

یکی از ضروریات دنیای برنامه نویسی امروز، داشتن یک الگوی مناسب می‌باشد. یکی از الگوهای مناسب برای وب فرم‌ها، استفاده از الگوی MVP است.

اگر در خلال پیاده سازی، گاهی اوقات نیاز به handle کردن رخدادها را داشته باشید بدین منظور به روش زیر عمل می‌کنیم: (توجه: شیء مورد نظر ما در این پست RadGrid از کنترل‌های Telerik در نظر گرفته شده است).

```
// ASPX page
<telerik:RadGrid ID="RadGrid1" runat="server"></telerik:RadGrid>
// Asp.Net Code Behind
protected void Page_Load(object sender, EventArgs e)
{
    GridPresenter presenter = new GridPresenter(this);
}
// view interface
public interface IGridView
{
    Telerik.Web.UI.RadGrid myGrid { get; }
}
// presenter
protected readonly IGridView _view;
public GridPresenter(IGridView view)
{
    _view = view;
    _view.myGrid.UpdateCommand += new Telerik.Web.UI.GridCommandEventHandler(onUpdateCommand);
    _view.myGrid.InsertCommand += new Telerik.Web.UI.GridCommandEventHandler(onInsertCommand);
    _view.myGrid.EditCommand += new Telerik.Web.UI.GridCommandEventHandler(onEditCommand);
}
private void onUpdateCommand(object sender, Telerik.Web.UI.GridCommandEventArgs e)
{
    // Code for updating
}
private void onInsertCommand(object sender, Telerik.Web.UI.GridCommandEventArgs e)
{
    // Code for inserting
}
private void onEditCommand(object sender, Telerik.Web.UI.GridCommandEventArgs e)
{
    // Code for editcommand
}
```

نظرات خوانندگان

نویسنده: حسین مرادی نیا
تاریخ: ۱۳۹۱/۰۳/۲۹ ۰:۴۹

خوبه
اما من به قسمت رو متوجه نشدم. ببینید ما می‌گیم (MVP) (Model View Presenter).
خب حالا سوال اینجاست که در بخش Asp.net Code Behind از چه کدی استفاده می‌کنیم؟
یعنی کدهای View و Presenter چه موقع فراخوانی میشوند؟
از کجا اجرا میشود و به کجا ختم میشود؟
مدل ما کجاست؟
و ...

نویسنده: امیر هاشم زاده
تاریخ: ۱۳۹۱/۰۳/۲۹ ۱۱:۱

ابتدا وب فرم مورد نظر اینترفیس IGridView را پیاده سازی می‌کند و سپس در Code behind در Page_Load یک وهله از presenter با پارامتر this ایجاد می‌کنید، در نتیجه کلیه رخدادهای presenter مدیریت می‌شوند.
نکته در این است که چون هر بارگذاری صفحه نیاز به متصل کردن رخدادهای به شیء هستیم مجبوریم در Page_Load یک وهله از presenter ایجاد کنیم.
در این پست هدف پیاده سازی مدل نبوده است، ولی شما می‌توانید مدل خود را با توجه به الگوی MVP طراحی کنید و از آن در presenter استفاده کنید.

نویسنده: حسین مرادی نیا
تاریخ: ۱۳۹۱/۰۳/۲۹ ۱۴:۲۶

مرسی
مشکل قطعه کد زیر بود که باعث شده بود که نحوه استفاده رو متوجه نشم!

```
GridPresenter presenter = new GridPresenter(this);
```

در مورد Model هم شما درست می‌گید. هدف این پست استفاده از Model نبود.

نویسنده: جلال
تاریخ: ۱۳۹۱/۰۷/۲۲ ۱۲:۴۱

سلام،
با این قضیه ارجاع مستقیم به یک کنترل خاص در IView می‌تونم کنار بیام، ولی با ارجاع به اون در CPresenter نه!
دلیل خاصی داشته؟ آخه معمولاً برای رخدادهای رو در IView معرفی می‌کنن و در CPresenter استفاده می‌کنند.

البته در [این مقاله](#)، کلاً منکر استفاده از EventHandlerها در IView میشه و ترجیح میده فقط تابع Presenter رو از Code Behind فراخوانی کنه!

نویسنده: امیر هاشم زاده
تاریخ: ۱۳۹۱/۰۸/۱۶ ۱۸:۴۸

یکی از دلایل استفاده از این روش، تمیز نگه داشتن ویو (با توسعه برنامه پارامترهای لازم برای 4 عمل اصلی اضافه می‌شد

که نگهداری پارامترهای رخداد گردانها پیچیده و دشوار می شد) بود، زمانی که داشتم این پیاده سازی رو در پروژه انجام می دادم به این نتیجه رسیده بودم بهتر است کلیه عملیات های افزودن، حذف، ویرایش و حذف شیء تلریک گرید و یو رو در presenter انجام بدم چون قرار بود یک سری عملیات منطقی بر روی 4 عمل اصلی انجام بدم.

یکی از ضروریات نرم افزارها وجود گزارشات مختلف در قالب لیست‌ها ، نمودارها و ... در آنها می‌باشد. یک نرم افزار خوب باید توانایی ارائه گزارشات خوب و زیبا را نیز داشته باشد. گزارشات در حقیقت نمایشی از داده‌ها هستند که عموماً به چاپ می‌رسند. مورد دیگری که در خصوص گزارشات حائز اهمیت می‌باشد ، تبدیل آنها به فرمت‌های مختلف جهت حمل و جابه جایی آسان از سیستمی به سیستم دیگر می‌باشد. برای مثال تبدیل گزارشات به قالب Pdf مزایایی بسیاری از نظر قابل حمل بودن در پی خواهد داشت. در برنامه نویسی دات نت گزارشات را می‌توان توسط دستورات چاپ در دات تهیه ، نمایش و چاپ نمود. اما تهیه گزارش توسط دستورات دات نت کاری مشکل و طاقت فرسا می‌باشد. همچنین امکان تبدیل این گزارشات به فرمت‌های دیگر نظیر Pdf ، به راحتی انجام نمی‌شود و باید از کلاس‌ها و ابزارهای جانبی که برای این کار تهیه شده اند استفاده نمود . از این رو ابزارهای مختلفی در جهت تهیه گزارشات به وجود آمدند. ابزارهایی نظیر :

• Crystal Report

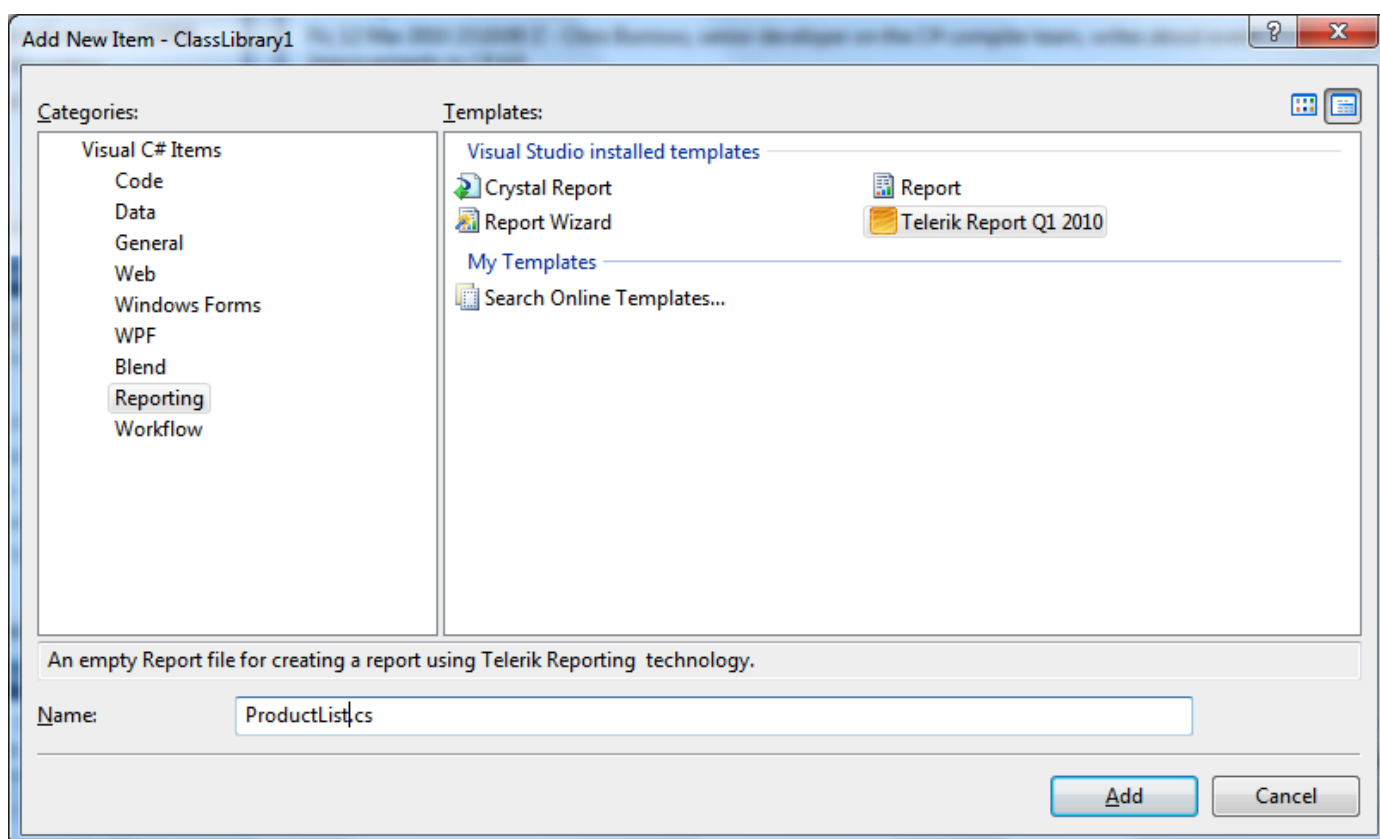
• Stimul Report

• Telerik Reporting

• ...

در ادامه این سری آموزش‌ها قصد داریم Telerik Reporting و نحوه تهیه گزارش با آن را مورد بررسی قرار دهیم. این ابزار امکانات بسیاری در خصوص تهیه گزارش برنامه‌های دات نت نظیر Asp.net ، Windows Form و ... در اختیار ما قرار می‌دهد. در ادامه و برای شروع ، ساخت یک گزارش ساده در این محیط را بررسی می‌کنیم.

نکته : گزارشاتی که توسط Telerik Reporting تهیه می‌شوند به وسیله کدهای C# جنریت می‌شوند. بنابراین همیشه توصیه می‌شود گزارشات خود را درون یک یا چند پروژه Class Library قرار دهیم و از این پس ، این گزارشات از درون پروژه‌های دیگر (ویندوزی ، وب و ...) در دسترس هستند. کافی ست پروژه Class Library را به عنوان Reference به پروژه مورد نظر خود اضافه کنیم.. برای شروع می‌توان یک پروژه جدید از نوع Class Library ایجاد کرد. پس از آن روی نام پروژه راست کلیک کنید و گزینه Telerik Report را انتخاب نمایید. پس از تعیین نام گزارش کلید Ok را کلیک نمایید.

