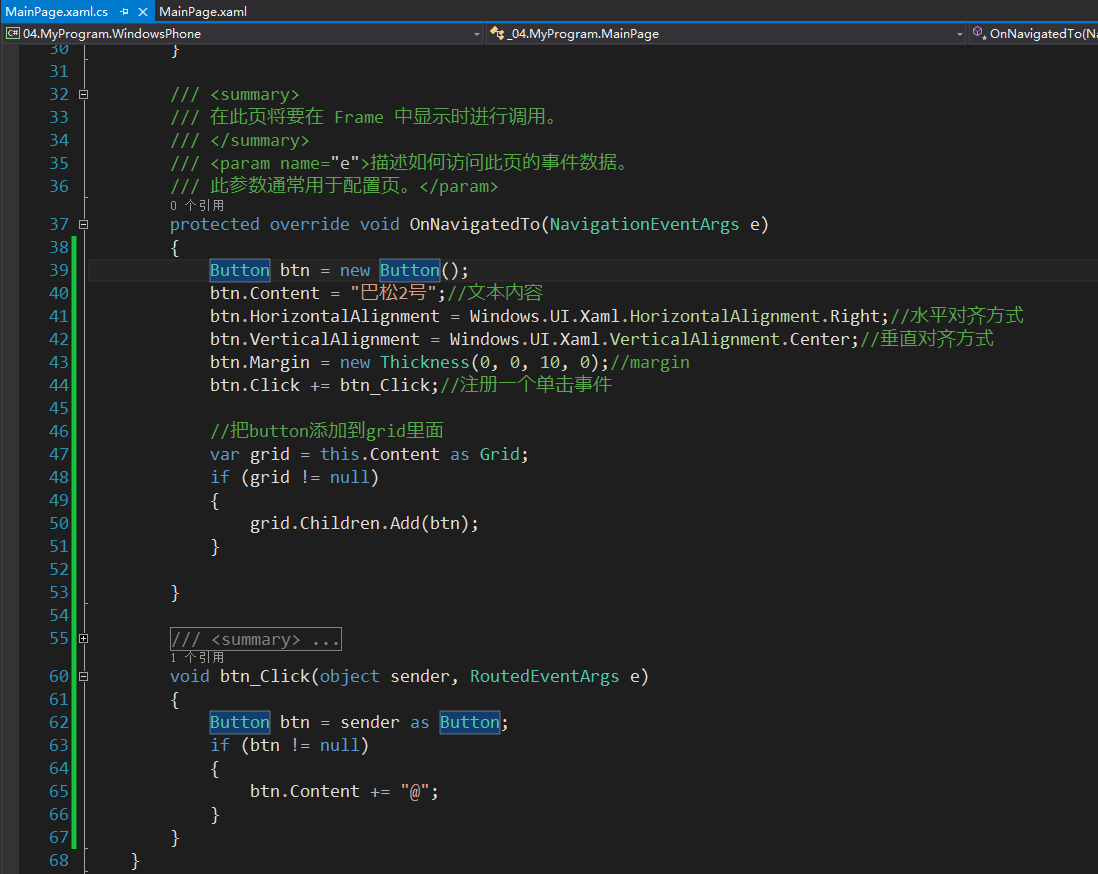
[06.移动先行之谁主沉浮----我的代码我来写（Xaml的优势）](http://www.cnblogs.com/dunitian/p/4876893.html)

如果移动方向有任何问题请参考===> [**异常处理汇总-移动系列（点）**](http://www.cnblogs.com/dunitian/p/4867701.html)

