四川奔腾信息产业公司

**报表处理\_工具说明文档**

|  |  |
| --- | --- |
| 编 制 | 虞晓东 |
| 审 核 |  |
| 批 准 |  |

二〇〇九年六月二十四日

修改历史

|  |  |  |  |
| --- | --- | --- | --- |
| **日 期** | **作者/修订者** | **版本** | **修改原因** |
| 2009-6-24 | 虞晓东 | 1.0 | 创建 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目 录

[1、目的 1](#_Toc235525000)

[2、名称与编号 1](#_Toc235525001)

[3、应用环境 1](#_Toc235525002)

[4、适用范围 1](#_Toc235525003)

[5、输入参数 1](#_Toc235525004)

[6、输出结果 1](#_Toc235525005)

[7、涉及的其它工具 1](#_Toc235525006)

[8、使用说明 1](#_Toc235525007)

[9、使用示例 2](#_Toc235525008)

# 1、目的

提供报表处理使用的说明。

# 2、名称与编号

名称：报表处理

编号：

# 3、应用环境

本工具应用J2EE环境,需要Jasperreport第三方开源组件支持。

# 4、适用范围

本工具适用J2EE运行环境。

# 5、输入参数

# 6、输出结果

# 7、涉及的其它工具

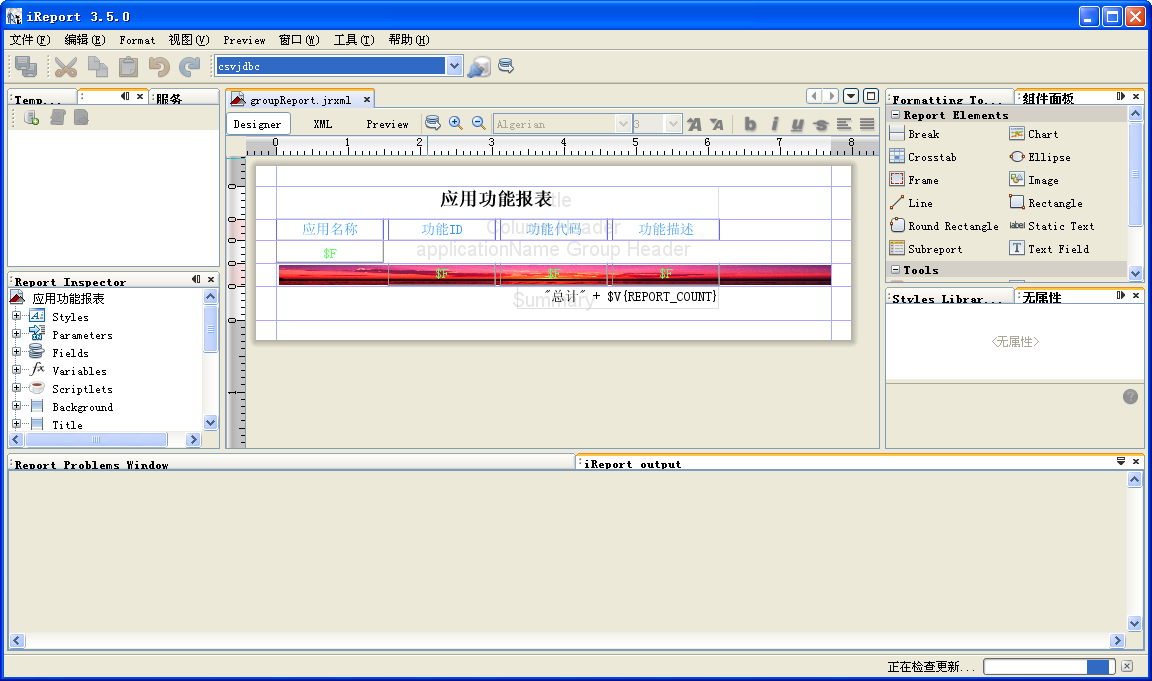
暂无。

# 8、使用说明

1. 设计报表布局模板，可通过Ireport设计或编程实现；
2. 构造数据源，有三种方式：JDBC的connection连接、数组、集合；
3. 构造报表需要的参数Map
4. 构造JasperReportHelper实例
5. 输出指定格式(HTML、PDF、EXCEL)的报表