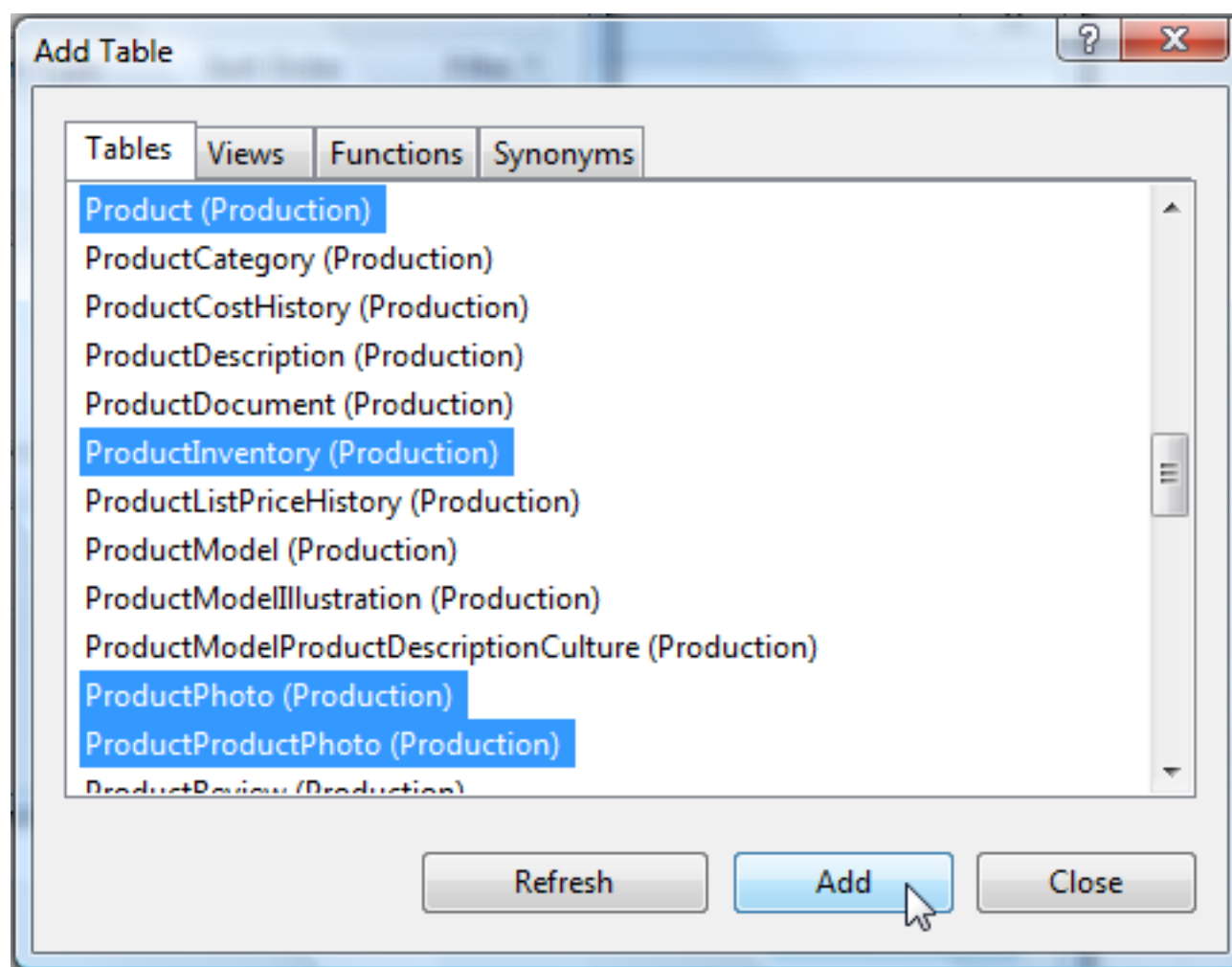


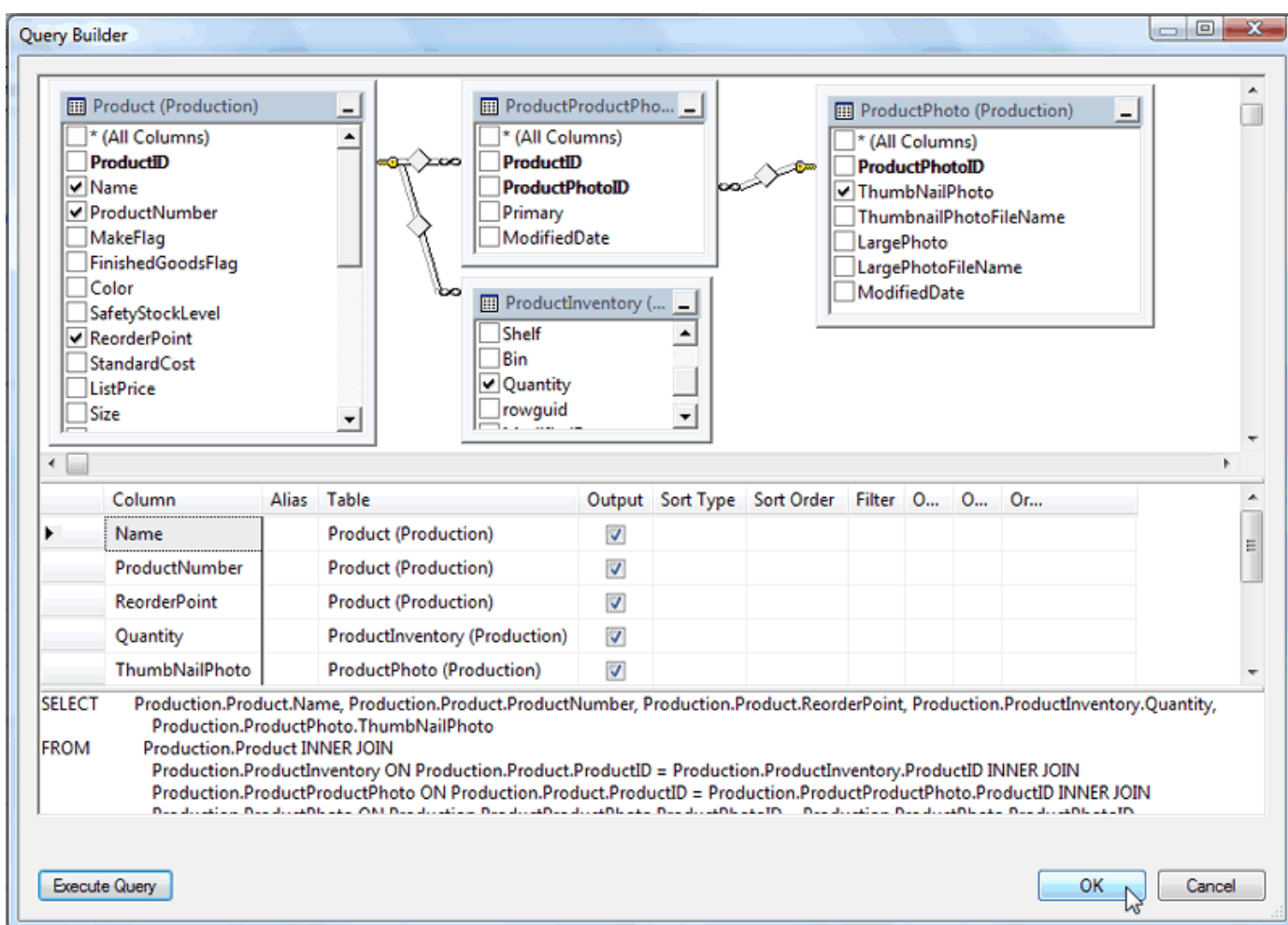
در این حالت فایل گزارش به پروژه افزوده می‌شود. در ادامه می‌توانید توسط ویزاردی که نمایش داده می‌شود کارهای عمومی مربوط به پیاده سازی گزارش (انتخاب منبع داده (Data Source)، ساخت Query جهت بارگذاری اطلاعات، فیلدهایی که باید نمایش داده شوند، گروه بندی داده ها و ...) را توسط این ویزارد انجام دهید. برای اینکار در پنجره ای که نمایش داده می‌شود بر روی کلید Next کلیک نمایید.

جهت ایجاد یک گزارش جدید در پنجره Report Choose Page گزینه New Report را انتخاب نموده و کلید Next را کلیک نمایید. جهت انتخاب منبع داده گزارش در پنجره Choose Data Source گزینه Add New Data Source را انتخاب نمایید. در این حالت می‌توانید گزینه‌های متفاوتی را به عنوان منبع داده گزارش خود انتخاب نمایید. گزینه‌های نمایش داده شده به شرح ذیل است:

- Sql Data Source : جهت اتصال مستقیم به بانک اطلاعات Microsoft Sql Server
- Object Data Source : جهت اتصال به کلاس‌های لایه Business و بارگذاری داده از این کلاس ها
- Entity Data Source : جهت اتصال به Entity Framework
- Open Access Data Source : جهت اتصال به Open Access ORM ساخت شرکت Telerik
- Cube Data Source : جهت اتصال و نمایش داده‌های تحلیل شده

در ادامه برای اینکه بتوان مستقیماً به Sql Server وصل شد و Queryهای مربوط به گزارش را روی آن اجرا نمود؛ می‌توان گزینه New Sql Data Source را انتخاب نمود و بر روی کلید Ok کلیک کرد. سپس در پنجره Choose Your Data Connection گزینه New Connection را کلیک کنید و یک اتصال به بانک مورد نظر خود ایجاد کنید. پس ایجاد و تست Connection ساخته شده روی Next کلیک کنید. در پنجره Save the connection string می‌توان نامی را جهت Connection string انتخاب کرد تا Connection string با همان نام در فایل Config پروژه ذخیره شود. در ادامه کلید Next را کلیک کرده و وارد مرحله بعد شوید. در پنجره Configure Data Source Command گزینه Query Builder را جهت ساخت Query مورد نظر برای بارگذاری داده‌ها انتخاب نمایید.





پس از ساخت Query مورد نظر کلید Ok را کلیک نمایید. در پنجره Configure Data Source Command کوئری ساخته شده به شما نمایش داده می‌شود. کلید Next را کلیک کنید.

سپس وارد مرحله Preview Data Source Result می‌شوید که در آن قادر خواهید بود پیش نمایشی از داده‌هایی که بعداً توسط Query ساخته شده بارگذاری خواهند شد را مشاهده نمایید. Next را کلیک نموده تا وارد مرحله بعد شوید. مرحله بعد Standard Report Type می‌باشد که در این مرحله شما می‌توانید نوع گزارش خود را انتخاب نمایید و کلید Next را فشار دهید. در بخش Design Data Layout چند فیلد را از بخش سمت چپ (Available Fields) انتخاب نموده و کلید Details را کلیک نمایید. فیلدهای انتخاب شده به بخش Details گزارش اضافه خواهند شد. در ادامه Next را کلیک کنید تا وارد بخش Choose Report Layout شوید. شما می‌توانید در این بخش یک حالت نمایشی را برای گزارش خود انتخاب نمایید و Next را کلیک نمایید. در بخش Choose Report Style یک قالب بندی جهت گزارش خود انتخاب نمایید. در ادامه Next و سپس Finish را کلیک نمایید. کدهای گزارش Generate شده می‌توان در قسمت Designer گزارش را مشاهده نمود.

