



主页 | 博客 | 相册 | 个人档案 | 好友

查看文章

## 使用JFreeReport生成报表

2007-09-03 20:30



### 文章摘要

JFreeReport用于生成报表。JFreeReport的数据继承自Swing组件的TableModel接口，使用基于XML的报表格式定义文件对报表进行格式化。JFreeReport生成的报表可以分页预览、打印或者保存为多种格式的文件如pdf、Excel、html等。作者在本文中详细的介绍了如何定义报表格式定义文件、如何使用JFreeReport生成 / 预览报表和将报表转为其它各种格式。

### 关键词

### JFreeReport 报表生成 预览

报表生成一直是企业信息化过程中重要的一环，也是目前比较难于实现的一环，今天作者给大家介绍一种新的报表生成组件——JFreeReport。JFreeReport是JFreeReport.Org基于LGPL授权协议提供的一组java包，用于生成各类报表，JFreeReport的数据继承自Swing组件的TableModel接口，使用基于XML的报表格式定义文件对报表进行格式化。JFreeReport生成的报表可以分页预览、打印，而且支持导出为多种格式的文件如pdf、Excel、CSV、html等。更重要的是，JFreeReport不仅支持基于C/S结构的系统，而且支持基于B/S结构的系统中的在线报表显示。更详细的关于JFreeReport的介绍请大家访问JFreeReport的[官方网站JFree.org](http://www.jfree.org)。

#### 1 环境准备

##### 1.1 JFreeReport组件

请大家到[http://prdownloads.sourceforge.net/jfreereport/jfreereport-0.8.4\\_7.zip?download](http://prdownloads.sourceforge.net/jfreereport/jfreereport-0.8.4_7.zip?download)下载JFreeReport组件，下载的是一个ZIP文件，然后将ZIP文件解压缩到c:\jfreereport（后面的章节中将使用%jfreereport\_home%表示这个目录）目录下。

##### 1.2 JFreeReport扩展组件

请大家到[http://www.jfree.org/jfreereport/jfreereport-ext-0.8.4\\_7.zip](http://www.jfree.org/jfreereport/jfreereport-ext-0.8.4_7.zip)下载JFreeReport扩展组件，他用于支持JFreeReport组件生成的报表的在线显示。请大载后解压缩到c:\jfreereport-ext目录下（后面的章节中将使用%jfreereport\_ext\_home%表示这个目录）

##### 1.3 Ant工具

Apache公司提供的—个基于JAVA的自动化脚本引擎，请大家到<http://ant.apache.org/>下载ant的可执行文件，关于如何使用ant请大家查看ant的帮助文档或者<http://ant.apache.org/>网站上的在线帮助文档。示例中主要是用ant来负责编译java代码。

##### 1.4 作者提供的代码

为了运行本文中作者提到的例子和相关资源文件，请大家下载作者提供的vivianjDemo.zip文件和中文转换工具gb2unicode.jar。然后解压缩到%jfreereport\_home%\vivianjDemo（后面的章节中将使用%demo\_home%表示这个目录）目录下。

