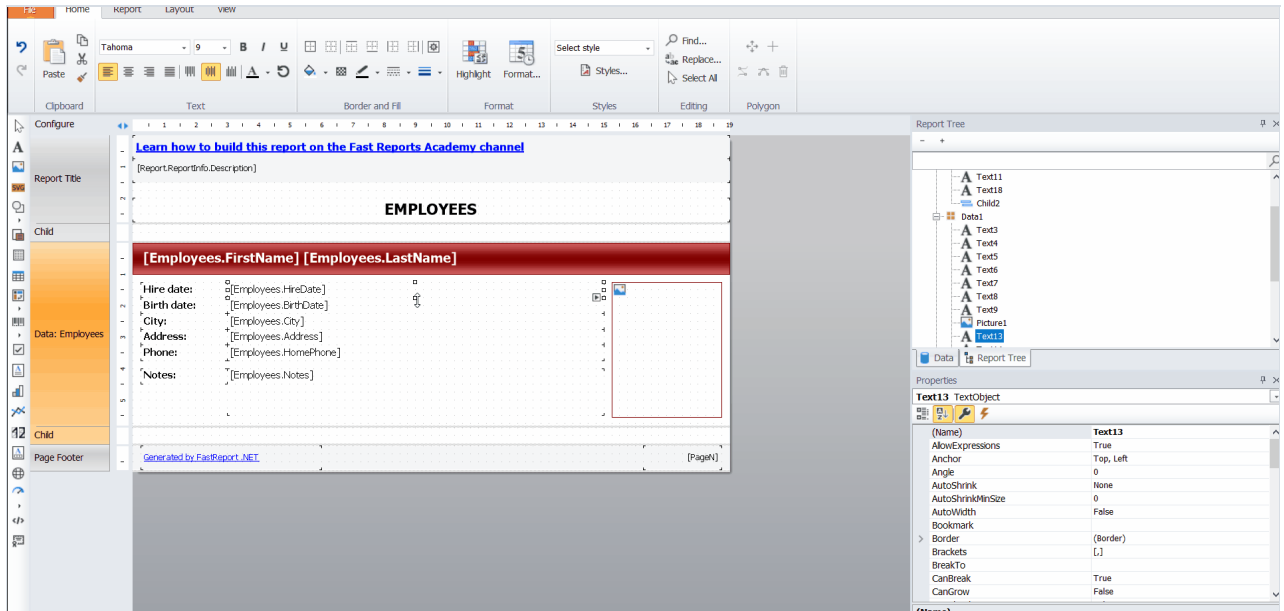
- FastReport.Net paketlerinde FastReport.Compat sürüm hatası giderildi.

Rapor sayfasına nesne yerleştirilirken, çıkan nesneler ile bant ve sayfa sınırlarını aşan nesneler renkli olarak vurgulanır. Doğru bir raporda bu tür nesneler bulunmamalıdır; uyulmaması raporların hazırlanması ve dışa aktarımı sırasında çeşitli sorunlara yol açabilir.

Varsayılan olarak bu seçenek kapalıdır ve Tasarımcı Ayarları (Dosya → Ayarlar) üzerinden aktif hale getirilebilir.



Aşağıda vurgulama işlevselliğinin bir demosu yer almaktadır.



RichObject Editöründe Kılavuzlu Cetvel

Yeni araç sayesinde RichObject düzenlenirken girintiler ve tab pozisyonlarını kolayca ayarlayabilirsiniz. Detaylar için makaleye göz atın.

RTF Dönüştürme Konsol Aracı

Bu araç sayesinde RTF dosyalarını rapor şablonlarına kolayca dönüştürebilirsiniz.

XLSX Dosyalarını Veri Kaynağı Olarak Kullanma İmkânı

Artık Excel 2007 dosyalarından veriler, bir veritabanı gibi alınarak raporlarda kullanılabilir.

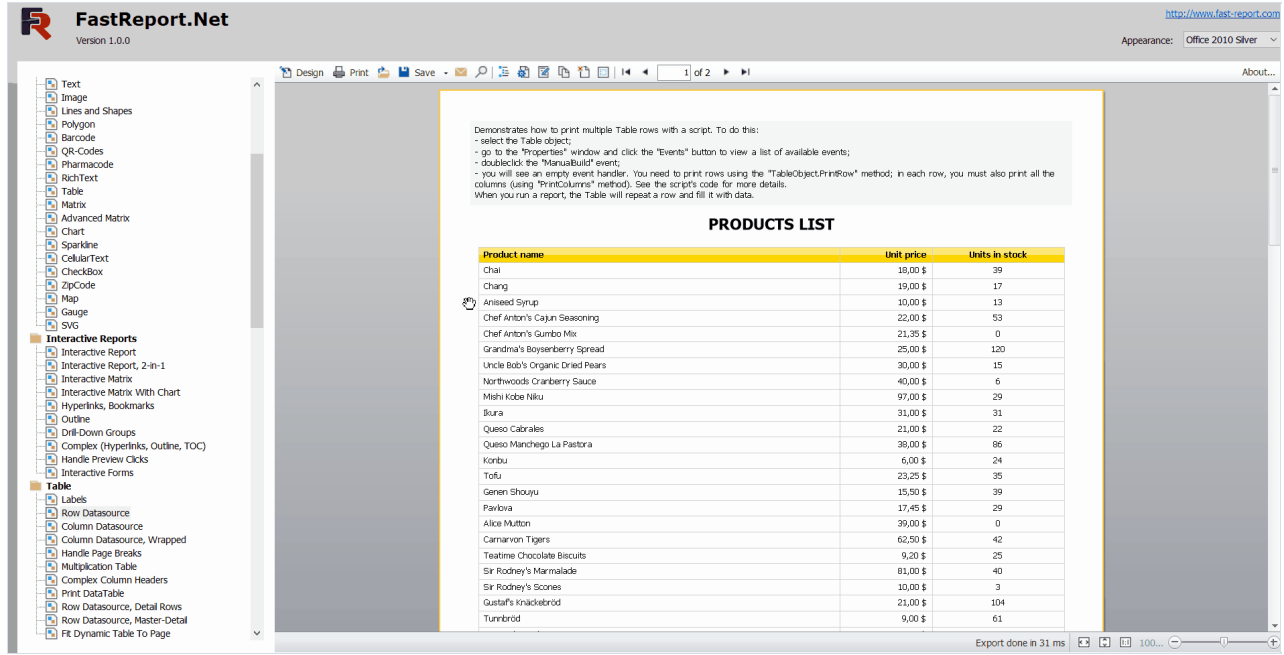
Detaylar için makaleye göz atın.

Dışa Aktarma İyileştirmeleri

Excel 2007 Dışa Aktarımında “Hücreleri Dondur” Seçeneği

Bu özellik, kaydırma sırasında her zaman görünür kalacak olan sayfa alanını belirlemenizi sağlar. Dondurulabilecek seçenekler arasında: ilk satır, ilk sütun ya da belirli sayıda satır ve sütun bulunur.

Ayarlar şu şekilde görünür:



Ve bu seçenek şu şekilde çalışır:

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
Mishi Kobe Niku	97,00 \$	29
Iura	31,00 \$	31
Queso Cabrales	21,00 \$	22
Queso Manchego La Pastora	38,00 \$	86
Korbu	6,00 \$	24
Tofu	23,25 \$	35
Genen Shouyu	15,50 \$	39
Pavlova	17,45 \$	29
Alice Mutton	39,00 \$	0
Carnarvon Tigers	62,50 \$	42
Teatime Chocolate Biscuits	9,20 \$	25
Sir Rodney's Marmalade	81,00 \$	40
Sir Rodney's Scones	10,00 \$	3
Gustaf's Knäckebröd	21,00 \$	104
Turnbrod	9,00 \$	61
Guaraná Fantástica	4,50 \$	20
NuNuCa Nuß-Hougat-Creme	14,00 \$	76
Gumbär Gummibärchen	31,23 \$	15
Schoggi Schokolade	43,90 \$	49
Rössle Sauerkraut	45,60 \$	26

ZPL'e Dışa Aktarımda ZPL II Desteği

HTML Dışa Aktarımında “Yüksek Kaliteli SVG” Seçeneği

Bu seçenek etkinleştirildiğinde, SVG formatındaki görüntülerin kalitesi dışa aktarım sırasında önemli ölçüde artar. Ancak, bunun bellek kullanımı miktarını da artıracakını göz önünde bulundurmalısınız.