# 9、使用示例

1. 设计报表布局模板，可通过Ireport设计或编程实现；
2. 通过Ireport设计，如下图



1. 编程实现，采用JasperDesign方式如下

private JasperDesign createJasperDesign(HttpServletRequest request) throws JRException{

//新建一个设计器

JasperDesign jasperDesign = new JasperDesign();

jasperDesign.setName("销售数据列表");

//设置SQL语句

JRDesignQuery query = new JRDesignQuery();

query.setText("SELECT COMM\_BARCODE,NAME,AREA\_CODE,COMM\_SORT,SALE\_TIME,UNIT,AMOUNT,PRICE,SUM\_PRICE FROM \"ROLLUP\_ENTERPRISE\_COMMBARCODE.txt\"");

jasperDesign.setQuery(query);

JRDesignBand title = new JRDesignBand();

title.setHeight(50);

//定义报表的Title

JRDesignStaticText titleText = new JRDesignStaticText();

titleText.setText("销售数据列表");

titleText.setX(230);

titleText.setFontSize(20);

titleText.setHeight(50);

titleText.setWidth(300);

titleText.setFontName("宋体");

titleText.setFontSize(14);

titleText.setPdfFontName("STSong-Light");

titleText.setPdfEncoding("UniGB-UCS2-H");

titleText.setPdfEmbedded(true);

title.addElement(titleText);

jasperDesign.setTitle(title);

String[] headers = { "COMM\_BARCODE", "NAME"

, "AREA\_CODE", "COMM\_SORT"

, "SALE\_TIME", "UNIT"

, "AMOUNT"

, "PRICE", "SUM\_PRICE"};

String[] headersname = { "条码", "名称"

, "地区编码", "种类编码"

, "销售时间", "单位"

, "数量"

, "价格", "总价"};

JRDesignBand columnHeader = new JRDesignBand();

columnHeader.setHeight(30);

JRDesignBand detail = new JRDesignBand();

detail.setHeight(30);

//定义一种显示边框的风格

JRDesignStyle style = new JRDesignStyle();

JRLineBox lineBox = style.getLineBox();

lineBox.setPadding(1);

JRPen pen = lineBox.getPen();

pen.setLineColor(Color.BLUE);

pen.setLineWidth(JRPen.LINE\_WIDTH\_1);

pen.setLineStyle(JRPen.LINE\_STYLE\_SOLID);

style.setName("style");

style.setFontName("宋体");

style.setFontSize(14);

style.setPdfFontName("STSong-Light");

style.setPdfEncoding("UniGB-UCS2-H");

style.setPdfEmbedded(true);

JRDesignImage image = new JRDesignImage(jasperDesign);

image.setX(1);

image.setY(1);

image.setHeight(29);

image.setWidth(250);

image.setScaleImage(JRImage.SCALE\_IMAGE\_FILL\_FRAME);

image.setLazy(true);

JRDesignExpression imageExpression = new JRDesignExpression();

imageExpression.setValueClass(String.class);

imageExpression.setText("\"http://" + request.getServerName() + ":" + request.getServerPort() + request.getContextPath() + "/images/jasperreport/Sunset.jpg\"");

image.setExpression(imageExpression);

detail.addElement(image);

//加入风格的设计中

jasperDesign.addStyle(style);

for (int i = 0; i < headers.length; i++) {

// add column headers

JRDesignStaticText staticText = new JRDesignStaticText();

staticText.setText(headersname[i]);

staticText.setFontSize(16);

staticText.setHeight(30);

staticText.setWidth(250);

staticText.setX(250 \* i);