در این حالت کارهای زیر توسط Wizard به صورت اتوماتیک انجام خواهد شد:

- بایند شدن اتوماتیک فیلدهای گزارش به ستون‌های مرتبط
 - اعمال قالب بندی انتخاب شده برای صفحه و سر ستونها
 - افزودن تاریخ و شماره صفحه به پایین گزارش
- در ادامه پروژه را Rebuild کرده و گزینه Preview را در Designer جهت نمایش، پیش نمایش گزارش کلیک نمایید.



نکته : در هر برنامه‌ی گزارش سازی بخش Designer گزارش به 4 بخش کلی تقسیم می‌شود:

- Report Header: مواردی که در این بخش از گزارش قرار می‌گیرند در بالای صفحه اول گزارش نمایش داده می‌شوند.
- Page Header: مواردی که در این بخش از گزارش قرار می‌گیرند در بالای همه صفحات گزارش قرار گرفته و تکرار می‌شوند.
- Details: داده‌های اصلی گزارش که شامل جزئیات و بخش اصلی گزارش می‌باشند و سطر به سطر نیز تکرار می‌شوند در این بخش قرار می‌گیرند.

• Page Footer: مواردی که در این بخش از گزارش قرار می‌گیرند در پایین همه صفحات نمایش داده می‌شوند.

• Report Footer: مواردی که در این بخش قرار می‌گیرند در پایین صفحه آخر گزارش نمایش داده می‌شوند.

ادامه دارد ...

نظرات خوانندگان

نویسنده: **ramin_rp**
تاریخ: ۱۳۹۱/۰۴/۱۱ ۸:۳۰

سلام دوست عزیز
ممنون بابت مطلب خوبتون.
یک پیشنهاد، اگه براتون مقدوره تو یه مطلب چند تا از ابزارهای گزارش سازی رو با هم مقایسه کنید (مثلا , Telerik , stim1 , devexpress, crystal ...)

نویسنده: **اژدری**
تاریخ: ۱۳۹۱/۰۶/۱۳ ۱۰:۳۰

با سلام
از نظر قدرت مانور در گزارش گزارش ساز devexpress از سایر ابزارها مثل Telerik , crystal , ssrs فوق العاده قوی تره و امکان نمایش داده ها به صورت پدر و فرزندی رو تا n لایه ارائه میده چیزی ، امکان تغذیه گزارش با داده هایی که خود شما می خواید مثل یک IQueryable یا list یا datatable و میتونه کانکشن نداشته باشه و در هر سطحی از گزارش شما امکانات کامل کد نویسی رو برای هر رویدادی در گزارش دارید که فوق العاده است و یک گزارش هم در وب و هم در ویندوز قابل استفاده مجده ولی مشکلی که هست و باعث شد بنده فعلا با ssrs کار کنم اینه که در وب چیدمان گزارش به هم میخوره که در مثال های شرکت سازنده چنین چیزی دیده نمیشه - رتبه دوم متعلق به ssrs هست که ابزاری پر قدرت با قابلیت برنامه نویسی توکار vb.net و درج هر گونه کتابخانه خارجی در صورت نیاز است ، تا اونجایی که بنده اطلاع دارم Telerik یکی از ضعیف ترین نوع گزارشات رو در بین بقیه شرکت ها ارائه میده

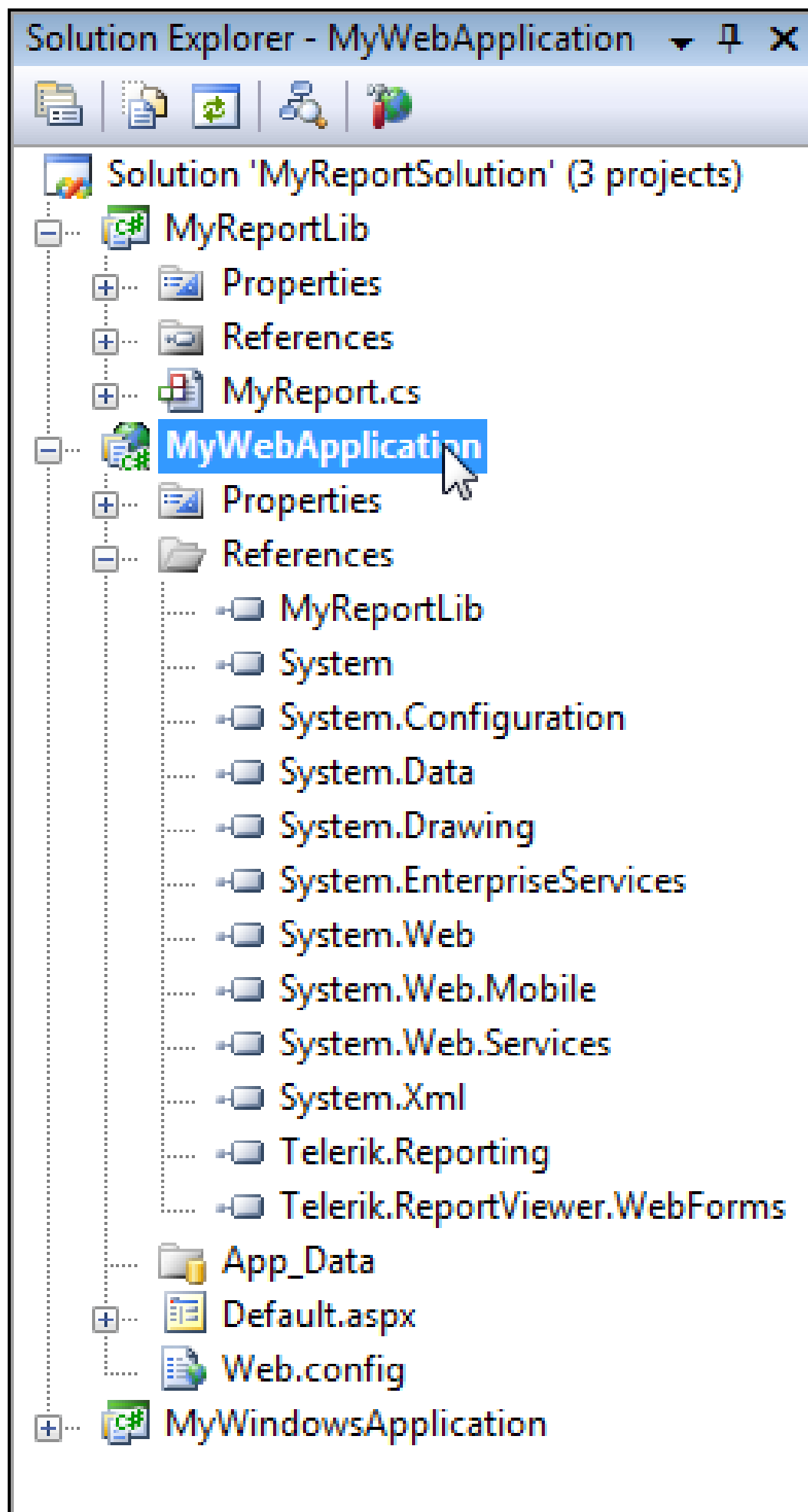
نویسنده: **جمشیدی فر**
تاریخ: ۱۳۹۲/۰۸/۳۰ ۱۶:۸

چطور میشه با Telerik Report نمودار دایره ای بارنگ های متفاوتی که خودمان مشخص می کنیم، رسم کرد؟ درحالتی که نمودار درحالت Design ایجاد نشده و Programmatically ایجاد شده.

نویسنده: **حسین**
تاریخ: ۱۳۹۳/۰۷/۰۲ ۱۴:۲

البته در این مورد که فرمودین DevExpress کا مله من هم موافقم ، جالب تر اینکه ساخت گزارش و نمایش ان در MVC بسیار راحت تر است اما ایراد بزرگی که داره اینه که خیلی سنگین و کندتر از تلریک عمل میکنه که منو مجبور کرد برگردم دوباره سراغ Telerik

در این قسمت به بررسی مجدد یک نکته یا Best Practice در حین کار با Telerik Reporting می‌پردازیم. همانطور که گفته شد روش توصیه شده جهت استفاده از Telerik Reporting ، ساخت یک پروژه Class Library و قرار دادن گزارشات در آن می‌باشد. در ادامه می‌توان پروژه Class Library ساخته شده را به هر تعداد پروژه ویندوزی و یا وب متصل کرد و گزارشات ساخته شده را به راحتی در محیط‌های مختلف نمایش داد. در تصویر زیر ساختار یک پروژه را بر اساس مباحث گفته شده مشاهده می‌کنید:



- MyReportLib: پروژه Class Library شامل گزارشات Telerik Reporting می باشد. در اینجا پروژه فقط یک گزارش با نام "Report1.cs" دارد که در تصویر مشاهده می کنید.
- MyWebApplication: یک پروژه از نوع Asp.net می باشد که گزارشات ساخته شده را می توان در آن به کاربر نمایش داد(در قالب صفحات HTML). توجه داشته باشید که این پروژه یک Reference به پروژه Class Library حاوی گزارشات دارد.
- MyWindowsApplication: یک پروژه از نوع Windows Application است که برای نمایش گزارشات ساخته شده در محیط ویندوز می توان از آن استفاده کرد. توجه داشته باشید که این پروژه هم برای نمایش گزارشات ساخته شده یک Reference به پروژه Class Library دارد.

نظرات خوانندگان

نویسنده: امیر هاشم زاده
تاریخ: ۱۵:۴۳ ۱۳۹۲/۱۲/۱۲

چه خوب بود که توضیحاتتون را خیلی کامل تر و همراه با سناریوهای فرضی قرار می دادید.

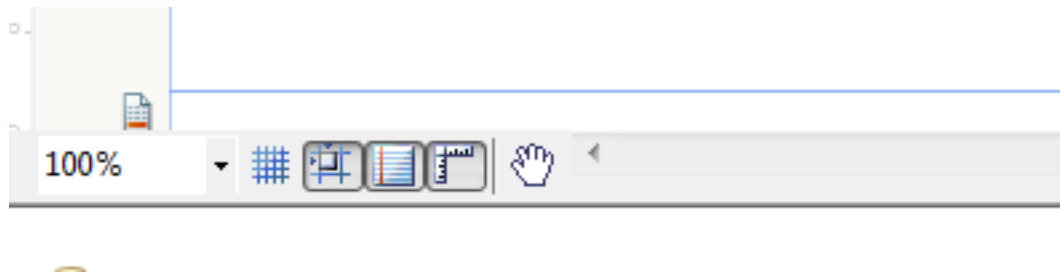
در این بخش قصد داریم به طور خلاصه تعدادی از ویژگی‌های Telerik Reporting را جهت ساخت گزارشات مورد بررسی قرار دهیم.

