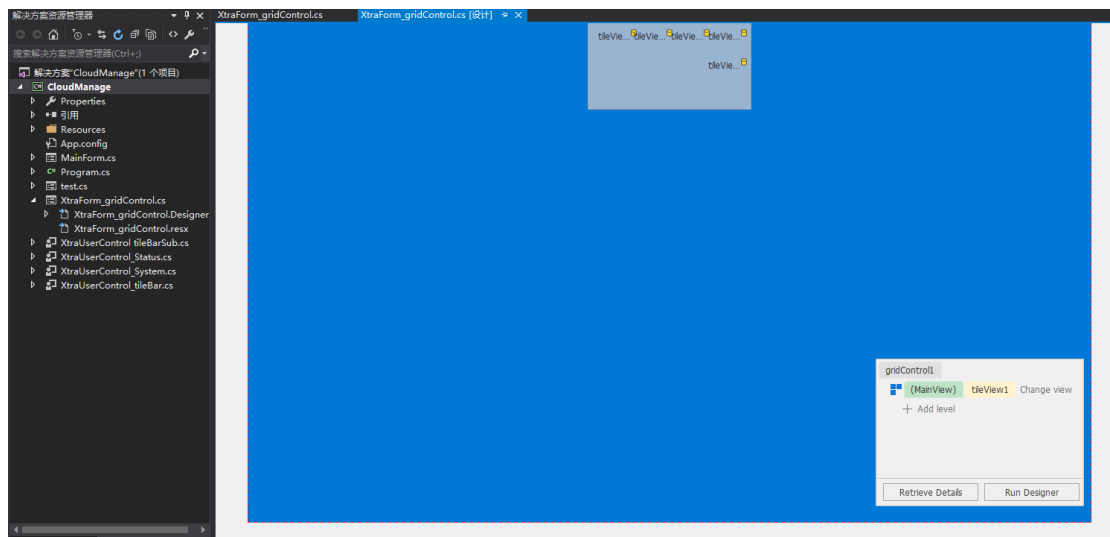


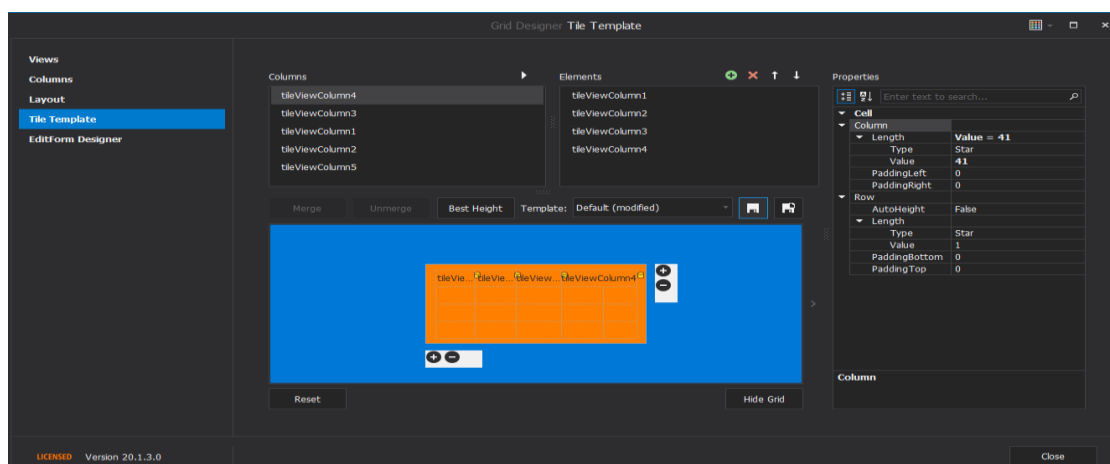
DevExpress20.1_V1.0

1、增加 Grid 行、列及选中高亮显示、增加图像方法；



1.1 点击 **Run Desinger** 进行 **Tile Template**（布局）设计()；

默认有四个表格, 如下图所示: (一行四列排布; row:1;column:4), 其中每个表格中又有两列 (column\column2) ;



Code:

```
DataTable dt = new DataTable("dtName");//global_variable;
private void initGridControl()
{
    //创建一个datatable
    //添加列
    dt.Columns.Add("col1", typeof(String)); //Add() 参数: 列名, 该列存放的数据的
    类型;(string-->image亦可)
    dt.Columns.Add("col2", typeof(String));

    ////添加列--add--0517
    //dt.Columns.Add("col3", typeof(String)); //Add() 参数: 列名, 该列存放的数据
    的类型;(string-->image亦可)
    //dt.Columns.Add("col4", typeof(String));

    ////add-image-0517;
    //dt.Columns.Add("col5", typeof(Image));

    //添加行
    for (int i = 0; i < 4; i++)//default--column=4;
    {
        DataRow dr = dt.NewRow();    //创建dt的一个行对象dr

        dr["col1"] = "Order";        //给dr的两个列赋值
        dr["col2"] = "Brand";

        //添加列--add--0517
        //dr["col3"] = "Weight";        //给dr的两个列赋值
        //dr["col4"] = "Weight+";

        ////add-image-0517
        //dr["col5"] = CloudManage.Properties.Resources.LG_LOGO;
        //drTemp["deviceImgTop"] =
        global::CloudManage.Properties.Resources.ZJ17_PROTOS70_336x140;

        dt.Rows.Add(dr);            //将行dr添加到表dt中
    }

    //Click-Grid, Obtain MessageBox Information;
    private void tileView1_Click(object sender, EventArgs e)
    {
        int[] selrow = tileView1.GetSelectedRows();
        //MessageBox.Show(selrow[0].ToString());
    }
}
```

```
if(selrow[0]==0)
{
    MessageBox.Show("row0-clo0-a00");
}

if (selrow[0] == 1)
{
    MessageBox.Show("row0-clo1-a01");
}

if (selrow[0] == 2)
{
    MessageBox.Show("row0-clo2-a02");
}

if (selrow[0] == 3)
{
    MessageBox.Show("row0-clo3-a03");
}

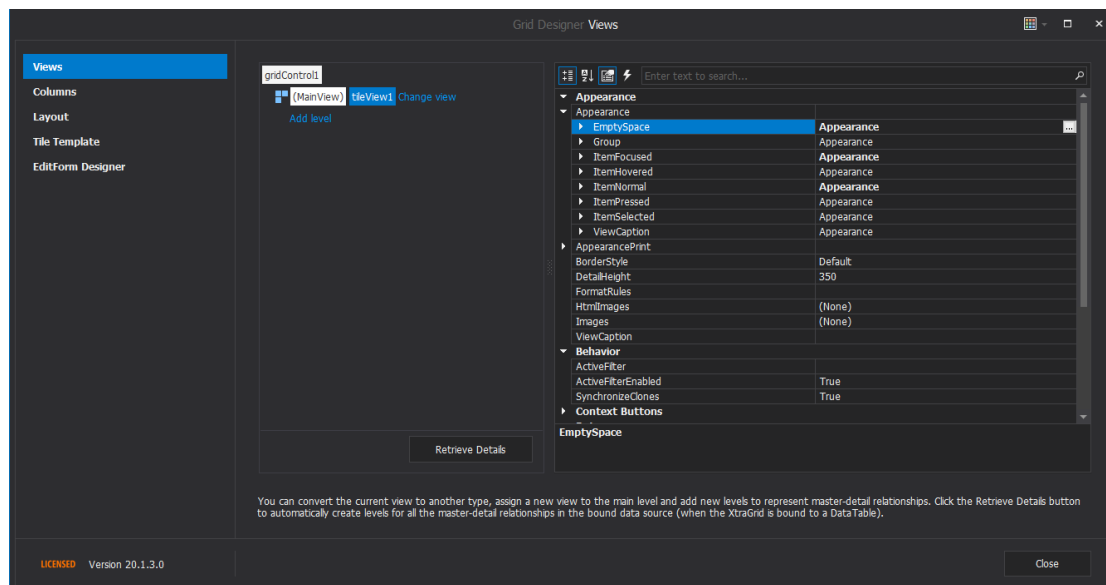
//if (selrow[0] == 4)
//{
//    MessageBox.Show("row1-clo0-a10");
//}

//if (selrow[0] == 5)
//{
//    MessageBox.Show("row1-clo1-a11");
//}

//if (selrow[0] == 6)
//{
//    MessageBox.Show("row1-cl2-a12");
//}

//if (selrow[0] == 7)
//{
//    MessageBox.Show("row1-clo3-a13");
//}
```

1.2 Grid 颜色显示，背景颜色、Grid 颜色默认设置及选中高亮显示：



1.2.1 背景颜色——EmptySpace(default--Highlight);

1.2.2 Grid 颜色——ItemNormal(default--ActiveCaption);

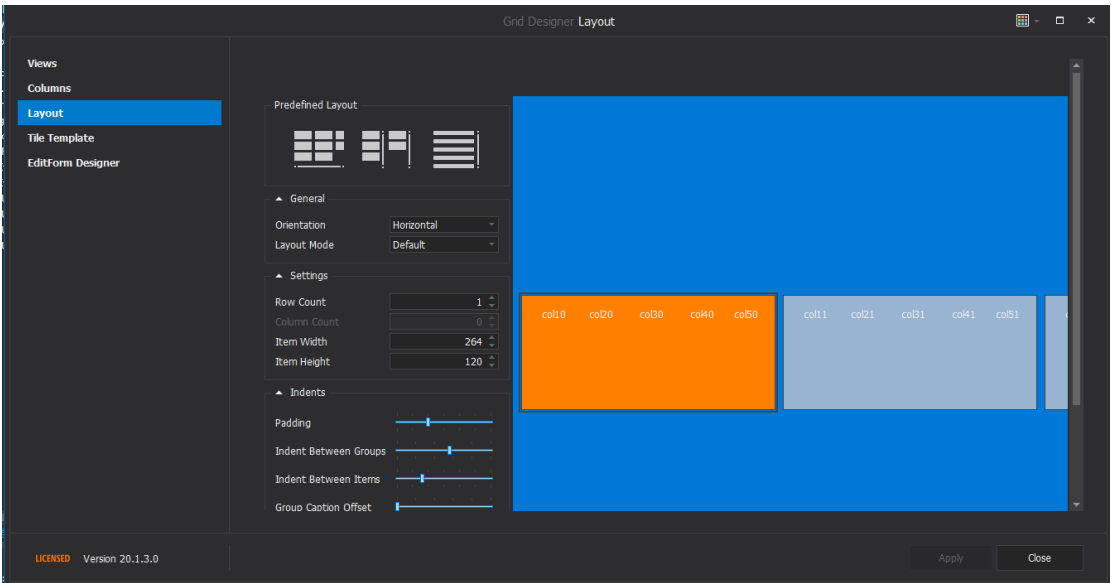
1.2.3 选中高亮显示——ItemFocused(default--255, 128, 0);