//引用风格style

staticText.setStyleNameReference("style");

columnHeader.addElement(staticText);

// define fields

JRDesignField field = new JRDesignField();

field.setName(headers[i]);

field.setValueClass(String.class);

jasperDesign.addField(field);

// add text fields for displaying fields

JRDesignTextField textField = new JRDesignTextField();

JRDesignExpression expression = new JRDesignExpression();

expression.setText("$F{" + headers[i] + "}");

expression.setValueClass(String.class);

textField.setExpression(expression);

textField.setFontSize(14);

textField.setHeight(30);

textField.setWidth(250);

textField.setX(250 \* i);

//引用风格style

textField.setStyleNameReference("style");

detail.addElement(textField);

}

jasperDesign.setColumnHeader(columnHeader);

jasperDesign.setDetail(detail);

//jasperDesign.setPageHeight(12000);

return jasperDesign;

}

1. 构造数据源，有三种方式：JDBC的connection连接、数组、集合；
2. JDBC的connection连接，如下

private Connection getConnection(HttpServletRequest request) throws ClassNotFoundException, SQLException {

Class.forName("jstels.jdbc.csv.CsvDriver");

Properties props = new java.util.Properties();

props.put("separator","|"); // separator is a bar

props.put("suppressHeaders","false"); // column headers are on the first line

props.put("fileExtension",".txt"); // file extension is .txt

props.put("charset","UTF-8"); // file encoding is "ISO-8859-2"

props.put("commentLine","--"); // string denoting comment line is "--"

// date/time format

props.put("dateFormat","yyyy-MM-dd HH:mm | dd/MM/yyyy");

//创建连接

Connection connection = DriverManager.getConnection("jdbc:jstels:csv:http://" + request.getServerName() + ":" + request.getServerPort() + request.getContextPath() + "/jasperreport/csvfiles", props);

return connection;

}

1. 数组，如下

@SuppressWarnings("unchecked")

private Object[] createArray() {

Object[] objs = new Object[3]();

DataBean dataBean0 = new DataBean();

dataBean0.setName("测试1");

dataBean0.setDesc("描述1");

objs[0] = dataBean0;

DataBean dataBean1 = new DataBean();

dataBean1.setName("测试2");

dataBean1.setDesc("描述2");

objs[1] = dataBean1;

DataBean dataBean2 = new DataBean();

dataBean2.setName("测试3");

dataBean2.setDesc("描述3");

objs[2] = dataBean2;

return objs;

}

1. 集合，如下

@SuppressWarnings("unchecked")

private Collection createColl() {

Collection coll = new ArrayList();

DataBean dataBean = new DataBean();

dataBean.setName("测试1");

dataBean.setDesc("描述1");

coll.add(dataBean);

dataBean = new DataBean();

dataBean.setName("测试2");

dataBean.setDesc("描述2");

coll.add(dataBean);

dataBean = new DataBean();

dataBean.setName("测试3");

dataBean.setDesc("描述3");

coll.add(dataBean);

return coll;

}

1. 构造报表需要的参数Map

//构造参数

Map parameterMap = new HashMap();

//得到报表的绝对路径

String subJasperFileName = approot + "jasperreport\\report\\connection\\";

parameterMap.put("SUBREPORT\_DIR", subJasperFileName);

1. 构造JasperReportHelper实例

JasperReportHelper jasperReportHelper = new JasperReportHelper(jasperFileName, parameterMap, connection);

