

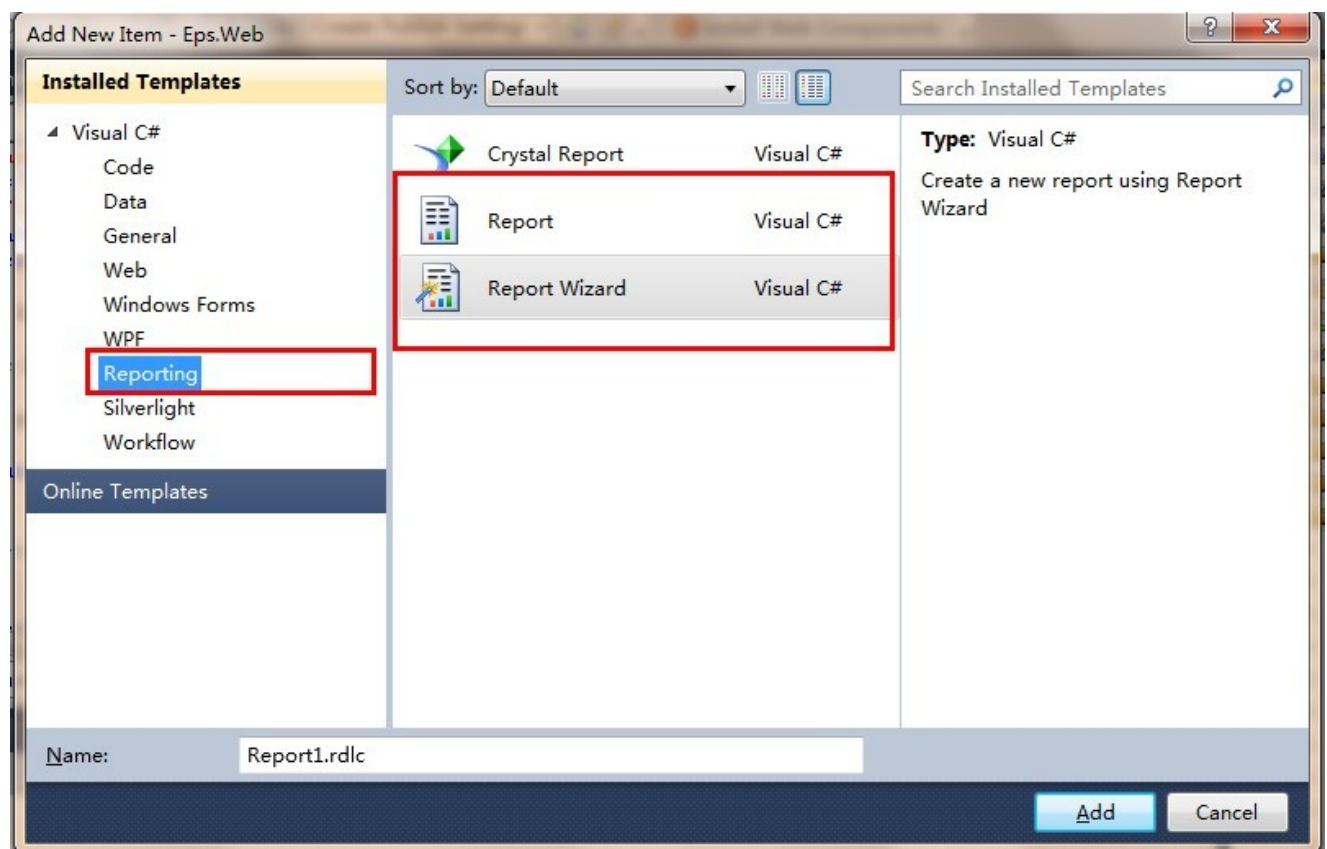
layout: post  
title: "C# RDLC报表 "  
category : C#

tags : [csharp, report]

