1. ChartFactory类及常用方法
   1. 柱形图 public static JFreeChart createBarChart() 创建一个常规的柱形图对象

public static JFreeChart createBarChart3D() 创建一个3D效果的柱形图对象

* 1. 饼形图 public static JFreeChart createPieChart() 创建一个常规的饼形图对象

public static JFreeChart createPieChart3D() 创建一个3D 效果的饼形图对象

* 1. 区域图 public static JFreeChart createAreaChart() 创建一个常规的区域图对象
  2. 折线图 public static JFreeChart createLineChart() 创建一个常规的折线图对象

public static JFreeChart createLineChart3D() 创建一个3D 效果的折线图对象

* 1. 时序图 public static JFeeChart createTimeSeriesChart() 创建一个常规的时序图对象

实例：

JFreeChart chart=ChartFactory.createPieChart3D(

"饼形图", //图表的标题

initPieData() , //饼形图的数据集对象

true , //是否显示图列

true , //是否显示提示文

false); //是否生成超链接

1. 数据集对象
   1. DefaultCategoryDataset类 默认的类别数据集合对象，可用于创建柱形图、区域图数据集合等
   2. DefaultPieDataset类 默认的饼形图数据集合对象，可用于创建饼形图数据集合
   3. XYSeriesCollection类 描述坐标轴序列类型的数据集合，可用于创建折线图等数据集合
   4. TimeSeriesCollection类 描述时间序列的数据集合，可用于创建时序图等数据集合

实例：

DefaultCategoryDataset dataset=new DefaultCategoryDataset(); //创建数据集合

dataSet.addValue(100,"长春","土豆"); //向数据集合中添加数据

1. 绘图区对象

Plot

* 1. public void setBackgroundImage(Image image) 设置数据区的背景图片
  2. public void setBackgroundImageAlignment(int alignment) 设置数据区的背景图片对齐方式
  3. public void setBackgroundAlpha(float alpha) 设置数据区的背景透明度，范围在0.0～1.0间
  4. public void setForegroundAlpha(float alpha) 设置数据区的前景透明度，范围在0.0～1.0间
  5. public void setDataAreaRatio(double ratio) 设置数据区占整个图表区的百分比
  6. public void setOutLinePaint(Paint paint) 设置数据区的边界线条颜色
  7. public void setNoDataMessage(String message) 设置没有数据时显示的消息

PiePlot

1. public void setDataset(PieDataset dataset) 设置绘制图表所需要的数据集合
2. public void setDataset(PieDataset dataset) 设置绘制图表所需要的数据集合
3. public void setCircular(boolean flag) 设置饼形图是否一定是正圆
4. public void setStartAngle(double angle) 设置饼形图的初始角度
5. public void setDirection(Rotation direction) 设置饼形图的旋转方向
6. public void setExplodePercent(int section,double percent) 设置抽取图表的部分及距离
7. public void setLabelFont(Font font) 设置分类标签字体（3D效果下无效）
8. public void setLabelPaint(Paint paint) 设置分类标签字体颜色（3D效果下无效）

CategoryPlot

1. public void setDataset(PieDataset dataset) 设置绘制图表所需要的数据集合
2. public void setColumnRenderingOrder(SortOrder order) 设置数据分类的排序方式
3. public void setAxisOffset(Spacer offset) 设置坐标轴到数据区的间距
4. public void setOrientation(PlotOrientation orientation) 设置数据区的方向（横向或纵向）
5. public void setDomainAxis(CategoryAxis axis) 设置数据区的分类轴
6. public void setRangeAxis(ValueAxis axis) 设置数据区的数据轴
7. public void addAnnotation(CategoryAnnotation annotation) 设置数据区的注释

XYPlot

1. public ValueAxis getDomainAxis 返回X轴
2. public ValueAxis getRangeAxis() 返回Y轴
3. public void setDomainAxis(ValueAxis axis) 设置X轴
4. public void setRangeAxis(ValueAxis axis) 设置Y轴
5. 坐标轴对象

Axis

* 1. public void setVisible(boolean flag) 设置坐标轴是否可见
  2. public void setAxisLinePaint(Paint paint) 设置坐标轴线条颜色，此设置在3D效果下无效
  3. public void setAxisLineVisible(boolean visible) 设置坐标轴线条是否可见
  4. public void setLabel(String label) 设置坐标轴标题
  5. public void setLabelFont(Font font) 设置坐标轴标题字体
  6. public void setLabelPaint(Paint paint) 设置坐标轴标题颜色
  7. public void setLabelAngle(double angle) 设置坐标轴标题旋转角度

