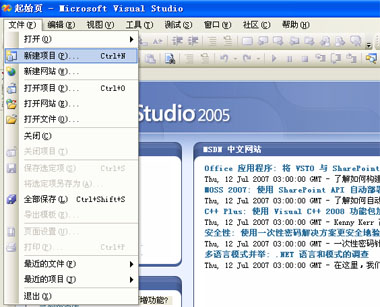
**ReportViewer教程(1)-新建 工程，添加Form和报表浏览器**

打开Visual Studio后，选择"文件"—>"新建项目"，打开新建项目对话框:



打开新建项目对话框后，在“项目类型”中选“Visual Basic"中的

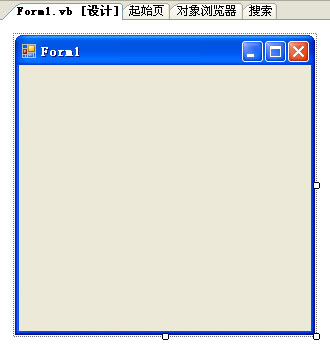
Windows:

(也可在“项目类型”中选其它语言中的“Visual C#")

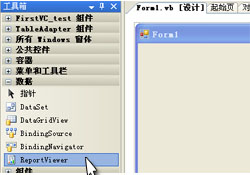
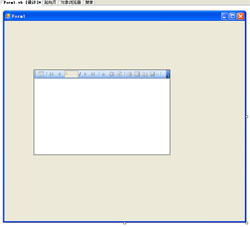




按确定后，显示Form的设计画面：



将工具箱中ReportViewer拖入至Form1中,将Form1适当的调大一点:

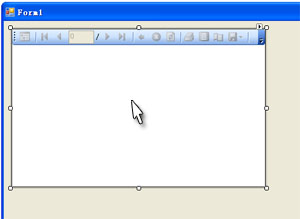
## ReportViewer教程(2)-添加 空报表Report1.rdlc

这样就在Form1上创建了一个默认名为"ReportViewer1"的控件，这个控件

就是用来浏览报表的控件。

(如果是用C#语言就创建了默认名为"reportViewer1"的控件,是第一个字母的大小写有区别）

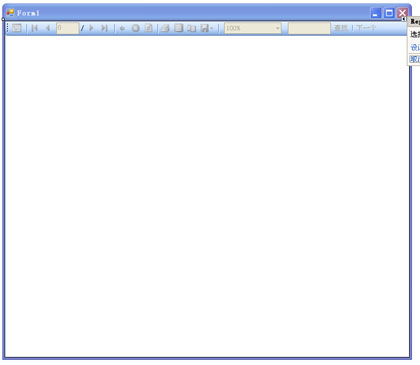
点击"ReportViewer1",在它的右上角出现了一个小三角形：



点击右上角的小三角形,点选"在父容器中停靠":

出现以下画面：

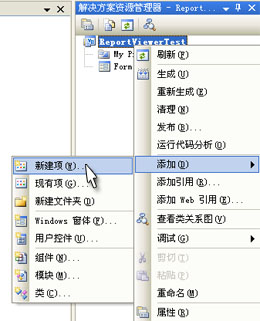


打开解决方案资源管理器(按Ctrl+Alt+L或在菜单->视图->解决方案资源管理器):

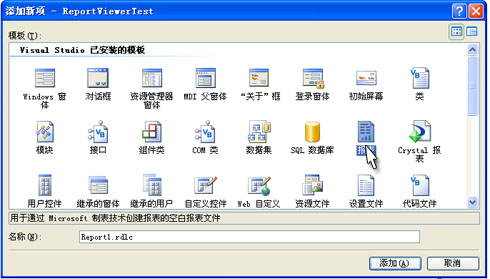


右击这个项目名称"ReportViewerTest",在出来的菜单上选：

  添加->新建项,如下图所示:



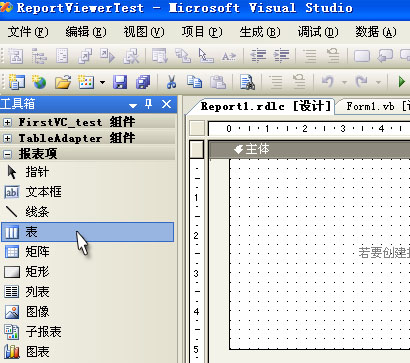
在添加新项对话框中选"报表"，名称就用默认的Report1.rdlc:



再点添加后,出现Report1.rdlc的设计画面：



选择工具箱中报表项下的"表",将"表"拖到Report1.rdlc设计窗体中:



将"表"拖到Report1.rdlc设计窗体中后，会自动生成“table1" 表,如下图：



红框以内就是"table1" 表。

## ReportViewer教程(3)-添加 空报表Report1.rdlc

在红框以内是报表的空白区：

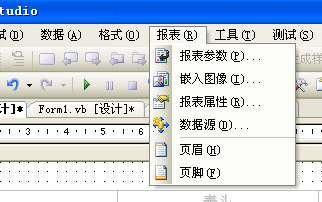


下面我们来为报表添加页眉、页脚:

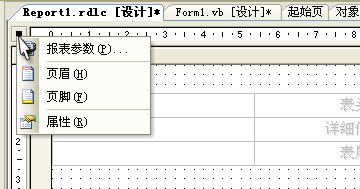
在报表空白区右击调出菜单来添加页眉、页脚:



可以点击"菜单->报表"调出菜单来添加页眉、页脚:



还可以点击下图的小方块调出菜单来添加页眉、页脚:



有了这个菜单，分别点击页眉、页脚,这样就为这个报表添加了页眉、页脚：



    在上图中有"页眉"字样的横条,我们称为页眉横条，以此类推；

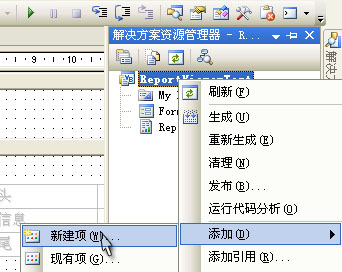
    在页眉横条下方带阵点的部分称为页眉区域，在页脚横条下方带阵点的部分称为页脚区域,

主体横条下方带阵点的部分称为主体区域，现在主体区域中还有一个"table1"表。

**ReportViewer教程(4)-为报表打造数据源**

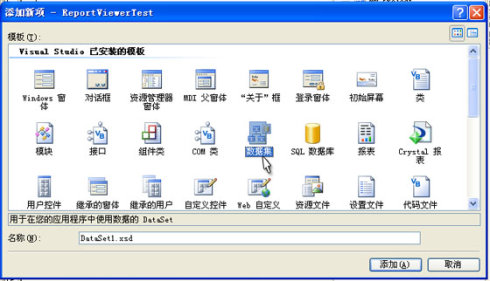
    页眉、页脚增加好了后，我们还要为报表做一个数据 源，

在解决方案资源管理器中，右击工程"ReportViewerTest"->添加->新建项：



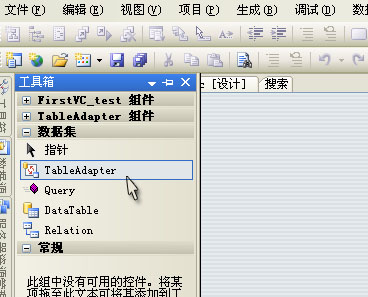
点击“新建项”后，在出来的“添加新项”对话框的模板中选“数据集"

名称就用默认的DataSet1.xsd 。



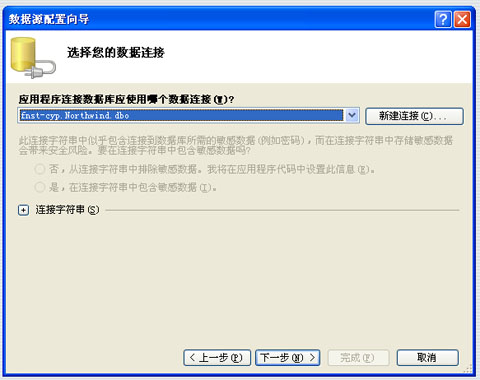
点击“添加”按钮后，出现DataSet1.xsd设计画面。

打开工具箱，将TableAdapter拖到DataSet1.xsd设计画面。



将TableAdapter拖到DataSet1.xsd设计画面后会出现数据源配置向导，

再点击"新建连接":



在出来的“添加连接”对话框中，服务器名中输入"(local)"(因为在本地安装了

Sql Server 2005),在“选择或输入一个数据库名”中下拉按钮后，显示了本地所

有数据库，我们选择"Northwind"库:

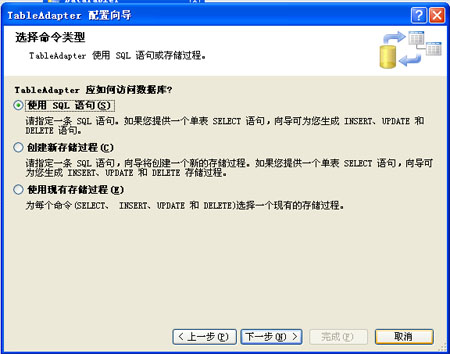


选择好Northwind库后，按确认，出现下面的画面：



在这个域中自动填入了"NorthwindConnectionString (MySettings)",

点下一步后出现下面画面，这里我们就选“使用 SQL 语句”，再点下一步：





出现上面的画面后，在其中填入下面的SQL语句:

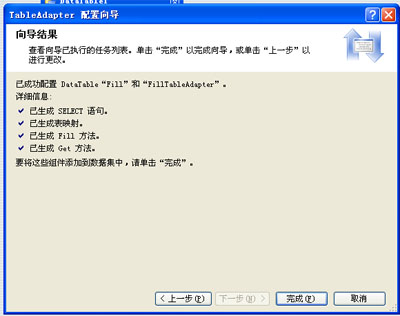
SELECT  c.CompanyName AS 公司名称, od.OrderID AS 定单号, p.ProductName AS 产品名称,          od.UnitPrice AS 单价, od.Quantity AS 数量,  
         od.UnitPrice \* od.Quantity AS 总价, Orders.OrderDate AS 定货日期  
FROM   [Order Details] AS od INNER JOIN  
       Orders ON od.OrderID = Orders.OrderID INNER JOIN  
       Customers AS c ON Orders.CustomerID = c.CustomerID INNER JOIN  
       Products AS p ON od.ProductID = p.ProductID

填入Sql语句后，点下一步，出现下面的画面:



它已经勾选了"填充 DataTable",方法名: "Fill" 和勾选了"返回 DataTalbe",

方法名称: "GetData"; 就用它的这些默认的，点下一步，显示了“向导结果”，如下面的画面：



在上面画面中点击完成，显示了下面的画面：



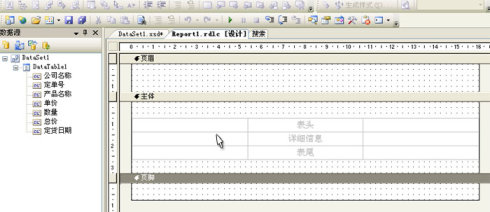
如果右面的“数据源”窗口没有显示，可按Shift+Alt+D调出来,

至此报表的“数据源”就做好了。

## ReportViewer教程(5)-将数 据源字段放入报表

如果“数据源”窗口没有显示，可按Shift+Alt+D调出来,并将DataTable1展开，

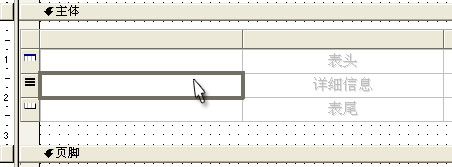
回到"Report.rdlc [设计]"窗口，



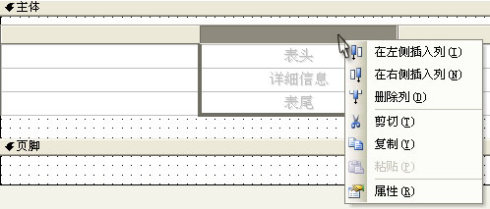
    因为数据源中有7个字段，所以我们要为"table1"表再创建4列。

如上图，在table1表中的任意一个地方点击(也选中了table1表中的一个元素）；

点击后如下图：

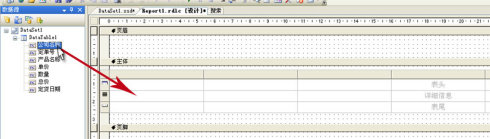


然后再右击table1表的其中一列的列头，如下图:



选择"在右侧插入列",一共插入4列，再将数据源中“公司名称”拖曳到table1表中的

第1列中间一行（详细信息行）,如红色箭头所示：



将"定单号"拖曳到table1表中的第2列中间一行（详细信息行）,

将"产品名称"拖曳到table1表中的第3列中间一行（详细信息行）,

将"单价"拖曳到table1表中的第4列中间一行（详细信息行）,

将"数量"拖曳到table1表中的第5列中间一行（详细信息行）,

将"总价"拖曳到table1表中的第6列中间一行（详细信息行）,

将"定货日期"拖曳到table1表中的第7列中间一行（详细信息行），

 拖完后如下图：



在适当调整列的宽度，光标放在列头上的列与列之间拖曳。

好了，我们已经将数据源字段放入到报表中去了。

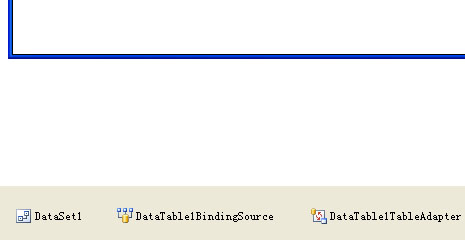
## ReportViewer教程(6)-将报表浏览器与报表联系

调出Form1.vb的设计器,并选中报表浏览器ReportViewer控件（在上面点击一下即选中),

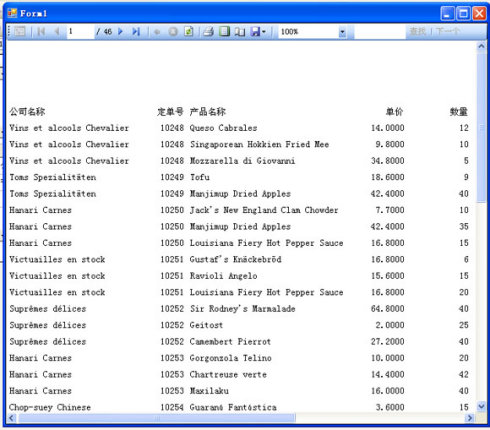
再点击ReportViewer右上角的小山角形如下图,在"选择报表中"选ReportViewerTest.Report1.rdlc：



选择ReportViewerTest.Report1.rdlc后，将自动创建DataSet1,DataTable1BindingSource,DataTable1TableAdapter 3个实例，显示在Form1.vb的设计器的下方：



现在可以运行程序了，按F5运行：



根据运行结果，适当再调整各列的列宽。

## ReportViewer教程(7)-对报表作一些调整(格式和属性）

对日期格式化，下面是程序运行后日期显示的部分，

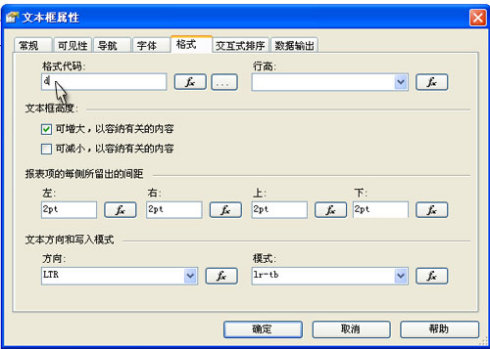


后面的小时，分，秒我们并不需要，要去掉；

调出Report1.rdlc报表设计器画面,右击定货日期这列中间行(详细信息行)：



再点属性，调出"文本框属性"对话框,在格式选项卡的格式代码中输入"d" ：



下面是(中国区域)格式代码和显示格式对照：

格式代码   显示格式

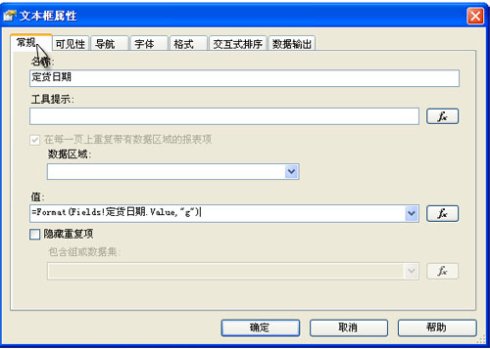
(d) Short date: . . . . . . . 2010/1/22  
(D) Long date:. . . . . . . . 2010年1月22日  
(t) Short time: . . . . . . . 9:28  
(T) Long time:. . . . . . . . 9:28:51  
(f) Full date/short time: . . 2010年1月22日 9:28  
(F) Full date/long time:. . . 2010年1月22日 9:28:51  
(g) General date/short time:. 2010/1/22 9:28  
(G) General date/long time: . 2010/1/22 9:28:51  
    (default):. . . . . . . . 2010/1/22 9:28:51 (default = 'G')  
(M) Month:. . . . . . . . . . 1月22日  
(R) RFC1123:. . . . . . . . . Fri, 22 Jan 2010 09:28:51 GMT  
(s) Sortable: . . . . . . . . 2010/01/22T09:28:51  
(u) Universal sortable: . . . 2010/01/22 09:28:51Z (invariant)  
(U) Universal sortable: . . . 2010年1月22日 1:28:51  
(Y) Year: . . . . . . . . . . 2010年1月

