AePlus

Reporting System

(Miscellaneous Printings

Using PeopleCode)

SRI Technologies Pty Ltd
Web Site: www.sritech.biz

Email: sritech@sritech.biz

Table Of Contents

AePlus Reporting System

Miscellaneous Printing using PeopleCode

In this tutorial, we will demonstrate some miscellaneous printings in PDF document using PeopleCode.

Pre-requisites:

The developer should go thru

- my_first_pdf_report_using_peoplecode tutorial document.
- aep_printing_invoice_using_peoplecode tutorial document.

Requirement:

The Requirement is to produce a PDF document demonstrating:

1. Some text printing using variety of Fonts.
2. Print Bar Code using Custom Font.
3. Print JPG image.
4. Print PNG image.
5. Print BMP image.
6. Print some text in a grid with following alignments:
    a. Left
    b. Centre
    c. Right

Let's get on with the steps that would be required to develop this report using AePlus Reporting System.

Fields and Records Creation:

Report Name:

The very first thing is to give a name to this report that will be identified uniquely in the PeopleSoft database. Let us name it as AEP_MISC_PDF. The length of this name must not be more than 15 characters as there will be technical objects created with this name in the database. You may choose to follow any naming conventions. We have started this with AEP – meaning AePlus Report.

Fields Creation:

Assuming that following fields are already in the system (created while my_first_pdf_report_using_peoplecode tutorial), we will not be creating any additional fields.

Field Name	Long Name	Short Name	Field Type	Size
AEP_RPT_EVENTS	Aep Report Events	Rpt Events	Char	1

Records Creation:

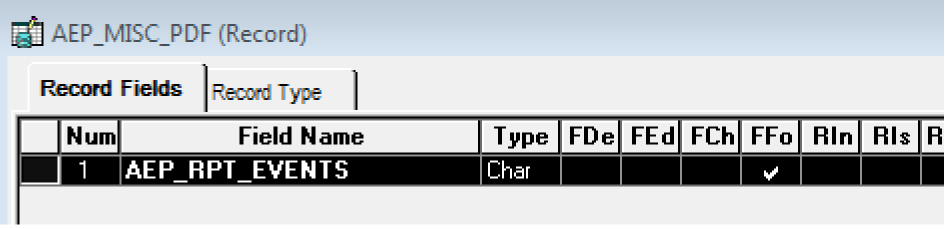
Create AEP_MISC_PDF(***rpt_aet_rec***) record with Record Type as Derived/Work. Please add following field to the record. We do not need any ***rpt_dat_rec*** record and will use ***rpt_aet_rec*** instead. i.e. In this case, ***rpt_aet_rec*** and ***rpt_dat_rec*** records will be the same.

Field Name
AEP_RPT_EVENTS

That's all about fields and records creations. Now is the time to get into PeopleCode stuff. We will generate PeopleCode Template using AePlus Workbench.

PeopleCode Template Creation:

AePlus workbench will create various *FieldFormula* PeopleCode functions in AEP_MISC_PDF.AEP_RPT_EVENTS. The record definition will look like:



Num	Field Name	Type	FDe	FEd	FCh	FFo	RIn	RIs	R
1	AEP_RPT_EVENTS	Char				✓			

The AePlus workbench creates following dummy functions for given ***rpt_aet_rec*** Record (i.e. for AEP_MISC_PDF record).

Report Events:	rpt_begin(…),rpt_end(…)
Page Events:	rpt_page_header(…),rpt_page_footer(…)
Data Events:	rpt_init_variables, rpt_skip, rpt_detail_section

In addition it also generates AEP_MISC_PDF.AEP_MISC_PDF Application Class that drives AePlus Reporting System. Both Application Class and FuncLib PeopleCode are tightly coupled and must migrate in tandem from one environment to other.

These Event Functions are triggered by AePlus Reporting System and developer should not remove or call them directly.

PeopleCode Development/Maintenance (FuncLib):

The next step will be fill-in these template functions. Let's do it one by one.

Function rpt_begin (…):

This function is called once by the AePlus Reporting System before we start printing on the report. We will leave thisas blank as we do not intend to do anything here for this report.

Function rpt_page_header(…):

We are generating a single page with some text and images. We will not have anything in the header to print. We will leave this as blank.

Function rpt_page_footer(…):

We are generating a single page and will have nothing in footer to print. We will leave this as blank.