1.3 添加列、图像；

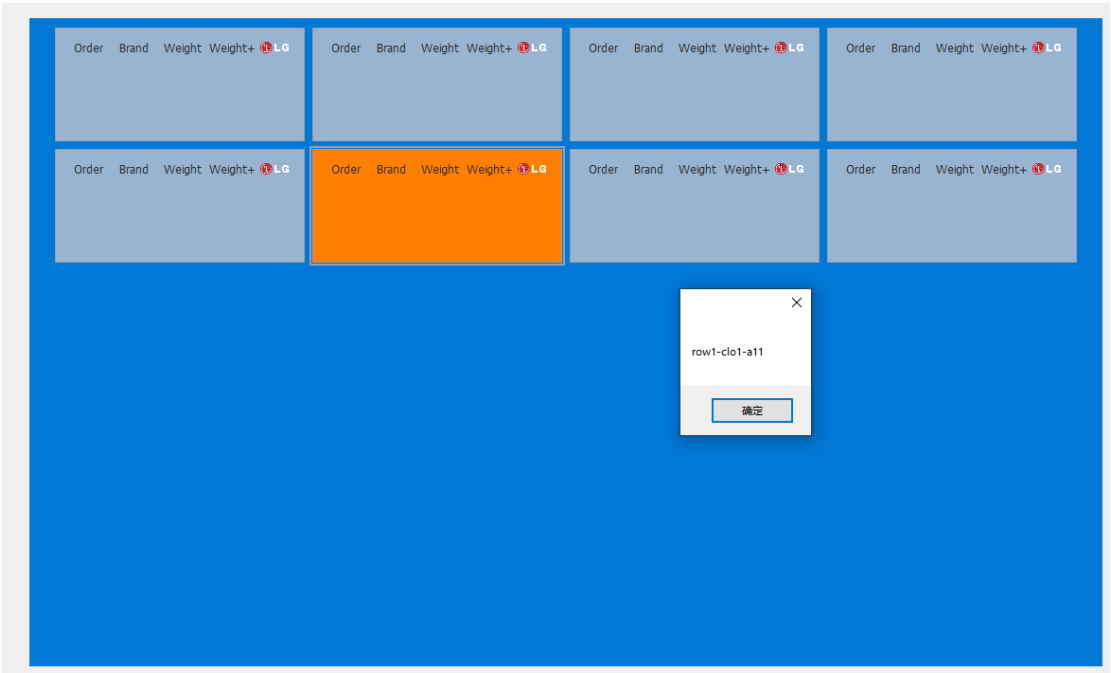
```
//添加列--add--0517
//dr["col3"] = "Weight";           //给dr的两个列赋值
//dr["col4"] = "Weight+";

//dt.Columns.Add("col5", typeof(Image));
////add-image-0517
//dr["col5"] = CloudManage.Properties.Resources.LG_LOGO;
```

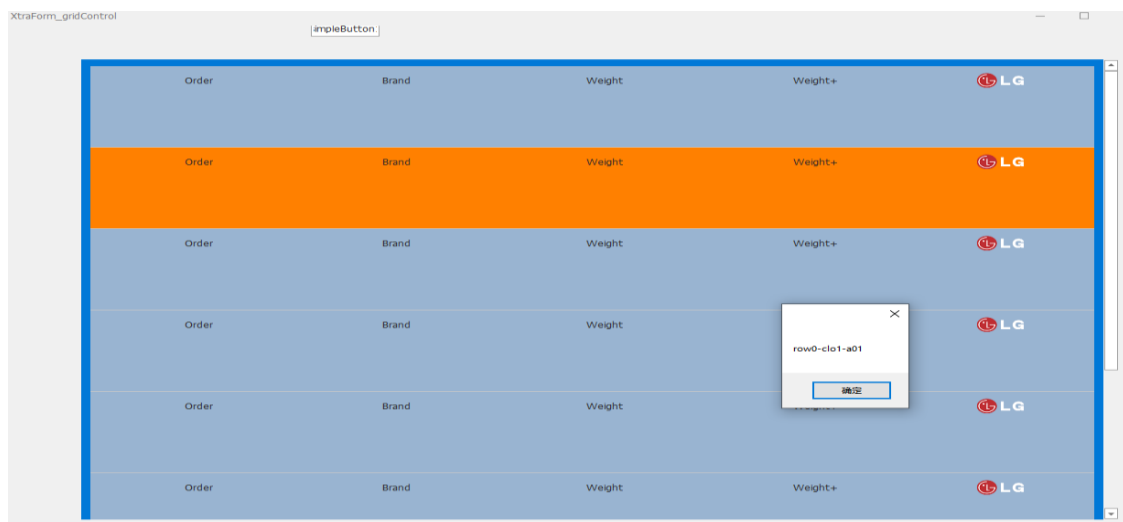
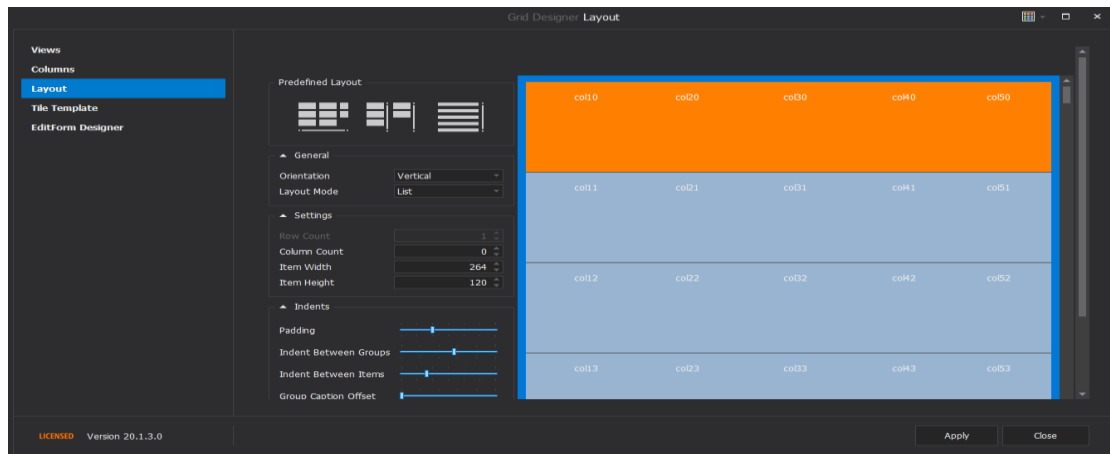
1.3.1 Grid 排列方向: (Orientation: Horizontal;Vertical);



1.3.2 选择 **Vertical**, 并在选择 **Grid** 同时，高亮显示及显示 **MessageBox** 信息;



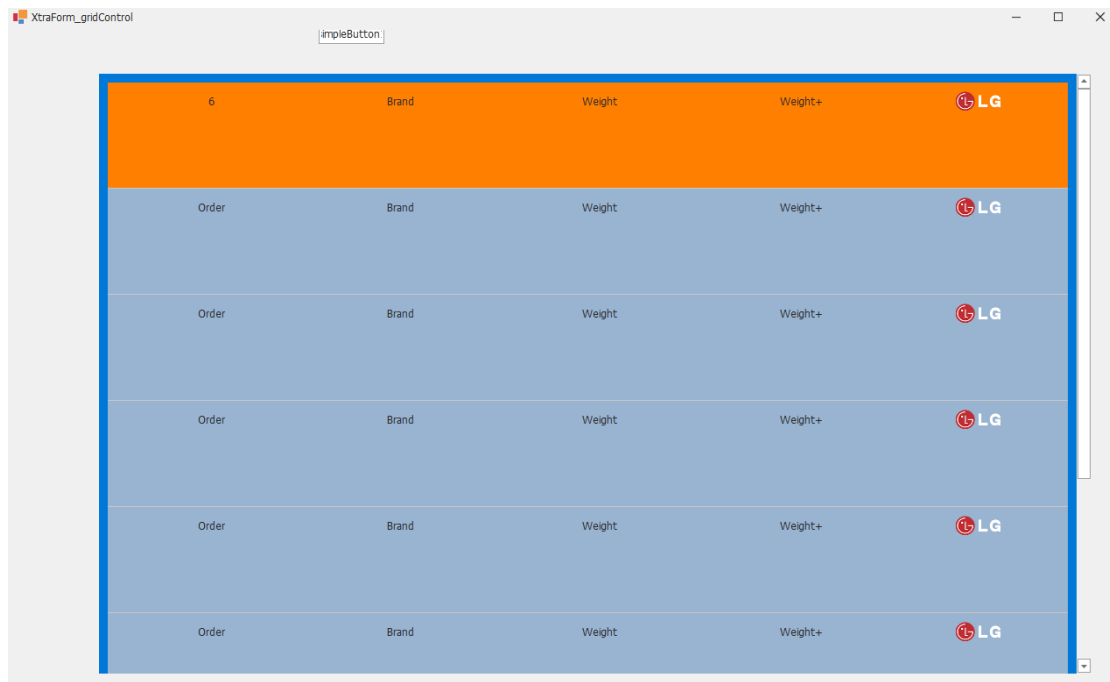
1.3.3 布局-Layout;Layout Mode---List;



1.4 在 Grid 中写入事件;

//在row0-clo0-a00中写入计数事件;

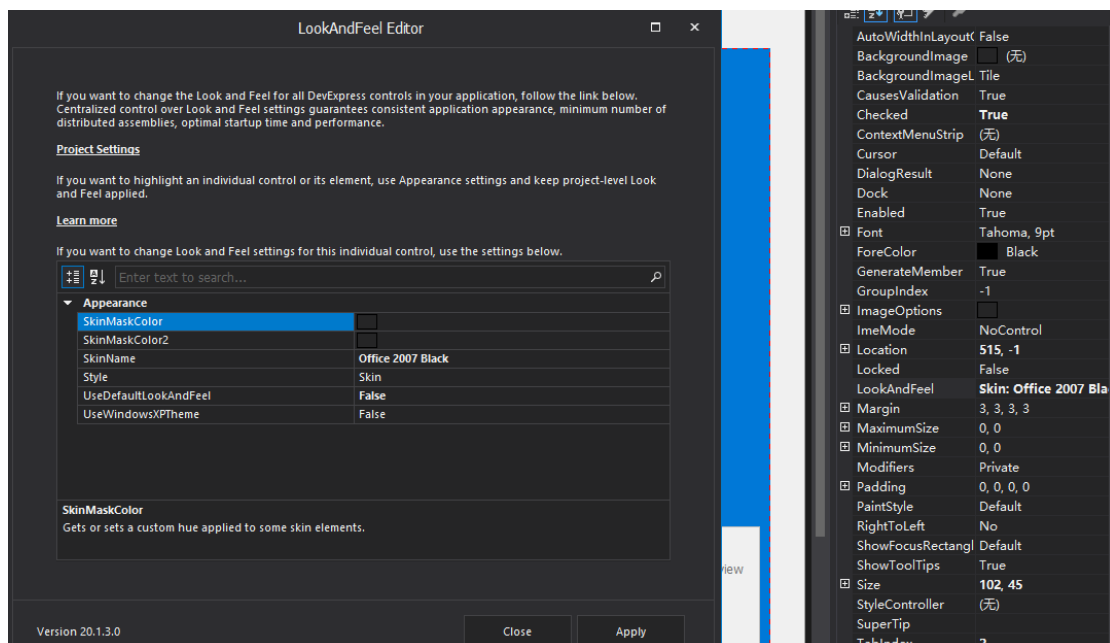
```
int i = 0;
private void simpleButton1_Click(object sender, EventArgs e)
{
    dt.Rows[0]["col1"] = (i++).ToString();
}
```