Word'e Dışa Aktarımda Yer İmleri ve Dahili Bağlantılar

PDF, Word, HTML ve RTF Dışa Aktarımında Tab Karakter Genişliği

SVG Dışa Aktarımında Yeni PrefixStyle Özelliği

Bu özellik, SVG dışa aktarımında tüm stiller için bir örnek belirlemenizi sağlar.

Excel 97 Dışa Aktarımında Sayı ve Tarih Formatı

Tam Değişiklik Listesi

[Engine]

- + Raporun rastgele verilerle kaydedilmesi imkanı eklendi;
- * ExportBand metodunda artık Base yerine BandBase argümanı kullanılıyor;
- ContainerObject nesnesinde AfterData, BeforePrint ve AfterPrint olaylarının çift çağrılmasıyla ilgili hatalar giderildi;
- Diyalog formları içeren raporlar başlatılırken System.NullReferenceException**'a yol açan hata düzeltilildi;
- Alt raporlar ve sayfalarda VisibleExpression özelliğinin çalışmaması hatası giderildi;
- RTF dönüşümü sırasında kesişmeyen nesnelerin dikey kaymasıyla ilgili hata düzeltilildi;
- Sınırsız genişlik ve yatay düzenli sayfalarda sağ çapa ile ilgili hata düzeltilildi;
- RTF dönüşümünde listelerin hatalı işlenmesi düzeltilildi;
- RichObject.AllowExpressions özelliğinin çalışmamasına neden olan hata düzeltilildi;
- Hazırlanmamış sınırsız sayfanın çiziminde System.OverflowException'a yol açan hata giderildi;

[Designer]

- + Çakışan nesnelerin vurgulanması eklendi;
- + RichObject editöründe kılavuzlu cetvel eklendi;
- + Nesne denetleyicisinde şifre karakterleri nokta ile değiştirildi;
- + Tekrarlayan bantlarda Matrix veya AdvMatrix yerleştirilirken olası yığın taşması konusunda uyarı eklendi;
- Barkod metni bir ifade içeriyorsa hata mesajı kaldırıldı;
- Kısa yol tuşlarını devre dışı bırakma seçeneğiyle ilgili hata düzeltilildi;
- LineStyle ve LineWidth düğmelerine tıklandığında açılan menü hatası düzeltilildi;
- Tasarımcıdaki veri görüntülemesi düzeltilildi;
- AdvMatrix'e nesne sürüklerken System.NullReferenceException hatası giderildi;
- Sınır editöründe gölge ayarlarının hatalı görüntülenmesi hatası düzeltilildi;

[Preview]

- Düzenlenebilir metin nesnesine tıklandığında **System.NullReferenceException**'a neden olan hata düzeltilildi;
- Çok sütunlu bant içeren raporlarda çalışmayan hiperlink hatası giderildi;
- Rapor dışa aktarılırken hazırlanan raporun kaydedilmesine neden olan hata düzeltilildi;
- PreviewControl'de mevcut dışa aktarma listesi ve bulut dışa aktarma seçenekleri ayarıyla ilgili hata giderildi;

[Exports]

- + ZPL'e dışa aktarımda ZPL II eklendi;
- + HTML dışa aktarımında "Yüksek Kaliteli SVG" seçeneği eklendi;
- + Excel 2007'e dışa aktarımda "Hücreleri Dondur" seçeneği eklendi;
- + Excel 2007 dışa aktarımında baskı ölçeklendirme imkanı eklendi;
- + Word'e dışa aktarımda yer imleri ve dahili bağlantılar eklendi;
- + Excel 97 dışa aktarımında sayı ve tarih formatının dışa aktarımı eklendi;
- + E-posta dışa aktarımında kullanıcının kişisel verilerinin şifrelenmesi eklendi;
- + RTF dışa aktarımında RichObject girintisinin aktarımı eklendi;
- + RTF dışa aktarımında RichObject satır sonu aktarımı eklendi;
- + Word'e dışa aktarımda TextObject girintisinin aktarımı eklendi;
- + PDF, Word, HTML ve RTF dışa aktarımında tab karakter genişliğinin aktarımı eklendi;
- + SVG dışa aktarımında tüm stiller için örnek belirleme imkanı sağlayan PrefixStyle özelliği eklendi;
- * Excel 2007'e RichObject dışa aktarımı geliştirildi;
- * FastReport Cloud ve XMPP dışa aktarımları kaldırıldı;
- Önceki sayfalardan alınan stillerin kullanılması durumunda HTML baskısında yatay düzenli sayfaların yanlış döndürülmesi düzeltilildi;
- PDF dışa aktarımında yazı tipi ölçeği hatası giderildi;
- HTML'de "Yüksek Kaliteli SVG" seçeneği ile SVG nesnelerinin dışa aktarımında oluşan bellek sızıntısı

HTML dışı aktarımında SVG seçeneği ile SVG nesnelerinin dışı aktarımında oluşan sorun giderildi;
düzeltildi;
- Paketlemeye izin verilmeyen yazı tiplerinin PDF dışı aktarımında gömülmesi sırasında oluşan hata giderildi;
- Word dışı aktarımında tab karakterlerinin aktarımı hatası düzeltildi;
- HTML dışı aktarımında resim arka planı dolgusu ve line-height özelliği düzeltildi;
- PDF dışı aktarımında SVGObject için özel kesikli çizgi hatası giderildi;
- SVG dışı aktarımında birleşik hücrelerin sınırlarının aktarımı hatası düzeltildi;

[WebReport]
+ WebReport'ta Geliştirilmiş Matris etkileşimi eklendi;
- OnFormClosing olayında web diyalogu kapatma iptalinin işlenmesi düzeltildi;

[.NET Core]
- .NET Core'da "Dışa Aktarmadan Sonra Aç" seçeneğinin çalışmaması hatası giderildi;

[WebReport Core]
- Artık DatePicker simgesi tüm tarayıcılarda aynı görünüyor;

[Demos]
+ Blazor için, iki raporla çalışmayı gösteren yeni bir demo eklendi;
- Yeni demoda, bağlantılar üzerine gelindiğinde imlecin değişmemesine neden olan hata giderildi;
- Yeni demoda AdvMatrix ile ilgili hata düzeltildi;

[Plugins]
+ Excel'e bağlantı eklendi;
- FastReport.Core, FastReport.CoreWin ve FastReport.OpenSource için SQLite bağlantısı düzeltildi;
- Firebird bağlantı dizesi düzeltildi;

[Extras]
+ RTF belgelerini rapor şablonlarına dönüştürmek için bir araç eklendi (\Extras\Misc\rtf2frx).