Function rpt_skip(…):

We are not doing any consolidation/aggregation, so this function should always return false value. We will leave this function as default.

Function rpt_detail_section(…):

This function is called by the AePlus Reporting System to print each data row received from input data stream. Here we will put the PeopleCode to print some sample text and images. We will also put some PeopleCode to print text with some alignment.

```
Function rpt_detail_section(&AEP_RPT As AEP_MISC_PDF:AEP_MISC_PDF, &Rec As Record);

   Local number &i, &lines, &CurrLineNum;
   Local string &PageAtr;
   Local string &PrintAtr, &Msg, &Msg2;
   Local number &xPoint, &yPoint, &xPoint2, &yPoint2;

&PrintAtr = "FontName=Helvetica,FontSize=12";
&PageAtr = "PageSize=A4, PageOrientation=P";
&AEP_RPT.NewPage(&Rec, &PageAtr);

&AEP_RPT.NewLine(1, &Rec);
&AEP_RPT.PrintStr(1, "Misc Printing: ", "FontName=Helvetica,FontSize=12");

&AEP_RPT.NewLine(1, &Rec);
&AEP_RPT.PrintStr(1, "Page Attribute: " | &PageAtr, &PrintAtr);

&AEP_RPT.NewLine(1, &Rec);
&AEP_RPT.PrintStr(1, "Text Printed in Courier", "FontSize=12");
&AEP_RPT.NewLine(1, &Rec);

&AEP_RPT.NewLine(1, &Rec);
&AEP_RPT.PrintStr(1, "Text Printed in Helvetica", "FontName=Helvetica,FontSize=12");
&AEP_RPT.NewLine(1, &Rec);

&AEP_RPT.NewLine(1, &Rec);
&AEP_RPT.PrintStr(1, "Text Printed in Times-Roman", "FontName=Times-
Roman,FontSize=12");
&AEP_RPT.NewLine(1, &Rec);

&AEP_RPT.NewLine(1, &Rec);
&Msg = "ABC123";
&AEP_RPT.PrintStr(1, "Printing BarCode for: " | &Msg, "FontName=Times-
Roman,FontSize=12");
&AEP_RPT.PrintStr(30, "*" | &Msg | "*", "FontName=free3of9.ttf,FontSize=12");
&AEP_RPT.NewLine(1, &Rec);

&AEP_RPT.NewLine(1, &Rec);
&AEP_RPT.PrintStr(1, "Printing jpg image: ", "FontName=Times-Roman,FontSize=12");
&AEP_RPT.PrintImage(&AEP_RPT.ColPositionToPoint_X(35), (&AEP_RPT.get_CurrPointY() +
&AEP_RPT.get_CharHeight()), 0.2, "d:\sqrplus\docs\peoplesoft.jpg");
&AEP_RPT.NewLine(1, &Rec);

&AEP_RPT.NewLine(1, &Rec);
&AEP_RPT.PrintStr(1, "Printing png image: ", "FontName=Times-Roman,FontSize=12");
```

```
&AEP_RPT.PrintImage(&AEP_RPT.ColPositionToPoint_X(35), (&AEP_RPT.get_CurrPointY() +