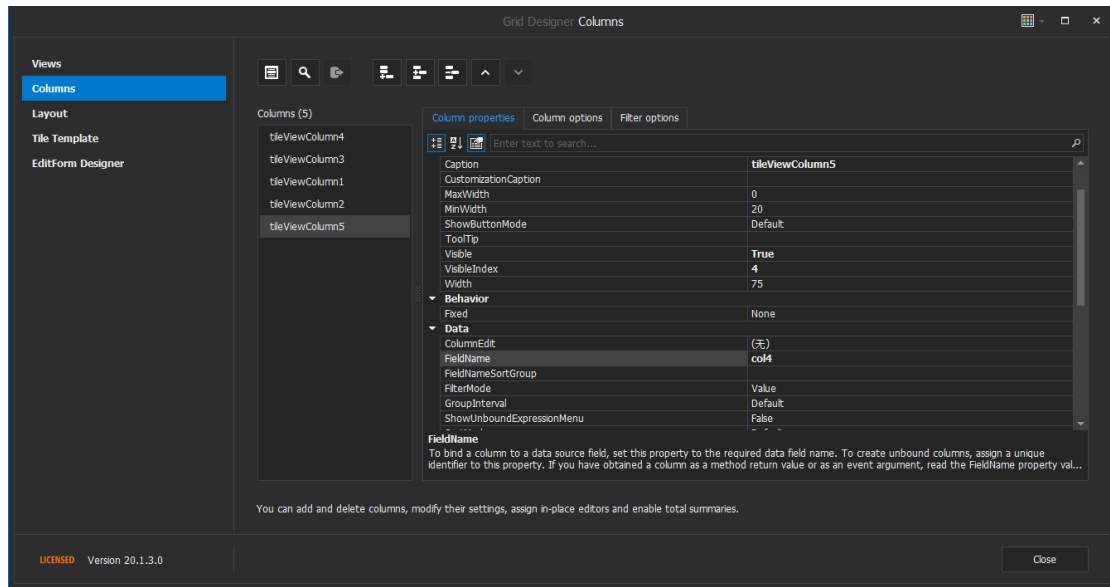
1.5 修改 button 控件颜色;

1) 属性—>appearance;

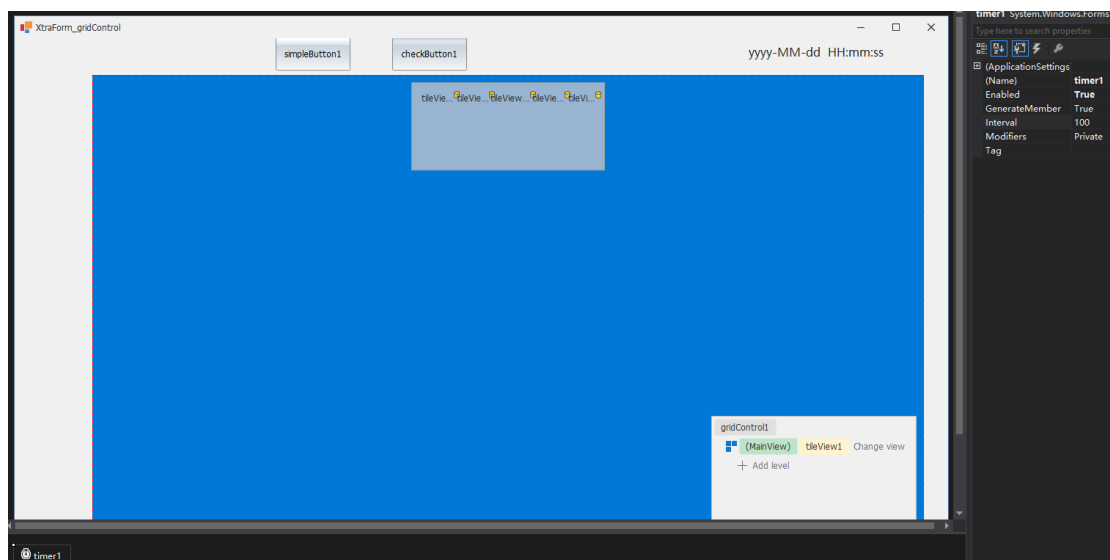
2) LookAndFeel—SkinName\Style\etc;



1.6 当修改 col 中数字时，将 **tileviewColumn** 的 **fieldname** 设为 **dt** 的列名（**fieldname** 要同列名相同否则不显示）；



1.7 添加系统时间后，需要将定时器开启：**Enable:true;**



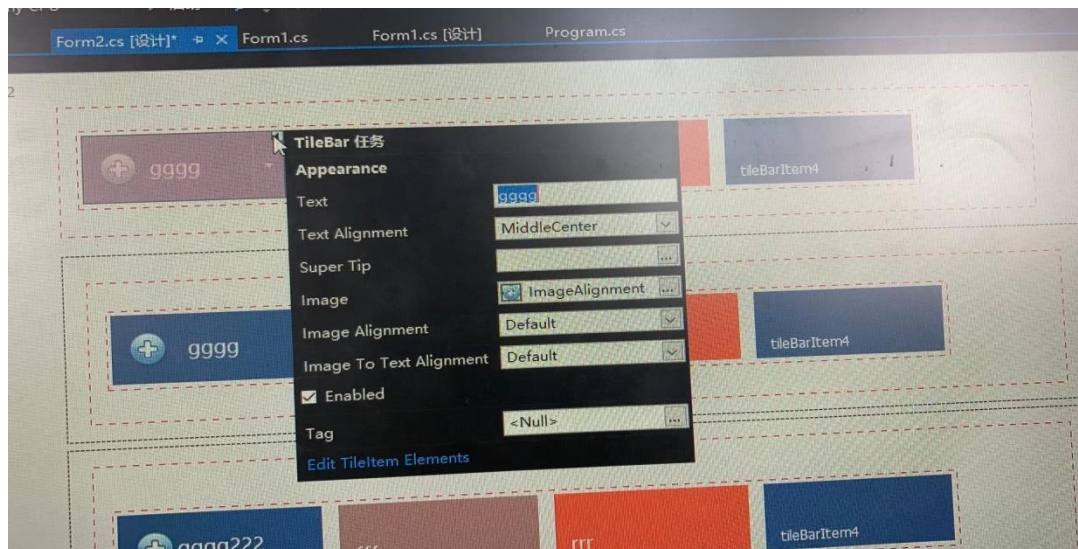
2、tileBar（利用 group）设置主页面；

① 工具箱搜索选择 **tileBar**，点击 ▷ 按钮，选择 **Add Group (Choose Dock Style :None; BackGround Image: 无 ; BackGround Image layout:Tile;Selection Color :White)**，在主页面中添加页面（此处添加四个）；

选择主页面中的第一个页面：点击 ▷

Text:gggg;Text;Alignment:MiddleCenter;Image:→DX Raster images→选择

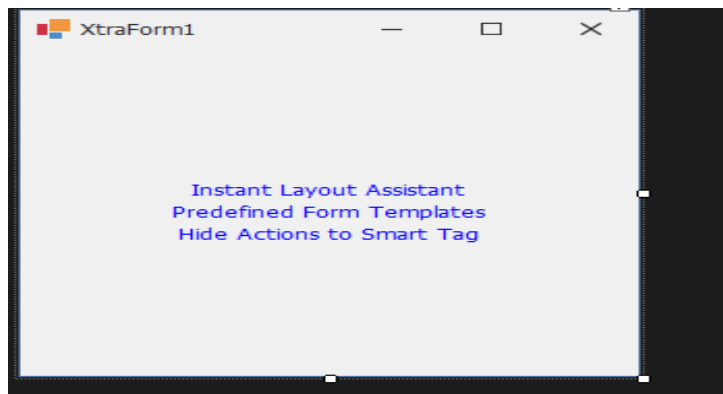
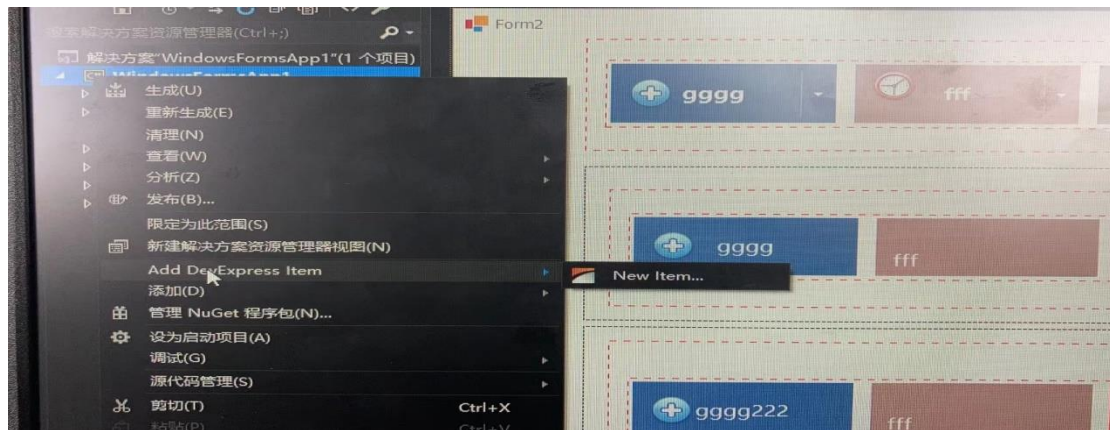
add_32X32.png; Image Alignment:default; Image To Text Alignment:Default;