Sürüm 2022.1

Yeni Özellikler

Yeni "Gelişmiş Matris" Nesnesi Eklendi:

	Anne Dodsworth	Michael Suyama	Steven Buchanan	Laura Callahan	Janet Leverling	Nancy Davolio	Robert King	Margaret Peacock	Andrew Fuller	Total
▼ Beverages (12)	\$19 642,56	\$11 538,20	\$30 998,53	\$23 297,85	\$45 297,41	\$46 725,36	\$66 404,83	\$52 324,21	\$40 463,25	\$336 692,18
1. Chai	3%	13%	3%	5%	4%	2%	1%	7%	4%	\$12 788,10
2. Chang	3%	16%	3%	5%	3%	7%	2%	9%	3%	\$16 374,96
3. Chartreuse verte	1%	39%	69%	29%	5%	4%	0%	7%	4%	\$42 732,54
4. Côte de Blaye	74%		20%		56%	50%	30%	56%	62%	\$144 728,74
5. Guarani Fantástica	1%	5%	1%	4%	1%	2%	1%	0%	1%	\$4 504,37
6. Ipoh Coffee				12%	14%	14%	3%	4%	9%	\$23 526,70
7. Lakkaikööri	5%	7%	2%	15%	7%	8%	1%	3%	3%	\$15 760,44
8. Laughing Lumberjack Lager				1%	0%		0%	1%	2%	\$2 396,80
9. Outback Lager	4%	5%		6%	3%	3%	2%	6%	3%	\$10 672,65
10. Rhönbräu Klosterbier	4%	8%	1%	6%	3%	3%	53%	2%	2%	\$43 212,49
11. Sasquatch Ale	1%	4%	0%	1%	2%		5%	1%	2%	\$6 350,40
12. Steeleye Stout	4%	4%	0%	17%	1%	6%	2%	3%	5%	\$13 644,00
► Condiments (12)	\$10 125,55	\$6 208,47	\$4 115,30	\$46 437,66	\$14 581,64	\$17 027,56	\$9 731,38	\$24 634,87	\$96 210,67	\$229 073,09
► Confections (13)	\$8 053,16	\$6 940,63	\$4 809,80	\$21 699,91	\$33 622,40	\$28 568,92	\$14 518,99	\$27 768,73	\$21 455,69	\$167 438,23
► Dairy Products (10)	\$21 101,13	\$17 039,04	\$21 769,63	\$56 269,47	\$32 320,84	\$36 022,98	\$27 621,86	\$33 549,80	\$23 812,55	\$269 507,29
► Grains/Cereals (7)	\$1 245,30	\$9 410,70	\$4 027,56	\$11 072,05	\$21 235,01	\$8 465,91	\$6 535,50	\$22 579,61	\$11 172,95	\$95 744,59
► Meat/Poultry (6)	\$8 676,66	\$9 003,69	\$11 488,20	\$16 395,28	\$20 502,62	\$15 038,47	\$101 176,72	\$70 867,14	\$129 873,60	\$383 022,36
► Produce (5)	\$314,81	\$11 560,71	\$18 734,02	\$12 016,52	\$11 960,85	\$32 206,25	\$10 753,38	\$17 186,56	\$24 376,48	\$139 109,58
► Seafood (12)	\$8 148,91	\$5 940,71	\$5 744,25	\$12 041,54	\$25 032,10	\$24 206,16	\$7 146,59	\$27 315,93	\$15 747,57	\$131 323,74
Total	\$77 308,07	\$77 642,13	\$101 687,28	\$199 230,28	\$204 552,84	\$208 261,60	\$243 889,24	\$276 226,85	\$363 112,76	\$1 751 911,04

İşte ana özelliklerinin listesi:

- Satır ve sütun başlıkları, rastgele sırada gruplar ve basit öğeler içerebilir. Bu, asimetrik raporlar oluşturmayı mümkün kılar;
- Katlama (collapse) düğmeleri, belirli öğelerin görünürliğini etkileşimli olarak yönetmeyi sağlar;
- Sıralama düğmeleri, matrisin seçilen değerlere – hatta özet (toplam) değerlere – göre etkileşimli olarak sıralanmasını sağlar;
- Top N gruplama, başlıkta N değeri görüntülemeye ve geri kalan değerleri açılabilir ayrı bir öge altında gruplayabilmeye olanak tanır;
- Matris başlıklarının basamaklı (kademe) biçimde görüntülenmesi;
- Başlıkların özet (toplam) değerlere göre sıralanması;
- Geniş bir yelpazede toplama (aggregate) fonksiyonları;
- Kullanıcı tanımlı toplama (aggregate) fonksiyonlarının desteği;
- Özet değerler, komşu hücre değerleri elde etme ve yüzdeleri hesaplama gibi işlevleri sunan geniş özel fonksiyon yelpazesi;
- Veri hücrelerinde "İskrografik" ve "İlerleme Göstergesi" nesnelerinin desteği.

TopN group with ability to expand/collapse its elements

Expand/collapse matrix elements

Block or stepped layout

Sort elements by the total value

Top 5 customers			
	Total	Others (84)	Total
▶ 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

Matrix header can contain different types of elements

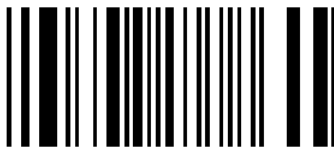
	2011	2012	2013	2014	2015	2012 to 2011 change	2012 to 2011 %
Andrew Fuller	3900	2100			1800	-1800	53,85%
Janet Leverling	6100	3200				-2900	52,46%
Nancy Davolio	3300	2700	3100		1700	-600	81,82%
Steven Buchanan			3999	8100			
Total	13300	8000	7099	8100	3500	-5300	60,15%

Special functions to get other cell's value

Nesnenin özellikleri hakkında daha fazla bilgi için dokümantasyona bakınız.

GS1 DataBar Barkodları Eklendi:

Limited, Omnidirectional, Stacked ve Stacked Omnidirectional barkodlar eklenmiştir.

GS1 DataBar Omnidirectional  (01)12345678901231	GS1 DataBar Limited  (01)12345678901231
GS1 DataBar Stacked Omnidirectional  (01)12345678901231	GS1 DataBar Stacked  (01)12345678901231

Yeni Özellikler: Config.CompilerSetting.ExceptionBehaviour ve

Config.CompilerSetting.Placeholder

Bu özellikler, veritabanı alanları ve tablolarında hatalı isimler nedeniyle oluşan istisnalar sırasında davranışı ayarlama imkanı sağlar.

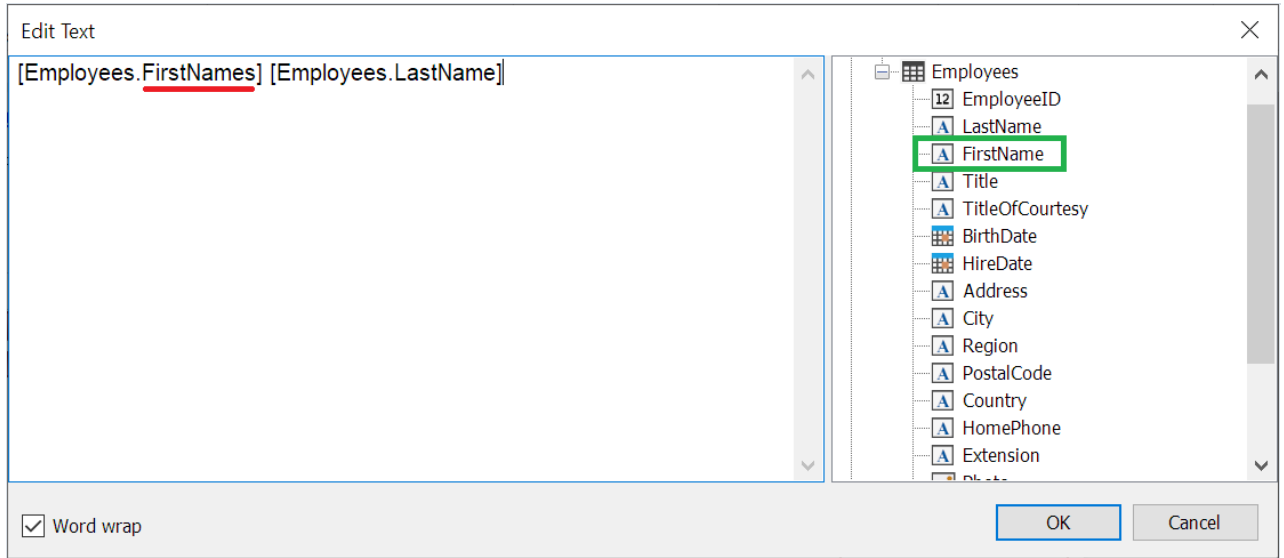
Config.CompilerSetting.Placeholder – Var olmayan isimlere sahip ifadelerin yerine kullanılacak olan dize değişkenidir. Varsayılan değeri boş dizgedir. **Config.CompilerSetting.ExceptionBehaviour** aşağıdaki değerleri alabilir: **ExceptionBehaviour.Default** – Önceden uygulandığı gibi standart davranış. Hatalı isim hatalarında hata mesajı gösterilir ve rapor hazırlanması durdurulur.