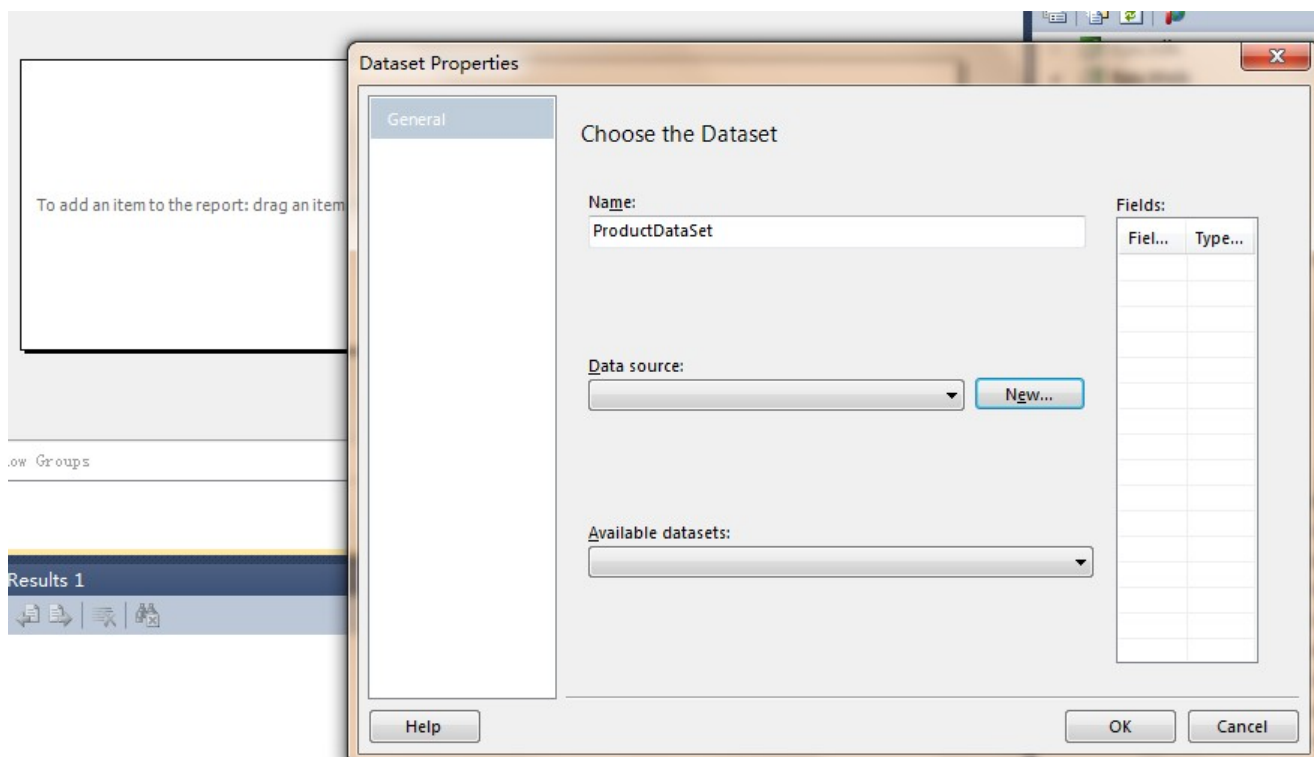
本文原发表在[ITEYE](#)上,均为本作者原创。

## 创建报表

可以通过Report或者Report Wizard创建报表



右键 Insert-->Table, 可以为插入的Table选择或创建数据源



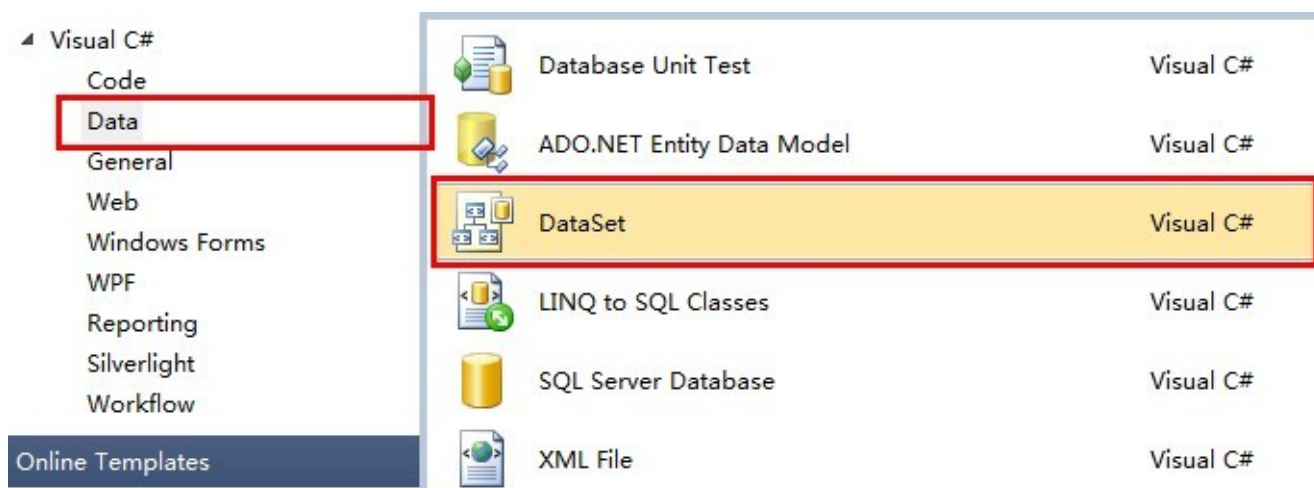
我们也可以跳过这一步，暂且生成一个空报表，动态地为报表设置数据源，关于这个在后边会提到。