以下是添加 Image\text 后的页面；



在解决方案中添加一个新的页面（Form）；

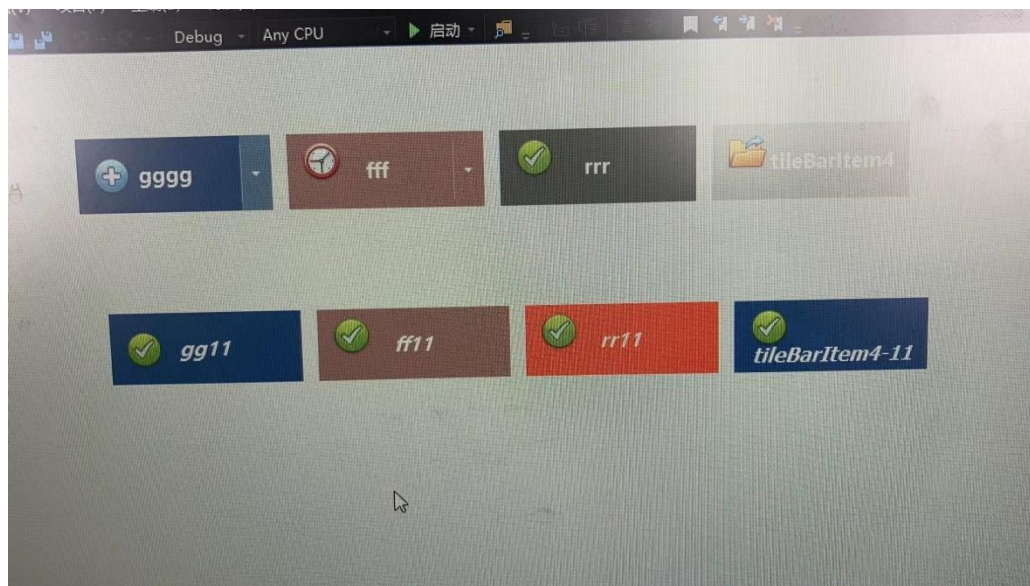


② 在①中设计生成的主页面中添加(包络)一个 **tileBarDropDownContainer**;
(工具箱中搜索添加); 拖拽 **tileBarDropDownContainer** 至 **Form** 并将①中
生成的页面复制粘贴到 **tileBarDropDownContainer** 中;

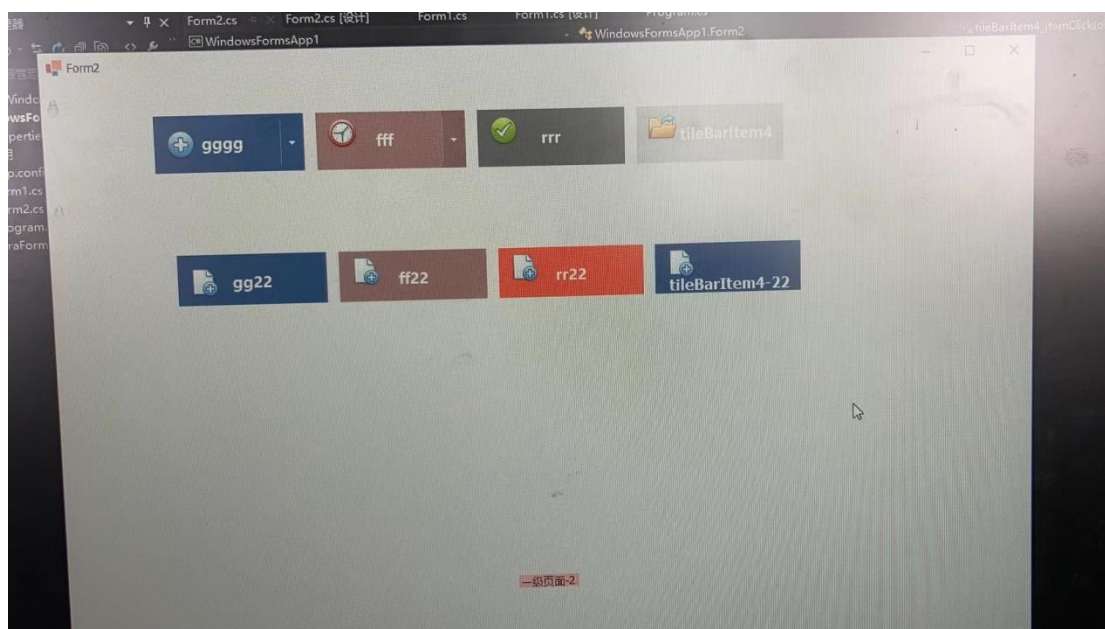
③ 关键字菜单; 使得①、②联动(对应); 设置一级界面:
DropDownControl:tileBarDropDownContainer1;

DropDownControl:tileBarDropDownContainer2;

点击第一个 **tileBarItem1**: (gggg) ;子页面 gg11\ff11;

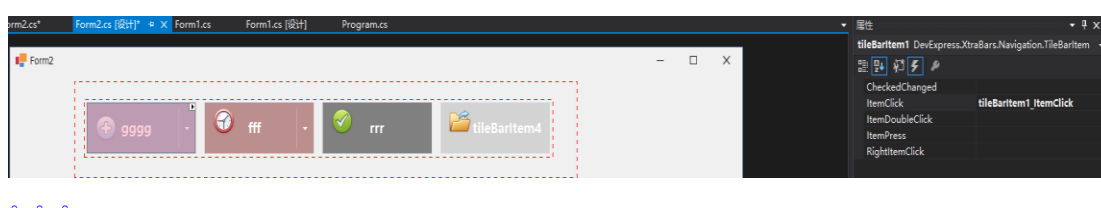


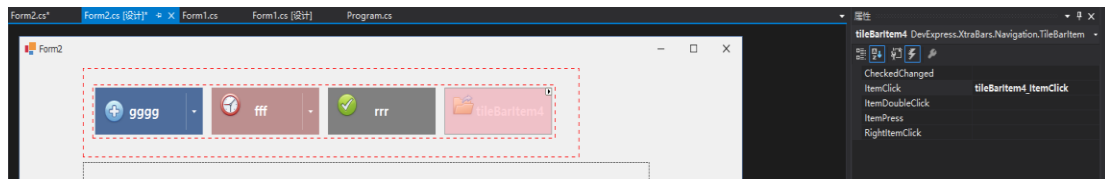
点击第一个 tileBarItem1: (fff) ; 子页面 gg22\ff22;



④ （一级页面翻页属性）通过设置 **NavigationFrame** 设计 **navigationPage** （通过+可自增，可嵌套（Page 中嵌套 Frame））;

tileBarItem1_ItemClick--- tileBarItem4_ItemClick 设置主页面联动一级页面;





```
private void tileBarItem1_ItemClick(object sender, TileItemEventArgs e)
```

```
{
```

```
    this.navigationFrame1.SelectedPage = navigationPage1;
```

```
}
```

```
.....
```

```
.....
```

```
private void tileBarItem4_ItemClick(object sender, TileItemEventArgs e)
```

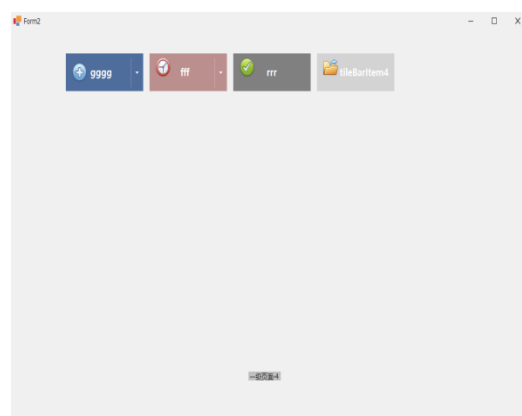
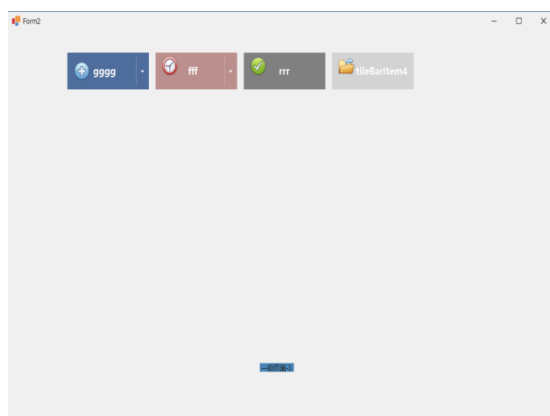
```
{
```

```
    this.navigationFrame1.SelectedPage = navigationPage4;
```

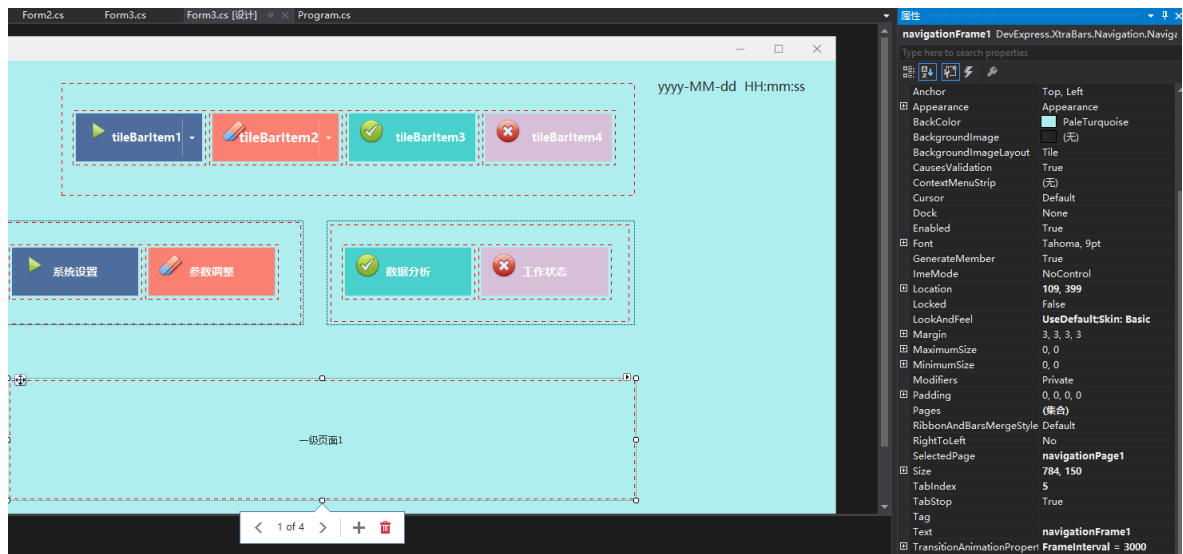
```
}
```

```
• • •
```

```
.....
```



3、 初始页面设置及翻页速度；



3. 1 NavigationFrame1→SelectedPage→navigationPage1;

3. 2 TransitionAnimationProperties(调整翻页速度)→ FrameInterval = 3000(默认

FrameInterval = 10000);