ExceptionBehaviour.ReplaceExpressionWithExceptionMessage – Hatalı ifadeler, hata mesajı metniyle değiştirilir. Hata mesajı gösterilmez; rapor hazırlanması devam eder.

ExceptionBehaviour.ShowExceptionMessage – Hata mesajı görüntülenir; OK düğmesine basıldığında rapor hazırlanması devam eder. Bu durumda, hatalı ifadeler Placeholder değişkeninin değeriyle değiştirilir.

ExceptionBehaviour.ReplaceExpressionWithPlaceholder – Hatalı ifadeler doğrudan Placeholder ile değiştirilir. Hata mesajı gösterilmez ve rapor hazırlanması kesintiye uğramaz.

Örnek: **ExceptionBehaviour = ExceptionBehaviour.ReplaceExpressionWithPlaceholder Placeholder = "NO DATA!"**



Burada tabloda **FistName** adlı bir alan mevcut olmasına rağmen, ifade hatalı şekilde belirtilmiştir.

EMPLOYEES

NO DATA! Fuller

Hire date: 14.08.2009
Birth date: 19 февраля 1972 г.
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.



NO DATA! Dodsworth

Hire date: 15.11.2011
Birth date: 27 января 1986 г.
City: London
Address: 7 Houndstooth Rd.
Phone: (71) 555-4444

Notes: Anne has a BA degree in English from St. Lawrence College. She is fluent in French and German.



Böylece, daha önce hata nedeniyle hazırlamak mümkün olmayan rapor başarıyla oluşturulabilmektedir.

RTF'den Rapor Nesnelerine Çevirme Kalitesi Artırıldı

RTF'nin rapor nesnelerine dönüştürülmesi optimize edilmiştir. Tablolardaki hücrelerde RTF çevirisi eklendi. Ayrıca, birçok hata giderilmiştir.

Dışa Aktarım İyileştirmeleri

Word ve RTF'de Filigran Dışa Aktarımı.

Filigranlar (watermark) Word ve RTF formatlarına başarıyla dışa aktarılmaktadır.

SVG Görsellerin Ölçeklendirilmesi.

Dışa aktarım matrisinde SVG görsellerin ölçeklendirilmesi eklenmiştir. Bu, Word ve Excel'e dışa aktarım sırasında görsellerin kalitesini artırır; ancak çıktı dosyasının boyutu artar. Bu özelliği kullanmak için dışa aktarımda "Baskı İçin" seçeneği etkinleştirilmelidir.

Excel 2007 Grupların ayrı sayfalara dışa aktarılması gerçekleştirilmiştir.

Excel 2007'de, dışa aktarım sırasında görüntünün boyutunu ve konumunu belirleyen özellik dışa aktarılabilir hale getirilmiştir.

Artık, hücrenin konumu ve boyutu değiştirildiğinde görüntünün:

- Hücreyle birlikte taşınarak boyutu değiştirilebilir;
- Hücreyle birlikte taşınabilir ancak boyutu değişmez;
- Ne taşınır ne de boyutu değişir şeklinde davranışı tanımlanabilir.

Excel 97: İzgara (grid) çizgilerinin dışa aktarımda gizlenmesi veya gösterilmesi imkanı eklenmiştir.

HTML Dışa Aktarımı: "Baskı sırasında yatay sayfaları döndürmeme" seçeneği eklenmiştir. Önceden, tarayıcıların hem dikey hem de yatay oryantasyona sahip raporları doğru şekilde yazdıramaması nedeniyle yatay sayfalar zorunlu olarak döndürülüyordu. Artık bu davranış ayarlanabilir; ayrıca, dikey sayfa olmadığı durumlarda bile yatay sayfaların döndürülmesi hatası giderilmiştir.

.NET 6 Desteği

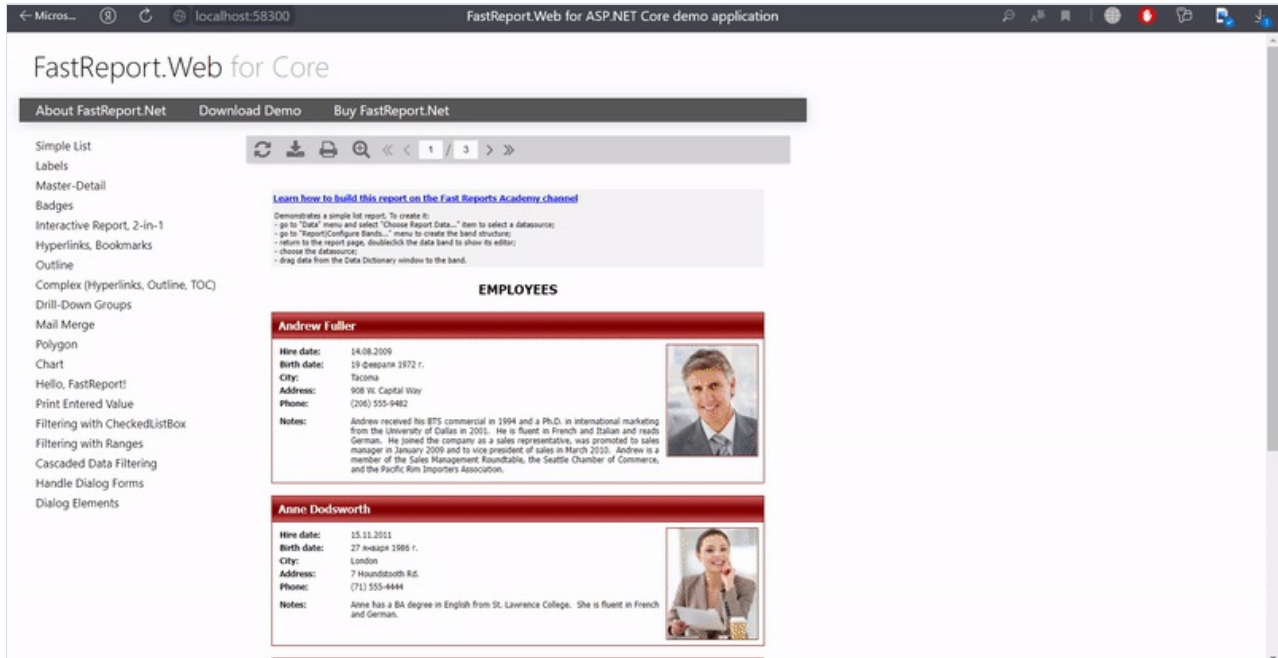
FastReport.Core ve FastReport.CoreWin için .NET 6 desteği eklenmiştir.

WebReport for Core ve Blazor Server İyileştirmeleri

Dışa Aktarım Ayarları:

WebReport üzerinden raporu istenen formata dışa aktarmayı ayarlama imkanı eklenmiştir.