&AEP_RPT.get_CharHeight()), 0.2, "d:\sqrplus\docs\peoplesoft.png");
&AEP_RPT.NewLine(1, &Rec);

&AEP_RPT.NewLine(1, &Rec);
&AEP_RPT.PrintStr(1, "Printing bmp image: ", "FontName=Times-Roman,FontSize=12");
&AEP_RPT.PrintImage(&AEP_RPT.ColPositionToPoint_X(35), (&AEP_RPT.get_CurrPointY() +
&AEP_RPT.get_CharHeight()), 0.2, "d:\sqrplus\docs\peoplesoft.bmp");
&AEP_RPT.NewLine(1, &Rec);

&AEP_RPT.NewLine(1, &Rec);
&AEP_RPT.PrintStr(1, "Printing Grid: ", "FontName=Times-Roman,FontSize=12");
&AEP_RPT.NewLine(1, &Rec);

&AEP_RPT.NewLine(1, &Rec);
&CurrLineNum = &AEP_RPT.get_CurrLineNumber();
&AEP_RPT.PrintStr(1, "Printing in Grid, Current Line: " | &CurrLineNum,
"FontName=Times-Roman,FontSize=12");

&AEP_RPT.SetCurrPointY_AtLine(&CurrLineNum);
&AEP_RPT.AdjustCurrPointY(8);
&Msg = "This is a text line that is ";
&Msg2 = "justified with wrap 1234 5678 aaaabbbb ccc, CurrLine: ";
&lines = &AEP_RPT.PrintStrWrap( False, &Rec, 1, 26, &Msg | "left" | &Msg2 |
&CurrLineNum | ", CurrY: " | &AEP_RPT.get_CurrPointY(),
"fontname=Helvetica,FontSize=12", "-");
&AEP_RPT.SetCurrPointY_AtLine(&CurrLineNum);
&lines = &AEP_RPT.PrintStrWrap( False, &Rec, 31, 60, &Msg | "centre" | &Msg2 |
&CurrLineNum | ", CurrY: " | &AEP_RPT.get_CurrPointY(),
"fontname=Helvetica,FontSize=12,centre=", "-");
&AEP_RPT.SetCurrPointY_AtLine(&CurrLineNum);
&lines = &AEP_RPT.PrintStrWrap( False, &Rec, 62, 90, &Msg | "right" | &Msg2 |
&CurrLineNum | ", CurrY: " | &AEP_RPT.get_CurrPointY(),
"fontname=Helvetica,FontSize=12,right", "-");

&AEP_RPT.SetCurrPointY_AtLine(&CurrLineNum);
&xPoint = &AEP_RPT.get_CurrLeftMargin();
&yPoint = &AEP_RPT.get_CurrPointY() + (&AEP_RPT.get_CharHeight() * 1.5);
&xPoint2 = &AEP_RPT.get_PageBottomRight_x() - (&AEP_RPT.get_CharWidth() * 1.0);
&yPoint2 = &yPoint - ((&lines + 3) * &AEP_RPT.get_CharHeight());
&AEP_RPT.DrawBox(&xPoint, &yPoint, &xPoint2, &yPoint2, "lineThickness=0");

&xPoint = &AEP_RPT.ColPositionToPoint_X(27);
&xPoint2 = &xPoint;
&AEP_RPT.DrawLine(&xPoint, &yPoint, &xPoint2, &yPoint2, "lineThickness=0");

&xPoint = &AEP_RPT.ColPositionToPoint_X(60);
&xPoint2 = &xPoint;
&AEP_RPT.DrawLine(&xPoint, &yPoint, &xPoint2, &yPoint2, "lineThickness=0");


End-Function;
```