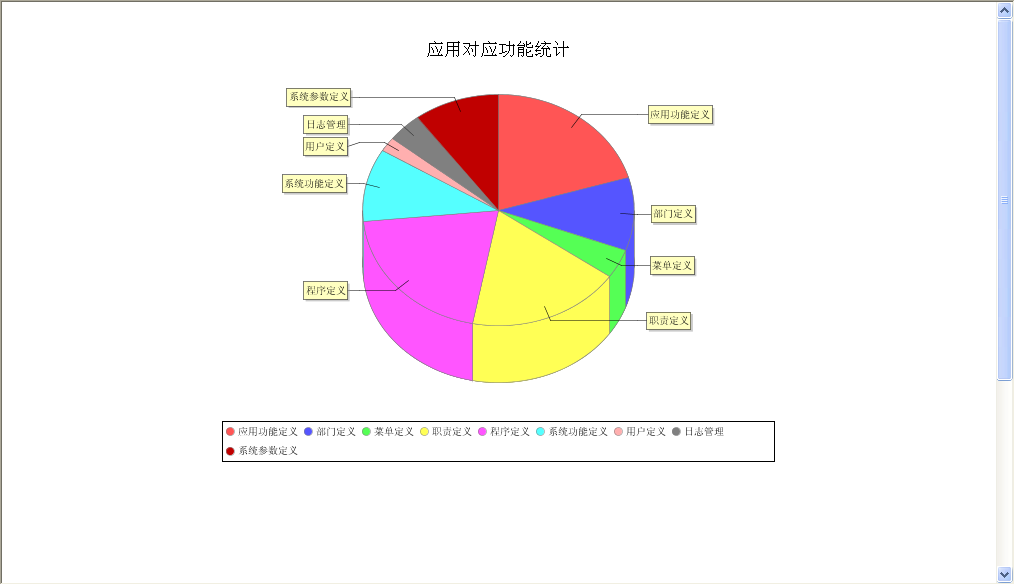
1. 输出指定格式(HTML、PDF、EXCEL)的报表

…

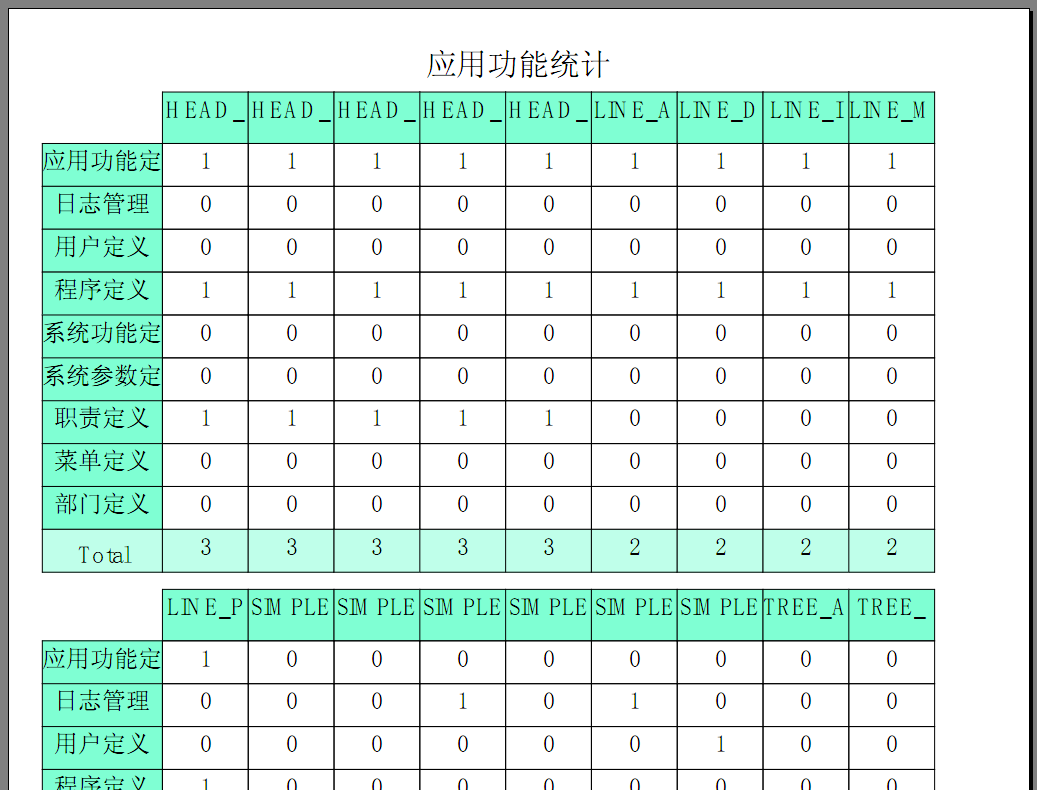
jasperReportHelper.exportReportToPdf(servletOutputStream);