4、折线图、饼状图、散点图设计;

dev ChartControl 的最全使用方法 有横轴纵轴高亮显示:

https://blog.csdn.net/MRX220518/article/details/76974808?spm=1001.2101.3001.6650.11&utm_medium=distribute.pc_relevant.none-task-blog-2~default~BlogCommendFromBaidu~Rate-11.pc_relevant_paycolumn_v3&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2~default~BlogCommendFromBaidu~Rate-11.pc_relevant_paycolumn_v3&utm_relevant_index=16

4.1 折线图(Spline)设置;

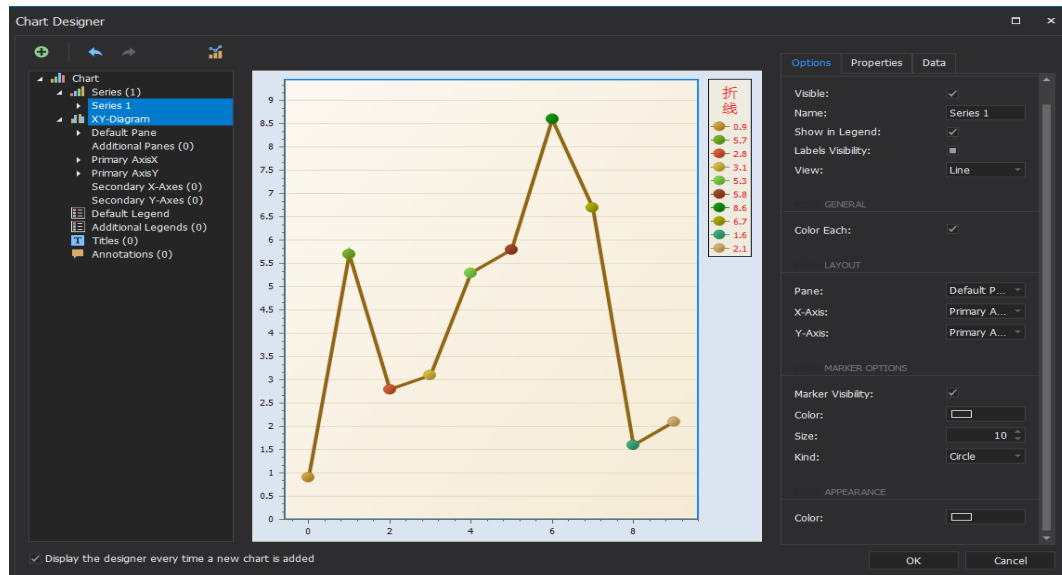
https://blog.csdn.net/qq_18854761/article/details/120126721

https://blog.csdn.net/xiaochenXIHUA/article/details/120307234?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidulandingword~default-0-120307234-blog-120126721.pc_relevant_paycolumn_v3&spm=1001.2101.3001.4242.1&utm_relevant_index=2

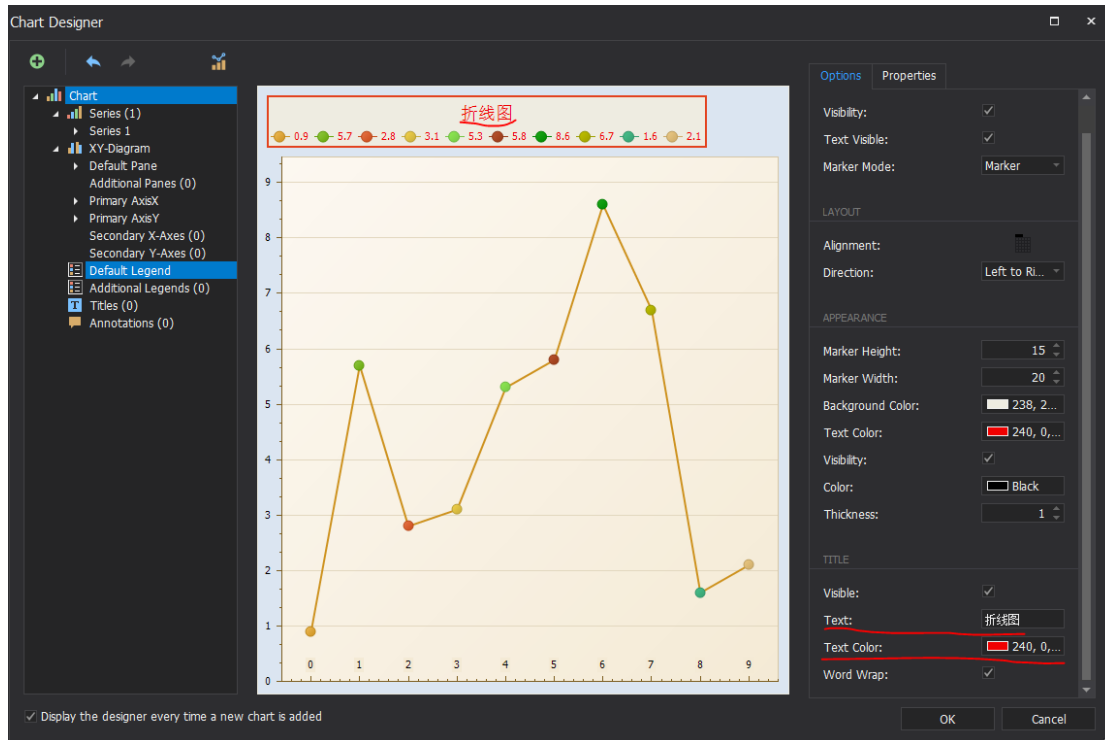
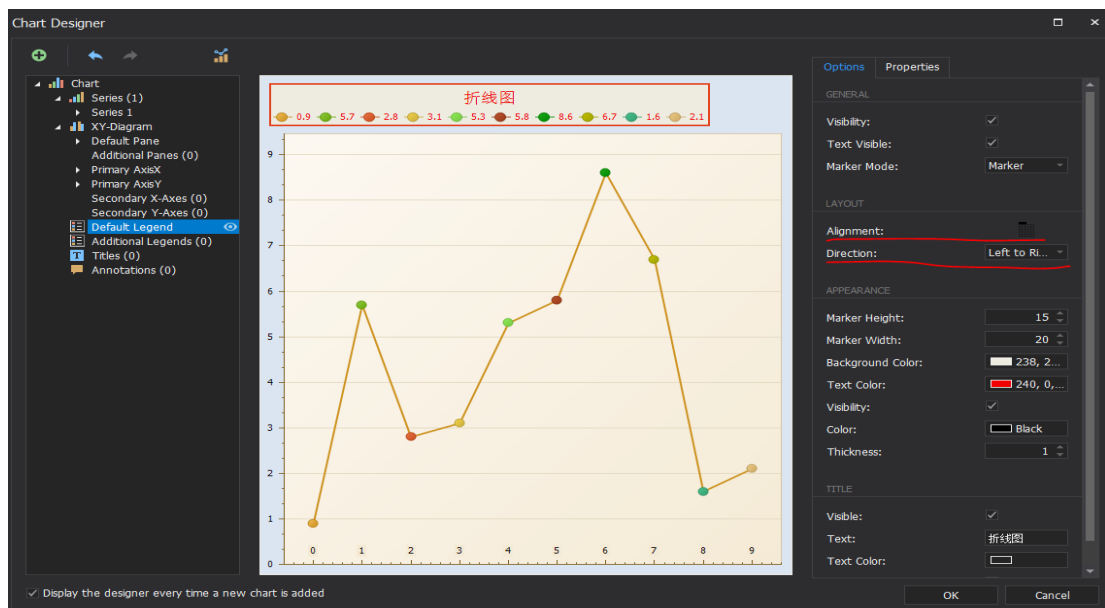
工具箱→ChartControl1→Title(chart title:在 Options 中修改表名) →Properties(修改表头字体、颜色及背景颜色等等)→Series("+",添加不同类型折线图);

- 1、Marker Visibility——曲线上是否显示小圆点;
- 2、Color——小圆点颜色 (Automatic 时跟随曲线颜色或由 Color Each 自动生成);
- 3、Size——小圆点尺寸;
- 4、Kind——小圆点的类型;

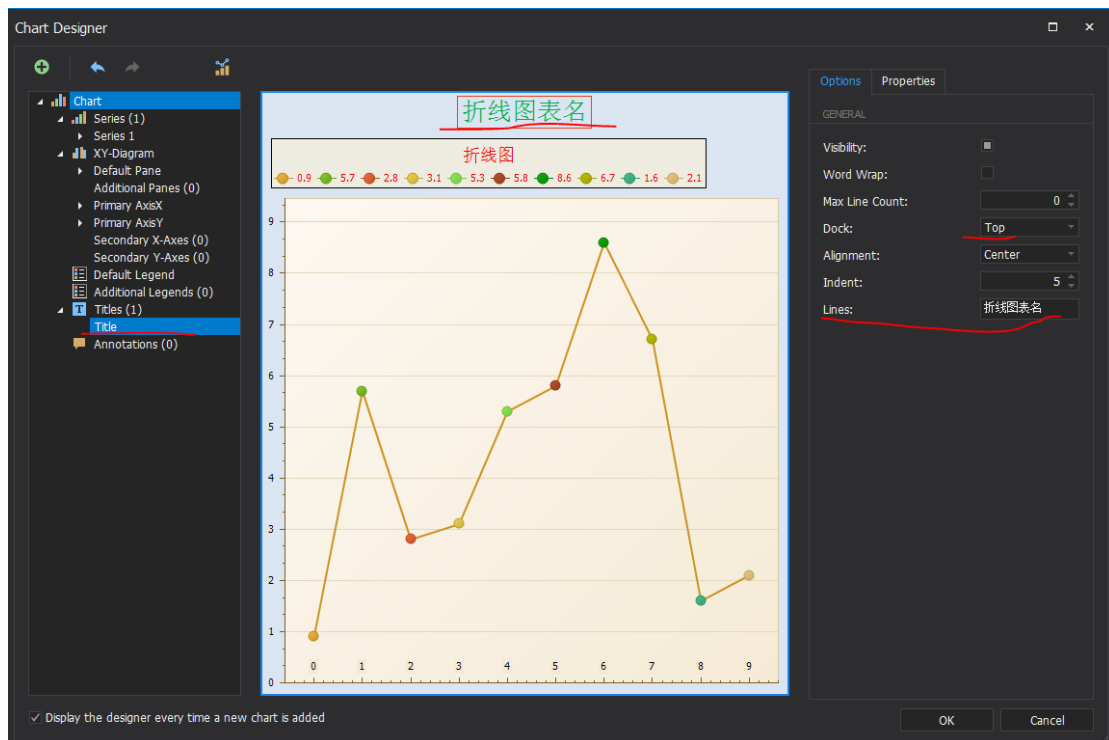
- 5、Visibility——是否可见
- 6、Text Visible——文本数字是否可见
- 7、Marker Mode——选择类型
- 8、Alignment——Legend 相对图表的位置
- 9、Direction——排列方向;