它符合String.Format的格式。

还可以这样格式化日期，在文本框属性中选"常规"选项卡,"值"中输入：

=Format(Fields!定货日期.Value,"g")

如下图所示：



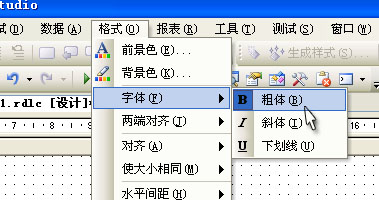
  下面再将"表头"行显示为粗体并居中：

点击"表头"行行头:



选中"表头"行行头后，再选择菜单中格式->字体->粗体,

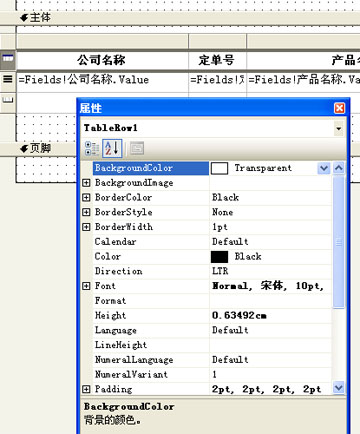
再选择菜单中格式->两端对齐->居中,



这样就调整好行头格式了。

**下面来设置显示报表的页宽页高:**

**按F4**调出"属性页"窗口，

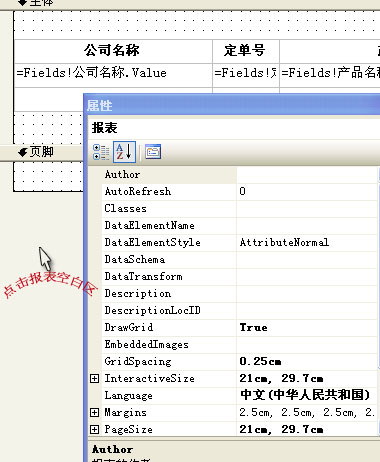


    因为刚才是选中的"表头"行行头,所以调出的属性页是

"表头"行行头（TableRow1)属性页，在报表空白区(关于

报表空白区见ReportViewer教程(3)-添加空报表 Report1.rdlc ）

点一下即换成的报表属性页如下图：



在“报表”的属性页中展开InteractiveSize，

InteractiveSize的Height决定了打印的行数，

显示出来的行数=InteractiveSize.Height/TableRow2.Height

假如要显示25行，根据公式：

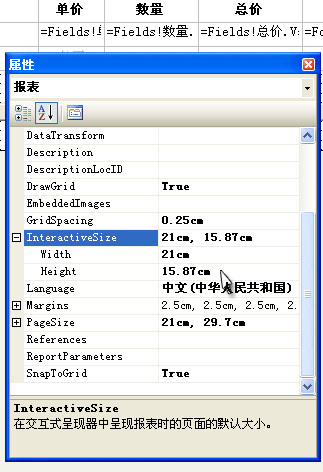
InteractiveSize.Height=25 \* 0.63492

得出15.87,

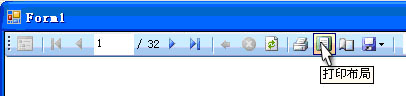
我们把Interactive.Height设 成15.87

## ReportViewer教程(8)-对报表作一些调整(格式和属性)

把Interactive.Height设成15.87后如下图:



**设打印布局中行宽高：**

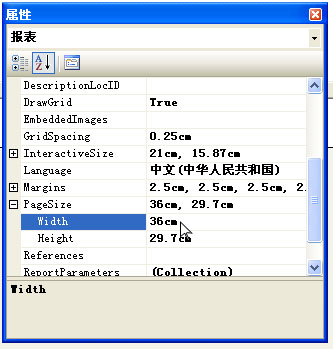
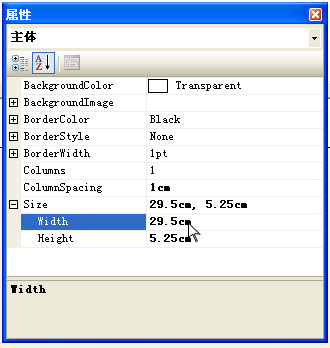
****

   在运行程序后，导航条上有个打印布局（见上图）可能不正常，需要设置一下,

   报表属性中的PageSize决定了打印布局中的行宽高：

注意：报表属性中的PageSize.Width-margins\*2 >=主体属性中Size.Width

      否则在显示打印布局时会将主体中没有被显示出来的页面变成第2页显示出来。

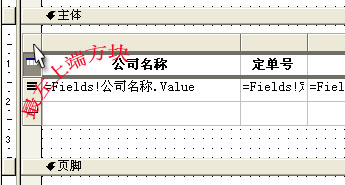


根据公式，我们将报表中PageSize的宽设成36，这样打印布局就正常了。

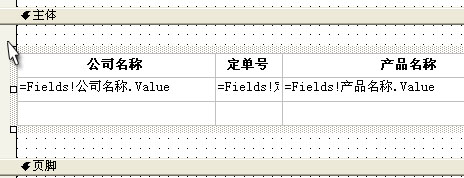
   再将报表网格化并将“table1 表”向右移动到主体中间,以免显示出来太靠右，