…

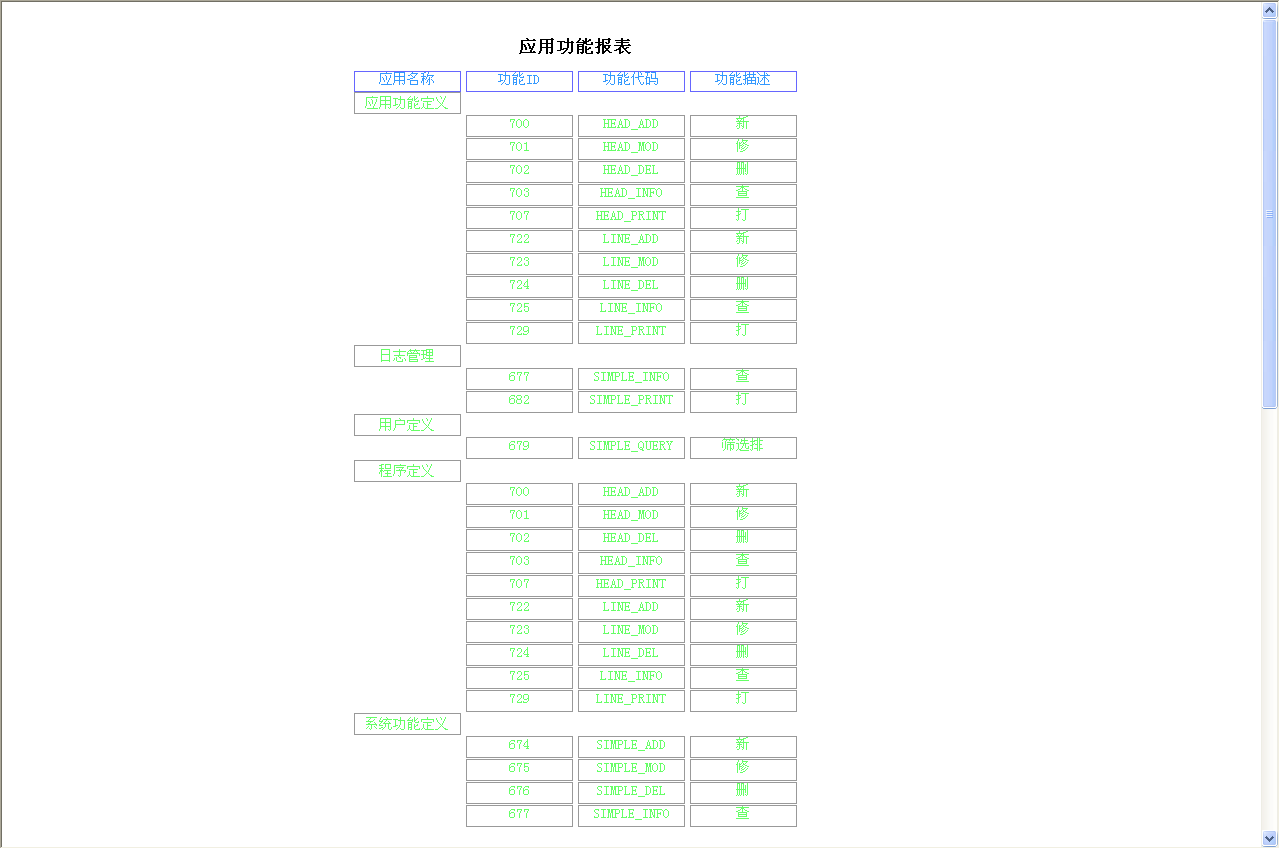
结果效果图：



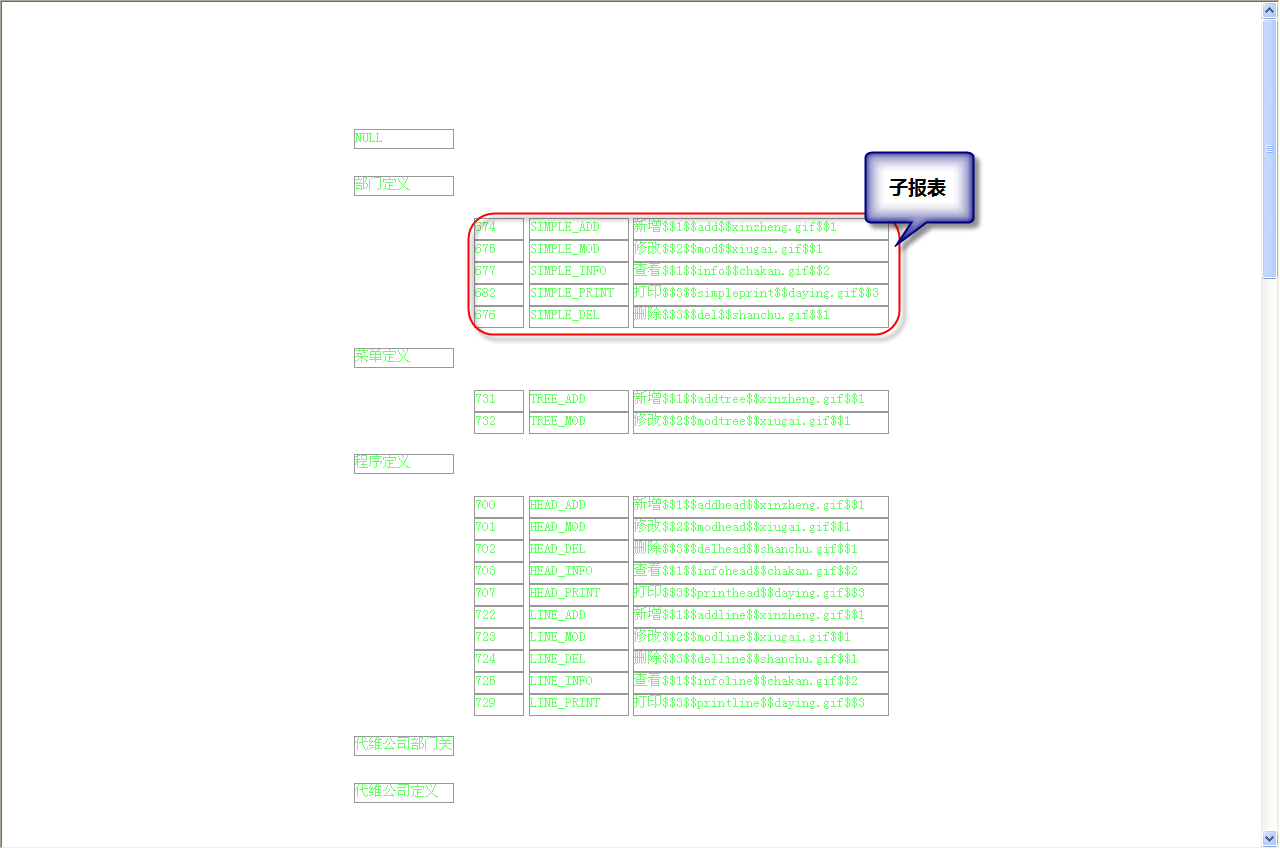