ویژگی‌های مورد بحث شامل موارد زیر می‌باشند:

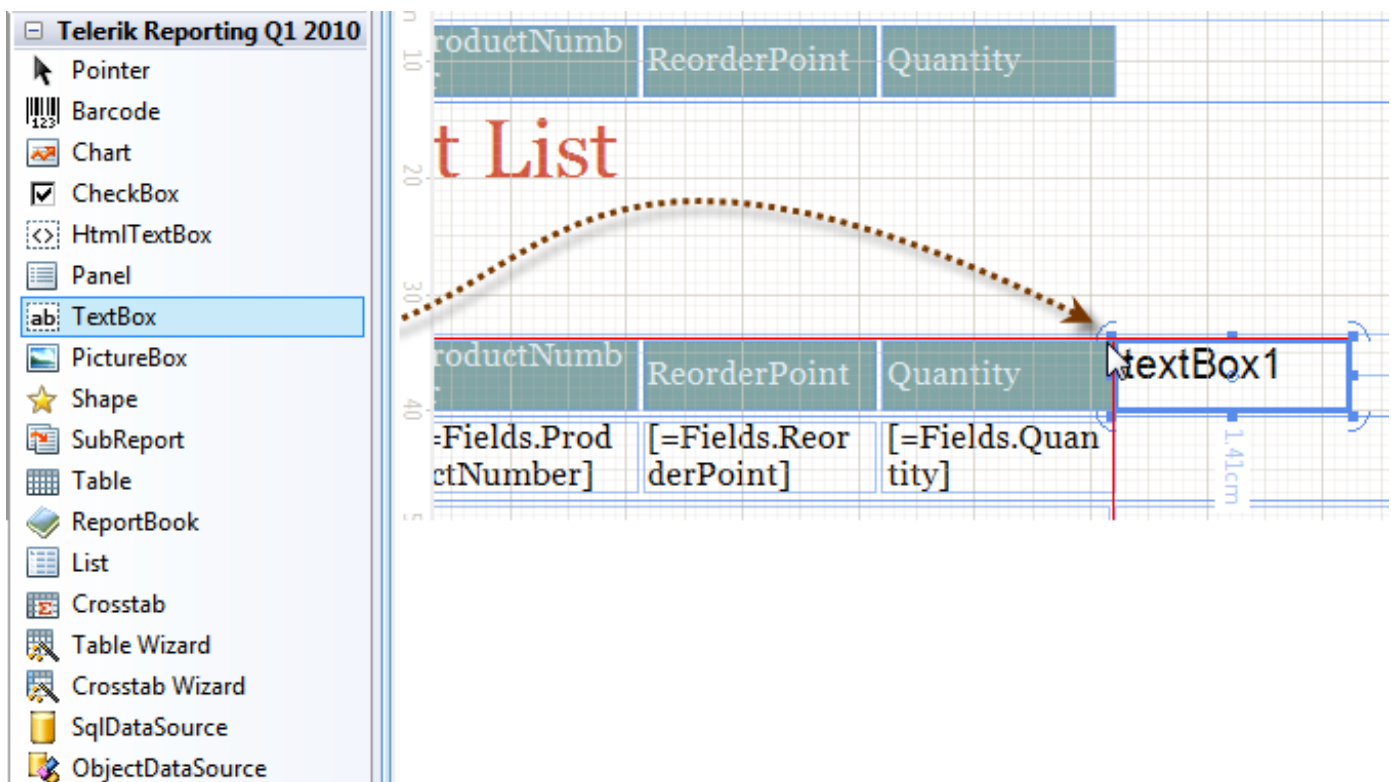
- ویرایش TextBox ها در محیط Designer
- Copy و Paste کردن Style ها از یک کنترل به کنترل دیگر
- قالب بندی شرطی
- پیمایش و تغییر اندازه گزارش و آیتم‌های آن
- تغییر اندازه بخش‌های مختلف گزارش نظیر Page Header ، Detail و ...
- افزودن TextBox Shape و PictureBox درون Designer گزارش
- استفاده از پنجره Data Explorer
- استفاده از پنجره Report Explorer

در ادامه به بررسی موارد ذکر شده جهت طراحی گزارش می‌پردازیم.

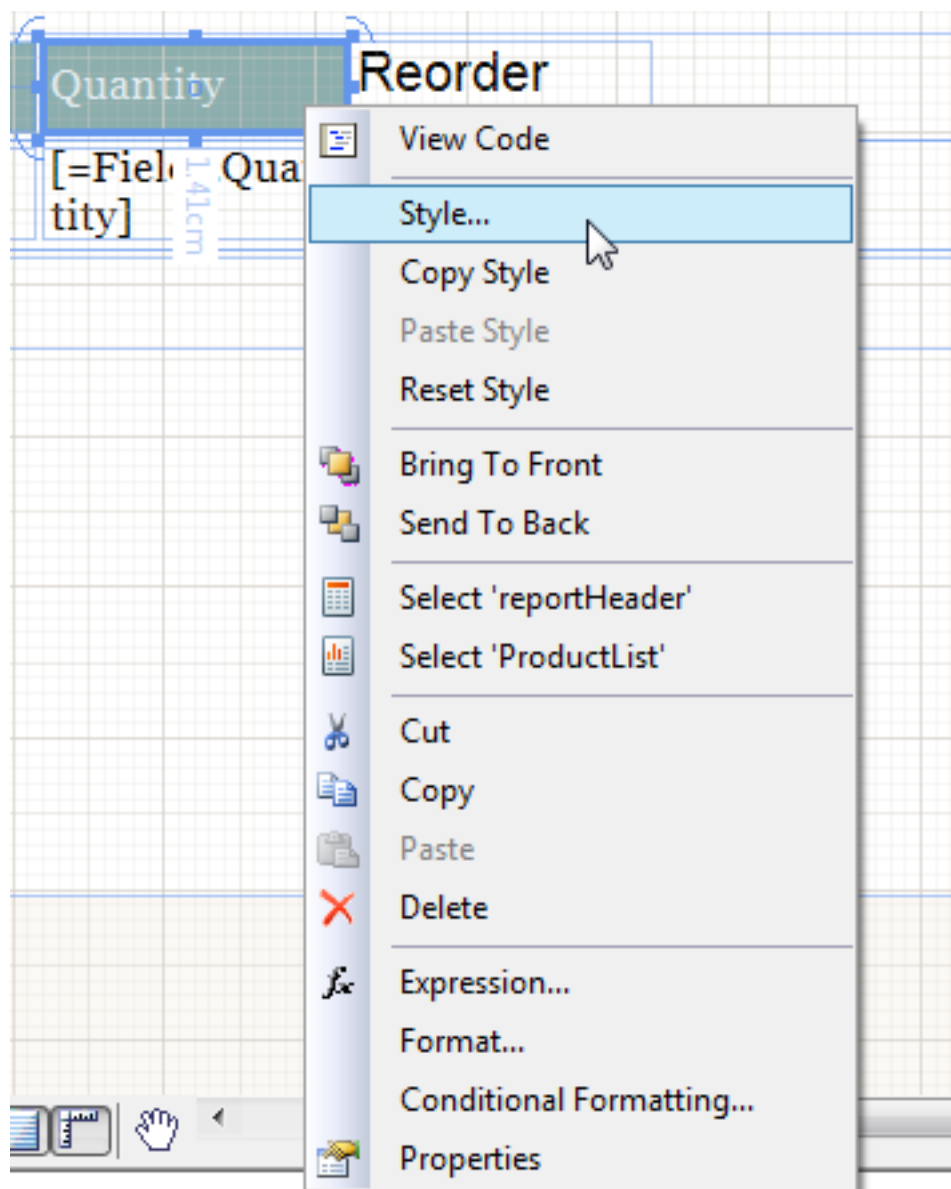
1. جهت تغییر محتوای یک TextBox می‌توان روی آن دوبار کلیک نمود. پس از آن TextBox به حالت ویرایش می‌رود و می‌توان متن درون آن را به سادگی تغییر داد. این کار در محیط Designer انجام می‌شود و نیازی نیست برای تغییر محتوای TextBox به پنجره Properties بروید و متن آن را تغییر دهید.
2. در محیط Designer به راحتی می‌توانید گزارش خود را Zoom نمایید. این کار توسط ComboBox مربوطه در پایین سمت چپ Designer انجام می‌شود.



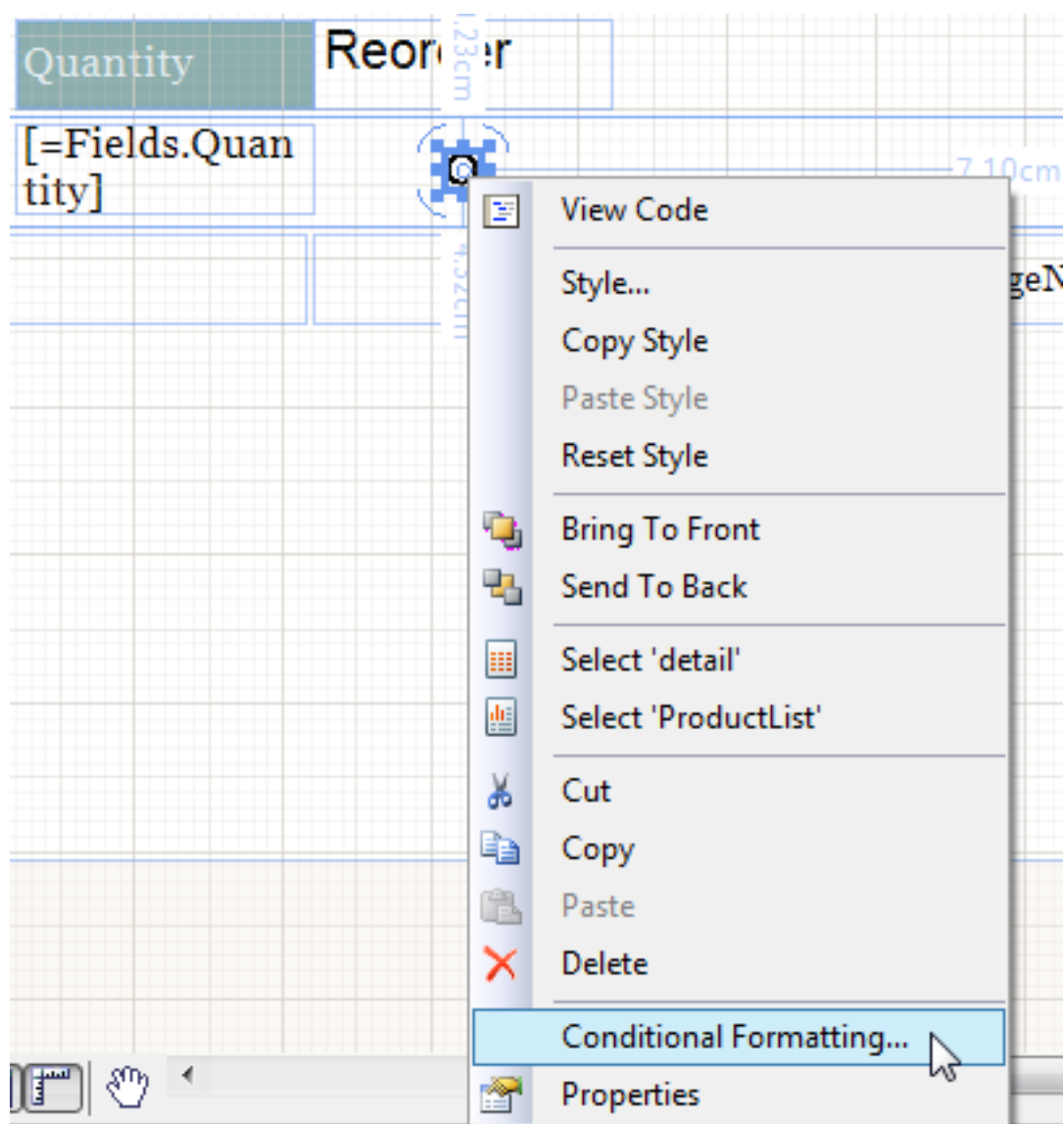
3. اگر قرار باشد TextBox جدیدی به گزارش خود اضافه نمایید، کافی است آن را از بخش Toolbox به محیط گزارش بکشید و سپس به چپش آن بپردازید.