先要选中"table1 表",要选中 "table1 表",点击"table1 表"的最左面上端方块，

如下图(也可用F4主调出属性页,在其中选择“table1 表”）：



选中"table1 表"后显示如下：

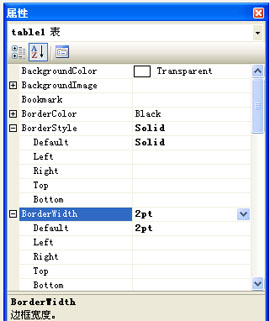


   可以看到"table1 表"被虚框包围着，这就代表"table1 表"被选中了。

   我们可以将鼠标对准虚框，就可以拖动"table1 表"了，也可以上下移动键来

移动"table1 表"。我们将"table1 表"向右面移动到在主体上水平居中的样子，

按F4调出"table1 表"的属性页：



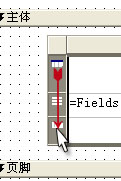
将BorderStyle设成"Solid",将BorderWidth设成"2pt",

再在"table1 表"内部点击一下，选中"table1 表"中的元素，

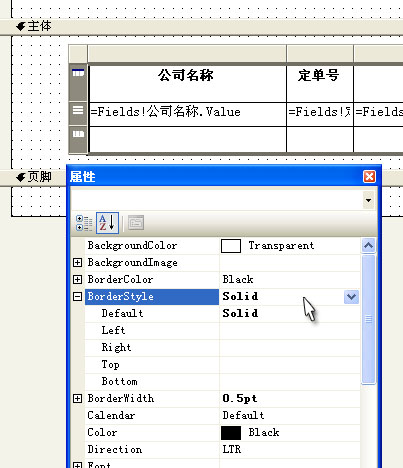
如图：



用鼠标在行头拖动,下图中红色箭头经过的地方表示鼠标拖动路线:



经过鼠标拖动表示选中了所有这3行,调出这3行的属性页，



将BorderStyle 设成"Solid",BorderWidth设成0.5,

按F5运行程，如下图：



## ReportViewer教程(9)-给报表增加页打印日期编号

**增加报表页和打印日期：**

从工具箱拖曳两个文本框到页眉区域：



在两个文本框中分别输入（见上图)：

="第" & Globals.PageNumber & "页，共" & Globals.TotalPages & "页"

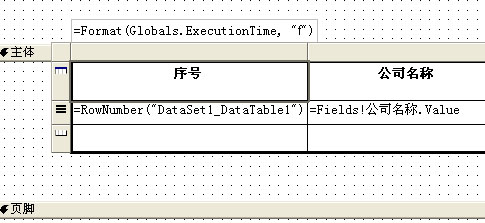
=Format(Globals.ExecutionTime, "f")

调整它们的大小并将其拖到适当的位置：



在“公司名称”列左边插入一列，在表头行输入“序号”，详细信息行输

入: =RowNumber("DataSet1\_DataTable1")   如下图：

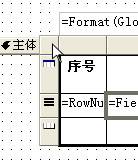


调整“序号”列的宽度。

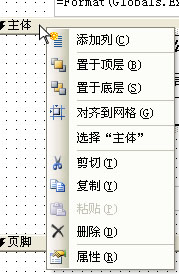
## ReportViewer教程(10)-给报表分组合计

选择主体内的"table1"表（点击它一下即选中),

再右击它的最左上角方块：



右击后调出菜单，如下图：

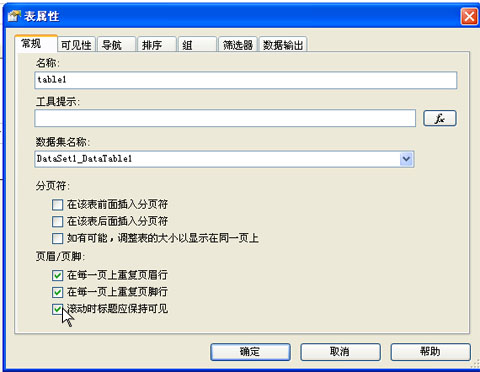


选择菜单上“属性”，调出“表属性”对话框：

勾选“常规”选项卡中：

    在每一页上重复页眉行，在每一页上重复页脚行，滚动时标题应保持可见

  如下图:

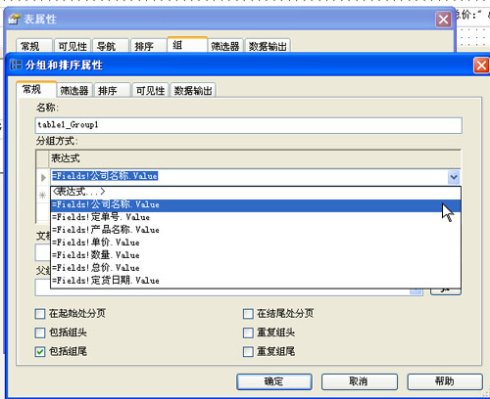


    再在“表属性”对话框上选“组”选项卡并点击添加，调出“分组和排序属性”对话框，

名称默认为table1\_Group1,勾选"包括组尾",再在"表达式"中下拉菜单中选择：

=Fields！公司名称.Value

这将按公司名称进行分组，如下图：



再选择"分组和排序属性"对话框的排序选项卡，在下拉菜单中选择:

=Fields！公司名称.Value

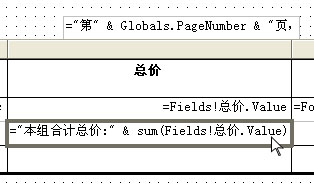


选择好后按确定，关闭“分组和排序属性”对话框，

再在“表属性”对话框上按确定,关闭“表属性”对话框。

在总价那一列，新出来的一行（组尾)中输入：

="本组合计总价:" & sum(Fields!总价.Value)



**页汇总，全部汇总：**

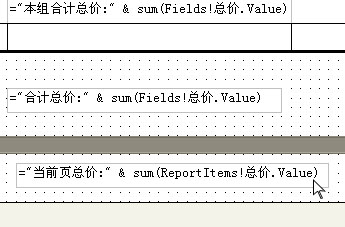
   从工具箱中拖出两个文本框分别到主体内的table1表下（在页脚横条以上）

和页脚区内（页脚横条以下）,在两个文本框内分别输入：

 ="合计总价:" & sum(Fields!总价.Value)

 ="当前页总价:" & sum(ReportItems!总价.Value)

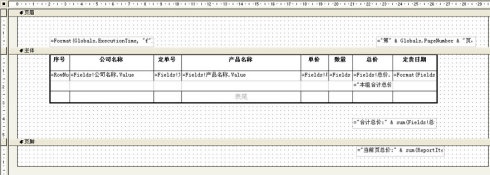
  如下图：



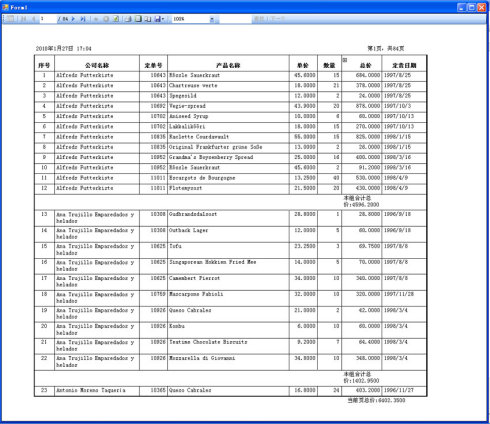
再选中“talbe1”表，右击表尾那一行行头,将那行的属性页调出来，

将属性中Visibility中Hidden设成True,让它不可见。

全部设好后如图：



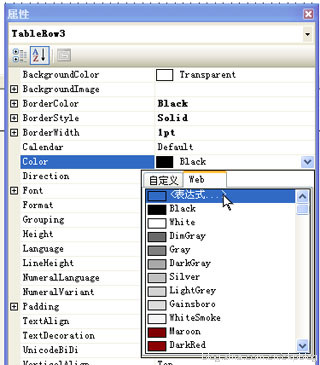
按F5运行后如图：



## ReportViewer教程(11)-给报表特定的内容设置颜色

假如要对报表中某些记录的字体设颜色，例如对总价小于100的这一行字体设红色，总价大于2000的设水鸭色。

由于是要对一行字体设颜色，所以要选中一行，点击“table1”表的详细行行头，如下图：  
  
选中后，按F4调出这一行的属性页，点击属性中“Color”中的表达式：

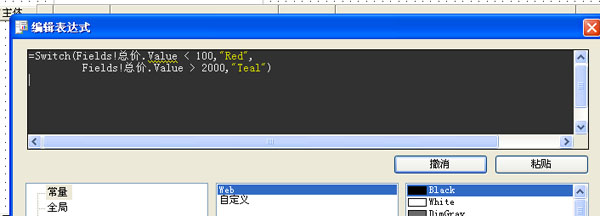


点击表达式后出来“编辑表达式”对话框：

在出来的“编辑表达式”对话框中输入：

=Switch(Fields! 总价.Value < 100,"Red",

         Fields!总价.Value > 2000,"Teal")



或输入:

=Iif(Fields! 总价.Value < 100,"Red",

     Iif(Fields! 总价.Value > 2000,"Teal","Black"))

点击确定后就完成了。

**ReportViewer教程(12)-按组折叠**

我们已经将每个公司为一组排列显示了报表。

现在将每个公司折叠起来显示(如下图）：

  
  
步骤如下：

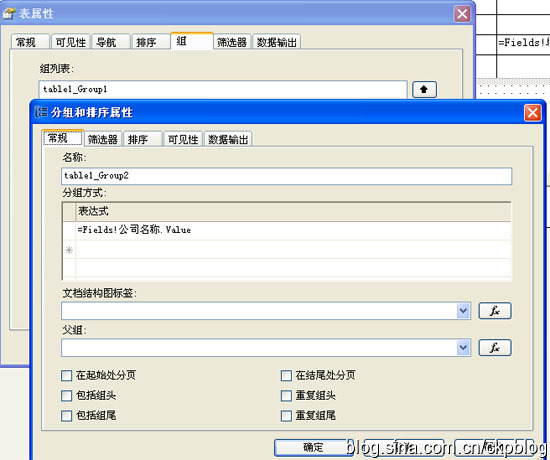
   在“table1\_Group1"组头那行、总价那一列的单元格中输入:

    =sum(Fields!总价.Value)



**添加组2：**

  按 ReportViewer教程(10) 所示的方法调出 “表属性”：

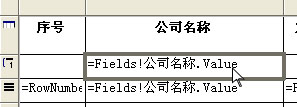
  
  
在“表属性”中选分组选项卡，点添加后调出“分组和排序属性”对话框，

要添加的组名称为"table1\_Group2",表达式 为"=Fields!公司名称.Value",其它不选，

再在该对话框中点"可见性"选项卡，如下图：

  
  