关于报表的创建，可以参考

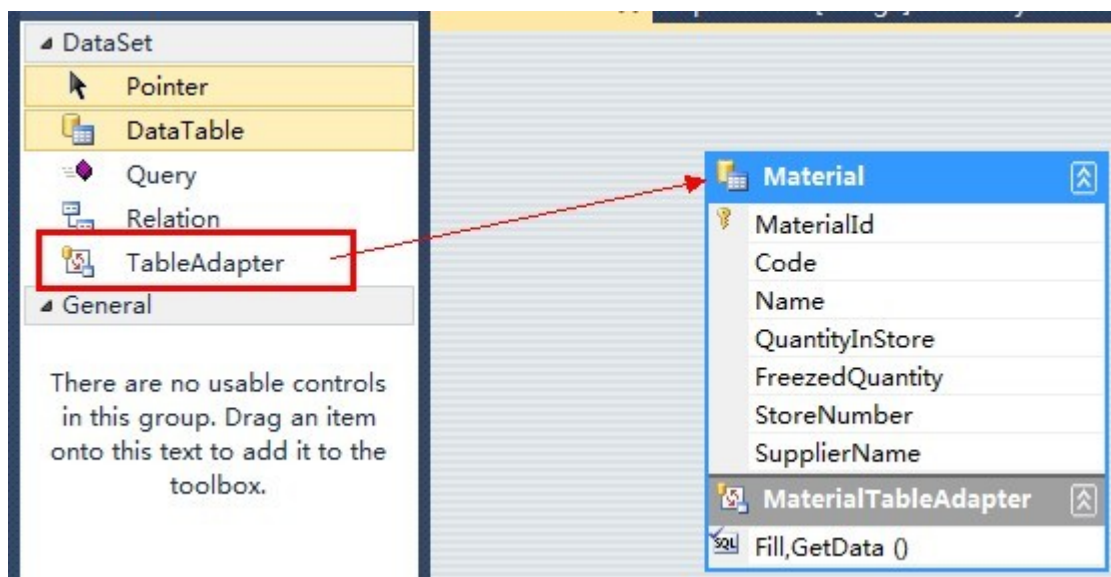
- [概述](#)
- [Walkthrough: Creating a ReportViewer Report](#)
- [RDLC 报表系列（一） 创建一个报表](#)
- [WinForm中创建Rdlc报表](#)
- [Windows Forms Control in Local Processing Mode](#)