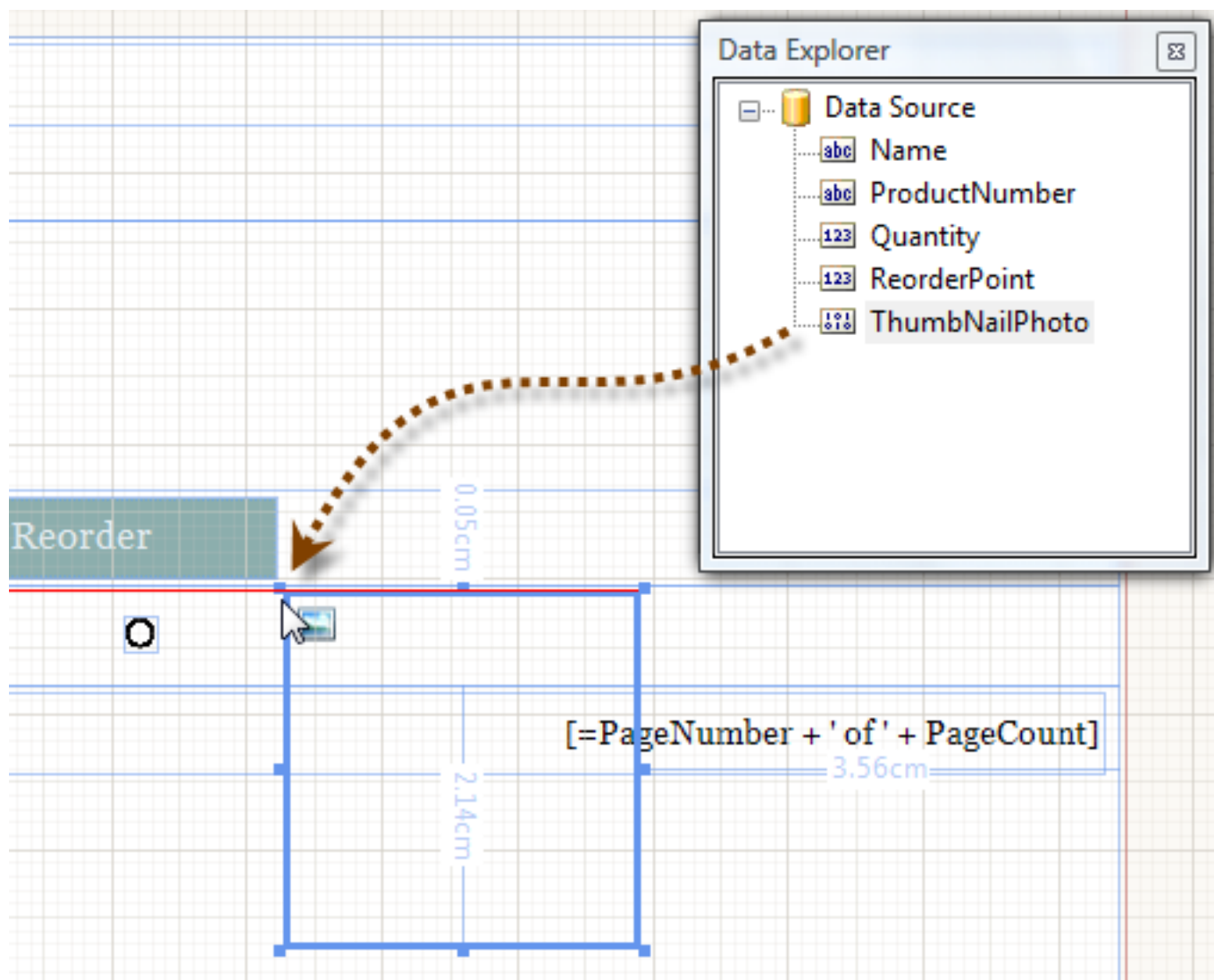
4. در محیط طراحی گزارش Telerik می‌توانید به راحتی یک قالب بندی و یا Style را از کنترلی به کنترل دیگر کپی نمایید و در وقت خود جهت طراحی گزارش صرفه جویی نمایید. برای انجام اینکار کافی است روی کنترلی که قرار است Style آن را کپی نمایید راست کلیک نموده و پس از آن از منوی ظاهر شده گزینه Copy Style را انتخاب نمایید. در ادامه می‌توانید کنترل و یا کنترل هایی که قرار است قالب بندی را به آنها اعمال کنید انتخاب نموده ، روی آنها راست کلیک نمایید و گزینه Paste Style را انتخاب کنید. با این کار Style ی که در مرحله قبل از کنترلی دیگر کپی کرده بودید به کنترل یا کنترل های انتخاب شده اعمال می‌شود.



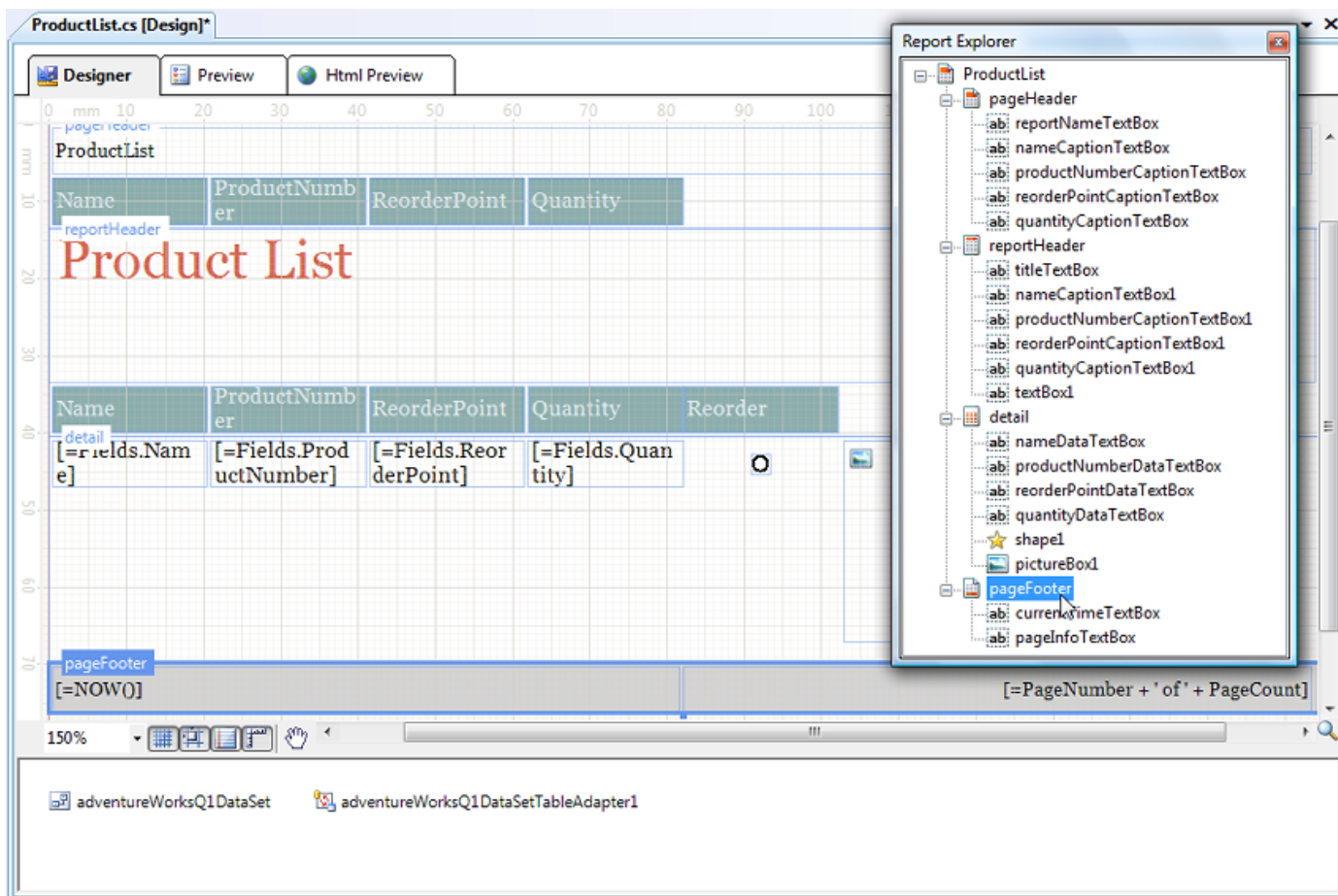
5. یکی دیگر از امکانات Telerik Reporting امکان قالب بندی شرطی (Conditional Formating) می باشد. یعنی Style یک کنترل توسط شرطها تعیین می شود. برای مثال می توانیم بگوییم که اگر مقدار فروش بیشتر از مبلغ خاصی بود ، عدد نمایش داده شده با رنگ سبز نمایش داده شود و یا اینکه از فونت و یا اندازه دیگری جهت نمایش آن استفاده شود (به طور کلی با توجه به شرطهای تعیین شده نمایش آن کنترل با یک Style متفاوت صورت گیرد). در قسمت های آینده به بررسی کامل این قابلیت نیز خواهیم پرداخت.



