[Java读写Excel之POI超入门](http://rensanning.iteye.com/blog/1538591)

Apache POI 是用Java编写的免费开源的跨平台的 Java API，Apache POI提供API给Java程式对Microsoft Office格式档案读和写的功能。   
  
Apache POI 是创建和维护操作各种符合Office Open XML（OOXML）标准和微软的OLE 2复合文档格式（OLE2）的Java API。用它可以使用Java读取和创建,修改MS Excel文件.而且,还可以使用Java读取和创建MS Word和MSPowerPoint文件。Apache POI 提供Java操作Excel解决方案（适用于Excel97-2008）。   
  
如果处理.xlsx、docx、pptx的话可以试试Docx4j 。   
[Docx4j](http://www.docx4java.org/) is a Java library for creating and manipulating Microsoft Open XML (Word docx, Powerpoint pptx, and Excel xlsx) files. 

* HSSF － 提供读写Microsoft Excel XLS格式档案的功能。
* XSSF － 提供读写Microsoft Excel OOXML XLSX格式档案的功能。
* HWPF － 提供读写Microsoft Word DOC格式档案的功能。
* HSLF － 提供读写Microsoft PowerPoint格式档案的功能。
* HDGF － 提供读Microsoft Visio格式档案的功能。
* HPBF － 提供读Microsoft Publisher格式档案的功能。
* HSMF － 提供读Microsoft Outlook格式档案的功能。

<http://poi.apache.org/>   
  
[Busy Developers' Guide to HSSF and XSSF Features](http://poi.apache.org/spreadsheet/quick-guide.html)   
  
版本：poi-3.8-20120326.jar   
  
1、生成Workbook

**Java代码  收藏代码**

1. //生成Workbook
2. HSSFWorkbook wb = **new** HSSFWorkbook();
4. //添加Worksheet（不添加sheet时生成的xls文件打开时会报错）
5. @SuppressWarnings("unused")
6. Sheet sheet1 = wb.createSheet();
7. @SuppressWarnings("unused")
8. Sheet sheet2 = wb.createSheet();
9. @SuppressWarnings("unused")
10. Sheet sheet3 = wb.createSheet("new sheet");
11. @SuppressWarnings("unused")
12. Sheet sheet4 = wb.createSheet("rensanning");
14. //保存为Excel文件
15. FileOutputStream out = **null**;
17. **try** {
18. out = **new** FileOutputStream("c:\\text.xls");
19. wb.write(out);
20. } **catch** (IOException e) {
21. System.out.println(e.toString());
22. } **finally** {
23. **try** {
24. out.close();
25. } **catch** (IOException e) {
26. System.out.println(e.toString());
27. }
28. }

2、生成Workbook OOXML形式(.xlsx)

**Java代码  收藏代码**

1. //生成Workbook
2. XSSFWorkbook wb = **new** XSSFWorkbook();
4. //......

3、打开Workbook

**Java代码  收藏代码**

1. //方法一：使用WorkbookFactory
2. FileInputStream in = **null**;
3. Workbook wb = **null**;
5. **try** {
6. in = **new** FileInputStream(TEST\_WORKBOOK\_NAME);
7. wb = WorkbookFactory.create(in);
8. } **catch** (IOException e) {
9. System.out.println(e.toString());
10. } **catch** (InvalidFormatException e) {
11. System.out.println(e.toString());
12. } **finally** {
13. **try** {
14. in.close();
15. } **catch** (IOException e) {
16. System.out.println(e.toString());
17. }
18. }
20. System.out.println("====================Workbook====================");
21. System.out.println("Number of Sheets:" + wb.getNumberOfSheets());
22. System.out.println("Sheet3's name:" + wb.getSheetName(3));
23. System.out.println();
25. //方法二：使用POIFSFileSystem
26. **try** {
27. in = **new** FileInputStream(TEST\_WORKBOOK\_NAME);
28. POIFSFileSystem fs = **new** POIFSFileSystem(in);
29. wb = **new** HSSFWorkbook(fs);
30. } **catch** (IOException e) {
31. System.out.println(e.toString());
32. } **finally** {
33. **try** {
34. in.close();
35. } **catch** (IOException e) {
36. System.out.println(e.toString());
37. }
38. }
40. System.out.println("====================Workbook====================");
41. System.out.println("Number of Sheets:" + wb.getNumberOfSheets());
42. System.out.println("Sheet3's name:" + wb.getSheetName(3));
43. System.out.println();

4、打开加密的Workbook(读加密)

**Java代码  收藏代码**

1. FileInputStream input = **new** FileInputStream(TEST\_WORKBOOK\_NAME\_ENCRYPTED);
2. BufferedInputStream binput = **new** BufferedInputStream(input);
3. POIFSFileSystem poifs = **new** POIFSFileSystem(binput);
5. Biff8EncryptionKey.setCurrentUserPassword(TEST\_WORKBOOK\_PASSWORD);
7. HSSFWorkbook wb = **new** HSSFWorkbook(poifs);
9. System.out.println("====================EncryptedWorkbook====================");
10. System.out.println("Number of Sheets:" + wb.getNumberOfSheets());
11. System.out.println("Sheet0's name:" + wb.getSheetName(0));
12. System.out.println();