## 报表数据源

### 添加DataSet数据源

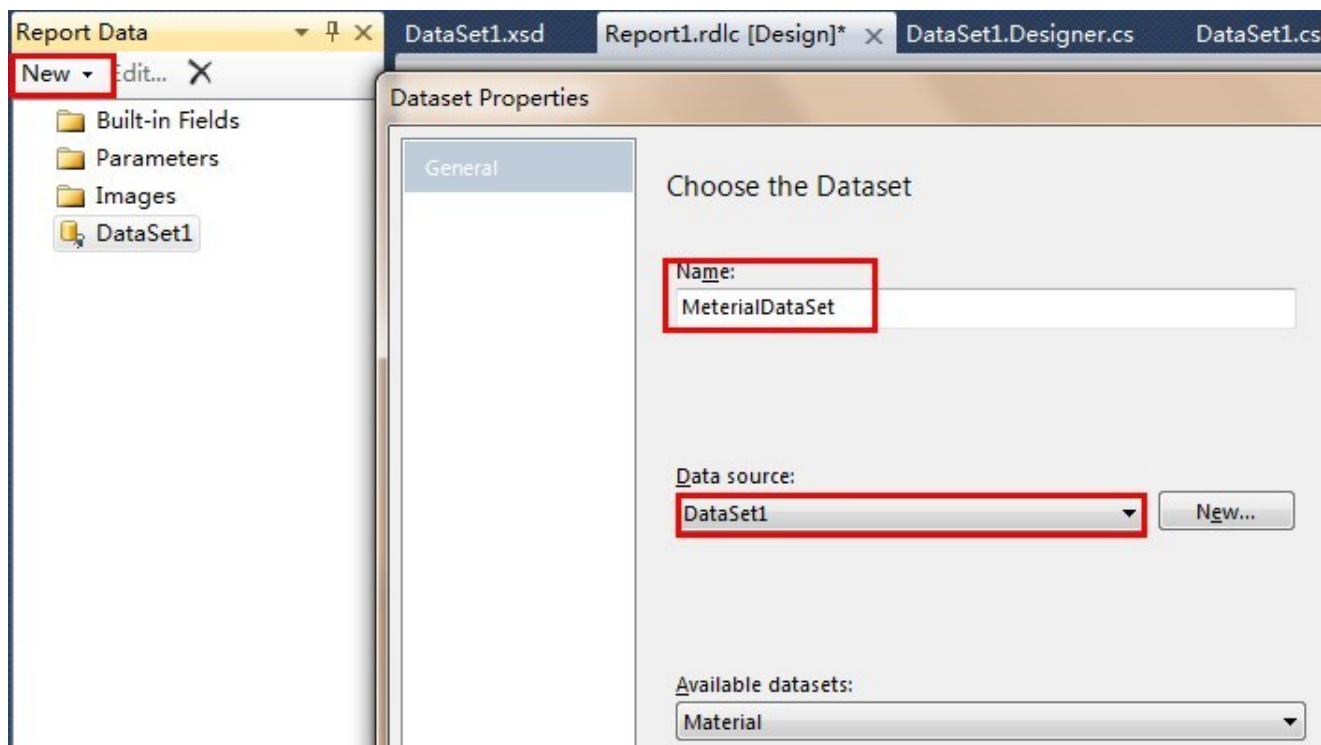


在DataSet中新建一个TableAdapter，创建第一个DataAdapter时，会提示创建数据库连接。



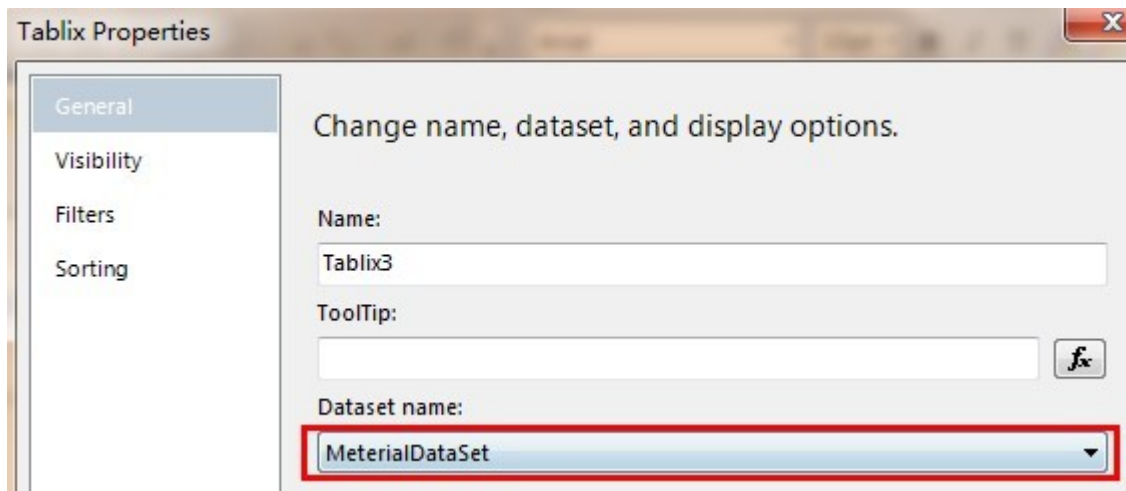
当然，我们也可以不映射数据库中的字段，直接手动添加列，之后为这些DataSet复制，可以参考[这里](#)。