6. یکی از امکاناتی که در هنگام طراحی گزارش در اختیار ما قرار میگیرد پنجره Data Explorer می باشد. توسط این پنجره می توان فیلدهای یک منبع داده (DataSource) را مشاهده نمود و برای اینکه بتوان از آنها در محیط طراحی استفاده کرد بر روی محیط طراحی درگ نمود. DataSource ها انواع مختلفی دارند که در قسمت اول این آموزش به معرفی آنها پرداختیم و از نمونه آن نیز جهت طراحی یک گزارش ساده استفاده کردیم. در ادامه نیز با این موارد بیشتر آشنا خواهید شد. در تصویر زیر نحوه ی درگ کردن یک فیلد تصویر را از پنجره Data Explorer مشاهده می نمایید.



7. یکی دیگر از اجزای Reporting پنجره Report Explorer می باشد. توسط این پنجره می توان دسترسی سریعی به اجزای درون گزارش داشت. برای مثال می توان به راحتی یک بخش درون گزارش را انتخاب نمود و در پنجره Properties تغییراتی در آن اعمال نمود.



ادامه دارد ...

نظرات خوانندگان

نویسنده: علیرضا صالحی
تاریخ: ۱۳:۶ ۱۳۹۱/۰۴/۱۷

به نظرم در معرفی کامپوننت‌ها قبل از هر موضوعی باید به این نکته پرداخت که سطح پشتیبانی کامپوننت در زمینه unicode و نمایش محتویات rtl و bidi چگونه است.

نویسنده: حسین مرادی نیا
تاریخ: ۱۴:۱۱ ۱۳۹۱/۰۴/۱۷

سلام
بله ؛ حق با شماست
Telerik Reporting از Unicode و rtl به خوبی پشتیبانی میکند و در کل سیستم گزارش ساز قوی هستش.

نویسنده: پژمان
تاریخ: ۲۳:۱۹ ۱۳۹۱/۰۶/۰۱

به نظر جالب میاد. خروجی pdf اون فارسی رو هم پشتیبانی می‌کنه؟

نویسنده: حسین مرادی نیا
تاریخ: ۱:۲ ۱۳۹۱/۰۶/۰۲

بله
تمام خروجی هایی که داره (Word , Excel , PDF و ...) همگی فارسی رو به خوبی پشتیبانی میکنن.

نویسنده: مهدی
تاریخ: ۱۸:۵۳ ۱۳۹۲/۰۲/۰۷

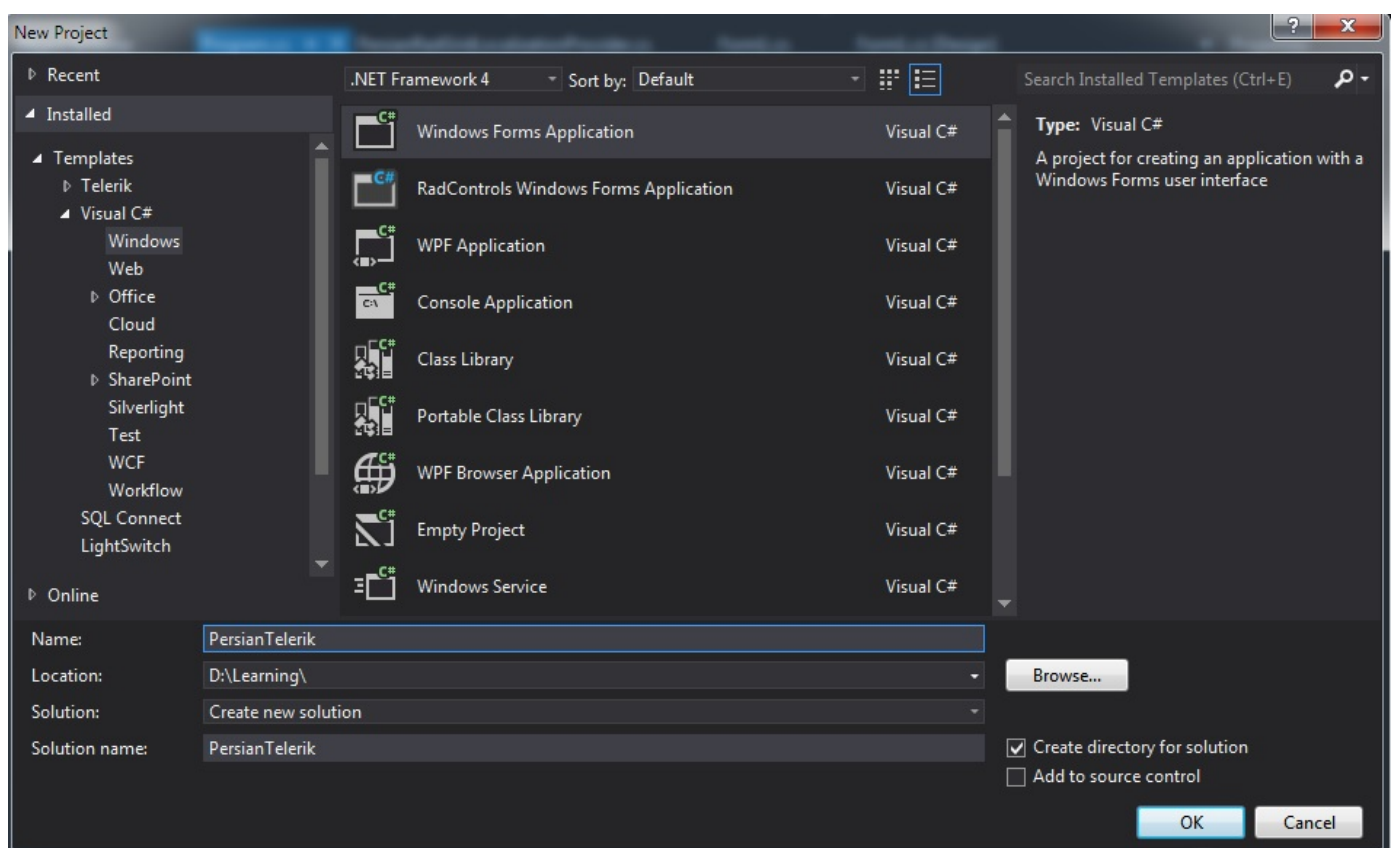
با سلام
امکان این هست که آموزش استفاده از telerik reporting را رو وب سایت بزارید به صورتی که از ویزاد استفاده نشه

و کلا در code behind فرم مون پیام از دیتابیس اطلاعات را بخونیم و تو گزارش قرار بدیم

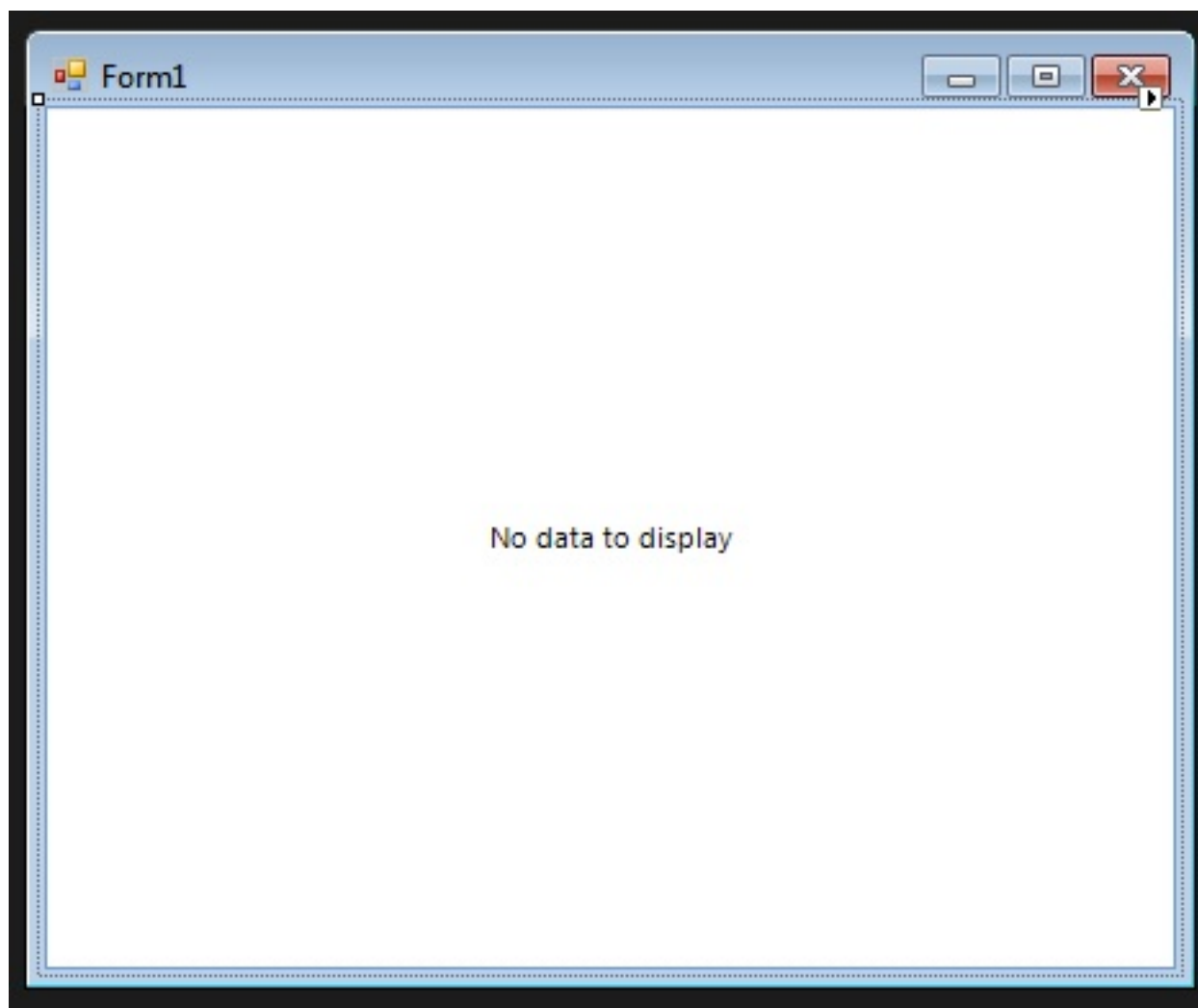
با تشکر

در پروژه های ویندوزی یکی از بیشترین ابزار کاربردی گریدویو تلریک Telerik GridView میباشد و اینکه تمامی امکانات گرید مانند گروه بندی ، فیلترینگ و ... همه فارسی باشند خیلی برای پروژه خوب است. منم در یکی از پروژه ها نیاز به فارسی کردن این ابزار پرکاربرد ویندوزی داشتم و توانستم این مورد را حل کنم . نحوه فارسی کردن این ابزار به شرح ذیل میباشد:

1- یک پروژه جدید ویندوزی در visual studio ایجاد میکنیم



2- اضافه کردن یک radGridView به فرم و خاصیت Dock آن را به حالت Fill و خاصیت RightToLeft را Yes قرار میدهیم :



3- حال برای اینکه یک سری اطلاعاتی داخل این گرید نمایش بدهیم یک کلاس در همان فرم درست میکنیم مشابه کد ذیل :