5、追加Sheet

**Java代码  收藏代码**

1. Sheet sheet = wb.createSheet("append sheet");

6、复制Sheet

**Java代码  收藏代码**

1. wb.cloneSheet(1);

7、修改Sheet名称

**Java代码  收藏代码**

1. wb.setSheetName(i, "SheetName new");

8、删除Sheet

**Java代码  收藏代码**

1. wb.removeSheetAt(1);

9、设置下部Sheet名的Tab的第一个可见Tab

**Java代码  收藏代码**

1. //设置下部Sheet名的Tab的第一个可见Tab（以左的Sheet看不见）
2. wb.setFirstVisibleTab(2);

10、调整Sheet顺序

**Java代码  收藏代码**

1. wb.setSheetOrder("SheetName3", 1);
2. wb.setSheetOrder(wb.getSheetName(4), 0);

11、设置当前Sheet   
t.setActiveSheet();

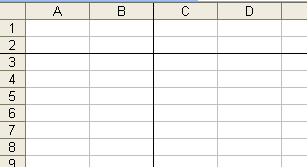
**Java代码  收藏代码**

1. //设置当前Sheet
2. wb.setActiveSheet(wb.getNumberOfSheets() - 1);
3. //（Excel的当前Sheet被设置，需要结合setSelected使用，不然下部Sheet名的Tab还是默认为第一个）
4. //（需要选择多个Sheet的话，每个Sheet调用setSelected(true)即可）
5. wb.getSheetAt(wb.getNumberOfSheets() - 1).setSelected(**true**);

12、固定窗口

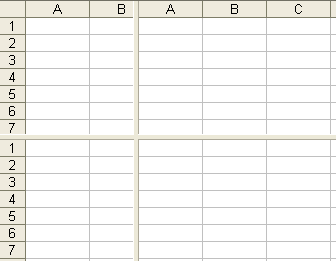
**Java代码  收藏代码**

1. wb.getSheet("SheetName4").createFreezePane(2, 2);

   
  
13、分割窗口

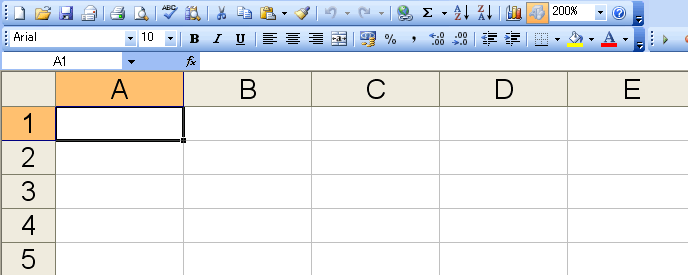
**Java代码  收藏代码**

1. wb.getSheet("SheetName5").createSplitPane(2000, 2000, 0, 0, HSSFSheet.PANE\_LOWER\_LEFT);

   
  
14、Sheet缩放

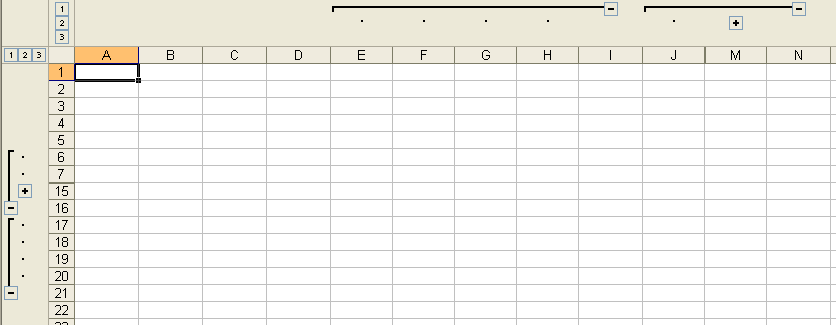
**Java代码  收藏代码**

1. //setZoom(int numerator, int denominator)
2. //"numerator"÷"denominator"  例如： 3÷1＝3 那就是设置为300%
4. //扩大(200%)
5. wb.getSheet("sheetname1").setZoom(2, 1);
6. //缩小(50%)
7. wb.getSheet("sheetname2").setZoom(1, 2);

   
  
15、行列分组

**Java代码  收藏代码**

1. wb.getSheet("sheetname3").groupColumn(4, 7);
2. wb.getSheet("sheetname3").groupColumn(9, 12);
3. wb.getSheet("sheetname3").groupColumn(10, 11);
5. wb.getSheet("sheetname3").groupRow(5, 14);
6. wb.getSheet("sheetname3").groupRow(7, 13);
7. wb.getSheet("sheetname3").groupRow(16, 19);

  
  
16、关闭分组

**Java代码  收藏代码**

1. wb.getSheet("sheetname3").setColumnGroupCollapsed(10, **true**);
2. wb.getSheet("sheetname3").setRowGroupCollapsed(7, **true**);