## 为Rdlc报表生成数据源

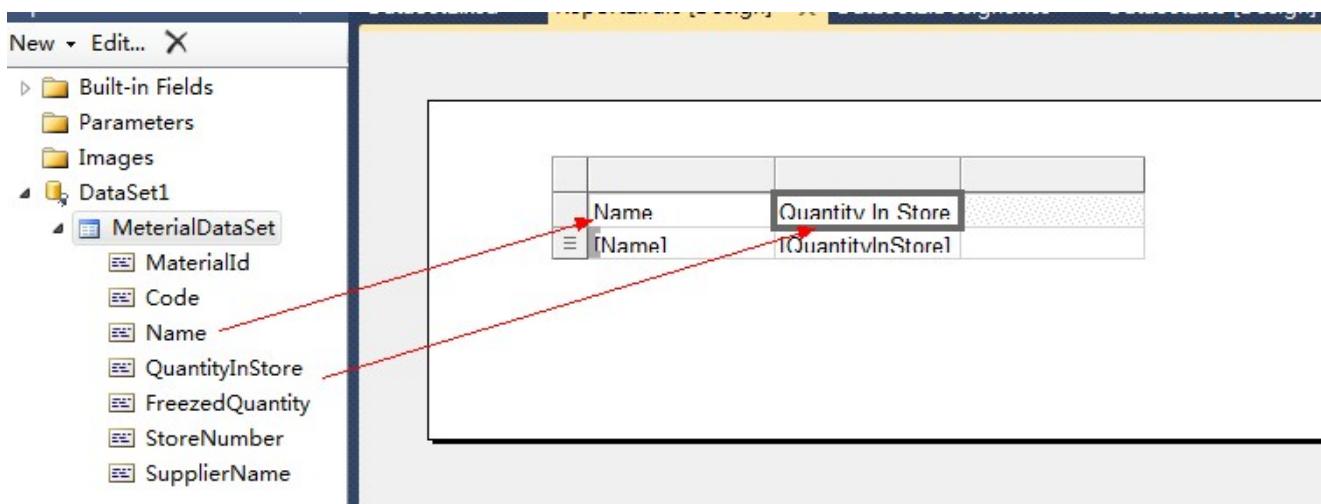


配置Rdlc的数据源，当然可以在为Rdlc创建Table时根据Wizard配置好：

右键Table-->Tablix Properties-->General-->Data Set Name



接下来就可以把DataSet中的字段往Table中拖拽或者直接编辑该列的Expression。



## 添加页眉和页脚

Report-->Add Report Header, Add Report Foo。

如果不能在每页都显示页眉或页脚，请在RDLC源码After后加上After，如：

```

<TablixRowHierarchy>
  <TablixMembers>
    <TablixMember>
      <KeepWithGroup>After</KeepWithGroup>
      <RepeatOnNewPage>true</RepeatOnNewPage>
    </TablixMember>
    <TablixMember>
      <Group Name="详细信息" />
    </TablixMember>
    <TablixMember>
      <KeepWithGroup>Before</KeepWithGroup>
    </TablixMember>
  </TablixMembers>
</TablixRowHierarchy>
<TablixRowHierarchy>
  <TablixMembers>
    <TablixMember>
      <KeepWithGroup>After</KeepWithGroup>
      <RepeatOnNewPage>true</RepeatOnNewPage>
    </TablixMember>
    <TablixMember>
      <Group Name="详细信息" />
    </TablixMember>
    <TablixMember>
      <KeepWithGroup>Before</KeepWithGroup>
    </TablixMember>
  </TablixMembers>
</TablixRowHierarchy>

```

报表的数据源还可以是某个类的方法，这个方法应该返回一个集合，可以参考[这里](#)，如：

```

/// <summary>
/// The data source.
/// </summary>
public class DataSource
{
    /// <summary>
    /// The get Materials.
    /// </summary>
    /// <returns>
    /// </returns>
    public static IEnumerable<Material> GetMaterials()
    {
        return null;
    }

    public static IEnumerable<Material> GetMaterials2()
    {
        return null;
    }
}

```

```

/// <summary>
/// The Material
/// </summary>
public class Material
{
    /// <summary>
    /// Gets or sets Name.
    /// </summary>
    public string Name { get; set; }

    /// <summary>
    /// Gets or sets Email.
    /// </summary>
    public string Email { get; set; }
}

```

```

/// <summary>
/// The data source.
/// </summary>
public class DataSource
{
    /// <summary>
    /// The get Materials.
    /// </summary>
    /// <returns>
    /// </returns>
    public static IEnumerable<Material> GetMaterials()
    {
        return null;
    }

    public static IEnumerable<Material> GetMaterials2()

```

```

    {
        return null;
    }

}

/// <summary>
/// The Meterial
/// </summary>
public class Meterial
{
    /// <summary>
    /// Gets or sets Name.
    /// </summary>
    public string Name { get; set; }

    /// <summary>
    /// Gets or sets Email.
    /// </summary>
    public string Email { get; set; }
}

```

## 在aspx页面中显示报表

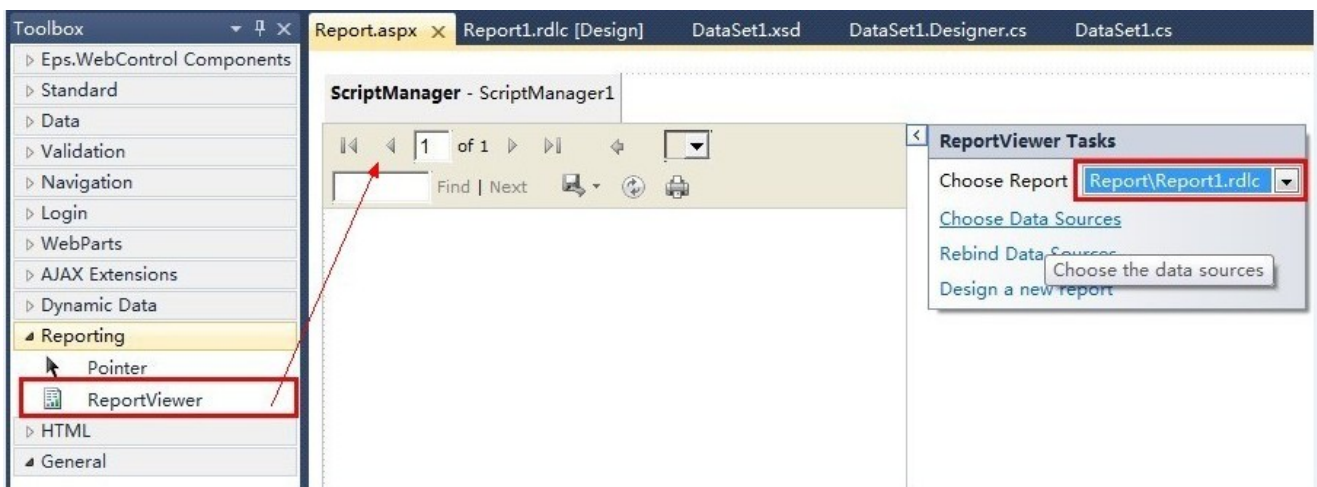
新建Aspx页面，拖拽一个ScriptManager(在Toolbox-->Ajax Extensions中)到具有runat="server"的控件之下，如

```

<form id="form1" runat="server">
    <asp:ScriptManager ID="ScriptManager1" runat="server">
    </asp:ScriptManager>
</form>
<form id="form1" runat="server">
    <asp:ScriptManager ID="ScriptManager1" runat="server">
    </asp:ScriptManager>
</form>

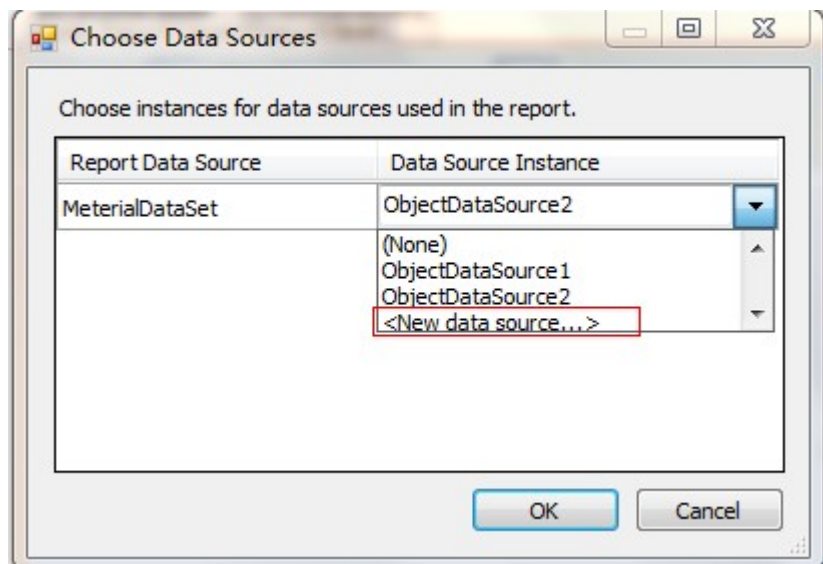
```