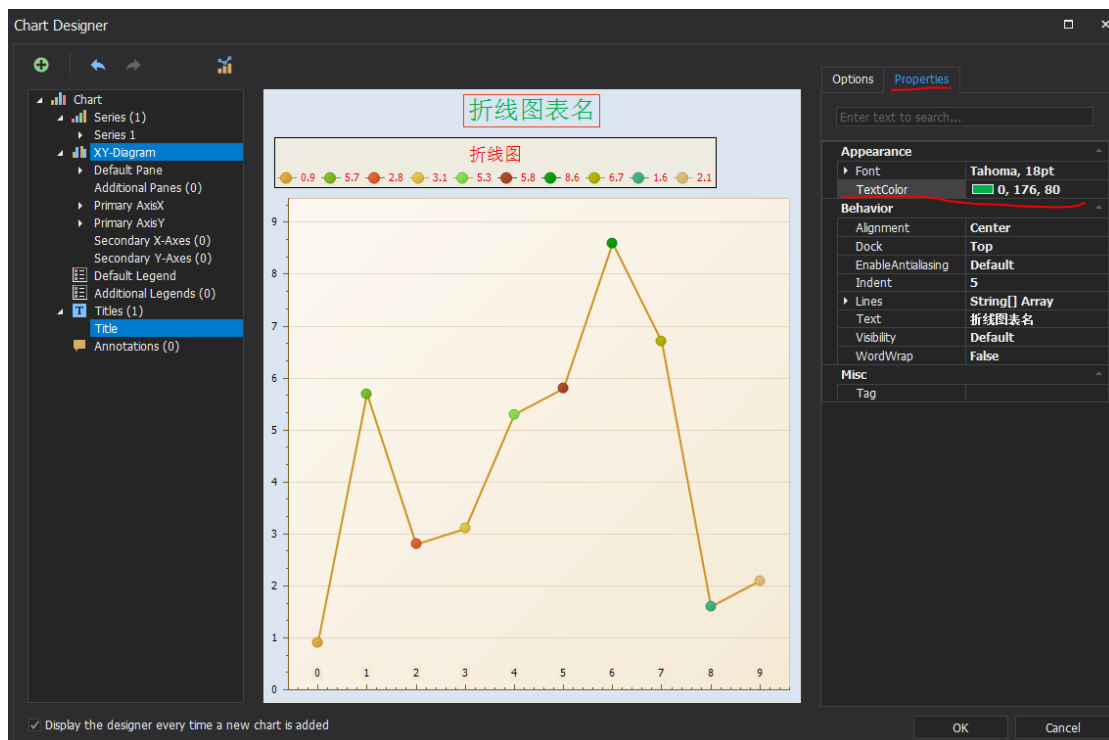
- A. Marker Height——高度(图例-Legend: 折线图);
- B. Marker Width——宽度(图例-Legend: 折线图);
- C. Background Color——Legend 框背景色;
- D. Text Color——Legend 数字颜色;
- E. Visibility——Legend 边框是否可见;
- F. Color——Legend 边框颜色;
- G. Thickness——Legend 边框粗细;
- H. Visible——Legend 标题是否可见;
- I. Text——Legend 标题文本;
- J. Text Color——Legend 标题文本颜色;
- K. Word Wrap——Legend 标题文本换行;
- L. **Alignment——Legend 相对图表的位置;**
- M. **Direction (left-right) ——排列方向;**



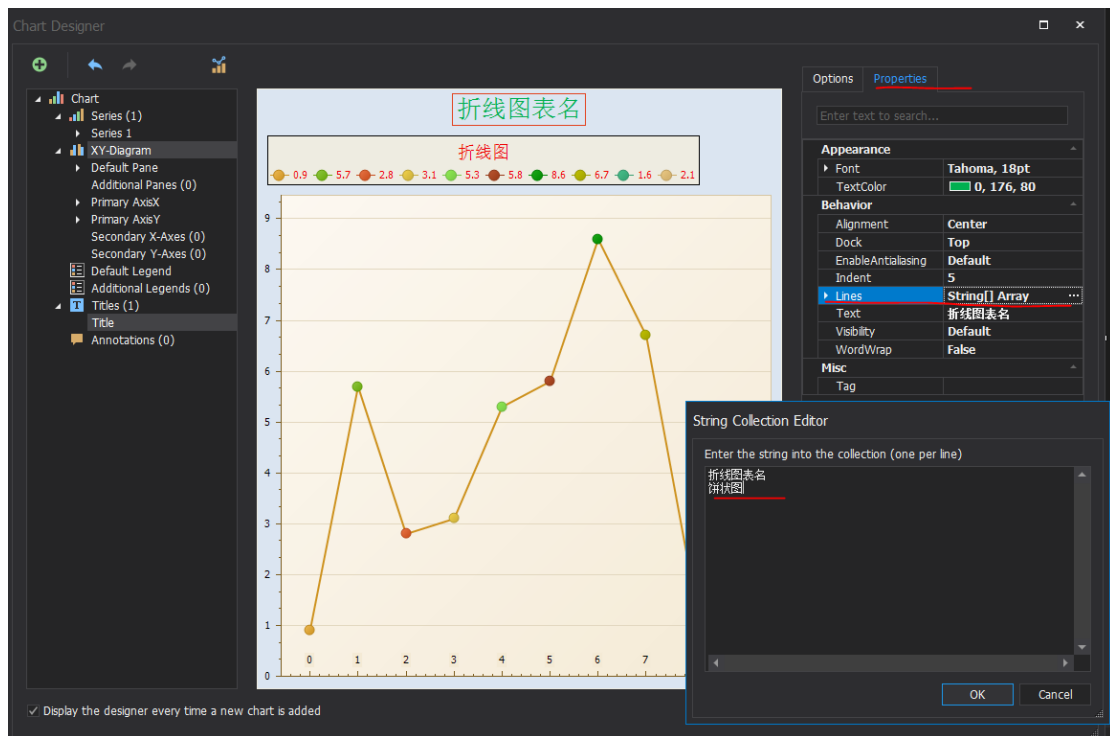
Text:文本信息（折线图）； Text Color（Legend:color）；



TextColor——标题颜色 **Alignment**——标题位置：居左、居中、居右；
Dock——图表的上下左右;**Visibility**——标题是否可见；多行标题;



Properties:设置 **Chart_Title_Color**;

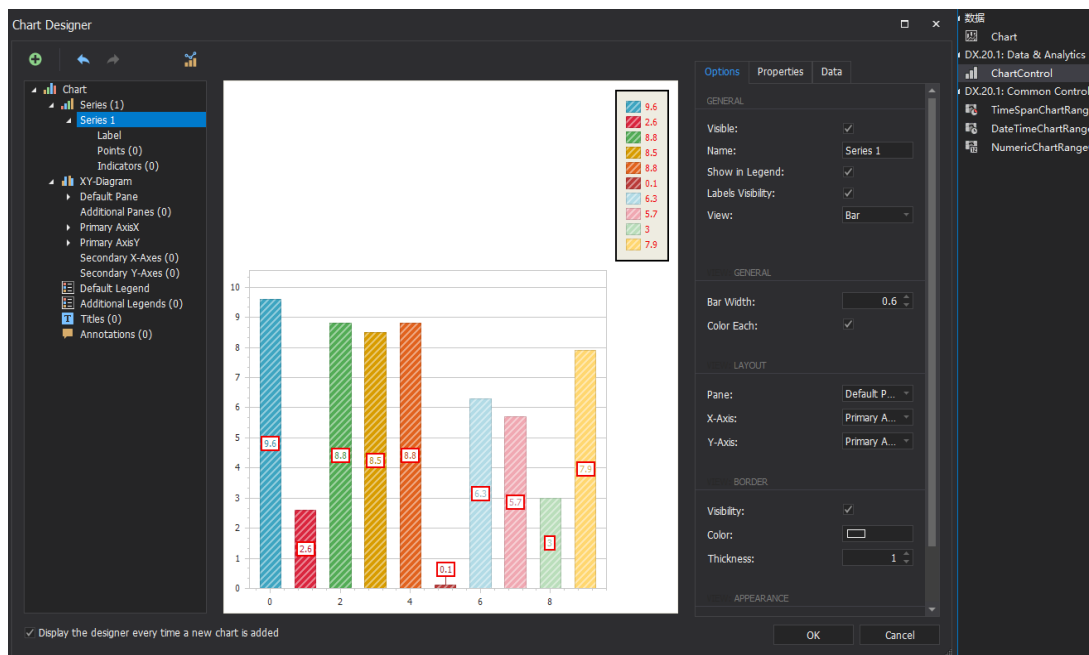


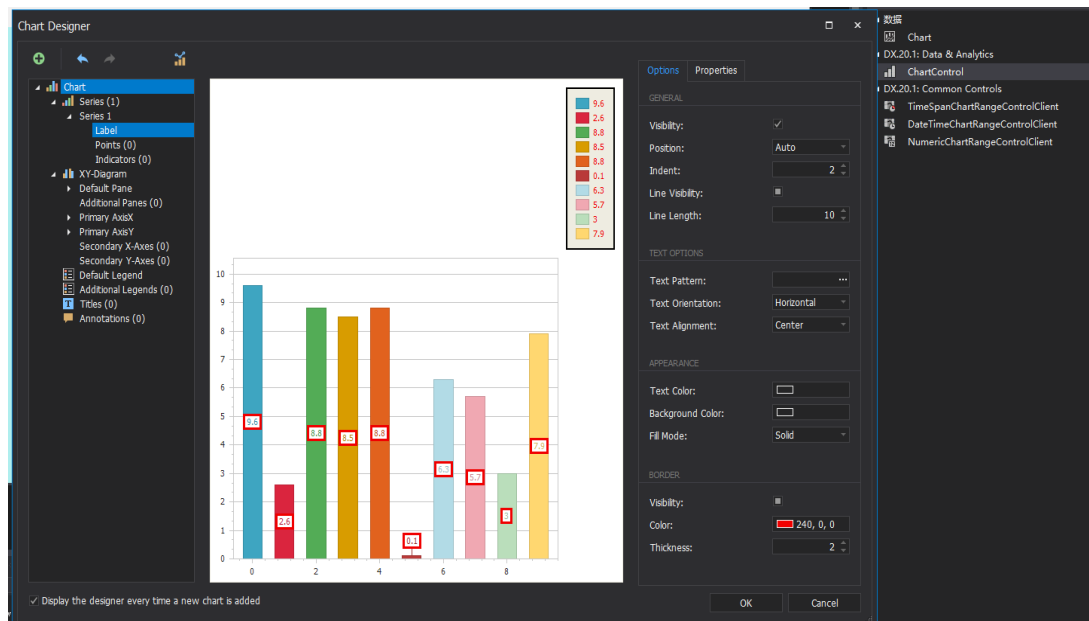
多行标题;