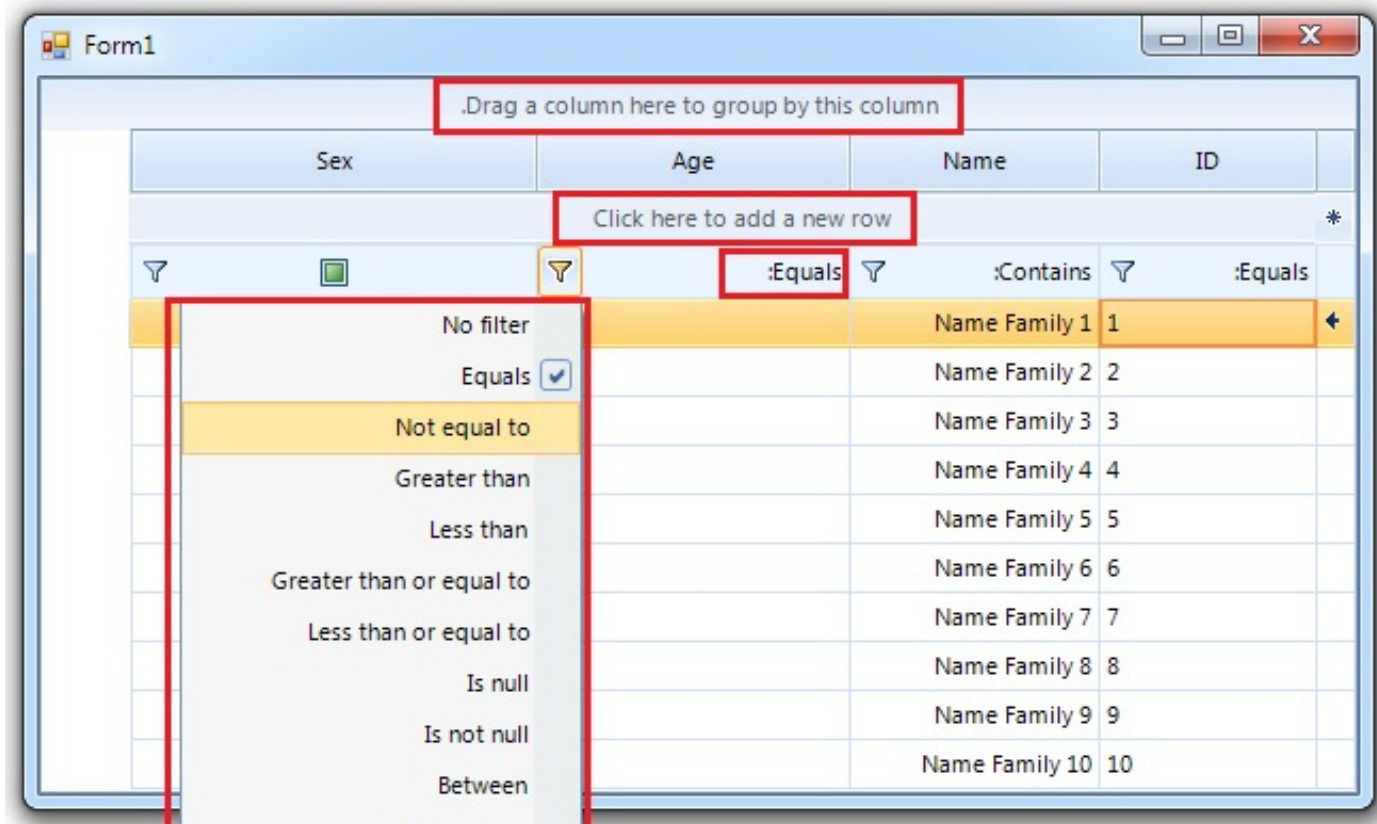
```
public List<MyCustomData> GetData() {  
    List<MyCustomData> myList = new List<MyCustomData>();  
    for (int i = 1; i < 11; i++)  
    {  
        myList.Add(new MyCustomData() {  
            ID = i,  
            Name = "Name Family " + i.ToString(),  
            Age = 29  
        });  
    }  
    return myList;  
}  
  
public class MyCustomData  
{  
    public int ID { get; set; }  
    public string Name { get; set; }  
    public int Age { get; set; }  
    public bool Sex { get; set; }  
}
```

4 - حال برای اینکه این اطلاعات را در گرید نمایش دهیم کد زیر را در بخش Load_Form1 مینویسیم :

```
private void Form1_Load(object sender, EventArgs e)  
{  
    radGridView1.DataSource = GetData();  
}
```

}

5 - در این صورت اگر برنامه را اجرا کنیم بدین صورت در گرید نمایش داده میشود که هیچ کدام از موارد فارسی نیستند:



6 - برای اینکه این موارد فارسی شوند نیاز به یک کلاس یا Provider داریم که این عمل ترجمه را انجام دهد که حتی در سایت خود [تلریک در بخش مربوطه](#) نیز ارائه شده است. بنده این کلاس را کپی کرده و تمامی ترجمه های آنها را نیز نوشتم (اگر در ترجمه ایرادی بود به بزرگی خودتان بخشیده و تصحیح نمائید .) که کد آن را در زیر میتوانید در اختیار داشته باشید:

```
class PersianRadGridLocalizationProvider : RadGridLocalizationProvider
{
    public override string GetLocalizedString(string id)
    {
        switch (id)
        {
            case RadGridStringId.FilterFunctionBetween: return "بین"; //Between
            case RadGridStringId.FilterOperatorBetween: return "بین";
            case RadGridStringId.FilterFunctionContains: return "حاوی";
            case RadGridStringId.FilterOperatorContains: return "حاوی";
            case RadGridStringId.FilterFunctionDoesNotContain: return "شامل نشود"; //Does not
            case RadGridStringId.FilterOperatorDoesNotContain: return "شامل نشود";
            case RadGridStringId.FilterFunctionEndsWith: return "پایان پذیرد با"; //Ends with
            case RadGridStringId.FilterOperatorEndsWith: return "پایان پذیرد با";
            case RadGridStringId.FilterFunctionEqualTo: return "برابر با"; //Equals
            case RadGridStringId.FilterOperatorEqualTo: return "برابر با";
            case RadGridStringId.FilterFunctionGreaterThan: return "بزرگتر از"; //Greater than
            case RadGridStringId.FilterOperatorGreaterThan: return "بزرگتر از";
            case RadGridStringId.FilterFunctionGreaterThanOrEqual: return "بزرگتر یا مساوی با";
            case RadGridStringId.FilterOperatorGreaterThanOrEqual: return "بزرگتر یا مساوی با";
            case RadGridStringId.FilterFunctionIsEmpty: return "خالی باشد"; //Is empty
            case RadGridStringId.FilterOperatorIsEmpty: return "خالی باشد";
            case RadGridStringId.FilterFunctionIsNull: return "تهی باشد"; //Is null
            case RadGridStringId.FilterOperatorIsNull: return "تهی باشد";
            case RadGridStringId.FilterFunctionLessThan: return "کمتر از"; //Less than
            case RadGridStringId.FilterOperatorLessThan: return "کمتر از";
            case RadGridStringId.FilterFunctionLessThanOrEqual: return "کمتر یا مساوی با"; //Less
        }
    }
}
```



```

than or equal to
    case RadGridStringId.FilterOperatorLessThanOrEqualTo: return "کمتر یا مساوی با";
    case RadGridStringId.FilterFunctionNoFilter: return "بدون شرط"; //No filter
    case RadGridStringId.FilterOperatorNoFilter: return "بدون شرط";
    case RadGridStringId.FilterFunctionNotBetween: return "نباشد بین"; //Not between
    case RadGridStringId.FilterOperatorNotBetween: return "نباشد بین"; //Operator
    case RadGridStringId.FilterFunctionNotEqualTo: return "برابر نباشد با"; //Not equal to
    case RadGridStringId.FilterOperatorNotEqualTo: return "برابر نباشد با";
    case RadGridStringId.FilterFunctionNotIsEmpty: return "خالی نباشد"; //Is not empty
    case RadGridStringId.FilterFunctionNotNull: return "خالی نباشد"; //Is not null
    case RadGridStringId.FilterFunctionStartsWith: return "شروع شود با"; //Starts with
    case RadGridStringId.FilterFunctionCustom: return "شرط دلخواه"; //Custom
    case RadGridStringId.CustomFilterMenuItem: return "شرط دلخواه منو"; //Custom
    case RadGridStringId.CustomFilterDialogCaption: return "انتخاب شرط دلخواه";
//RadGridView Custom Filter Dialog
    case RadGridStringId.CustomFilterDialogLabel: return "نشان دادن سطرهایی که:"; //Show
rows where:
    case RadGridStringId.CustomFilterDialogRbAnd: return "و"; //And
    case RadGridStringId.CustomFilterDialogRbOr: return "یا"; //Or
    case RadGridStringId.CustomFilterDialogBtnOk: return "تایید"; //OK
    case RadGridStringId.CustomFilterDialogBtnCancel: return "انصراف"; //Cancel
    case RadGridStringId.AddNewRowString: return "برای افزودن سطر جدید اینجا کلیک کنید";
    case RadGridStringId.ClearValueMenuItem: return "پاک کردن مقدار سلول";
    case RadGridStringId.DeleteRowMenuItem: return "حذف سطر"; //Delete Row
    case RadGridStringId.SortAscendingMenuItem: return "مرتب سازی صعودی"; //Sort
Ascending
    case RadGridStringId.SortDescendingMenuItem: return "مرتب سازی نزولی"; //Sort
Descending
    case RadGridStringId.ClearSortingMenuItem: return "حذف مرتب سازی"; //Clear Sorting
    case RadGridStringId.ConditionalFormattingMenuItem: return "قالب بندی مشروط";
//Conditional Formatting
    case RadGridStringId.GroupByThisColumnMenuItem: return "گروه بندی بر حسب این ستون";
//Group by this column
    case RadGridStringId.UngroupThisColumn: return "حذف این ستون از گروه بندی"; //Ungroup
this column
    case RadGridStringId.ColumnChooserMenuItem: return "انتخابگر ستون"; //Column Chooser
    case RadGridStringId.HideMenuItem: return "مخفی کردن ستون"; //Hide
    case RadGridStringId.UnpinMenuItem: return "حالت پیش فرض"; //Unpin
    case RadGridStringId.PinMenuItem: return "حالت ستون"; //Pin
    case RadGridStringId.PinAtLeftMenuItem: return "چسپیدن به سمت چپ";
    case RadGridStringId.PinAtRightMenuItem: return "چسپیدن به سمت راست";
    case RadGridStringId.PinAtTopMenuItem: return "چسپیدن به بالا";
    case RadGridStringId.PinAtBottomMenuItem: return "چسپیدن به پایین";
    case RadGridStringId.BestFitMenuItem: return "اندازه بهینه ستون"; //Best Fit
    case RadGridStringId.PasteMenuItem: return "چسپاندن"; //Paste
    case RadGridStringId.EditMenuItem: return "ویرایش"; //Edit
    case RadGridStringId.CopyMenuItem: return "کپی"; //Copy
    case RadGridStringId.ConditionalFormattingCaption: return "قالب بندی مشروط"; //Custom
Formatting Condition
    case RadGridStringId.ConditionalFormattingLblColumn: return "قالب بندی سلولهایی با"
Editor
    case RadGridStringId.ConditionalFormattingLblName: return "نام شرط"; //Name:
    case RadGridStringId.ConditionalFormattingLblType: return "مقدار سلول"; //Type:
    case RadGridStringId.ConditionalFormattingLblValue1: return "مقدار اول"; //Value 1:
    case RadGridStringId.ConditionalFormattingLblValue2: return "مقدار دوم"; //Value 2:
    case RadGridStringId.ConditionalFormattingGrpConditions: return "شرایط"; //Conditions
    case RadGridStringId.ConditionalFormattingGrpProperties: return "مشخصات";
//Properties
    case RadGridStringId.ConditionalFormattingChkApplyToRow: return "اعمال این شرط به کل"
    case RadGridStringId.ConditionalFormattingBtnAdd: return "افزودن شرایط"; //Add
    case RadGridStringId.ConditionalFormattingBtnRemove: return "حذف شرایط انتخابی";
//Remove
    case RadGridStringId.ConditionalFormattingBtnOK: return "تایید"; //OK
    case RadGridStringId.ConditionalFormattingBtnCancel: return "انصراف"; //Cancel
    case RadGridStringId.ConditionalFormattingBtnApply: return "اعمال قالب بندی"; //Apply
    case RadGridStringId.ColumnChooserFormCaption: return "انتخاب ستون ها"; //Column
Chooser
    case RadGridStringId.ColumnChooserFormMessage: return "برای حذف یکی از ستونها، آن ستون را به اینجا بکشید";
    case RadGridStringId.CompositeFilterFormErrorCaption: return "خطا";
    case RadGridStringId.ConditionalFormattingChooseOne: return "[یکی را انتخاب کنید]";
    case RadGridStringId.ConditionalFormattingContains: return "حاوی [مقدار اول]";
    case RadGridStringId.ConditionalFormattingDoesNotContain: return "حاوی [مقدار اول]";
    case RadGridStringId.ConditionalFormattingEndsWith: return "با [مقدار اول] پایان یابد";
    case RadGridStringId.ConditionalFormattingEqualsTo: return "برابر با [مقدار اول]";
    case RadGridStringId.ConditionalFormattingIsBetween: return "بین [مقدار اول] و [مقدار دوم]";
    case RadGridStringId.ConditionalFormattingIsGreaterThan: return "بزرگتر از [مقدار اول]";
    case RadGridStringId.ConditionalFormattingIsGreaterThanOrEqual: return "بزرگتر یا مساوی [مقدار اول]";