插入ReportView控件



为报表的数据源初始化实例

点击上图的"ReportView Tasks"-->Choose Data Source



在弹出的Wizard中，选择“Object”，输入Data Source Id， 点击OK确认，此时可以选择在“报表数据源”中创建的TableAdapter。当然，我们可以通过代码来创建数据源，代码如下：

```
ObjectResult<MaterialReturnResult> materials = DbEntities.BusinessScale();
var rdsMaterials = new ReportDataSource(
    "MaterialDataSet", materials );

localReport.DataSources.Clear();
localReport.DataSources.Add(rdsMaterials );
localReport.Refresh();

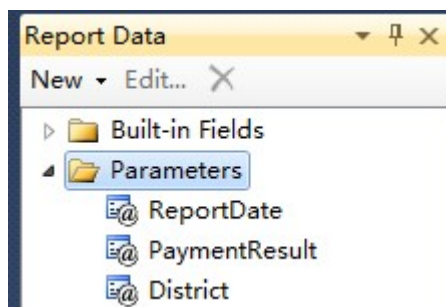
ObjectResult<MaterialReturnResult> materials = DbEntities.BusinessScale();
var rdsMaterials = new ReportDataSource(
    "MaterialDataSet", materials );

localReport.DataSources.Clear();
localReport.DataSources.Add(rdsMaterials );
localReport.Refresh();
```

我在以上代码中使用EF调用存储过程发挥的结果作为报表的数据源。

## 报表参数

报表参数的值只能通过程序来设定，在报表上无法接收参数值的输入，只能显示。通过右键“Report Data” View的“Parameters”添加参数，如：





设置报表参数的值：

```
var reportDateParameter = new ReportParameter("ReportDate", reportDate);
reportParameters.Add(reportDateParameter);
var reportDateParameter = new ReportParameter("ReportDate", reportDate);
reportParameters.Add(reportDateParameter);
```

多次添加相同参数名的报表参数，前者将会被覆盖。

## 报表变量和常量

- **ExecutionTime** 生成报表的时间
- **PageNumber** 当前的页号
- **ReportFolder** 包含报表的文件路径
- **ReportName** 报表的名称
- **ReportServerUrl** 执行报表Server的路径（这里没有用Server，所以没有，如果用ReportService那就就存在了）
- **TotalPages** 总行数
- **UserID** 当前执行报表的人
- **Language** 执行报表Server的系统语言

显示页码：

```
Globals!PageNumber.ToString() & "/" && Globals!TotalPages.ToString()
Globals!PageNumber.ToString() & "/" && Globals!TotalPages.ToString()
```

显示行号：

```
=RowNumber("MaterialDataSet")
=RowNumber("MaterialDataSet")
```

参考：[RDLC 报表系列（三） 参数、常量及常用表达式的使用](#)

## 报表表达式

回车拼接字符串：

```
=Fields!FirstName.Value & vbCrLf & Fields!LastName.Value
=Fields!FirstName.Value & vbCrLf & Fields!LastName.Value
```