4.2 柱形图 (Bar) 设置;

<https://blog.csdn.net/jameshelong/article/details/114682380>

工具箱→ChartControl1→Series(Bar) →Visible: √ ; Nmae:series1; Show in Lengend: √ ;Labels
Visibility: √ ; View:Bar; Color Each: √ ;





Chart_Histogram 将小方框中的数字由中间调至柱状图最上方；

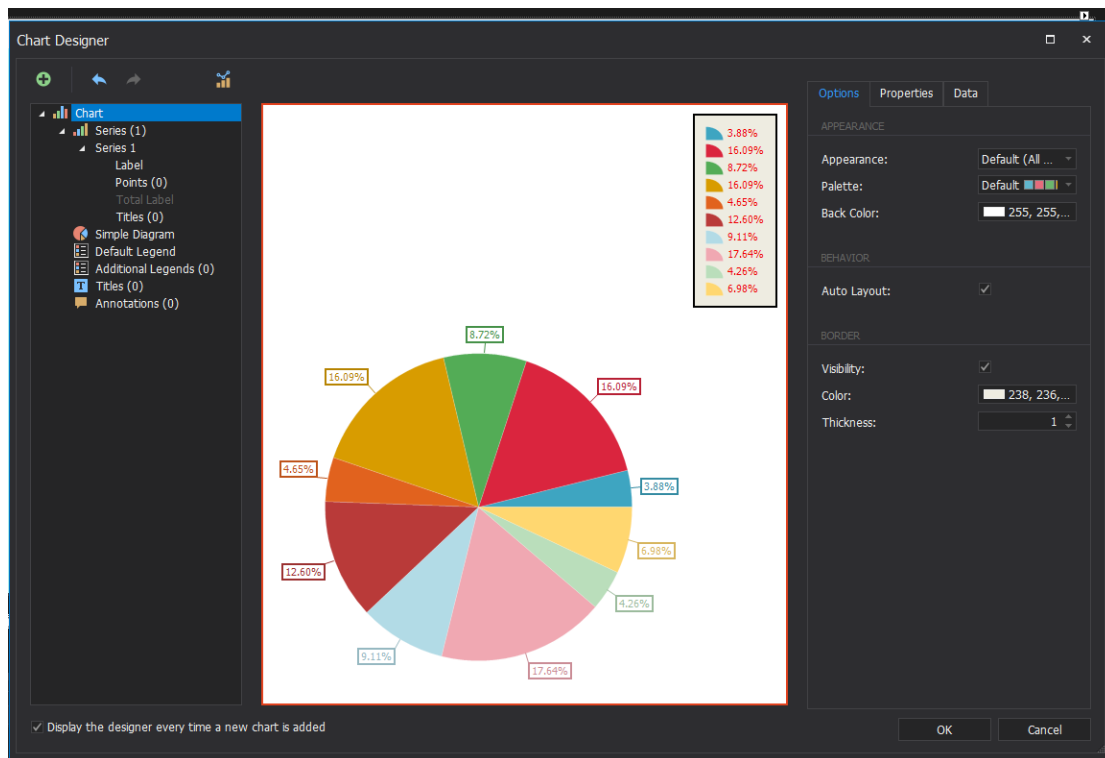


4.3 饼状图 (Pie) 设置;

<https://www.cnblogs.com/xuminyang/p/4600443.html>

<https://supportcenter.devexpress.com/ticket/details/t116685/how-to-convert-obsolete-pointoptions-and-legendpointoptions-properties-to-textpattern>

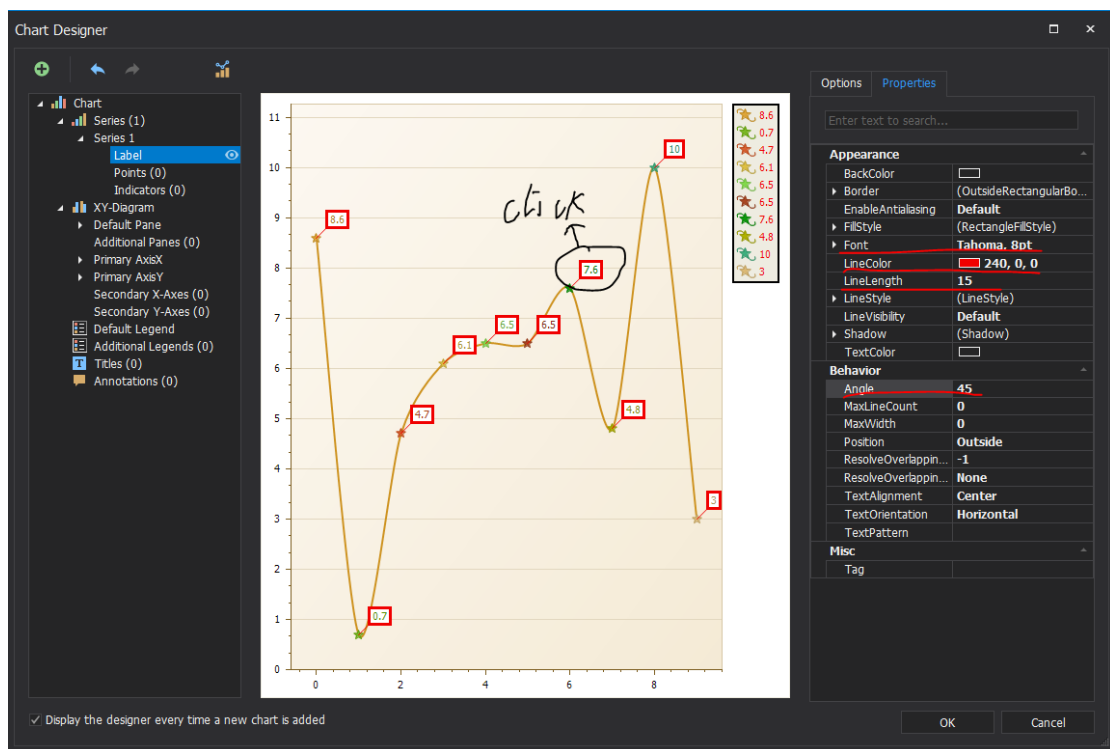
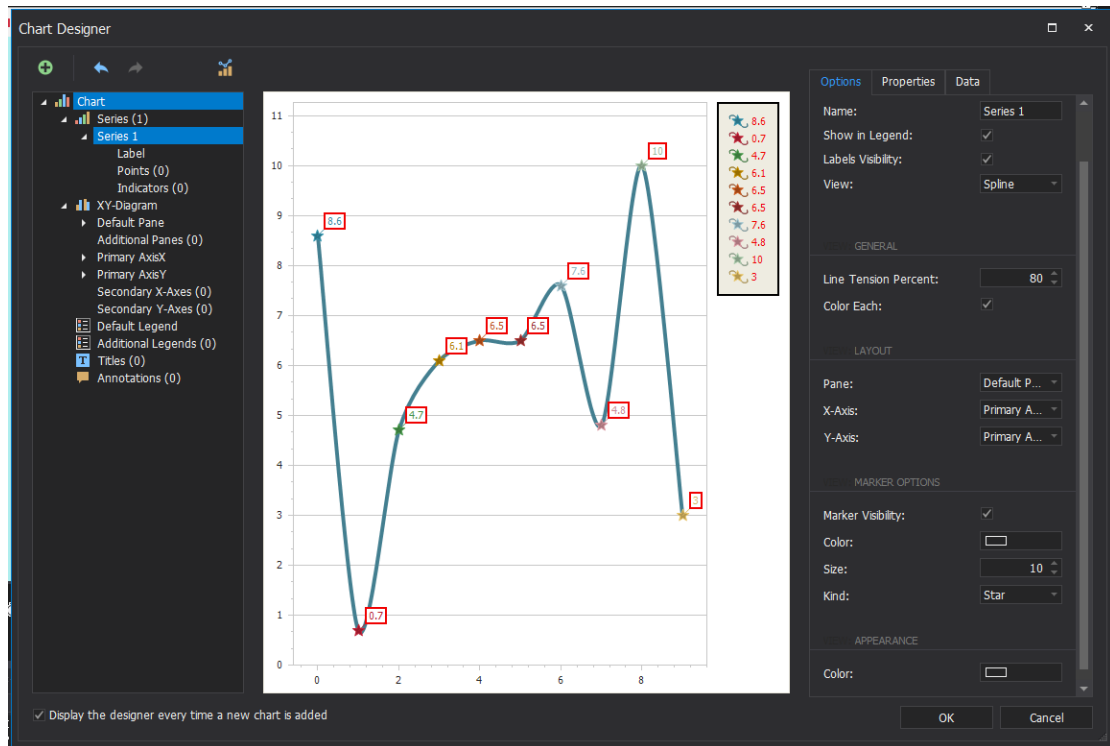
https://blog.csdn.net/qq_36248777/article/details/105745481



4.4 散点图 (point) ;

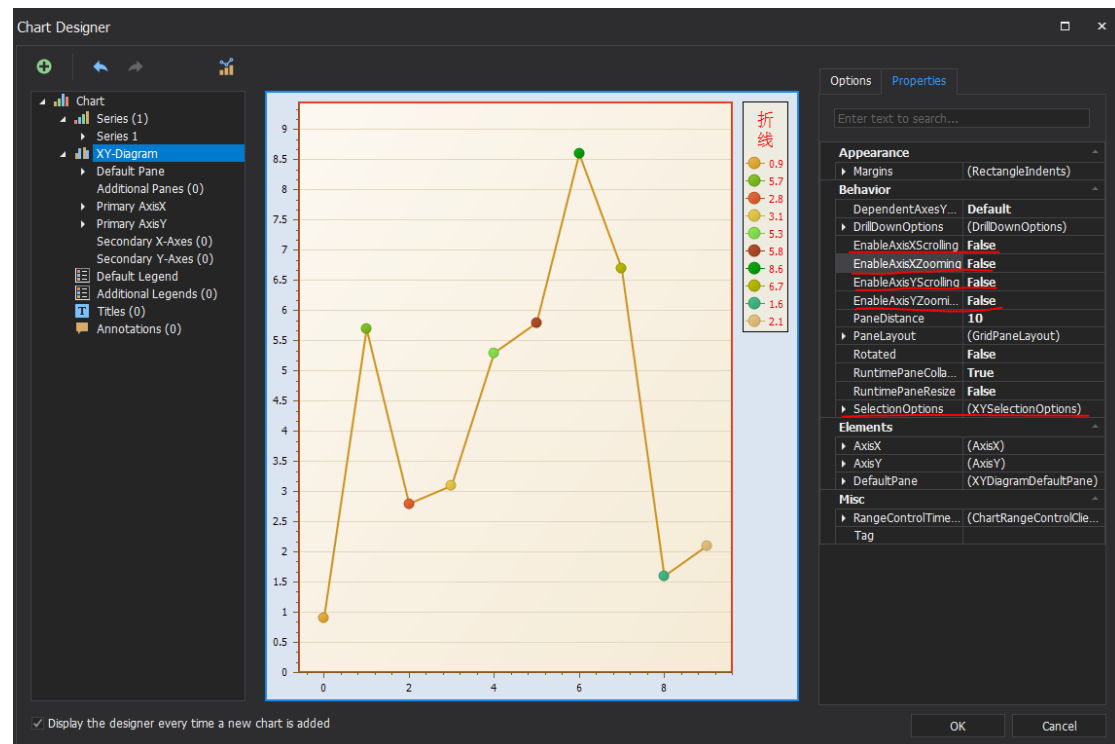
- 1、Marker Visibility——曲线上是否显示小圆点;
- 2、Color——小圆点颜色 (Automatic 时跟随曲线颜色或由 Color Each 自动生成);
- 3、Size——小圆点尺寸; 4、Kind——小圆点的类型;