17、插入行

**Java代码  收藏代码**

1. Row row1 = wb.getSheet("sheetname4").createRow(1);
2. Cell cell1\_1 = row1.createCell(1);
3. cell1\_1.setCellValue(123);
5. Row row4 = wb.getSheet("sheetname4").createRow(4);
6. Cell cell4\_3 = row4.createCell(3);
7. cell4\_3.setCellValue("中国");

18、删除行

**Java代码  收藏代码**

1. Row row = wb.getSheet("sheetname4").getRow(1);
2. wb.getSheet("sheetname4").removeRow(row);

19、移动行

**Java代码  收藏代码**

1. //\*\*\*\*\*\*移动行只移动内容，不牵扯行的删除和插入
3. //移动行（把第1行和第2行移到第5行之后）
4. wb.getSheet("sheetname5").shiftRows(0, 1, 5);
6. //移动行（把第3行和第4行往上移动1行）
7. wb.getSheet("sheetname5").shiftRows(2, 3, -1);

20、修改行高

**Java代码  收藏代码**

1. //设置默认行高
2. wb.getSheet("sheetname6").setDefaultRowHeight((**short**)100);
4. //设置行高
5. wb.getSheet("sheetname6").getRow(2).setHeight((**short**)(100 \* 20));

21、修改列宽

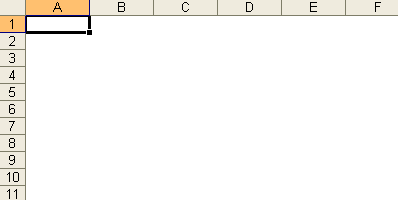
**Java代码  收藏代码**

1. //设置默认列宽
2. wb.getSheet("sheetname7").setDefaultColumnWidth(12);
4. //设置列宽
5. wb.getSheet("sheetname7").setColumnWidth(0, 5 \* 256);

22、不显示网格线

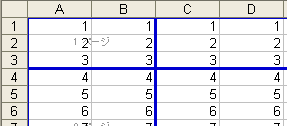
**Java代码  收藏代码**

1. //不显示网格线
2. wb.getSheet("sheetname8").setDisplayGridlines(**false**);

   
  
23、设置分页

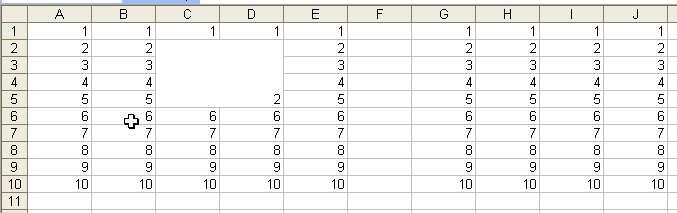
**Java代码  收藏代码**

1. //设置第一页：3行2列 （可以多次设置）
2. wb.getSheet("sheetname9").setRowBreak(2);
3. wb.getSheet("sheetname9").setColumnBreak(1);

   
  
24、添加，删除，合并单元格

**Java代码  收藏代码**

1. //追加行
2. **for** (**int** i = 0; i < 10; i++) {
3. Row row = wb.getSheet("sheetname10").createRow(i);
4. **for** (**int** j = 0; j < 10; j++) {
5. //添加单元格
6. Cell cell = row.createCell(j);
7. cell.setCellValue(i + 1);
8. }
10. //删除单元格
11. row.removeCell(row.getCell(5));
12. }
14. //合并单元格
15. //CellRangeAddress(int firstRow, int lastRow, int firstCol, int lastCol)
16. wb.getSheet("sheetname10").addMergedRegion(**new** CellRangeAddress(1, 4, 2, 3));

   
  
25、设置Header，Footer

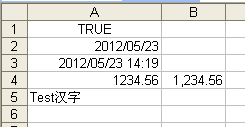
**Java代码  收藏代码**

1. //Header
2. Header header = wb.getSheet("sheetname11").getHeader();
3. header.setLeft(HSSFHeader.startUnderline() +
4. HSSFHeader.font("宋体", "Italic") +
5. "文字文字" +
6. HSSFHeader.endUnderline());
7. header.setCenter(HSSFHeader.fontSize((**short**)16) +
8. HSSFHeader.startDoubleUnderline() +
9. HSSFHeader.startBold() +
10. "汉字汉字" +
11. HSSFHeader.endBold() +
12. HSSFHeader.endDoubleUnderline());
13. header.setRight("打印时间:" + HSSFHeader.date() +  " " + HSSFHeader.time());
15. //Footer
16. Footer footer = wb.getSheet("sheetname11").getFooter();
17. footer.setLeft("Copyright @ rensanning");
18. footer.setCenter("Page:" + HSSFFooter.page() + " / " + HSSFFooter.numPages());
19. footer.setRight("File:" + HSSFFooter.file());

http://dl.iteye.com/upload/attachment/0068/5430/163956a5-f277-30d5-91ef-656c7375127d.gif   
http://dl.iteye.com/upload/attachment/0068/5428/23f093c1-f22c-3a3e-867a-11445622c816.gif   
  