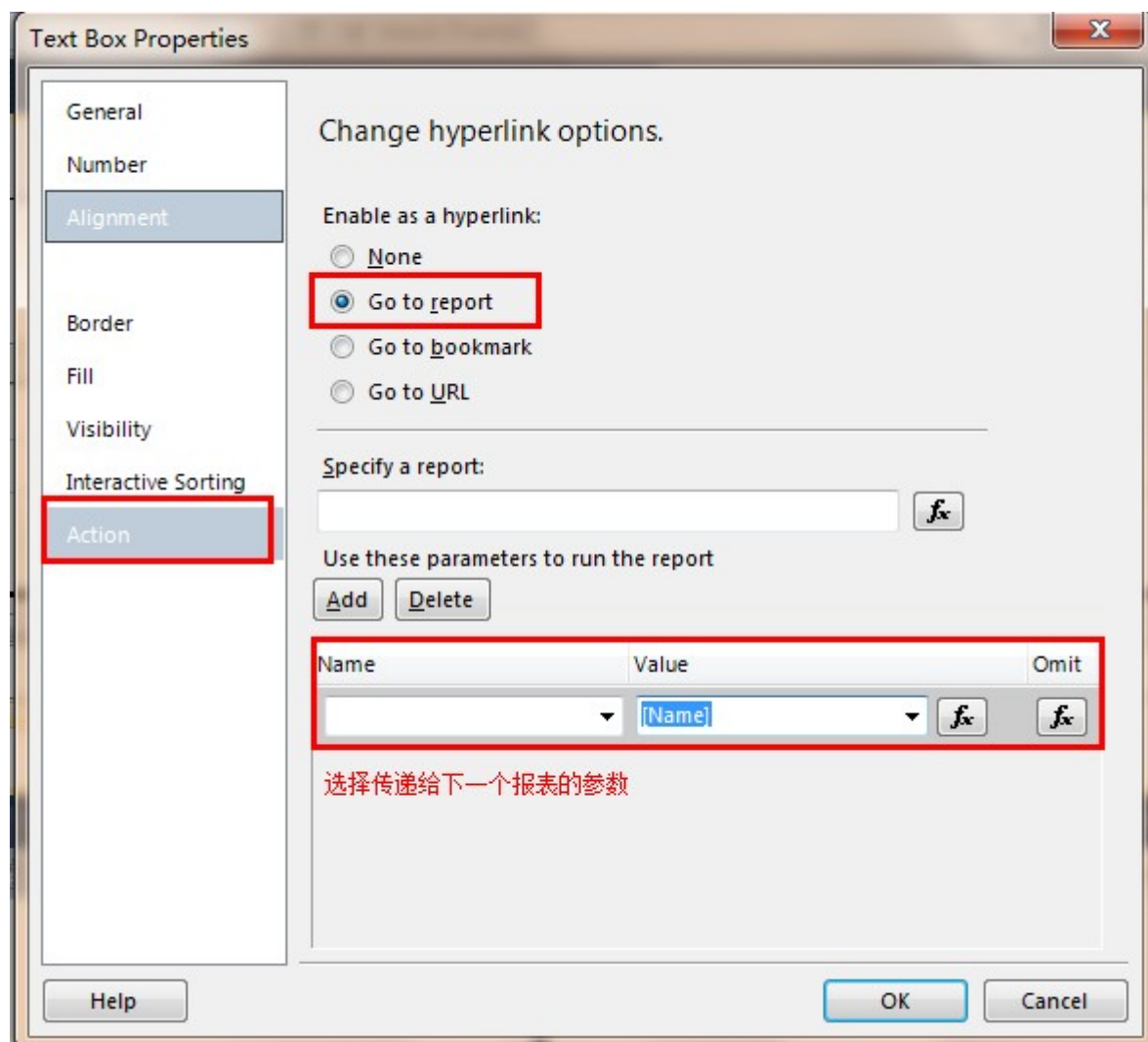
其他表达式请参考：[rdlc报表表达式应用\(字符串和转换\)](#)

可以两个小应用可以参考：

- [RDLC报表中给Matrix矩阵控件中的小计subtotal行加上背景色](#)
- [RDLC报表中Matrix矩阵控件空值替换成0](#)

## 子报表

如果希望单击报表中某个值后，弹出一个新的报表，即钻取报表或自报表，则右键报表的某个单元格-->Text Box Property-->制定钻取报表和传递给钻取报表的参数