#### 2 JFreeReport生成报表的基本步骤

我们首先演示一个简单的例子，说明使用JFreeReport生成报表的一些必要的步骤。

##### 2.1 实例说明

在这个例子中，我们将循环生成100条数据放入TableModel中，然后使用JFreeReport组件提供的预览功能在屏幕上显示生成的报表。

【注】为了简化，这里仅仅是逐条显示数据，不作任何修饰和统计工作，所以也不使用报表格式定义文件。

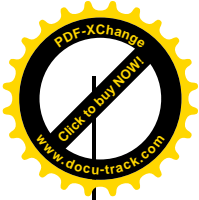
##### 2.2 代码编制

整个演示实例（HelloWorld.java）的代码和相关注释如下，如果你执行了1.3中规定的步骤，你可以在%demo\_home%\src\org\vivianj\jfreereport/看到这个文件。

```
/**
 * HelloWorld.java
 */

package org.vivianj.jfreereport;

import java.awt.Color;
```



```
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;
import java.awt.geom.Point2D;
import javax.swing.table.DefaultTableModel;
import javax.swing.table.TableModel;

import org.jfree.report.Boot;
import org.jfree.report.ElementAlignment;
import org.jfree.report.JFreeReport;
import org.jfree.report.ReportProcessingException;
import org.jfree.report.elementfactory.TextFieldElementFactory;
import org.jfree.report.modules.gui.base.PreviewDialog;
import org.jfree.ui.FloatDimension;

/**
 * 使用JFreeReport生成报表的简单例子，用于演示使用JFreeReport生成报表的一些基本步骤
 *
 * 本例子中，为了简化操作，报表定义是使用java直接编码
 *
 * @ 作者：bookman
 */
public class HelloWorld
{
    /**
     * 处理窗口关闭事件
     */
    protected static class CloseHandler extends WindowAdapter
    {
        public void windowClosing(final WindowEvent event)
        {
            System.exit(0);
        }
    }

    /**
     * 创建和显示简单的报表
     */
    public HelloWorld()
    {
        // 获得创建报表需要用到的数据
        final TableModel data = createData();

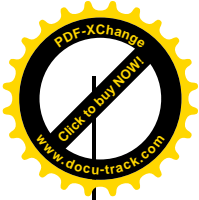
        //获得报表要用到的报表定义内容
        final JFreeReport report = createReportDefinition();

        //将报表定义和数据结合
        report.setData(data);

        try
        {
            //将生成的报表放到预览窗口中
            final PreviewDialog preview = new PreviewDialog(report);
            preview.addWindowListener(new CloseHandler());
            preview.pack();

            //显示报表预览窗口

```



```
preview.setVisible(true);

}

catch (ReportProcessingException e)

{

    System.out.println(e);

}

}

/**

 * 创建生成报表需要用到的数据

 *

 * @返回一个TableModel实例

 */

private TableModel createData()

{

    final Object[] columnNames = new String[]{"Column1", "Column2"};

    final DefaultTableModel result = new DefaultTableModel(columnNames, 100);

    int rownum = 0;

    int colnum = 0;

    for (;rownum < 100 ; rownum++)

    {

        result.setValueAt("say Hello " + rownum + "次", rownum, 0);

        result.setValueAt("say World " + rownum + "次", rownum, 1);

    }

    return result;

}

/**

 * 创建一个报表定义

 *

 * @返回一个报表定义实例

 */

private JFreeReport createReportDefinition()

{

    final JFreeReport report = new JFreeReport();

    report.setName("A Very Simple Report");

    /**

     * 定义要显示报表第一列的样式

     */

    TextFieldElementFactory factory = new TextFieldElementFactory();

    factory.setName("T1");

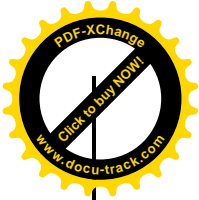
    factory.setAbsolutePosition(new Point2D.Float(0, 0));

    factory.setMinimumSize(new FloatDimension(150, 20));

    factory.setColor(Color.black);

    factory.setHorizontalAlignment(ElementAlignment.LEFT);

    factory.setVerticalAlignment(ElementAlignment.MIDDLE);
```



```
factory.setNullString("-");
factory.setFieldName("Column1");
report.getItemBand().addElement(factory.createElement());

/**
 * 定义要显示报表第二列的样式
 */

factory = new TextFieldElementFactory();
factory.setName("T2");
factory.setAbsolutePosition(new Point2D.Float(200, 0));
factory.setMinimumSize(new FloatDimension(150, 20));
factory.setColor(Color.black);
factory.setHorizontalAlignment(ElementAlignment.LEFT);
factory.setVerticalAlignment(ElementAlignment.MIDDLE);
factory.setNullString("-");
factory.setFieldName("Column2");
report.getItemBand().addElement(factory.createElement());

/**
 * 返回一个报表定义的实例
 */

return report;

}

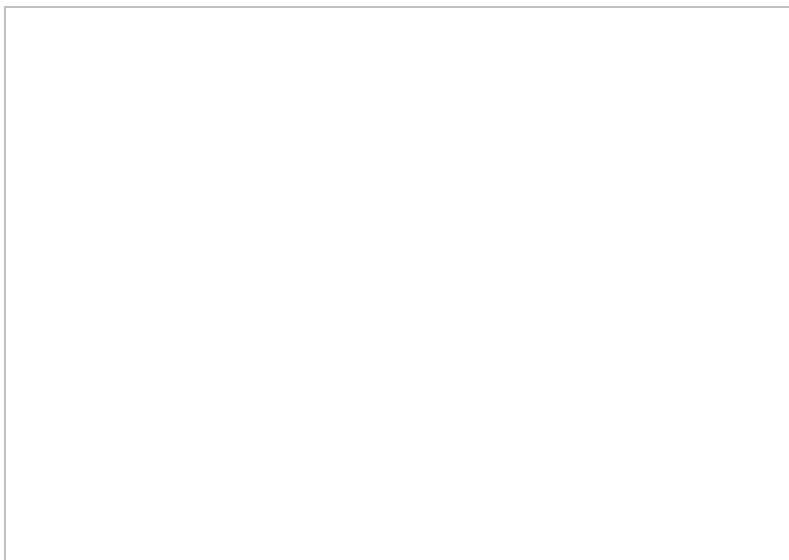
public static void main(final String[] args)
{
    // 初始化JFreeReport
    Boot.start();

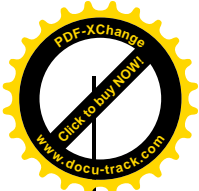
    //调用演示实例
    new HelloWorld();
}


}
```