26、设置单元格值

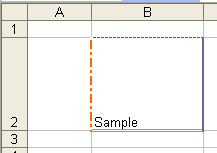
**Java代码  收藏代码**

1. //boolean
2. Cell cell00 = rows[0].createCell(0);
3. **boolean** val00 = **true**;
4. cell00.setCellValue(val00);
6. //Calendar 格式化
7. CellStyle styleCalendar = wb.createCellStyle();
8. DataFormat formatCalendar = wb.createDataFormat();
9. styleCalendar.setDataFormat(formatCalendar.getFormat("yyyy/mm/dd"));
10. Cell cell11 = rows[1].createCell(0);
11. Calendar val11 = Calendar.getInstance();
12. cell11.setCellStyle(styleCalendar);
13. cell11.setCellValue(val11);
15. //Date 格式化
16. CellStyle styleDate = wb.createCellStyle();
17. DataFormat formatDate = wb.createDataFormat();
18. styleDate.setDataFormat(formatDate.getFormat("yyyy/mm/dd hh:mm"));
19. Cell cell21 = rows[2].createCell(0);
20. Date val21 = **new** Date();
21. cell21.setCellStyle(styleDate);
22. cell21.setCellValue(val21);
24. //double
25. Cell cell30 = rows[3].createCell(0);
26. **double** val30 = 1234.56;
27. cell30.setCellValue(val30);
29. //double 格式化
30. CellStyle styleDouble = wb.createCellStyle();
31. DataFormat formatDouble = wb.createDataFormat();
32. styleDouble.setDataFormat(formatDouble.getFormat("#,##0.00"));
33. Cell cell31 = rows[3].createCell(1);
34. **double** val31 = 1234.56;
35. cell31.setCellStyle(styleDouble);
36. cell31.setCellValue(val31);
38. //String
39. Cell cell40 = rows[4].createCell(0);
40. HSSFRichTextString val40 = **new** HSSFRichTextString("Test汉字");
41. cell40.setCellValue(val40);

   
  
27、设置单元格边线

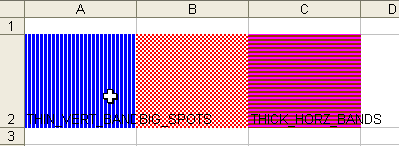
**Java代码  收藏代码**

1. wb.getSheet("sheetname2").setColumnWidth(1, 4096);
3. Row row1 = wb.getSheet("sheetname2").createRow(1);
4. row1.setHeightInPoints(70);
6. Cell cell1\_1 = row1.createCell(1);
7. cell1\_1.setCellValue("Sample");
9. CellStyle style = wb.createCellStyle();
11. style.setBorderTop(CellStyle.BORDER\_DASHED);
12. style.setBorderBottom(CellStyle.BORDER\_DOUBLE);
13. style.setBorderLeft(CellStyle.BORDER\_MEDIUM\_DASH\_DOT);
14. style.setBorderRight(CellStyle.BORDER\_MEDIUM);
16. style.setTopBorderColor(IndexedColors.MAROON.getIndex());
17. style.setBottomBorderColor(IndexedColors.SKY\_BLUE.getIndex());
18. style.setLeftBorderColor(IndexedColors.ORANGE.getIndex());
19. style.setRightBorderColor(IndexedColors.BLUE\_GREY.getIndex());
21. cell1\_1.setCellStyle(style);

   
  
28、设置单元格背景填充

**Java代码  收藏代码**

1. wb.getSheet("sheetname3").setColumnWidth(0, 4096);
2. wb.getSheet("sheetname3").setColumnWidth(1, 4096);
3. wb.getSheet("sheetname3").setColumnWidth(2, 4096);
5. Row row1 = wb.getSheet("sheetname3").createRow(1);
6. row1.setHeightInPoints(70);
8. Cell cell1\_0 = row1.createCell(0);
9. Cell cell1\_1 = row1.createCell(1);
10. Cell cell1\_2 = row1.createCell(2);
12. cell1\_0.setCellValue("THIN\_VERT\_BANDS");
13. cell1\_1.setCellValue("BIG\_SPOTS");
14. cell1\_2.setCellValue("THICK\_HORZ\_BANDS");
16. CellStyle style1 = wb.createCellStyle();
17. style1.setFillPattern(CellStyle.THIN\_VERT\_BANDS);
18. style1.setFillForegroundColor(IndexedColors.WHITE.getIndex());
19. style1.setFillBackgroundColor(IndexedColors.BLUE.getIndex());
21. CellStyle style2 = wb.createCellStyle();
22. style2.setFillPattern(CellStyle.BIG\_SPOTS);
23. style2.setFillForegroundColor(IndexedColors.RED.getIndex());
24. style2.setFillBackgroundColor(IndexedColors.WHITE.getIndex());
26. CellStyle style3 = wb.createCellStyle();
27. style3.setFillPattern(CellStyle.THICK\_HORZ\_BANDS);
28. style3.setFillForegroundColor(IndexedColors.PINK.getIndex());
29. style3.setFillBackgroundColor(IndexedColors.BROWN.getIndex());
31. cell1\_0.setCellStyle(style1);
32. cell1\_1.setCellStyle(style2);
33. cell1\_2.setCellStyle(style3);