前面几节课，我们都是在前台创建对象，进行一些设置，那么我们为什么不用传统的方法来编程呢？

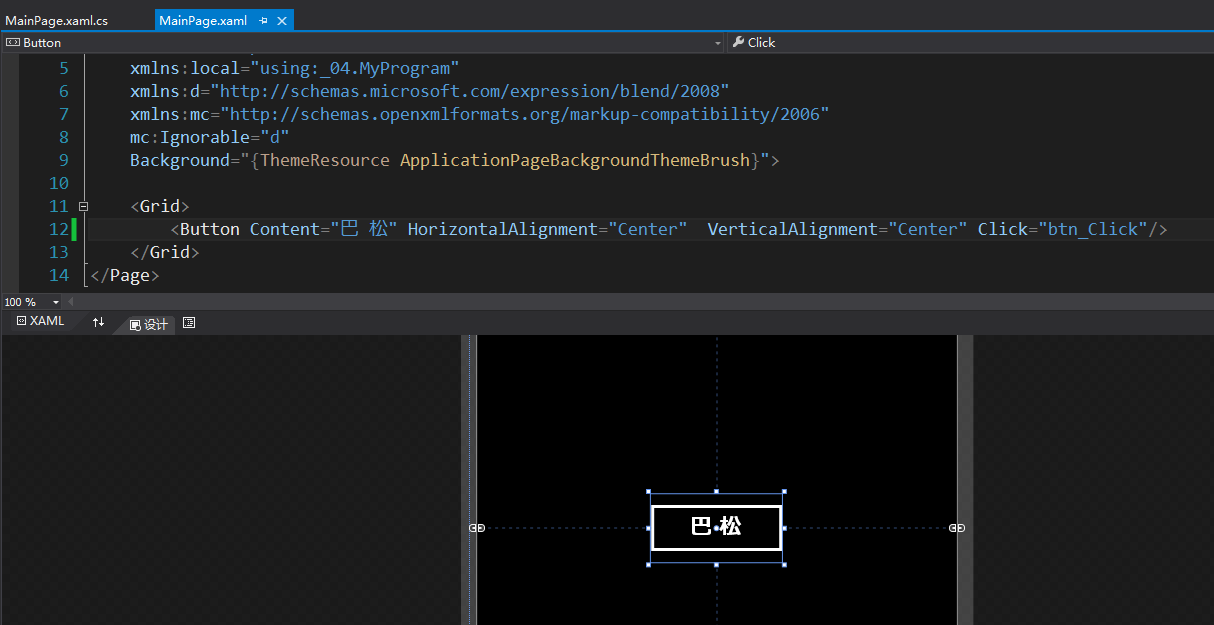
我们今天来试试你就明了了~~

打开MainPage.xaml.cs，自己创建一个button，进行一些常用设置



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36 | /// <summary>     /// 在此页将要在 Frame 中显示时进行调用。     /// </summary>     /// <param name="e">描述如何访问此页的事件数据。     /// 此参数通常用于配置页。</param>     protected override void OnNavigatedTo(NavigationEventArgs e)     {         Button btn = new Button();         btn.Content = "巴松2号";//文本内容         btn.HorizontalAlignment = Windows.UI.Xaml.HorizontalAlignment.Right;//水平对齐方式         btn.VerticalAlignment = Windows.UI.Xaml.VerticalAlignment.Center;//垂直对齐方式         btn.Margin = new Thickness(0, 0, 10, 0);//margin         btn.Click += btn\_Click;//注册一个单击事件           //把button添加到grid里面         var grid = this.Content as Grid;         if (grid != null)         {             grid.Children.Add(btn);         }       }       /// <summary>     /// 单击事件     /// </summary>     /// <param name="sender"></param>     /// <param name="e"></param>     void btn\_Click(object sender, RoutedEventArgs e)     {         Button btn = sender as Button;         if (btn != null)         {             btn.Content += "@";         }     } |

我们再用xaml进行另一个按钮的创建



|  |  |
| --- | --- |
| 1  2  3 | <Grid>      <Button Content="巴 松" HorizontalAlignment="Center"  VerticalAlignment="Center" Click="btn\_Click"/>  </Grid> |

繁易程度一目了然

效果如下：（我们单击事件是用的同一个事件代理函数）



可见Xaml的优势~~~~