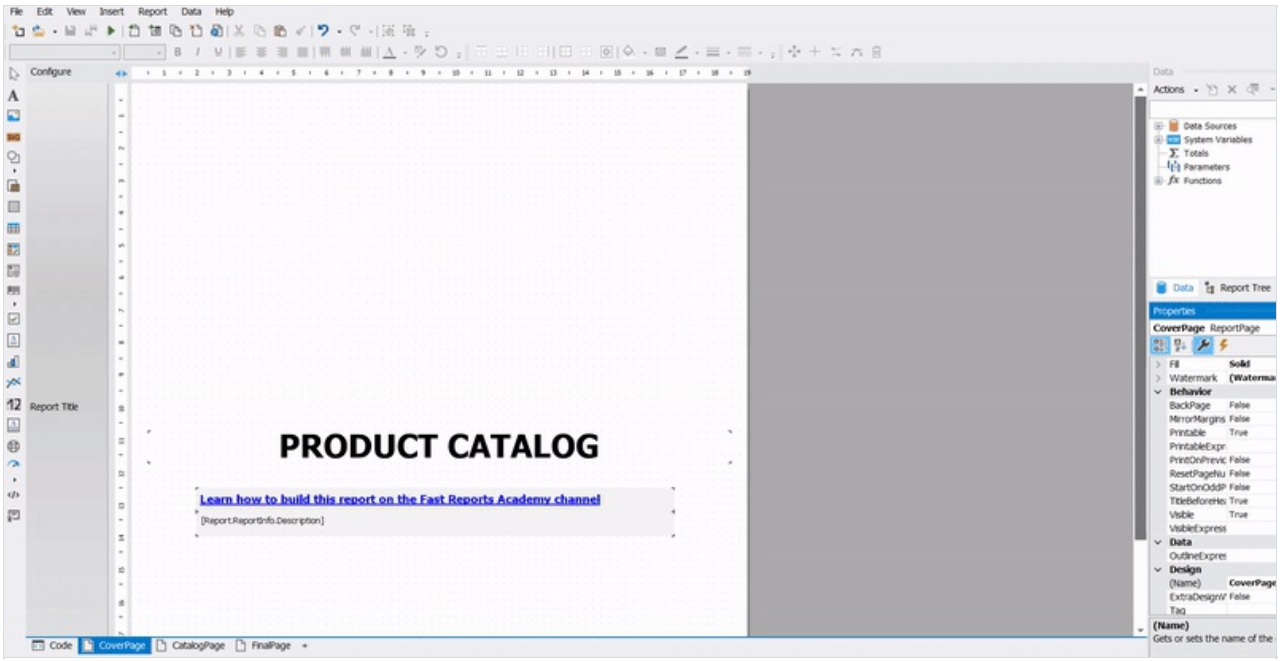
webReport.Toolbar.Exports.EnableSettings özelliği etkinleştirildiğinde, ilgili dışa aktarım düğmesinin yanında ayar simgesi belirecek; bu simgeye tıklandığında dışa aktarım yapılacak sayfalar, özellikler vb. seçeneklerin bulunduğu ayarlar penceresi açılacaktır.



Bu ayarlar penceresi, *webReport.Toolbar.Exports.Color* ve *webReport.Toolbar.Exports.FontSettings* özellikleriyle özelleştirilebilir.

Rapor Sayfalarının Sekmelerde Ayrılması

Raporun farklı sayfalarını (ReportPage) WebReport içerisinde ayrı sekmelerde açma imkanı eklenmiştir. Bu özelliği etkinleştirmek için *webReport.SplitReportPagesInTabs* seçeneği kullanılmalıdır.



Statik Stil İsimleri:

Araç çubuğu, ana hat (outline) ve diğer öğelerin varsayılan stillerini geçersiz kılmak amacıyla, statik sınıf isimleri eklenmiştir. Bunlar şunlardır: *fr-toolbar*, *fr-toolbar-item*, *fr-toolbar-narrow*, *fr-toolbar-dropdown-content*, *fr-toolbar-zoom-selected*, *fr-toolbar-pointer*, *fr-toolbar-notbutton*, *fr-toolbar-slash*.

Değişikliklerin Tam Listesi

```
[Engine]
+ Yeni AdvMatrixObject nesnesi eklendi;
+ GS1 Databar barkodları eklendi: Limited, Omnidirectional, Stacked ve Stacked Omnidirectional;
+ Yeni özellikler eklendi: Config.CompilerSetting.ExceptionBehaviour ve Config.CompilerSetting.Placeholder. Bu özellikler, veritabanı alanları ve tablolarındaki hatalı isimler nedeniyle oluşan istisnalar durumunda davranışı ayarlama imkanı sağlar;
+ RichObject'in TableCell içinde çevrimi eklendi;
* RichObject'in rapor nesnelere dönüştürülmesi yeniden düzenlendi;
- RTF'den çevrilen metin nesneleri için ShiftMode hatası düzeltildi;
- Aynı isimde iki parametre nedeniyle oluşan System.ArgumentException hatası giderildi;
- Çok sütunlu Databand içeren gömülü rapor hatası düzeltildi;
- Bant yüksekliğinin yanlış hesaplanması hatası giderildi;
- RTF'nin rapor nesnelere dönüştürülmesi sırasında hipermetin (link) gösterim hatası düzeltildi;
- Doküman, script üzerinden atanırsa RichObject çevirimi düzeltildi;
- Config.PrivateFontCollection'a eklenen yazı tipleriyle ilgili hata giderildi;

[.Net Core]
+ .NET 6 desteği eklendi;
- Bold-Italic yazı tiplerinin hatalı aranması düzeltildi;

[Designer]
+ SberBank QR kodu düzenleme penceresinde girilen verilerin kontrolü eklendi;
- Metin nesnesi düzenleyicisinde satır sonu hatası düzeltildi;
- Tablo hücreleri içinde matrisler içeren rdl raporlarının dönüştürülmesindeki hata giderildi;
- Designer içerisindeki rehber çizgiler ile ilgili hata düzeltildi;
- "Rapor Ağacı" penceresi hatası giderildi;
- Otomatik rehber (Auto-guidelines) seçeneği açıkken, designer'ın başlatılması sırasında oluşan System.NullReferenceException ve çökme hatası giderildi;

[Preview]
- Hazırlanan sayfa düzenlemesi sırasında, bant görünümü değiştirilirken nesnelerin pozisyon kayması hatası düzeltildi;

[Exports]
+ Word'de filigran dışa aktarımı gerçekleştirildi;
```

- + RTF'de filigran dışa aktarımı gerçekleştirildi;
- + HTML dışa aktarımında "Baskı sırasında yatay sayfaları döndürmeme" seçeneği eklendi;
- + E-posta ile gönderim sırasında eklenen dosyanın adını değiştirme imkanı sağlandı;
- + Dışa aktarım matrisinde SVG görsellerin ölçeklendirilmesi eklendi;
- + Excel 2007 dışa aktarımında, görüntünün boyutunu ve konumunu belirleyen özellik dışa aktarılabilir hale getirildi;
- + Excel 97 dışa aktarımında ızgara çizgilerinin gizlenmesi veya gösterilmesi imkanı sağlandı;
- + Excel'de grupların ayrı sayfalara dışa aktarımı gerçekleştirildi;
- + Word'de filigran görüntülerinin şeffaflık seviyesinin dışa aktarımı gerçekleştirildi;
- + RTF'de filigran görüntü boyutunun dışa aktarımı gerçekleştirildi;
- Satır sayısı birden az olan tabloların metin dışa aktarımında System.NullReferenceException hatası giderildi;
- Word dışa aktarımında tabloların hatalı sol boşluğu düzeltildi;
- HTML dışa aktarımında Wingdings yazı tipi kullanılırken oluşan hata giderildi;
- HTML dışa aktarımında Wingdings ve Webdings yazı tiplerinin dışa aktarımı ile ilgili hata düzeltildi;
- PowerPoint dışa aktarımında çerçeve genişliği hatası giderildi;
- RTF'de şeffaf dolguya sahip nesnelerin dışa aktarımında hata giderildi;
- Word'de şeffaf dolguya sahip nesnelerin dışa aktarımında hata giderildi;
- PDF dışa aktarımında System.OutOfMemoryException'a yol açan hata giderildi;
- HTML dışa aktarımında satır sonu (line break) hatalı görüntülenmesi düzeltildi;
- PDF dışa aktarımında bellek taşması giderildi;
- Windows dışındaki sistemlerde PDF dışa aktarımındaki hatalar düzeltildi;
- Word 2007'de sütun sayısı 63'ten fazla olan tabloların dışa aktarımında hata giderildi;
- "Metin Eğri" (Text on Curves) seçeneği açıkken, PDF dışa aktarımında bellek sızıntısı ve System.OutOfMemoryException hatası giderildi;
- HTML dışa aktarımında satır bölünmesi (line break) hatası düzeltildi;