   
  
29、设置单元格注释

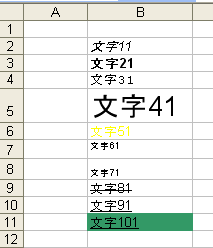
**Java代码  收藏代码**

1. HSSFCreationHelper createHelper =
2. (HSSFCreationHelper)wb.getCreationHelper();
3. Drawing patriarch = wb.getSheet("sheetname4").createDrawingPatriarch();
5. //注释
6. Row row = wb.getSheet("sheetname4").createRow(1);
7. Cell cell = row.createCell(1);
9. HSSFClientAnchor clientAnchor = **new** HSSFClientAnchor(0, 0, 0, 0,
10. (**short**) 4, 2, (**short**) 6, 5);
12. Comment comment = patriarch.createCellComment(clientAnchor);
13. comment.setString(createHelper.createRichTextString("注释注释111"));
14. comment.setAuthor("rensanning");
16. cell.setCellComment(comment);
18. //带字体的注释
19. Row row2 = wb.getSheet("sheetname4").createRow(2);
20. Cell cell2 = row2.createCell(1);
22. Font font = wb.createFont();
23. font.setFontName("宋体");
24. font.setFontHeightInPoints((**short**)10);
25. font.setBoldweight(HSSFFont.BOLDWEIGHT\_BOLD);
26. font.setColor(HSSFColor.RED.index);
28. Comment comment2 = patriarch.createCellComment(clientAnchor);
29. HSSFRichTextString text = **new** HSSFRichTextString("注释注释222");
30. text.applyFont(font);
31. comment2.setString(text);
32. comment2.setAuthor("rensanning");
34. cell2.setCellComment(comment2);

30、设置单元格字体（斜体，粗体，下线，取消线，字体，大小，背景色）

**Java代码  收藏代码**

1. Font font = **null**;
2. CellStyle style = **null**;
4. //斜体
5. font = wb.createFont();
6. font.setItalic(**true**);
7. style = wb.createCellStyle();
8. style.setFont(font);
10. wb.getSheet("sheetname5").getRow(1).getCell(1).setCellStyle(style);
12. //粗体
13. font = wb.createFont();
14. font.setBoldweight(HSSFFont.BOLDWEIGHT\_BOLD);
15. style = wb.createCellStyle();
16. style.setFont(font);
18. wb.getSheet("sheetname5").getRow(2).getCell(1).setCellStyle(style);
20. //字体名
21. font = wb.createFont();
22. font.setFontName("Courier New");
23. style = wb.createCellStyle();
24. style.setFont(font);
26. wb.getSheet("sheetname5").getRow(3).getCell(1).setCellStyle(style);
28. //字体大小
29. font = wb.createFont();
30. font.setFontHeightInPoints((**short**)20);
31. style = wb.createCellStyle();
32. style.setFont(font);
34. wb.getSheet("sheetname5").getRow(4).getCell(1).setCellStyle(style);
36. //文字颜色
37. font = wb.createFont();
38. font.setColor(HSSFColor.YELLOW.index);
39. style = wb.createCellStyle();
40. style.setFont(font);
42. wb.getSheet("sheetname5").getRow(5).getCell(1).setCellStyle(style);
44. //上标
45. font = wb.createFont();
46. font.setTypeOffset(HSSFFont.SS\_SUPER);
47. style = wb.createCellStyle();
48. style.setFont(font);
50. wb.getSheet("sheetname5").getRow(6).getCell(1).setCellStyle(style);
52. //下标
53. font = wb.createFont();
54. font.setTypeOffset(HSSFFont.SS\_SUB);
55. style = wb.createCellStyle();
56. style.setFont(font);
58. wb.getSheet("sheetname5").getRow(7).getCell(1).setCellStyle(style);
60. //删除线
61. font = wb.createFont();
62. font.setStrikeout(**true**);
63. style = wb.createCellStyle();
64. style.setFont(font);
66. wb.getSheet("sheetname5").getRow(8).getCell(1).setCellStyle(style);
68. //下划线
69. font = wb.createFont();
70. font.setUnderline(HSSFFont.U\_SINGLE);
71. style = wb.createCellStyle();
72. style.setFont(font);
74. wb.getSheet("sheetname5").getRow(9).getCell(1).setCellStyle(style);
76. //背景色
77. style = wb.createCellStyle();
78. style.setFillForegroundColor(HSSFColor.SEA\_GREEN.index);
79. style.setFillPattern(HSSFCellStyle.SOLID\_FOREGROUND);
81. style.setFont(font);
83. wb.getSheet("sheetname5").getRow(10).getCell(1).setCellStyle(style);