在"可见性"选项卡中，选“隐藏”和勾选“可由其他关联报表项确定可见性”，

下拉“报表项”，选择“textbox29",按确定。

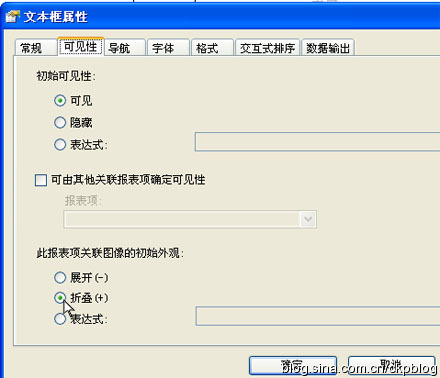


这里的“textbox29"是指 “公司名称”

那一列、“table1\_Broup1"组头那一行的单元格的名称，如上图。

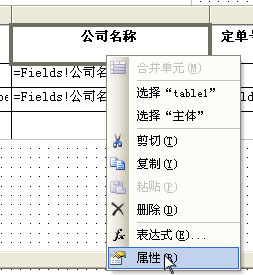
再右击这个单元格（"textbox29"),调出该单元格的"文本框属性"对话框,

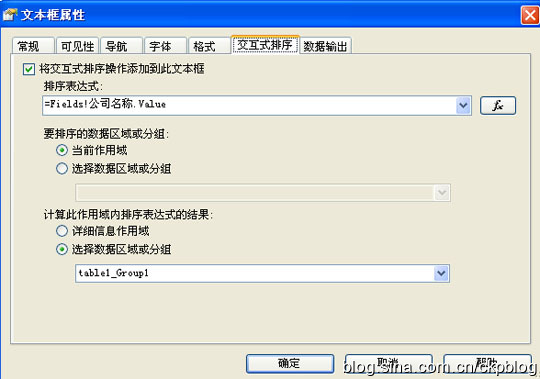
选“可见性”选项卡如下图：

  
  
因为在“table1\_Group2"中的可见性是隐藏的，所以这里选"折叠"（反之选"展开")。

按确定后就OK了。

**ReportViewer教程(13)-交互式排序**

  
  
右击“公司名称”单元格，选“属性”，如上图。

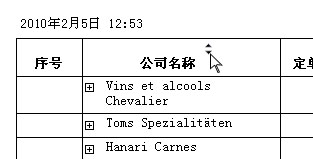
  
在调出的文本框属性中选择“交互式排序”，排序表达式为：

=Fields!公司名称.Value

选择数据区域或分组为：

table1\_Group1

点击确定。程序运行后，我们看到了报表中的可点击的排序标记：



**ReportViewer教程(14)-钻取报表和传入参数**

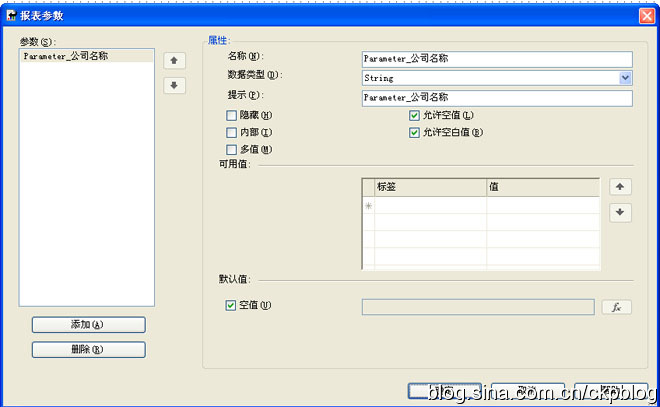
   当我们在前面做的报表上点击某一公司名称时，钻取了另一张报表并传递以该公司名称为内容的参数，这张表显示了该公司的详细资料。

我们按照  **ReportViewer教程(2)-添加空报表Report1.rdlc**和**ReportViewer 教程(3)-添加空报表Report1.rdlc**的方法添加一个名为Report2.rdlc的新报 表，从已有的数据源中拖入相应的字段到报表中。



在报表的空白处右击调出菜单，



选“报表参数”，调出“报表参数”对话框：  


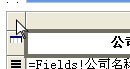
在名称和提示中分别输入“Parameter\_公司名称”，按确定。

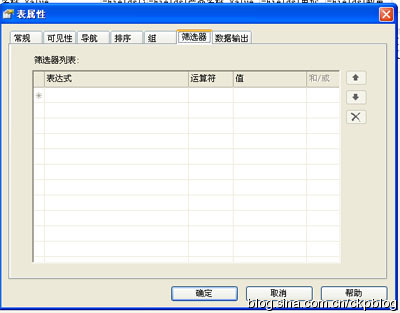
    从工具箱中拖入一个文本框到“table1”表的上方，在其中输入：

=Parameters!Parameter\_公司名称.Value & " 详细信息"

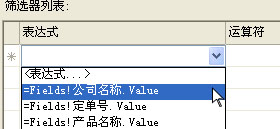
做好的表如下图：

  
  
