**ChartsMainDemo**

**可选的饼状图**

selection:

DevExpress Demos 14.2\Components\WinForms\CS\ChartsMainDemo\Modules\Miscellaneous

关键方法：

初始化饼图：void InitPieChart()

饼图点击事件：chartTotal.SeriesSelectionMode = SeriesSelectionMode.Point;

chartTotal.SelectedItemsChanged += ChartTotal\_SelectedItemsChanged;

以下是例子：

void 绑定投标情况饼状图(DeptItem pp投标单位, DataTable ppDT)

{

DataTable mmDT = new DataTable();

mmDT.Columns.Add("项目类型", typeof(String));

mmDT.Columns.Add("投标数量", typeof(Int32));

mmDT.Columns.Add("项目类型ID", typeof(String));

foreach (DataRow mmDR in ppDT.Rows)

{

if (mmDR["项目类型"].ToString() != "合计")

{

DataRow mmNewRow = mmDT.NewRow();

mmNewRow["项目类型"] = mmDR["项目类型"].ToString();

mmNewRow["投标数量"] = Common.StringToInt(mmDR["投标数量"].ToString());

mmNewRow["项目类型ID"] = mmDR["项目类型ID"].ToString();

mmDT.Rows.Add(mmNewRow);

}

}

Series \_pieSeries = new Series(pp投标单位.DeptName + " " + "投标情况", ViewType.Pie);

\_pieSeries.ValueDataMembers[0] = "投标数量";

\_pieSeries.ArgumentDataMember = "项目类型";

\_pieSeries.ArgumentScaleType = ScaleType.Qualitative;

\_pieSeries.DataSource = mmDT; //直接ppDT也行

ccPieChart.Series.Clear();

ccPieChart.Series.Add(\_pieSeries);

//----------------------------------------

\_pieSeries.PointOptions.PointView = PointView.ArgumentAndValues;

\_pieSeries.PointOptions.ValueNumericOptions.Format = NumericFormat.Percent;

\_pieSeries.PointOptions.ValueNumericOptions.Precision = 0;

ChartTitle chartTitle1 = new ChartTitle();

chartTitle1.Text = pp投标单位.DeptName + " " + "投标情况";

chartTitle1.Font = new Font("宋体", 14);

ccPieChart.Titles.Clear();

ccPieChart.Titles.Add(chartTitle1);

ccPieChart.SelectionMode = ElementSelectionMode.Single; //设置饼状图是否可选

ccPieChart.SeriesSelectionMode = SeriesSelectionMode.Point;

ccPieChart.SelectedItemsChanged += new SelectedItemsChangedEventHandler(ccPieChart\_SelectedItemsChanged);

}

string GetProjectTypeIDFromPie()

{

List<int> paletteIndexes = new List<int>();

List<DataFilter> filters = new List<DataFilter>();

//chartBars.SeriesTemplate.DataFilters.Clear();

string mm项目类型ID = string.Empty;

if (ccPieChart.SelectedItems.Count > 0)

{

DataRowView dataRow = (DataRowView)ccPieChart.SelectedItems[0];

if (dataRow != null)

{

mm项目类型ID = dataRow["项目类型ID"].ToString();

}

//foreach (DataRowView dataRow in ccPieChart.SelectedItems)

//{

//}

}

return mm项目类型ID;

}

void ccPieChart\_SelectedItemsChanged(object sender, SelectedItemsChangedEventArgs e)

{

string mm项目类型ID=GetProjectTypeIDFromPie();

if (!string.IsNullOrEmpty(mm项目类型ID))

{

DataTable mmDT\_Result = DbHelperMySQL.GetList("select t.\*,t.项目ID as ID,'' as ParentID,'2' as ImageIndex from VV\_项目基本信息 t where 2>1" + (!string.IsNullOrEmpty(mm项目类型ID) ? " and t.项目类型ID='" + mm项目类型ID + "'" : "") + \_Filter + \_DeptFilter + \_DeptFilter\_V2 + " order by t.招标时间 desc");

绑定历史报价列表(\_投标单位, mmDT\_Result);

}

}

private void ccPieChart\_ObjectSelected(object sender, HotTrackEventArgs e)

{

if (!(e.Object is Series))

{

e.Cancel = true;

ccPieChart.SelectedItems.Clear();

}

//else

//{

// Series \_pieSeries = (Series)e.Object;

//}

}

private void ccPieChart\_ObjectHotTracked(object sender, HotTrackEventArgs e)

{

//if (!(e.Object is Series))

//{

// e.Cancel = true;