[WebReport]

- Wingdings yazı tipi kullanılırken yeni satır karakteri hatası giderildi;

[WebReport Core/Blazor Server]

- + WebReport'ten rapor dışa aktarım özelliklerini ayarlama imkanı eklendi.
- webReport.Toolbar.Exports.EnableSettings etkinleştirildiğinde, ilgili dışa aktarım düğmesinin yanında ayar simgesi görünecektir;
- + Raporun farklı sayfalarını (ReportPage) WebReport'te farklı sekmelerde açmaya olanak tanıyan webReport.SplitReportPagesInTabs özelliği eklendi;
- + Araç çubuğu, ana hat ve diğer öğelerin varsayılan stillerini geçersiz kılmak amacıyla statik sınıf isimleri eklendi;
- DateTimePicker alanına klavyeden değer girildiğinde WebReport güncelleme hatası giderildi;
- Standart dışı rapor sayfa boyutlarında sekme genişliği hatası düzeltildi;

[Demos]

- + .NET 5 için WebReport Core kullanım demosu eklendi;
- + Angular için WebReport Core kullanım demosu eklendi;
- + Blazor Server için WebReport Blazor kullanım demosu eklendi.

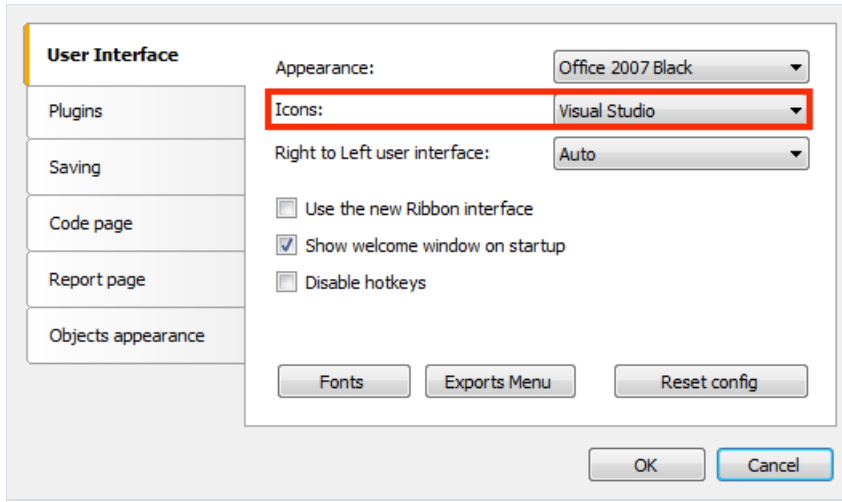
Sürüm 2021.4

Sürüm değişiklikleri

WinForms ve Win+WebForms sürümleri kaldırıldı. Bunun yerine, Windows Forms bileşenlerini, ASP.NET bileşenlerini ve .NET Core / .NET 5 / Blazor desteğini içeren yeni bir Standard sürüm eklendi.

Yeni Özellikler

- Visual Studio tarzında yeni simgeler eklendi. "Görünüm/Ayarlar/Arayüz" penceresinden (veya ribbon arayüzü kullanıyorsanız "Dosya/Ayarlar/Arayüz") simge setleri arasında geçiş yapabilirsiniz.



Yeni simgeler, yüksek DPI (hiDPI) ekranlar için uygundur.

- FastReport ortam ayarlarının, FastReport.config yapılandırma dosyasında saklanan, varsayılan hale sıfırlanması artık mümkündür. Bu işlem, tasarımcı arayüzündeki "Görünüm > Ayarlar > Arayüz" penceresindeki "Ayarları Sıfırla" düğmesi ile yapılabilir. Değişikliklerin etkili olması için tasarımcının yeniden başlatılması gerekir.
- Tasarım modunda "Metin" nesnesinde veritabanı alanlarının sadeleştirilmiş şekilde görüntülenmesi özelliği eklendi. Bu ayar, tasarımcı arayüzündeki "Görünüm > Ayarlar > Nesne Görünümü" penceresinden yapılabilir. Varsayılan olarak devre dışıdır; etkinleştirildiğinde, yalnızca bir veritabanı alanı içeren "Metin" nesneleri sade bir biçimde gösterilir:

CUSTOMERS ORDERS				
[Orders.Customers.CompanyName]				
OrderID	[Orders.OrderID]	OrderDate	[Orders.OrderDate]	ShippedDate [Orders.ShippedDate]
Product name			Unit Price	Quantity
[Order Details.Products.ProductName]			[Order Details.UnitPrice]	[Order Details.Quantity]
Total this order: [SumOfOrder]				
Generated by FastReport .NET				[PageN]

Bu, çok sayıda küçük alan içeren raporların görsel olarak daha kolay algılanmasını sağlar. Alanın tam içeriği ise hâlâ durum çubuğunda ve nesne metni düzenlenirken görüntülenir.

- "Matris" nesnesinin köşe hücreleri için ayrı ayrı özelleştirme imkanı eklendi. Bunun için hücrenin bağlam menüsünde yer alan "Hücreyi Böl" ve "Hücreleri Birleştir" komutlarını kullanabilirsiniz.

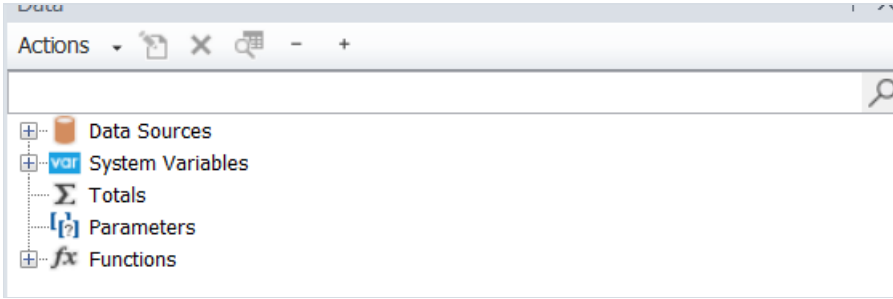
Name	[Year]		Total
[Name]	[Month]	Total	
Total	[Revenue]		

Generated by FastReport .NET

- ElasticSearch'e bağlanma özelliği eklendi. Bağlantı, veri sihirbazı üzerinden veya kod ile yapılabilir.
- Japanese Post 4-State Code barkodu eklendi.



- Tasarımcıda rapor ağacı ve veri ağacı için "Tümünü Genişlet / Tümünü Daralt" düğmeleri ve arama alanı eklendi. "+" simgesine tıklanınca ağaç genişler, "-" simgesine tıklanınca ağaç daralır.



Bu değişiklikler, çok sayıda nesne ve/veya veri kaynağı içeren raporlarla çalışmayı kolaylaştırmalıdır.

- RichText'ten rapor nesnelerine dönüştürücü önemli ölçüde geliştirildi ve optimize edildi.
- WebReport Core / Blazor Server sürümünde mevcut dış aktarma formatlarının sayısı önemli ölçüde artırıldı.
- FastReport Business Graphics entegrasyon nesneleri eklendi (`\Extras\Objects\FastReportBGObjects`).