图形报表（HTML格式）



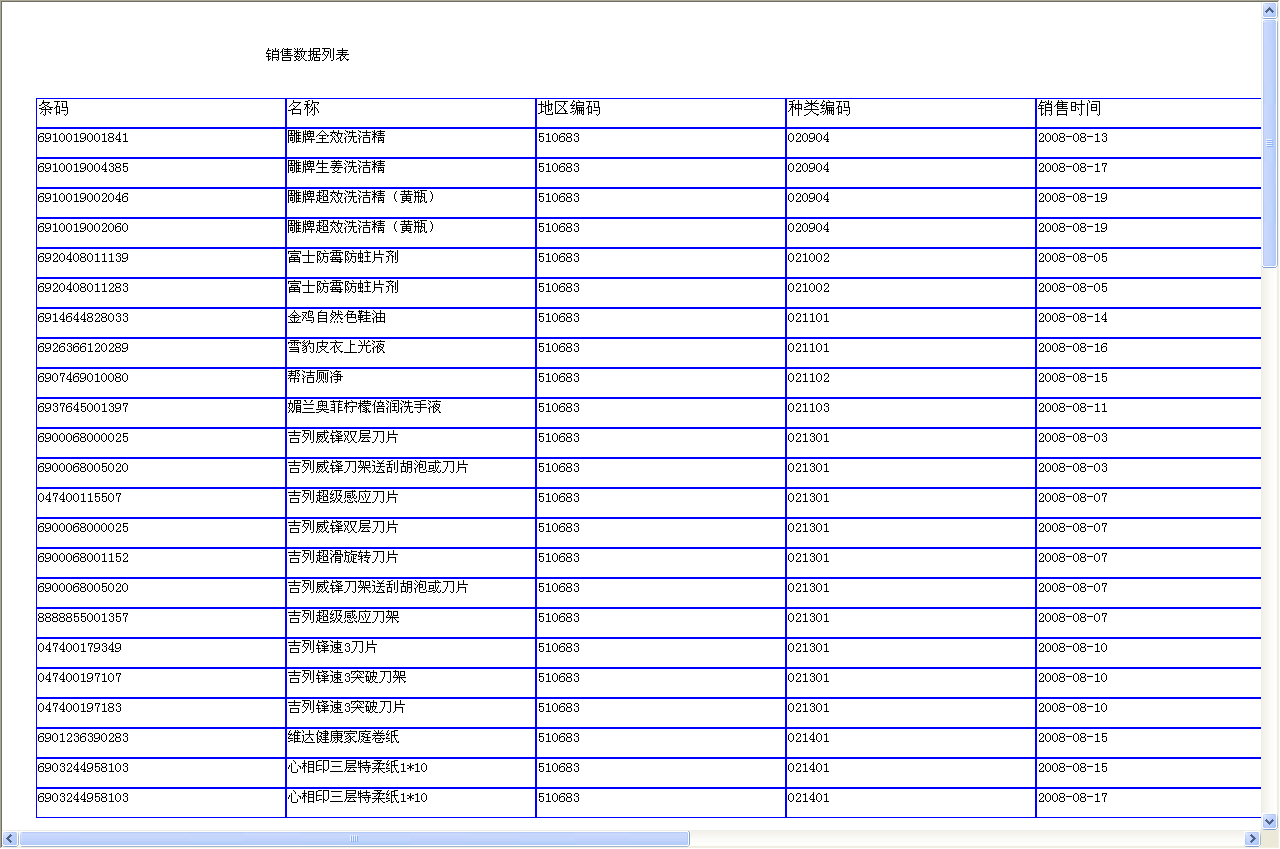
交叉报表(PDF格式)



分组报表（HTML格式）



子报表（HTML格式）



JasperDesign设计报表（HTML格式）

注：DEMO请参见UtilsWeb项目的WebRoot\jasperreport目录