   
  
31、设置超链接

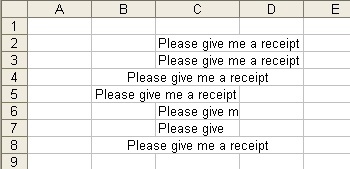
**Java代码  收藏代码**

1. HSSFCreationHelper createHelper =
2. (HSSFCreationHelper)wb.getCreationHelper();
4. CellStyle style = wb.createCellStyle();
5. Font font = wb.createFont();
6. font.setUnderline(HSSFFont.U\_SINGLE);
7. font.setColor(HSSFColor.BLUE.index);
8. style.setFont(font);
10. //追加行
11. Row[] rows = **new** Row[10];
12. **for** (**int** i = 0; i < 10; i++) {
13. rows[i] = wb.getSheet("sheetname6").createRow(i);
14. }
16. //URL
17. rows[0].createCell(0).setCellValue("URL Link");
19. HSSFHyperlink link1 = createHelper.createHyperlink(HSSFHyperlink.LINK\_URL);
20. link1.setAddress("http://poi.apache.org/");
21. rows[0].getCell(0).setHyperlink(link1);
22. rows[0].getCell(0).setCellStyle(style);
24. //Mail
25. rows[1].createCell(0).setCellValue("Email Link");
27. HSSFHyperlink link2 = createHelper.createHyperlink(HSSFHyperlink.LINK\_EMAIL);
28. link2.setAddress("mailto:poi@apache.org?subject=Hyperlinks");
29. rows[1].getCell(0).setHyperlink(link2);
30. rows[1].getCell(0).setCellStyle(style);
32. //File
33. rows[2].createCell(0).setCellValue("File Link");
35. HSSFHyperlink link3 = createHelper.createHyperlink(HSSFHyperlink.LINK\_FILE);
36. link3.setAddress("link.xls");
37. rows[2].getCell(0).setHyperlink(link3);
38. rows[2].getCell(0).setCellStyle(style);
40. //Workbook内
41. rows[3].createCell(0).setCellValue("Worksheet Link");
43. HSSFHyperlink link4 = createHelper.createHyperlink(HSSFHyperlink.LINK\_DOCUMENT);
44. link4.setAddress("sheetname1!A1");
45. rows[3].getCell(0).setHyperlink(link4);
46. rows[3].getCell(0).setCellStyle(style);

32、设置单元格横向对齐，纵向对齐

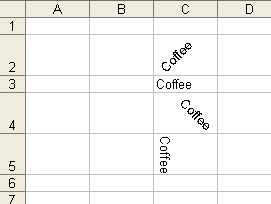
**Java代码  收藏代码**

1. //横向对齐
2. wb.getSheet("sheetname7").setColumnWidth(2, 3072);
4. Row[] row = **new** Row[7];
5. Cell[] cell = **new** Cell[7];
7. **for** (**int** i = 0 ; i < 7 ; i++){
8. row[i] = wb.getSheet("sheetname7").createRow(i + 1);
9. cell[i] = row[i].createCell(2);
10. cell[i].setCellValue("Please give me a receipt");
11. }
13. CellStyle style0 = wb.createCellStyle();
14. style0.setAlignment(CellStyle.ALIGN\_GENERAL);
15. cell[0].setCellStyle(style0);
17. CellStyle style1 = wb.createCellStyle();
18. style1.setAlignment(CellStyle.ALIGN\_LEFT);
19. cell[1].setCellStyle(style1);
21. CellStyle style2 = wb.createCellStyle();
22. style2.setAlignment(CellStyle.ALIGN\_CENTER);
23. cell[2].setCellStyle(style2);
25. CellStyle style3 = wb.createCellStyle();
26. style3.setAlignment(CellStyle.ALIGN\_RIGHT);
27. cell[3].setCellStyle(style3);
29. CellStyle style4 = wb.createCellStyle();
30. style4.setAlignment(CellStyle.ALIGN\_FILL);
31. cell[4].setCellStyle(style4);
33. CellStyle style5 = wb.createCellStyle();
34. style5.setAlignment(CellStyle.ALIGN\_JUSTIFY);
35. cell[5].setCellStyle(style5);
37. CellStyle style6 = wb.createCellStyle();
38. style6.setAlignment(CellStyle.ALIGN\_CENTER\_SELECTION);
39. cell[6].setCellStyle(style6);
41. //纵向对齐
42. Row row2 = wb.getSheet("sheetname8").createRow(1);
43. row2.setHeightInPoints(70);
44. Cell[] cell2 = **new** Cell[4];
46. **for** (**int** i = 0 ; i < 4 ; i++){
47. cell2[i] = row2.createCell(i + 1);
48. cell2[i].setCellValue("Please give me a receipt");
49. }
51. CellStyle style02 = wb.createCellStyle();
52. style02.setVerticalAlignment(CellStyle.VERTICAL\_TOP);
53. cell2[0].setCellStyle(style02);
55. CellStyle style12 = wb.createCellStyle();
56. style12.setVerticalAlignment(CellStyle.VERTICAL\_CENTER);
57. cell2[1].setCellStyle(style12);
59. CellStyle style22 = wb.createCellStyle();
60. style22.setVerticalAlignment(CellStyle.VERTICAL\_BOTTOM);
61. cell2[2].setCellStyle(style22);
63. CellStyle style32 = wb.createCellStyle();
64. style32.setVerticalAlignment(CellStyle.VERTICAL\_JUSTIFY);
65. cell2[3].setCellStyle(style32);