Universal plugins FastReport.Data

FastReport.Data eklentilerini içeren paketler güncellendi. Artık bu paketler, farklı FastReport sürümleri (.NET, Core, CoreWin, OpenSource) için eklentiler içeriyor ve kullandığınız ürüne bağlı olarak gerekli kütüphaneleri otomatik olarak bağlıyor. Doğru çalışması için FastReport 2021.3.0 veya daha yeni bir sürüm gereklidir.

FastReport.Core.Data, FastReport.CoreWin.Data ve FastReport.OpenSource.Data eklentileri artık kullanımdan kaldırılmıştır ve desteklenmemektedir.

FastReport kullanan kullanıcı uygulamalarının yayımlanmasında iyileştirmeler yapıldı.

.NET Core 3.0+ ve .NET 5+ üzerinde çalışan FastReport.Core, FastReport.CoreWin, FastReport.OpenSource tabanlı kullanıcı uygulamaları için Tek Dosyalı Uygulamalar (Single File Applications - SFA) desteği eklendi.

Ayrıca, kullanılmayan kütüphanelerin kaldırılmasıyla uygulamanın yayımlanmasını sağlayan PublishTrimmed

MSBuild özelliği de desteklenmektedir.

Uyarı! Bazı durumlarda .NET'in budamaması (trim) gereken derlemeleri açıkça belirtmeniz gerekebilir. Bu, rapor betiğinin bu kütüphaneleri kullanması ancak uygulama kodunuzun doğrudan kullanmaması durumunda önemlidir.

Bu işlem TrimmerRootAssembly özelliği kullanılarak yapılır. Örneğin, aşağıdaki şekilde `System.Security` kütüphanesinin budanmaması gerektiği belirtilmiştir:

```
<ItemGroup>
  <TrimmerRootAssembly Include="System.Security" />
</ItemGroup>
```

Yerelleştirmeler

Yerelleştirme mantığında küçük değişiklikler yapıldı.

1. FastReport.Localization paketi eklendi. Bu paket, FastReport.NET, FastReport.Core, FastReport.CoreWin, FastReport.Mono ve FastReport.OpenSource ürünleri için yerelleştirme dosyalarını içerir ve projeye eklendiğinde, kullanıcı projesinin çıkış dizininde Localization adlı bir klasör oluşturur.
2. Ayrıca, yerelleştirmeyi CultureInfo türü ile değiştirmek için yeni bir API eklendi:
FastReport.Utils.Res.LoadLocale(CultureInfo culture)

Bu yöntem çağrıldığında, FastReport belirtilen kültüre uygun yerelleştirme dosyasını arar. Yüklenen diller önbellege alınır. Bu yöntemin doğru çalışabilmesi için, projeye 1. adımda belirtilen FastReport.Localization paketinin yüklenmiş olması veya yerelleştirme dosyalarının bulunduğu klasör yolunun FastReport.Utils.Res.LocaleFolder özelliği aracılığıyla belirtilmesi gerekir.

WebReport Core / Blazor araç çubuğunda değişiklikler ve iyileştirmeler yapıldı.

- Araç çubuğu ayarları, WebReport sınıfından alınarak ToolbarSettings sınıfının WebReport.Toolbar özelliğine taşındı.
- Aşağıdaki yeni araç çubuğu ayarları eklendi: Konum, açılır menü rengi, yazı tipi, simge şeffaflığı, simge rengini değiştirme, içerik hizalamasını değiştirme. Bu özellikler artık `webReport.Toolbar` üzerinden yapılandırılabilir.
- Rapor yüklenirken araç çubuğu artık görüntülenmez.
- Toolbar nesnesine yeni bir özellik olan ShowOnDialogPage (varsayılan değeri `true`) eklendi. Bu sayede bir iletişim (dialog) penceresi açık olduğunda araç çubuğunun çizilmesi devre dışı bırakılabilir.
- Ayrıca, araç çubuğundaki açılır menüye daha fazla dışa aktarma seçeneği eklendi. Bu özellikler `webReport.Toolbar.Exports.ExportTypes` üzerinden yönetilir. Eklenen dışa aktarma türleri: HTML, Hpgl, Dxf, Json, LaTeX, Ppml, PS, Xaml, Zpl, Excel97, Svg.

Örnek kullanım:

```

ToolbarSettings toolbar = new ToolbarSettings()
{
    Color = Color.LightBlue,
    DropDownMenuColor = Color.LightBlue,
    ShowOnDialogPage = false,
    DropDownMenuTextColor = Color.Black,
    IconColor = IconColors.Black,
    Position = Positions.Right,
    FontSettings = new Font("Arial", 14, FontStyle.Bold),
    Exports = new ExportMenuSettings()
    {
        ExportTypes = Exports.Pdf | Exports.Excel97 | Exports.Rtf
    }
    // or
    //Exports = ExportMenuSettings.All
};
webReport.Toolbar = toolbar;

```

[Learn how to build this report on the Fast Reports Academy channel](#)

Demonstrates a simple list report. To create it:

- go to "Data" menu and select "Choose Report Data..." item to select a datasource;
- go to "Report|Configure Bands..." menu to create the band structure;
- return to the report page, doubleclick the data band to show its editor;
- choose the datasource;
- drag data from the Data Dictionary window to the band.

Export to PDF

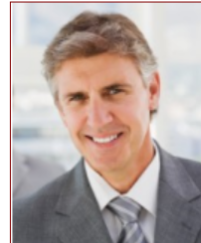
Export to Rich Text

Export to Excel 97

EMPLOYEES

Andrew Fuller

Hire date: 14.08.2009
Birth date: 19 февраля 1972 г.
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,



WebReport Core / Blazor Server iyileştirmeleri yapıldı.

FastReport.Core3.Web (CoreWin) paketi için Blazor Server bileşen desteği eklendi.

Diyalog rapor bileşenleri desteği geliştirildi:

- DateTimePicker bileşeni iyileştirildi. `DateTimePicker.Format.Time` modunda yalnızca saat, `DateTimePicker.Format.Short` modunda yalnızca tarih, `DateTimePicker.Format.Long` modunda ise hem tarih hem saat görüntülenir.

DateTimePicker1 Short:

02.09.2021



DateTimePicker2 Long:

11.03.2013



02:00:10



DateTimePicker3 Time:

14:24:33



- MaxLength özelliği desteği eklendi.
- Enabled (etkinleştirme/devre dışı bırakma) özelliği desteği eklendi.
- Arka plan rengi (background-color) desteği eklendi.

MaxLength is 21 symbo

Green background color

ListBox Value1
ListBox Value2
ListBox Value3
ListBox Value4

11.03.2013



Disabled:

☒ CheckBox1

☐ CheckBox2

☒ RadioButton1

☐ RadioButton2

Item 1

Disabled TextBox

02.09.2021

ListBox Value1
ListBox Value2
ListBox Value3
ListBox Value4

Cancel

Ok (disabled)

Düzeltilmeler

- Tablo/Matris hücresi içindeki nesnelerin Dock ve Anchor özellikleriyle ilgili hata düzeltildi.
- Sayfa altbilgisi bandına Alt Rapor (Subreport) nesnesi eklendiğinde oluşan stack overflow hatası giderildi.
- Sistem DPI ayarı 96DPI'den büyük olduğunda oluşan SVG dışa aktarma hatası düzeltildi.