CategoryAxis

1. public void setCategoryMargin(double margin) 设置分类轴边距
2. public void setLowerMargin(double margin) 设置分类轴下边距或左边距
3. public void setUpperMargin(double margin) 设置分类轴上边距或右边距
4. public void setVerticalCategoryLabels(boolean flag) 设置分类轴标题是否旋转到垂直
5. public void setMaxCategoryLabelWidthRatio(float ratio) 设置分类轴分类标签的最大宽度

ValueAxis

1. public void setAutoRange(boolean auto) 设置数据轴数据范围是否为自动
2. public void setFixedAutoRange(double length) 设置数据轴固定数据范围
3. public void setInverted(boolean flag) 设置数据轴是否反向
4. public void setLowerMargin(double margin) 设置分类轴下边距或左边距
5. public void setUpperMargin(double margin) 设置分类轴上边距或右边距
6. public void setLowerBound(double min) 设置数据轴上的显示最小值
7. public void setUpperBound(double max) 设置数据轴上的显示最大值

NumberAxis

1. public void setAutoRangeIncludesZero(boolean flag) 设置是否强制在自动选择的数据范围中包含0
2. public void setAutoRangeStickyZero(boolean flag) 设置是否强制在整个数据轴中包含0，即使0不在数据范围中
3. public void setNumberFormatOverride(NumberFormat formatter) 设置数据轴数据标签的显示格式

DateAxis

1. public void setMaximumDate(Date maximumDate) 设置日期轴上的最小日期
2. Public void setMinimumDate(Date minimumDate) 设置日期轴上的最大日期
3. public void setDateFormatOverride(DateFormat formatter) 设置日期轴日期标签的显示格式
4. public void setTickUnit(DateTickUnit unit) 设置日期轴的日期标签
5. 图片处理

JFreeChart组件可以能够生成.gpeg/.png类型的图片，并由JFreeChart对生成的图片进行管理并返回图片路径

生成及访问方式：

1. 调用 ServletUtilities 类的 saveChartAsJPEG() 方法

public static String saveChartAsJPEG(JFreeChart chart ,int width,int height ,HttpSession session )throws IOException

参数说明：

1. chart 制图对象JFreeChartford
2. width 所生成图片的宽度
3. height 所生成图片的高度
4. session HttpSession对象
5. Servlet配置

<servlet>

<servlet-name>DisplayChart</servlet-name>

<servlet-class>org.jfree.chart.servlet.DisplayChart</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>DisplayChart</servlet-name>

<url-pattern>/servlet/DisplayChart</url-pattern>

</servlet-mapping>

1. JDBC操作
   1. 直接通过JDBC操作数据集合

public JDBCPieDataset(String url,String driverName,String user,String password )

* 1. 执行dataset.executeQuery(query)得到结果集

1. 中文乱码问题解决

StandardChartTheme standardChartTheme = new StandardChartTheme("CN"); //创建主题样式

standardChartTheme.setExtraLargeFont(new Font("隶书", Font.BOLD, 20)); //设置标题字体

standardChartTheme.setRegularFont(new Font("宋体", Font.PLAIN, 15)); //设置图例的字体

standardChartTheme.setLargeFont(new Font("宋体", Font.PLAIN, 15)); //设置轴向的字体

ChartFactory.setChartTheme(standardChartTheme); //应用主题样式

1. 生成饼状图实例

<servlet>

<servlet-name>tp</servlet-name>

<servlet-class>org.jfree.chart.servlet.DisplayChart</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>tp</servlet-name>

<url-pattern>/tp</url-pattern>

</servlet-mapping>

public static JFreeChart createChart(){

JFreeChart jfreeChart = ChartFactory.createPieChart("饼状图", createDataset(), true, true, false); //创建JFreeChart对象

TextTitle localTextTitle = jfreeChart.getTitle(); //获取标题

localTextTitle.setFont(new Font("宋体",0,15)); //标题字体样式

PiePlot localPiePlot = (PiePlot)jfreeChart.getPlot(); //获取绘图区对象

localPiePlot.setLabelFont(new Font("宋体", 0, 12)); //设置分类标签的字体

localPiePlot.setCircular(false); //设置饼图是否一定是正圆

return jfreeChart;

}

private static PieDataset createDataset(){

DefaultPieDataset dataset = new DefaultPieDataset(); //创建饼图的数据集合

dataset.setValue("One", new Double(43.200000000000003D)); //向数据集合中添加数据

dataset.setValue("Two", new Double(10.0D));

dataset.setValue("Three", new Double(20.5D));

dataset.setValue("Four", new Double(14.5D));

dataset.setValue("Five", new Double(11.0D));

return dataset; //返回数据集合

}

String chartUrl = "localhost:8080/tp?filename=" + filename; //通过此路径可以访问图片