- 1、Font——字体；
- 2、LineColor——方框和点之间线的颜色；
- 3、LineLength——线长；
- 4、Angle——线的角度（方框与 kind:star 间的角度）；

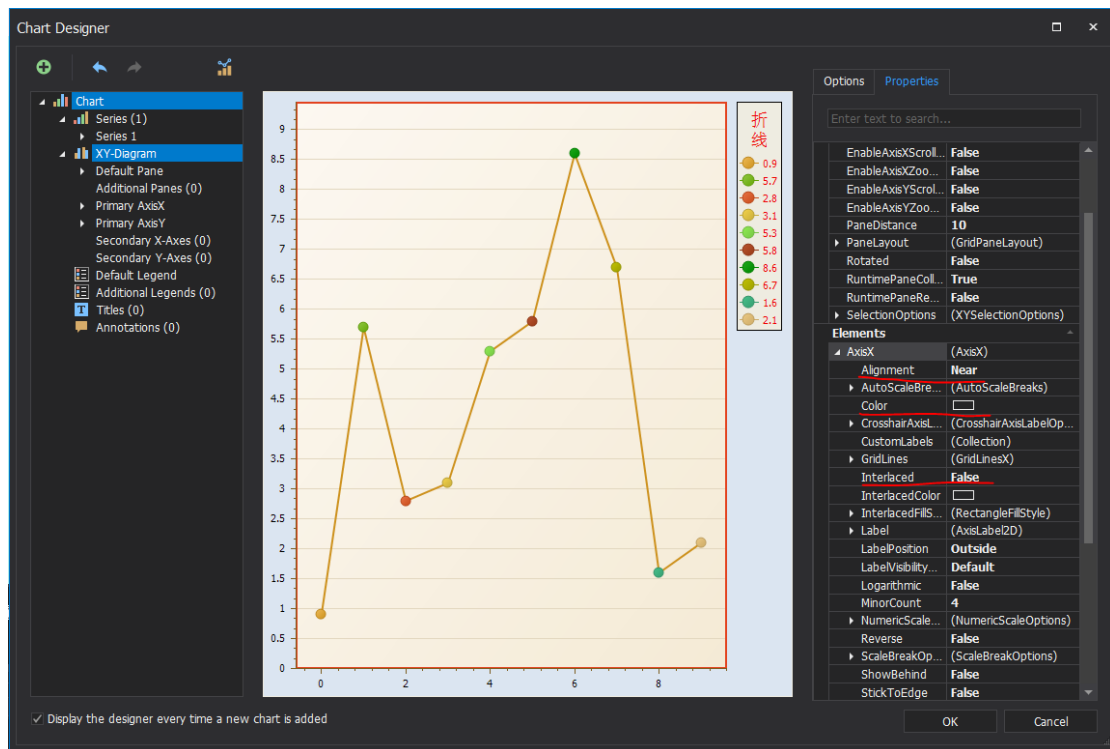
图表→X、Y 轴设置；



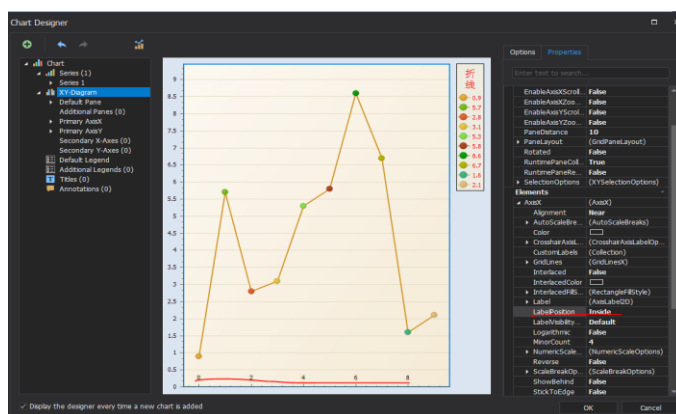
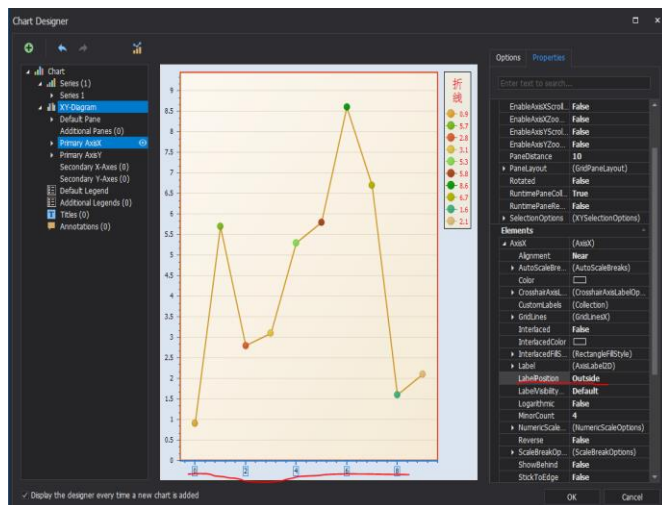
- A. EnableAxisXScrolling——X 轴是否可滚动
- B. EnableAxisXZooming——X 轴是否可缩放
- C. Rotated——翻转
- D. SelectionOptions——缩放快捷键设置
- E. 坐标轴设置

坐标轴设置：

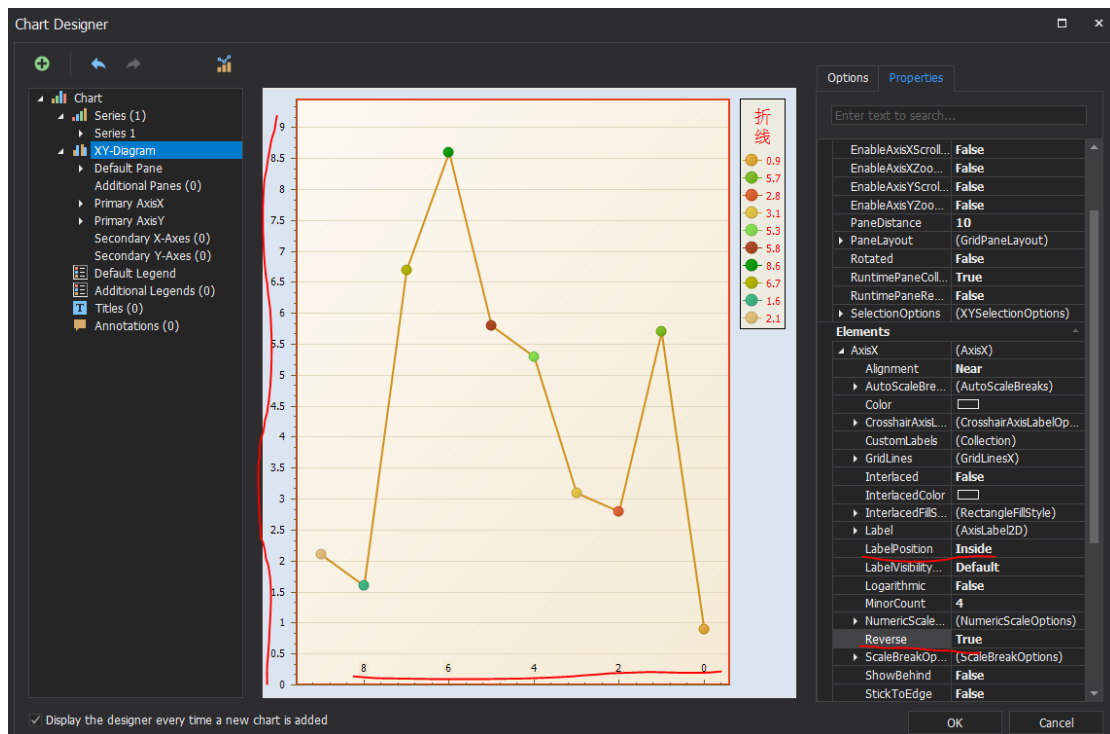
- 1、Alignment——坐标轴位置；
- 2、Color——坐标轴颜色；
- 3、Interlaced——False；



4、LabelPosition——数字标签位置在图表外还是内

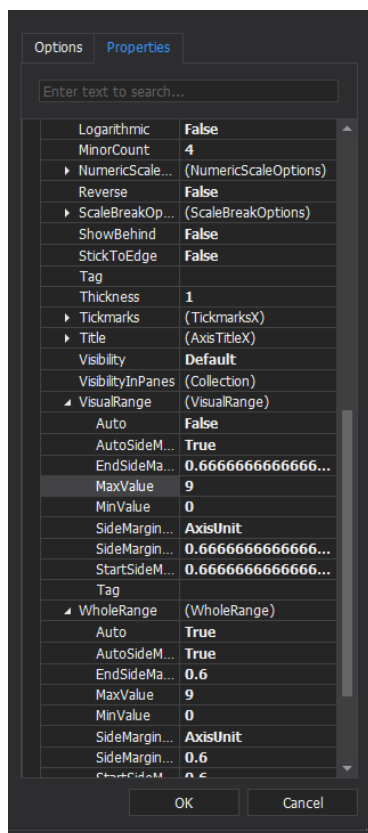


5、坐标轴翻转 (Reverse)；



6、Thickness——坐标轴粗细；

7、AxisX/AxisY；



只有设为 **Auto** 时代码设定的坐标轴范围和起点才能生效

```
XYDiagram lineDiagram = (XYDiagram)chartControl_line.Diagram;  
lineDiagram.AxisX.WholeRange.SetMinMaxValues(0, 200);  
lineDiagram.AxisX.WholeRange.SideMarginsValue = 1;  
lineDiagram.AxisY.WholeRange.SetMinMaxValues(0, 20);  
lineDiagram.AxisY.WholeRange.SideMarginsValue = 1;
```