Tüm Değişiklik Listesi

```
[Motor]
+ ElasticSearch bağlantısı eklendi;
+ Yeni barkod eklendi - Japanese PostNet;
+ Kültüre göre yerelleştirme değiştiren yeni yöntem eklendi: `Res.LoadLocale(CultureInfo)`. Yüklenen diller önbellege alınır. Doğru çalışması için FastReport.Localization paketinin projeye eklenmesi gerekir;
* RichText'ten rapor nesnelerine dönüştürücü optimize edildi ve hızlandırıldı;
```

RichText ten rapor nesnelerine dönüştürme optimize edildi ve değiştirildi;
- HtmlTextRenderer kullanıldığında sekme genişliği hatası düzeltildi;
- Alt rapor nesnesi (SubreportObject) sayfa altbilgisi bandında kullanıldığında oluşan StackOverflow hatası düzeltildi;
- Tablo/matris hücresi içindeki nesnelerde Dock ve Anchor özellikleriyle ilgili hata düzeltildi;
- Band dışında yer alan PictureBox çizilirken oluşan `System.ArgumentException` hatası giderildi;
- Sayfa genişliği sınırsız olduğunda Anchor = Right davranışı düzeltildi;
- Rapor hazırlanırken özel yazı tipinin varsayılan yazı tipiyle değiştirilmesi hatası düzeltildi;
- RTF dönüştürme sırasında varsayılan olarak dikey hizalamanın Ortadan Yukarı'ya (Top) çevrilmesi sağlandı;
- RTF tablolarını rapor nesnelerine dönüştürme hataları düzeltildi;

[Tasarımcı]

+ Veritabanı alanlarının basitleştirilmiş görünümü eklendi;
+ Rapor ve veri ağacı için "Tümünü Daralt / Genişlet" düğmesi ve arama alanı eklendi;
+ Yeni simgeler eklendi. "Görünüm > Seçenekler > Arayüz" bölümünden simge paketleri arasında geçiş yapılabilir;
- Veri kaynağındaki hatalı tablo nedeniyle rapor tasarımcısının çökmesine neden olan hata düzeltildi;

[Önizleme]

- Dönüştürülmüş RichObject içeren raporların kaydedilmesindeki hata giderildi;

[Dışa Aktarmalar]

+ Word 2007 dışa aktarmasında "Sayfa sonlarında bölüm sonu ekleme" seçeneği eklendi. Varsayılan olarak her ikisi de eklenir;
- HTML dışa aktarmasında sayfa sonu (`PageBreaks`) düzeltildi;
- SVG dışa aktarmada "Çoklu dışa aktarma" parametresi ile ilgili hata giderildi;
- HiDPI monitörlerde SVG dışa aktarma hatası düzeltildi;
- ZIP arşivinde kaydedilen dosya adları düzeltildi;
- RichObject dışa aktarımında sekme genişliği düzeltildi;
- Linux'ta oluşturulan XPS belgelerinin Windows'ta açılmaması sorunu düzeltildi;
- Sınırsız genişlikte sayfaların dışa aktarımında Dock ve Anchor hataları düzeltildi;
- Excel 2007 dışa aktarımında, `HtmlParagraph` tipi açık olan metin nesneleri için dışa aktarma hatası düzeltildi. WYSIWYG dışa aktarma seçeneği devre dışı bırakıldığında yalnızca metin dışa aktarılır;

[WebReport]

+ WebReport diyaloglarında arka plan rengi (background-color) desteği eklendi;
+ TextBox bileşeni için MaxLength desteği eklendi;
+ Araç çubuğu için yerelleştirme yükleme işlemi optimize edildi;
* Safari tarayıcılarda HTML/Blazor dışa aktarmasında sayfa arka plan rengi hatası düzeltildi;
- Oturumlarla birlikte çalışan WebReport'ta çevrimiçi tasarımcının kaydetme geri çağırma (save callback) hatası düzeltildi;
- Sınırsız genişlikte sayfalarda Anchor ve Dock özellikleriyle ilgili hatalar giderildi;
- Enabled özelliği desteği eklendi;

[Çevrimiçi Tasarımcı]

- Yatay (Landscape) sayfa yönlendirmesi ile kaydetme/önizleme hataları düzeltildi;

[.NET Core]

+ Single File Application (SFA) desteği eklendi;
* FastReport.Compat ve FastReport.DataVisualization bağımlılıkları güncellendi. `FastReport.Compat`, artık WinForms API kullanılabilirliğini doğru şekilde tespit eder. `FastReport.DataVisualization`, artık `System.Data.SqlClient` ve `System.Drawing.Common` bağımlılıklarını içermez;
- `CoreWin` sürümünde `ReferencedAssemblies` ile kaydedilmiş özel kütüphanelerden veri çekilememesi sorunu düzeltildi;
- Linux'ta bilinmeyen yazı tipiyle çoklu iş parçacığında rapor yüklenirken uygulamanın çökmesi düzeltildi;
- `System.Drawing.Design.UITypeEditor` tipi yüklenemedi hatası düzeltildi;
- `XmlDataConnection` ile tablo adlarının yüklenmesi düzeltildi;
- IIS / IIS Express kullanılarak yayımlama veya hata ayıklama sırasında rapor ve kaynakların yüklenmemesi sorunu düzeltildi. Bunun için `UseMvc/UseEndpoints` öncesinde `UseFastReport()` çağırısı yapılmalıdır;

[WebReport Core / Blazor Server]

+ FastReport.Core3.Web paketi için Blazor bileşen desteği eklendi;
_ Blazor'da çok satırlı metinlerin hatalı gösterimi düzeltildi;
+ Web kütüphanelerine XML açıklama (`DocumentationFile`) eklendi;
+ Araç çubuğunun diyalog sayfasında görünmesini devre dışı bırakmak için `webReport.Toolbar.ShowOnDialogPage` özelliği eklendi;
+ Araç çubuğu açılır menüsüne daha fazla dışa aktarma formatı eklendi (`webReport.Toolbar.Exports`);

- + Araç çubuğunun konumu, açılır menu rengi, yazı tıpi, simge saydamlığı, simge rengi ve içerik konumu gibi ayarlarını özelleştirme desteği eklendi (`webReport.Toolbar`);
- + WebReport'ta DateTimePicker bileşeni geliştirildi: `Format.Time`: sadece saat, `Format.Short`: sadece tarih, `Format.Long`: tarih ve saat birlikte;
- Label diyalog bileşeninde satır sonu eksikliği düzeltildi;

[Extras]

- + Çok dilli çalışmak için FastReport yerelleştirme dosyalarını içeren `FastReport.Localization` paketi eklendi;
- + `Extras\Objects\FastReportBGObjects` dizinine FastReport Business Graphics entegrasyon nesneleri eklendi;

[Demolar]

- + Yeni demoda klasöre tıklanınca rapor listesine geçiş uygulandı;
- * Hedef çerçeve .NET 4.7.2 olarak değiştirildi;
- * Küçük resim modunda pasif düğmelerin rengi, yakınlaştırma çubuğunun arka plan rengi, diyalog formu ve etkileşimli sekmelerin arka plan renkleri güncellendi;
- * Rapor küçük resimlerinin saklandığı klasör artık Program Files yerine AppData\Local dizininde;
- * Yeni demo arayüz öğelerinin görüntülenmesiyle ilgili sorunlar giderildi;
- * Kayıt menüsündeki açılır öğelere tıklanınırken hazırlanan raporun yanlışlıkla kaydedilmesi hatası düzeltildi;
- * Önizleme penceresinde raporların hizalanmasıyla ilgili hata düzeltildi;
- Rapor seçildiğinde diyalog formunun iki kez açılması sorunu düzeltildi;
- Küçük resimlerin orantısız büyütülmesi hatası düzeltildi;
- Demo penceresi taşınırken oluşan gecikme giderildi;
- Klasör ve rapor çubuğunda aynı anda küçük resimlerin gösterilmesi hatası düzeltildi;

[Eklentiler]

- * FastReport.Data eklenti paketleri güncellendi. Artık bu paketler .NET, Core, CoreWin, OpenSource sürümleri için eklentiler içeriyor ve kullanılan ürüne bağlı olarak gerekli kütüphaneleri otomatik olarak dahil ediyor;
- * PostgreSQL Npgsql sürümü 4.0.3'ten 3.2.7'ye düşürüldü.

İletişim ve Teknik Destek

Ürün kullanımıyla ilgili sorularınızı her zaman [e-posta](#), veya sitedeki [formlar aracılığıyla](#) iletebilirsiniz.

Ürünümüzle ilgili önerilerinizi de memnuniyetle karşılarız.