### 2.3 运行例子

如果你执行了1.3中规定的步骤，你可以进入命令行界面，然后进入%demo\_home%目录下，修改setenv.cmd中的相关设置，执行serenv.cmd设置环境变量。执行java org.vivianj.jfreereport.HelloWorld查看运行结果。下面这个图片是作者执行后结果的屏幕截图：





大家可以看到，JFreeReport已经自动帮我们实现了分页。上面这个图片显示的是第一页的数据，你可以通过工具栏中的查看其后面中的内容。

### 基本步骤解释

使用JFreeReport生成报表通常需要以下三个基本步骤：

生成可通过TableModel接口访问的数据，如本例中的createData方法完成的功能  
生成一个JFreeReport实例，他定义了我们如何格式化显示数据，如本例中的createReportDefinition方法完成的功能  
将数据和JFreeReport实例连接起来，并且将该JFreeReport实例传给PreviewDialog的一个实例显示给用户

### 3 使用JFreeReport生成复杂报表

#### 3.1 报表定义文件

报表定义文件是JFreeReport生成复杂报表的重要文件，他就是一个XML文档，主要描述如何使用指定的格式生成复杂的报表，同时使用报表定义文件也可以在报表格式需要修改时只需要更新该报表定义文件，而不需要修改应用代码。

##### 3.1.1 报表定义文件分类

JFreeReport中使用了两种基于XML的报表定义文件来保存报表定义信息：简单格式和扩展格式，很明显,简单格式不能够完全的描述JFreeReport支持的全部报表定义信息,但是他更易于上手使用。而扩展格式则能够对JFreeReport的报表定义提供完整的支持，但是扩展格式太详细了，不太容易使用。

关于这两种报表定义格式文件所支持的标签内容以及如何编写这两种格式的报表定义文件请大家参考%jfreereport\_home%下的jfreereport-0.8.3-A4.pdf中的相关部分，该文件附录中还包括了这两种格式的报表定义文件的DTD文档供大家参考，当然大家也JFreeReport的例子中提供了多种形式的报表定义文件，基本上涵盖了常用的报表格式定义，大家可以参考这些例子编写自己的报表定义文件，。