//}

}

**曲线图**

DevExpress Demos 14.2\Components\WinForms\CS\ChartsMainDemo\Modules

\DataAggregation\CurrencyExcangeRates.cs

**一个完整的例子**：

private DataTable CreateData()

{

DataTable dt = new DataTable();

dt.Columns.Add(new DataColumn("类型"));

dt.Columns.Add(new DataColumn("2005-1月", typeof(decimal)));

dt.Columns.Add(new DataColumn("2005-2月", typeof(decimal)));

dt.Columns.Add(new DataColumn("2005-3月", typeof(decimal)));

dt.Columns.Add(new DataColumn("2005-4月", typeof(decimal)));

dt.Columns.Add(new DataColumn("2005-5月", typeof(decimal)));

dt.Columns.Add(new DataColumn("2005-6月", typeof(decimal)));

dt.Rows.Add(new object[] { "员工人数", 437, 437, 414, 397, 387, 378 });

dt.Rows.Add(new object[] { "人均月薪", 3964, 3961, 3979, 3974, 3967, 3972 });

dt.Rows.Add(new object[] { "成本TEU", 3104, 1339, 3595.8, 3154.5, 2499.8, 3026 });

dt.Rows.Add(new object[] { "人均生产率", 7.1, 3.06, 8.69, 7.95, 6.46, 8.01 });

dt.Rows.Add(new object[] { "占2005年3月人数比例", 1.06, 1.06, 1, 0.96, 0.93, 0.91 });

return dt;

}

private void CreateChart(DataTable dt)

{

#region Series

//创建几个图形的对象

Series series1 = CreateSeries("员工人数", ViewType.Line, dt, 0);

Series series2 = CreateSeries("人均月薪", ViewType.Line, dt, 1);

Series series3 = CreateSeries("成本TEU", ViewType.Line, dt, 2);

Series series4 = CreateSeries("人均生产率", ViewType.Line, dt, 3);

Series series5 = CreateSeries("占2005年3月人数比例", ViewType.Line, dt, 4);

#endregion

List<Series> list = new List<Series>() { series1, series2, series3, series4, series5 };

ccDetailChart.Series.AddRange(list.ToArray());

ccDetailChart.Legend.Visible = false;

ccDetailChart.SeriesTemplate.LabelsVisibility = DevExpress.Utils.DefaultBoolean.True;

for (int i = 0; i < list.Count; i++)

{

list[i].View.Color = Common.colorlist[i];

CreateAxisY(list[i]);

}

}

/// <summary>

/// 根据数据创建一个图形展现

/// </summary>

/// <param name="caption">图形标题</param>

/// <param name="viewType">图形类型</param>

/// <param name="dt">数据DataTable</param>

/// <param name="rowIndex">图形数据的行序号</param>

/// <returns></returns>

private Series CreateSeries(string caption, ViewType viewType, DataTable dt, int rowIndex)

{

Series series = new Series(caption, viewType);

for (int i = 1; i < dt.Columns.Count; i++)

{

string argument = dt.Columns[i].ColumnName;//参数名称

decimal value = (decimal)dt.Rows[rowIndex][i];//参数值

series.Points.Add(new SeriesPoint(argument, value));

}

//必须设置ArgumentScaleType的类型，否则显示会转换为日期格式，导致不是希望的格式显示

//也就是说，显示字符串的参数，必须设置类型为ScaleType.Qualitative

series.ArgumentScaleType = ScaleType.Qualitative;

series.LabelsVisibility = DevExpress.Utils.DefaultBoolean.True;//显示标注标签

return series;

}

/// <summary>

/// 创建图表的第二坐标系

/// </summary>

/// <param name="series">Series对象</param>

/// <returns></returns>

private SecondaryAxisY CreateAxisY(Series series)

{

SecondaryAxisY myAxis = new SecondaryAxisY(series.Name);

((XYDiagram)ccDetailChart.Diagram).SecondaryAxesY.Add(myAxis);

((LineSeriesView)series.View).AxisY = myAxis;

myAxis.Title.Text = series.Name;

myAxis.Title.Alignment = StringAlignment.Far; //顶部对齐

myAxis.Title.Visible = true; //显示标题

myAxis.Title.Font = new Font("宋体", 9.0f);

Color color = series.View.Color;//设置坐标的颜色和图表线条颜色一致

myAxis.Title.TextColor = color;

myAxis.Label.TextColor = color;

myAxis.Color = color;

return myAxis;

}

上例具体的使用：

DataTable mmDT = CreateData();

CreateChart(mmDT);