右击“table1”表的右上角（如下图），选“属性”，

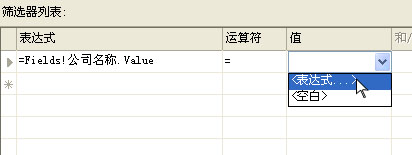


调出表属性对话框（如下图），再选"筛选器"选项卡  


在表达式的第一行中下拉后选:"=Fields!公司名称.Value"一项，如下图：

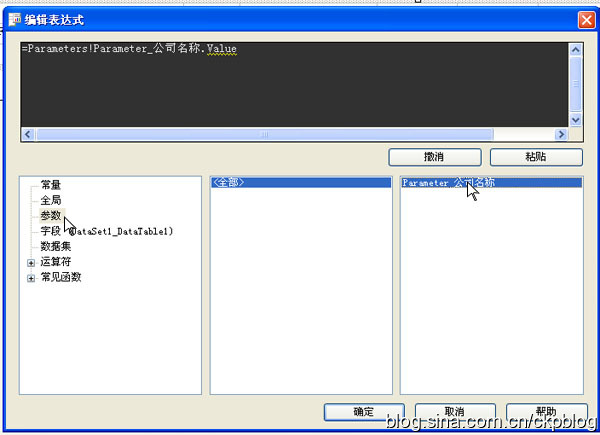


在"值"的第一行下拉后选"表达式",如下图：



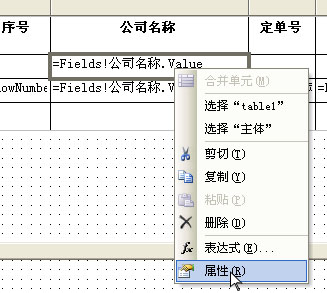
在出来的“编辑表达式”对话框中选参数，再双击参数“Parameter\_公司名称”参数，这样参数就自动填为：=Parameters!Parameter\_公司名称

如下图：

  
  
再按确定，退出“编辑表达式”对话框。再按确定，退出表属性对话框。

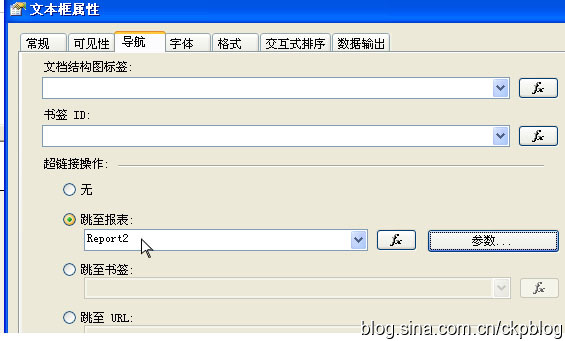
这时Report2.rdlc报表就做好了。

再调出Report1.rdlc报表设计画面，

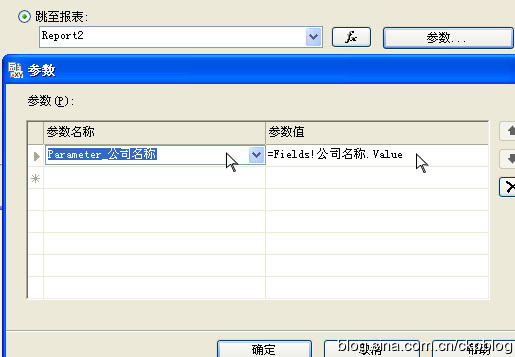


右击公司名称下面一个单元格，点属性，调出该单元格的“文本框属性”，

选“导航”选项卡，再选“跳至报表”，下拉后有“Report2”,将它选中：如下图：



再点击“跳至报表”右面的参数按钮：



在参数对话框中选 参数名称为：Parameter\_公司名称

参数值为：=Fields!公司名称.Value

按确定。再按确定退出该单元格的“文本框属性”对话框。

这样Report1.rdlc报表也就设置好了。

**二个表设好后还要输入一些引发钻取事件的代码。**

  调出Form1.vb代码编辑画面，添加一个钻取事件的方法“名为DemoDrillthroughEventHandler”：

如果是 .net VB ,添加代码如下：

    Private Sub DemoDrillthroughEventHandler(ByVal sender As System.Object, ByVal e As DrillthroughEventArgs)

        Dim myLocalReport As LocalReport  
        myLocalReport = e.Report  
        myLocalReport.DataSources.Add(New ReportDataSource("DataSet1\_DataTable1", Me.DataSet1.DataTable1))

    End Sub

在再Form1\_Load中添加一行代码：

**AddHandler ReportViewer1.Drillthrough, AddressOf DemoDrillthroughEventHandler**

如果是 .net C# ,添加代码如下：

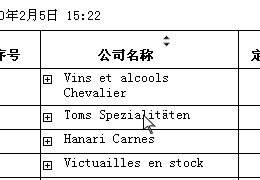
要有命名空间：using Microsoft.Reporting.WinForms;

    void DemoDrillthroughEventHandler(object sender, DrillthroughEventArgs e)  
    {  
        LocalReport localReport = (LocalReport)e.Report;  
        localReport.DataSources.Add(new ReportDataSource ("DataSet1\_DataTable1",this.DataSet1.DataTable1));  
    }

 在再Form1\_Load中添加一行代码：

 reportViewer1.Drillthrough += new DrillthroughEventHandler(DemoDrillthroughEventHandler);

代码添加完后运行程序：

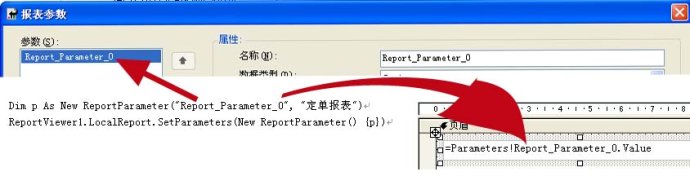
  
  
点击某一公司的名称（如上图），得到如下图的报表：



一切都OK了。

  上面是从一个报表向另一个报表传送参数，如果从程序的代码中向报表传送参数，请按下面的图示来做。

 下图说明了用.net vb 代码向Report1.rdlc的报表传送内容为“定单报表”的参数：



 如果是 .net C# 的话，用下面的代码：

  ReportParameter p = new ReportParameter("Report\_Parameter\_0", "定单报表");

  this.reportViewer1.LocalReport.SetParameters(new ReportParameter[]{p});