#### 3.2 代码编制

这个例子和2.2中的代码基本一致，只是报表定义内容不再由java编码实现，而是由报表定义文件提供，所以调用方面稍微有点不同，详细的代码如下，请大家注意其中加粗显示的部分：

```
/**
 * JFreeReport.java
 */

package org.vivianj.jfreereport;

import java.io.File;
import java.text.MessageFormat;
import javax.swing.table.TableModel;
import javax.swing.JFrame;

import org.jfree.ui.RefineryUtilities;
import org.jfree.report.Boot;
import org.jfree.report.modules.gui.base.PreviewFrame;
import org.jfree.report.modules.parser.base.ReportGenerator;
import org.jfree.report.JFreeReport;

import org.vivianj.jfreereport.tablemodel.SampleData;

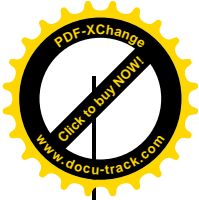
/**
 * 使用JFreeReport生成复杂报表的例子，
 * 用于演示使用JFreeReport生成复杂报表的一些基本步骤
 *
 * 本例子中，报表定义使用了报表定义文件，该文件是保存在c:\下的report3.xml文件
 * 本例中使用的报表定义使用了简单报表定义格式
 *
 * @ 作者：bookman
 */
public class JFreeTest
{

    public JFreeTest(final String urlName, final TableModel data)
    {

        //创建对报表定义文件的引用

        final File in = new File(urlName);

        if (in == null)
        {
```



```
18      System.out.print(" in is null");
19      return;
20  }

21
22      System.out.print("Processing Report: " + in);
23
24      final ReportGenerator gen = ReportGenerator.getInstance();
25
26
27      try
28      {
29          //从报表定义文件中获得要用到的报表定义内容
30
31          final JFreeReport report1 = gen.parseReport(in);
32
33          if (report1 == null)
34          {
35              System.out.print(" report1 is null");
36
37              return;
38          }
39
40          //将报表定义和数据结合
41
42          report1.setData(data);
43
44          //将生成的报表放到预览窗口中
45
46          final PreviewFrame frame1 = new PreviewFrame(report1);
47
48          frame1.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
49
50          frame1.pack();
51
52          RefineryUtilities.positionFrameRandomly(frame1);
53
54          frame1.setVisible(true);
55
56          frame1.requestFocus();
57
58      }
59
60      catch (Exception e)
61      {
62          System.out.print("report.definitionfailure-----\r\n" + e);
63      }
64
65  }
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

public static void main(String[] args)
{
    Boot.start();

    final TableModel data3 = new SampleData();
    JFreeTest jft = new JFreeTest("c:\\report3.xml", data3);
}
}
```

#### 4 中文乱码问题

大家在参看报表定义文件的时候会发现，里面的报表头定义中有一些显示内容，如果你直接将他修改成中文，显示在报表上就会乱码，这是因为这些报表定义文件都是XML文档，他的encoding默认设置是ISO-8859-1，所以出现了中文乱码的问题，解决办法有两个：

1.最简单的方法就是将encoding的设置修改为GB2312

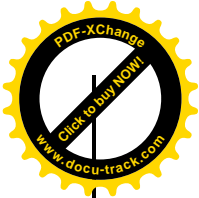
2.还有一个方法就是不修改encoding的设置，而是将这些中文内容使用他们的unicode码来代替。

**【注】** 作者提供了一个GUI的界面提供这种转化功能，你只需要输入中文就可以获得对应的unicode码，请大家下载作者提供的[gb2unicode.jar](#)，设置好java运行环境，然后执行java -jar gb2unicode.jar就可以了。

#### 5 总结

报表问题是企业信息化中比较关注的一个焦点，也比较难于实现客户化，作者在本文中给大家介绍了一个新的报表生成包-JfreeReport,他可以根据XML文档中定义的报表格式，生成符合客户要求的报表。文章中给出了2个简单的例子，演示了如何使用JFreeReport生成报表的详细步骤，希望能够帮助大家熟悉JFreeReport的工作方式。同时文章中给出了使用JFreeReport过程中可能遇到的中文问题的解决办法。

其实JFreeReport还支持在JSP、Servlet中显示报表，只是还需要用到他的扩展组件，作者这里没有给出例子，大家可以参考下载的JFreeReport的扩展组件中的例子。



类别: Eclipse | [分享](#) | [添加到收藏](#) | [分享到贴吧](#) | 浏览(1206) | 评论 (0)  
上一篇: [在Eclipse中导入/导出数据到Excel](#)... 下一篇: [利用iText在JSP中生成PDF报表](#)

最近读者:



[登录](#)后, 您就出现在这里。



[玫瑰、语风](#)



[yf550656330](#)



[peming\\_stone](#)



[uyo\\_fly](#)



[sunc\\_860415](#)



[oker0432](#)



[NCU行星地球](#)



[slier](#)

网友评论:

发表评论:

内 容:

[发表评论](#)