   
  
33、设置单元格旋转角度

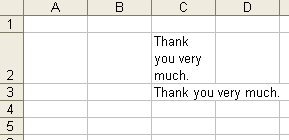
**Java代码  收藏代码**

1. Row[] row = **new** Row[4];
2. Cell[] cell = **new** Cell[4];
4. **for** (**int** i = 0 ; i < 4 ; i++){
5. row[i] = wb.getSheet("sheetname9").createRow(i + 1);
6. cell[i] = row[i].createCell(2);
7. cell[i].setCellValue("Coffee");
8. }
10. CellStyle style0 = wb.createCellStyle();
11. style0.setRotation((**short**)45);
12. cell[0].setCellStyle(style0);
14. CellStyle style1 = wb.createCellStyle();
15. style1.setRotation((**short**)0);
16. cell[1].setCellStyle(style1);
18. CellStyle style2 = wb.createCellStyle();
19. style2.setRotation((**short**)-45);
20. cell[2].setCellStyle(style2);
22. CellStyle style3 = wb.createCellStyle();
23. style3.setRotation((**short**)-90);
24. cell[3].setCellStyle(style3);

   
  
34、设置单元格自动折行

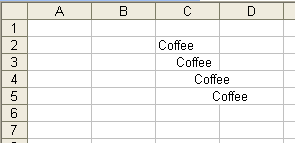
**Java代码  收藏代码**

1. Row[] row = **new** Row[2];
2. Cell[] cell = **new** Cell[2];
4. **for** (**int** i = 0 ; i < 2 ; i++){
5. row[i] = wb.getSheet("sheetname10").createRow(i + 1);
6. cell[i] = row[i].createCell(2);
7. cell[i].setCellValue("Thank you very much.");
8. }
10. CellStyle style0 = wb.createCellStyle();
11. style0.setWrapText(**true**);
12. cell[0].setCellStyle(style0);
14. CellStyle style1 = wb.createCellStyle();
15. style1.setWrapText(**false**);
16. cell[1].setCellStyle(style1);

   
  
35、设置单元格文字缩进

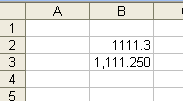
**Java代码  收藏代码**

1. Row[] row = **new** Row[4];
2. Cell[] cell = **new** Cell[4];
4. **for** (**int** i = 0 ; i < 4 ; i++){
5. row[i] = wb.getSheet("sheetname11").createRow(i + 1);
6. cell[i] = row[i].createCell(2);
7. cell[i].setCellValue("Coffee");
8. }
10. CellStyle style1 = wb.createCellStyle();
11. style1.setIndention((**short**)1);
12. style1.setAlignment(CellStyle.ALIGN\_LEFT);
13. cell[1].setCellStyle(style1);
15. CellStyle style2 = wb.createCellStyle();
16. style2.setIndention((**short**)2);
17. style2.setAlignment(CellStyle.ALIGN\_LEFT);
18. cell[2].setCellStyle(style2);
20. CellStyle style3 = wb.createCellStyle();
21. style3.setIndention((**short**)3);
22. style3.setAlignment(CellStyle.ALIGN\_LEFT);
23. cell[3].setCellStyle(style3);

   
  
36、自定义格式

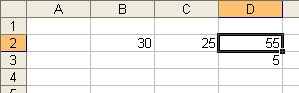
**Java代码  收藏代码**

1. Row[] rows = **new** Row[2];
2. **for** (**int** i = 0; i < rows.length; i++) {
3. rows[i] = wb.getSheet("sheetname12").createRow(i + 1);
4. }
5. DataFormat format = wb.createDataFormat();
7. CellStyle[] styles = **new** CellStyle[2];
8. **for** (**int** i = 0; i < styles.length; i++) {
9. styles[i] = wb.createCellStyle();
10. }
11. styles[0].setDataFormat(format.getFormat("0.0"));
12. styles[1].setDataFormat(format.getFormat("#,##0.000"));
14. Cell[] cells = **new** Cell[2];
15. **for** (**int** i = 0; i < cells.length; i++)  {
16. cells[i] = rows[i].createCell(1);
17. cells[i].setCellValue(1111.25);
19. cells[i].setCellStyle(styles[i]);
20. }

   
  