```



```

";با [مقدار اول
case RadGridStringId.ConditionalFormattingIsLessThan: return "[کوچکتر از] مقدار اول";
case RadGridStringId.ConditionalFormattingIsLessThanOrEqualTo: return "[کوچکتر یا مساوی] مقدار اول";
case RadGridStringId.ConditionalFormattingIsNotBetween: return "[بین مقدار اول] و [مقدار اول]";
case RadGridStringId.ConditionalFormattingIsNotEqualTo: return "[برابر با مقدار اول]";
case RadGridStringId.ConditionalFormattingRuleAppliesOn: return "[اعمال شرایط روی]";
case RadGridStringId.ConditionalFormattingStartsWith: return "[با مقدار اول] شروع";
";می شود
case RadGridStringId.CustomFilterDialogCheckBoxNot: return "[با این شرایط نباشد]";
case RadGridStringId.CustomFilterDialogFalse: return "False";
case RadGridStringId.CustomFilterDialogTrue: return "True";
case RadGridStringId.FilterCompositeNotOperator: return "[نباشد]";
case RadGridStringId.FilterLogicalOperatorAnd: return "و";
case RadGridStringId.FilterLogicalOperatorOr: return "یا";
case RadGridStringId.FilterMenuAvailableFilters: return "[فیلتر شده]";
case RadGridStringId.FilterMenuButtonCancel: return "[انصراف]";
case RadGridStringId.FilterMenuButtonOK: return "[تایید]";
case RadGridStringId.FilterMenuClearFilters: return "[پاک کردن فیلتر]";
case RadGridStringId.FilterMenuSearchBoxText: return "...جستجو";
case RadGridStringId.FilterMenuSelectionAll: return "[همه]";
//case RadGridStringId.FilterMenuSelectionAllSearched: return "[نتیجه همه جستجو]";
case RadGridStringId.FilterMenuSelectionNotNull: return "[خالی نباشد]";
case RadGridStringId.FilterMenuSelectionNull: return "[خالی باشد]";
case RadGridStringId.FilterOperatorCustom: return "[دلخواه]";
case RadGridStringId.FilterOperatorIsLike: return "[مانند]";
case RadGridStringId.FilterOperatorNotIsContainedIn: return "[نباشد در]";
case RadGridStringId.FilterOperatorNotIsEmpty: return "[خالی نباشد]";
case RadGridStringId.FilterOperatorNotIsLike: return "[نباشد شبیه]";
case RadGridStringId.FilterOperatorNotIsNull: return "[خالی نباشد]";
case RadGridStringId.FilterOperatorStartsWith: return "[شروع شود با]";
case RadGridStringId.GroupingPanelDefaultMessage: return "[برای گروه بندی ستونها، ستونی را]";
";به اینجا بکشید
case RadGridStringId.GroupingPanelHeader: return "[گروه بندی بر حسب]";
case RadGridStringId.NoDataText: return "[داده ای برای نمایش وجود ندارد]";
case RadGridStringId.UnpinRowMenuItem: return "[حالت پیش فرض]";
default:
    return base.GetLocalizedString(id);
    }
}
}

```

7 - حال اگر برنامه را اجرا کنید باز موارد انگلیسی گرید تلریک فارسی نمی شوند و باید در کلاس Program.cs پروژه این یک خط کد را هم اضافه نمائید.

```

//using Telerik.WinControls.UI.Localization;

RadGridLocalizationProvider.CurrentProvider = new PersianRadGridLocalizationProvider();

```

8 - حال اگر برنامه را اجرا نمایید تمامی موارد را فارسی مشاهده خواهید نمود (شکل ذیل)

Form1

برای گروه‌بندی ستون‌ها، ستونی را به اینجا بکشید

Sex	Age	Name	ID
برای افزودن سطر جدید اینجا کلیک کنید			
فیلتر	فیلتر	فیلتر	فیلتر
بدون شرط	برابر با:	Name Family 1	1
برابر با <input checked="" type="checkbox"/>		Name Family 2	2
برابر نباشد با		Name Family 3	3
بزرگتر از		Name Family 4	4
کمتراز		Name Family 5	5
بزرگتر یا مساوی با		Name Family 6	6
کمتراز یا مساوی با		Name Family 7	7
تهی باشد		Name Family 8	8
خالی نباشد		Name Family 9	9
بین		Name Family 10	10
نباشد بین			
شرط دلخواه			

لطفا ما را از نظرات سازنده خود بی نصیب نفرمائید. با تشکر

نظرات خوانندگان

نویسنده: ali

تاریخ: ۱۰:۴۹ ۱۳۹۲/۰۷/۱۱

امکانش هست فارسی کردن در برنامه های wpf هم توضیح بدید.

نویسنده: بهمن خلفی

تاریخ: ۱۲:۲۴ ۱۳۹۲/۰۷/۱۱

دوست عزیز

راهنمای فارسی کردن [کنترل های عمومی در سایت تلریک](#) و راهنمای [فارسی کردن گرید تلریک](#) در برنامه های wpf در دسترس می باشد.

نویسنده: میثم هوشمند

تاریخ: ۰:۳۷ ۱۳۹۲/۰۷/۱۲

با سلام

همان خط آخری که نوشتید باعث راست چین شدن متن ها شده؟
با تشکر

نویسنده: بهمن خلفی

تاریخ: ۶:۶ ۱۳۹۲/۰۷/۱۲

باسلام دوست عزیز بله خط آخری باعث شده گریدویو کلاس ایجاد شده توسط خودمان را بعنوان منبع اطلاعاتی کنونی خود در نظر بگیرد.

نویسنده: مرتضی

تاریخ: ۱۴:۲۹ ۱۳۹۲/۱۱/۱۱

درو؛ برای فارسی کردن وب ادیتور تلریک باید چکار کرد؟ من میخوام عنوان های موجود در دیالوگ هایپر لینک از وب ادیتور رو تغییر بدم.

نویسنده: بهمن خلفی

تاریخ: ۲۲:۲۴ ۱۳۹۲/۱۱/۱۱

با سلام خدمت شما

لطفا مشخص فرمائید که کنترل مورد نیاز شما در ASP.NET Web form یا ASP.NET MVC است؟ در ضمن تمامی راهنماهای مورد نیاز را میتوانید در سایت telerik.com بیابید بخصوص برای بومی سازی .
از طرفی لینک دموی کنترل مورد نظر خود را اعلام کنید تا بررسی صورت پذیرد در صورت امکان راهنمایی کامل را ارائه خواهم داد.

نویسنده: سعید

تاریخ: ۱۳:۵۲ ۱۳۹۳/۰۱/۱۷

باسلام ببخشید من در گزارش میخوام ستون های گزارش دیتا گرید تلریک از راست به چپ باشه ولی تلریک پیش فرض از چپ به راست است ستون ها را قرار میدهد. از شما خواهش میکنم راه حل این مشکل را در سایت قرار دهید. ضمناً از فارسی کردن دیتا گرید خیلی حال کردم. باتشکر

نویسنده: amirkhani

تاریخ: ۱۳۹۳/۰۲/۱۷ ۱۲:۵۳

با سلام؛ لطفا در مورد حل مشکل جستجوی ی و ک در گرید تلریک هم توضیح دهید. بنده نتونستم راه حلی براش پیدا کنم. با تشکر

نویسنده: محمدرضا رستمی
تاریخ: ۱۳۹۳/۰۵/۲۲ ۸:۱۷

سلام
جدا دستت درد نکه.
فقط اگه لطف کنی بگی چطور میشه فونت فارسی هم عوض کرد تکمیله دیگه من عوض کردم نشد

نویسنده: بهمن خلفی
تاریخ: ۱۳۹۳/۰۵/۲۲ ۱۲:۴۵

شما میتوانید بدین صورت عمل کنید :

```
font = New System.Drawing.Font("Arial", 3.0!, System.Drawing.FontStyle.Regular,  
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
```

```
Private Sub radGridView1_CellFormatting(ByVal sender As Object, ByVal e As CellFormattingEventArgs)  
    e.CellElement.Font = font  
End Sub
```

البته میتوانید از [ابزار ویژه طراحی](#) هم استفاده کنید .

نویسنده: مهرداد
تاریخ: ۱۳۹۳/۰۷/۰۸ ۱۲:۱۰

سلام.من هم مشکل فونت را دارم. فقط فونت های فیلتر (مساوی - شروع با و....) فونتش خیلی بده و نمیشه عوض کرد . میشه توضیح بدید ؟ ممنون