

PdWriter Class

Writing Paragraphs

The PdWriter class (com.baseprogramming.pdwriter.PdWriter) is a class that demonstrates how to use the Apache project PDFBox. More so, it demonstrates how PDFBox can be extended to provide a more user-friendly interface to write content to PDF--without having to worry about breaking up a large chunk of text such that it fits in a page.

This class attempts to emulate a basic word processor approach, where text is written in paragraphs(PdParagraph class), and each paragraph has settings such as font and spacing.

The PdWriter class has two constructors

```
public PdWriter(PDDocument document, Margin margin)
```

And

```
public PdWriter(PageMetadata meta, PDDocument document)
```

The Margin class stores margin information (Top, Left, Bottom, and Right). The margins are stored as a PdUnit. The concept of a PdUnit (as with all other code in this project) is an experimental concept; its goal is to provide a client with a wide range of options for units of measures. Currently, the available units of measure are: PdInch, PdMillimeters, PdPica, PdPixels, and Points. All units of measure convert the given value to points--the standard unit of measure in graphic systems; the PdPoints class merely echos the value given.

The PageMetadata class has basic page information(Margin and PDRectangle), and has methods to compute page boundaries. The default PDRectangle is PDRectangle.LETTER

To get started with the PdWriter class, create an instance:

```
Margin margin= new Margin(0.75f, 0.2f, 0.5f, 0.25f);  
PdWriter writer= new PdWriter(pdDoc, margin);
```

Then create one (or more) PdParagraph objects:

```
PdParagraph heading=writer.createParagraph();  
heading.setFont(PDType1Font.TIMES_BOLD);  
heading.setFontSize(24);  
heading.setAboveSpacing(new PdInch(0.75f));  
heading.setBelowSpacing(new PdInch(0.75f));  
  
PdParagraph body = writer.createParagraph();  
body.setFirstLineIndent(new PdInch(0.3f));  
body.setBelowSpacing(new PdInch(0.17f));  
  
PdParagraph code=writer.createParagraph();  
code.setFont(PDType1Font.COURIER);  
code.setBeforeTextIndent(new PdInch(0.5f));
```

```
code.setAboveSpacing(new PdInch(0.3f));
code.setBelowSpacing(new PdInch(0.3f));
```

Write text (paragraphs, by calling the method `PdWriter.write(PdParagraph,String)`):

```
writer.write(body,"Write text (paragraphs, by calling the method
PdWriter.write(PdParagraph,String):");
```

Note that where is a `write(String)` method, that creates its own `PdParagraph` instance with the default values.

Numbered and Bullet Lists

It is also possible to write a numbered or bullet list--use the class `PdList`, which extends `PdParagraph`. Use the `createNumberedPdList` method of the `PdWriter` class to create list paragraph style:

```
PdList list= writer.createNumberedPdList();
```

Or

```
PdList list= writer.createBulletPdList();
```

to create a bullet list. The line of code:

```
writer.write(list, "Java", "C++", "Python");
```

generates the following numbered list:

1. Java
2. C++
3. Python

A List of String can also be passed

To create a bullet list, use the `createBulletPdList` method:

```
list=writer.createBulletPdList();
```

and call the `PdWriter.write(PdList,...)` method as with the numbered list:

```
*Java
*C++
*Python
```

Printing Tables

It is also possible to print a data table. This is a more complex process, and requires the uses of three additional classes:

1. `PdColumn`
2. `PdTableHeader`
3. `PdTable`

The PdColumn class holds basic column information: label, name (for later reference and matching to the row data). and width. The PdTableHeader class contains the column definitions (PdColumn list) as well as header font information. Finally, the PdTable, which extends PdParagraph, contains the table header as well as additional border information. This class is used by the PdWriter class to write the table--given the data set. At the moment, the data set must be passed as a list of maps, with the map representing the a row, with keys that correspond to the name in the PdColumn entries in the PdTableHeader. A PdTable instance can be created by calling one of the helper methods in the PdWriter class:

```
PdTable table = writer.createTable("First Name", "Middle Name", "Last  
Name", "D.O.B", "Memo");
```

Or

```
PdTable table = writer.createTable()
```

The columns must be created and added later when using this method.

The table exterior borders, cell padding, cell spacing, row border, column borders can be specified as follows:

```
Borders border= new Borders(3, 1, 2, 1);  
table.setCellPadding(new PdPoints(10));  
table.setRowBorder(1);  
table.setColumnBorder(1);  
table.setBorder(border);
```

Consider that there exists a method called getDataTable(int) that generates the list of maps needed to supply the table data:

```
List<Map<String, Object>> data=getDataTable(10);
```

The column widths can be calculated automatically by calling the PdTable method calculateColumnWidths:

```
table.calculateColumnWidths(data, 5);
```

This method uses row count specified (5 in this example) to calculate an adequate column width based on the column data size. After having calculated, individual column widths can be updated as follows:

```
PdTableHeader header=table.getHeader();  
header.setFont(PDType1Font.TIMES_BOLD);  
PdColumn memo=header.getColumn("Memo");  
memo.setWidth(new PdInch(2));
```

Finally, the table can be printed as follows:

```
writer.write(table, data);
```

First Name	Middle Name	Last Name	D.O.B	Memo
Adaline	Mira	Bussinger	1997-04-14	Tarski rabbinistical central chimpanzee remeasure

Shizuko	Murray	Matacale	1921-03-25	deferentially contour curtain Gaston Heiney
Gaylord	Lucie	Dimon	1995-09-09	Brentano Andrew
Amparo	Arlinda	Herold	1982-06-12	spinus tumefied
Jerry	Tawnya	Aris	1934-03-15	Ginni Dathe
Elena	Jenna	Litterer	1993-08-14	Cameron Vonschiltz family cephalotaxaceae resettling necrose
Cory	Esteban	Holl	1948-06-12	kelpwort Madera County bloodroot tu-whit tu-whoo Brahmans Gilbert
Dora	Venetta	Zehe	1998-12-11	liberate stackyard lady-killing Rocklake city Brown township
Exie	Nikole	Mayo	1971-10-07	pronunciation gossypium peruvianum undercitizen Poughkeepsie city foreknow Wonacott
Faith	Danica	Sandus	1997-08-16	ten-gallon hat lofty Bluff Springs township Portland city (pt.)