在钻取报表中添加一个参数已接收主报表传递过来的参数，参数名和主报表传递给钻取报表的参数名相同。

捕获主报表的Drillthrough事件，实例化钻取报表的数据源，如：

```
private void reportViewer1_Drillthrough(object sender, DrillthroughEventArgs e)
{
    LocalReport lp = (LocalReport)e.Report;
    string customerid = lp.GetParameters()["customerid"].Values[0].Trim();

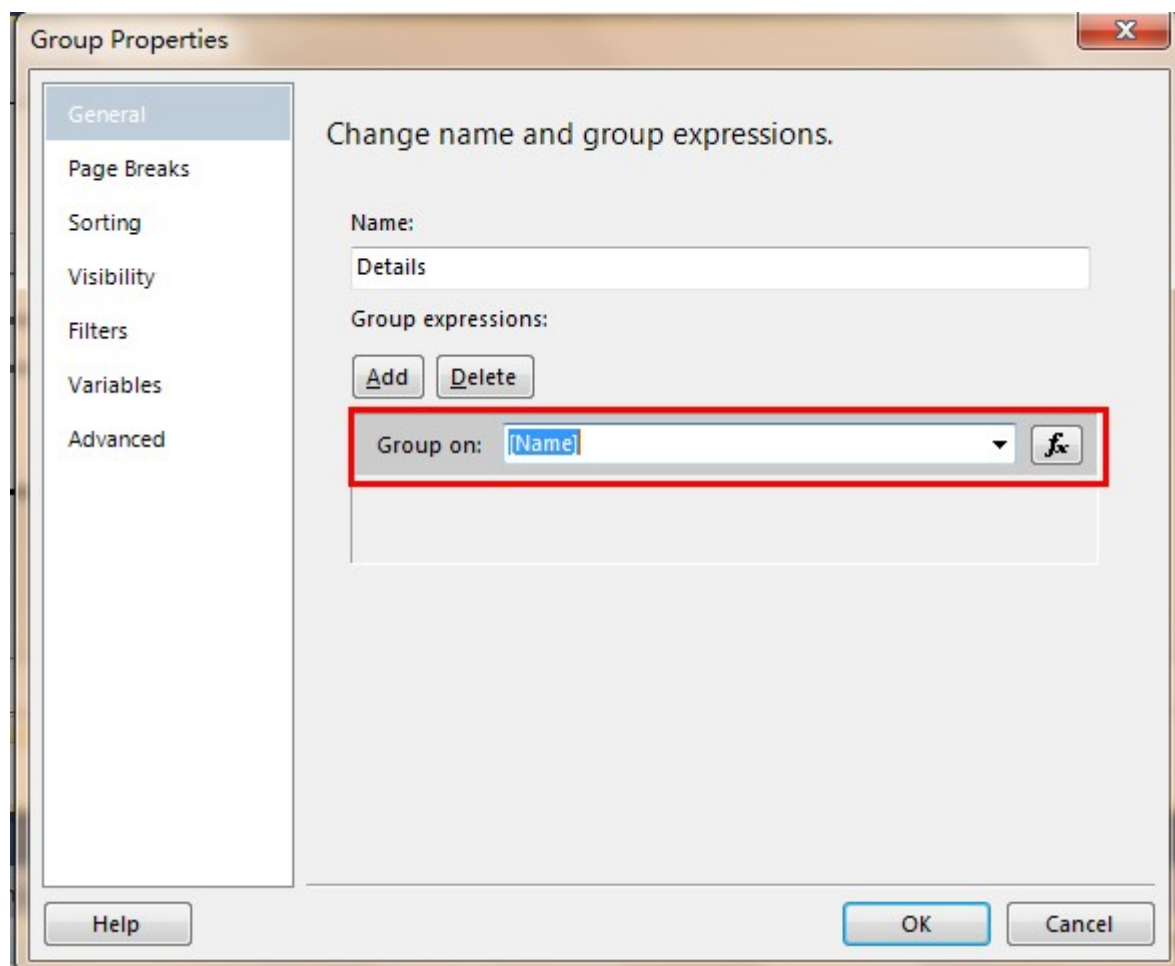
    lp.DataSources.Clear();
    lp.DataSources.Add(new ReportDataSource("NorthwindDataSet_Orders",
        new NorthwindDataSetTableAdapters.OrdersTableAdapter().GetDataByCustomerID(customerid)));
}
private void reportViewer1_Drillthrough(object sender, DrillthroughEventArgs e)
{
    LocalReport lp = (LocalReport)e.Report;
    string customerid = lp.GetParameters()["customerid"].Values[0].Trim();

    lp.DataSources.Clear();
    lp.DataSources.Add(new ReportDataSource("NorthwindDataSet_Orders",
        new NorthwindDataSetTableAdapters.OrdersTableAdapter().GetDataByCustomerID(customerid)));
}
```

详细请参考: [使用RDLC报表\(四\)-钻取式报表](#)

## 分组

对于新建的Table, 有一个默认的分組, 名为Details, 但是没有制定分组的字段, 我们可以为这个分组设置分组字段, 如:



## 在报表中插入图片

---

将图片复制到工程，右键“Report Data”视图中的Image，选择“Add”，将图片引入报表。之后便可以将图标拖到报表中，我们这里插入图片的方式为Embedded，参考[这里](#)。

## 阅读

---

- [蜡人张](#)
- [gotreportviewer](#)