37、设置公式

**Java代码  收藏代码**

1. Row row1 = wb.getSheet("sheetname13").createRow(1);
2. Row row2 = wb.getSheet("sheetname13").createRow(2);
4. Cell cell1\_1 = row1.createCell(1);
5. Cell cell1\_2 = row1.createCell(2);
6. Cell cell1\_3 = row1.createCell(3);
7. Cell cell2\_3 = row2.createCell(3);
9. cell1\_1.setCellValue(30);
10. cell1\_2.setCellValue(25);
11. cell1\_3.setCellFormula("B2+C2");
12. cell2\_3.setCellFormula("MOD(B2,C2)");

   
  
38、画直线，圆圈（椭圆），正方形（长方形）,Textbox

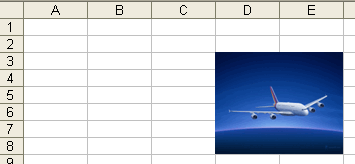
**Java代码  收藏代码**

1. HSSFPatriarch patriarch = ((HSSFSheet)wb.getSheet("sheetname14")).createDrawingPatriarch();
3. //直线
4. HSSFClientAnchor clientAnchor1 = **new** HSSFClientAnchor(0, 0, 0, 0,
5. (**short**) 4, 2, (**short**) 6, 5);
6. HSSFSimpleShape shape1 = patriarch.createSimpleShape(clientAnchor1);
7. shape1.setShapeType(HSSFSimpleShape.OBJECT\_TYPE\_LINE);
9. //圆圈（椭圆）
10. HSSFClientAnchor clientAnchor2 = **new** HSSFClientAnchor(0, 0, 0, 0,
11. (**short**) 8, 4, (**short**) 6, 5);
12. HSSFSimpleShape shape2 = patriarch.createSimpleShape(clientAnchor2);
13. shape2.setShapeType(HSSFSimpleShape.OBJECT\_TYPE\_OVAL);
15. //正方形（长方形）
16. HSSFClientAnchor clientAnchor3 = **new** HSSFClientAnchor(0, 0, 0, 0,
17. (**short**) 12, 6, (**short**) 6, 5);
18. HSSFSimpleShape shape3 = patriarch.createSimpleShape(clientAnchor3);
19. shape3.setShapeType(HSSFSimpleShape.OBJECT\_TYPE\_RECTANGLE);
21. //Textbox
22. HSSFClientAnchor clientAnchor4 = **new** HSSFClientAnchor(0, 0, 0, 0,
23. (**short**) 14, 8, (**short**) 6, 5);
24. HSSFTextbox textbox = patriarch.createTextbox(clientAnchor4);
25. textbox.setString(**new** HSSFRichTextString("This is a test"));

39、插入图片

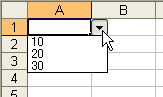
**Java代码  收藏代码**

1. //需要commons-codec-1.6.jar
2. FileInputStream jpeg = **new** FileInputStream("resource/test.jpg");
3. **byte**[] bytes = IOUtils.toByteArray(jpeg);
4. **int** pictureIndex = wb.addPicture(bytes, HSSFWorkbook.PICTURE\_TYPE\_JPEG);
5. jpeg.close();
7. HSSFCreationHelper helper = (HSSFCreationHelper) wb.getCreationHelper();
9. HSSFPatriarch patriarch = ((HSSFSheet)wb.getSheet("sheetname15")).createDrawingPatriarch();
11. HSSFClientAnchor clientAnchor = helper.createClientAnchor();
13. clientAnchor.setCol1(3);
14. clientAnchor.setRow1(2);
16. HSSFPicture picture = patriarch.createPicture(clientAnchor, pictureIndex);
17. picture.resize();

   
  
40、设置可输入List

**Java代码  收藏代码**

1. CellRangeAddressList addressList = **new** CellRangeAddressList(
2. 0,
3. 0,
4. 0,
5. 0);
7. **final** String[] DATA\_LIST = **new** String[] {
8. "10",
9. "20",
10. "30",
11. };
12. DVConstraint dvConstraint =
13. DVConstraint.createExplicitListConstraint(DATA\_LIST);
15. HSSFDataValidation dataValidation = **new** HSSFDataValidation(addressList, dvConstraint);
16. dataValidation.setSuppressDropDownArrow(**false**);
18. wb.getSheet("sheetname16").addValidationData(dataValidation);

   
  
41、设置输入提示信息

**Java代码  收藏代码**

1. CellRangeAddressList addressList = **new** CellRangeAddressList(
2. 0,
3. 0,
4. 0,
5. 0);
7. **final** String[] DATA\_LIST = **new** String[] {
8. "10",
9. "20",
10. "30",
11. };
12. DVConstraint dvConstraint = DVConstraint.createExplicitListConstraint(DATA\_LIST);
14. HSSFDataValidation dataValidation =
15. **new** HSSFDataValidation(addressList, dvConstraint);
16. dataValidation.setSuppressDropDownArrow(**false**);
17. dataValidation.createPromptBox("输入提示", "请从下拉列表中选择！");
18. dataValidation.setShowPromptBox(**true**);
20. wb.getSheet("sheetname17").addValidationData(dataValidation);