Function rpt_init_variables(…):

Since we are not doing and consolidation, we can leave this function as blank.

Function rpt_end(…):

We are leaving this function as blank. But you may want to print some summary information here.

Miscellaneous code in FuncLib(…):

That’s the end of the report.

PeopleCode Development/Maintenance (AppEngine):

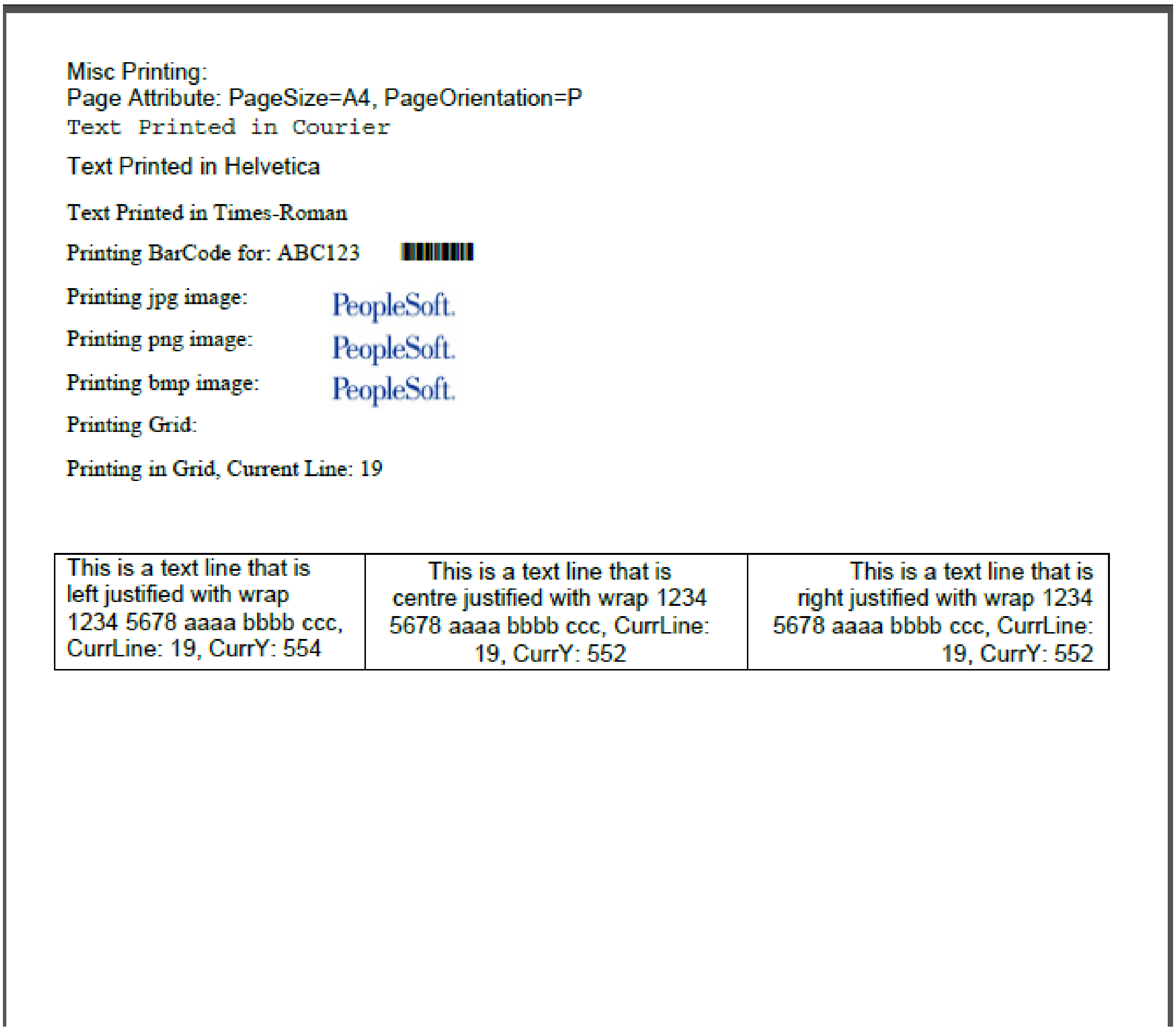
That’s all about putting code against various events and now is the time to write some code to initiate the report. We are using Application Engine PeopleCode for this. Let’s create a new Application Engine MISC_PDF. Create a PeopleCode step and add following code:

```
import AEP_MISC_PDF:*;

Local AEP MISC PDF:AEP_MISC_PDF &my_rpt;
Local Record &Rec;
Local string &RptAetRecName, &RptDatRecName, &ArgList;
Local boolean&Status;

&my_rpt = create AEP_MISC_PDF:AEP_MISC_PDF(“”);

rem build argument list;
&RptAetRecName = "AEP MISC PDF";
&RptDatRecName = &RptAetRecName;
&ArgList = "rpt aet rec=" | &RptAetRecName | ",";
&ArgList = &ArgList | "rpt_dat_rec=" | &RptDatRecName | ",";

If &my_rpt.InitReport(&ArgList) = True Then

&Status = &my_rpt.LoadCustFont("D:\sqrplus\docs\free3of9.ttf");

&Rec = CreateRecord(Record.AEP_MISC_PDF);

&my_rpt.SubmitData(&Rec, "");

&my_rpt.FinalizeReport("", &Rec);

End-If;

&my_rpt.ToLog("", "End");
```

Running the Report:

That’s all – now you can run the App Engine to generate your PDF report. We have not created Process Definition for this but you can run the Application Engine on the client using AePlus Workbench.

After running this app engine and all goes well, following output will be produced:



Misc Printing:
Page Attribute: PageSize=A4, PageOrientation=P
Text Printed in Courier
Text Printed in Helvetica
Text Printed in Times-Roman
Printing BarCode for: ABC123
Printing jpg image:
Printing png image:
Printing bmp image:
Printing Grid:
Printing in Grid, Current Line: 19

This is a text line that is left justified with wrap 1234 5678 aaaa bbbb ccc, CurrLine: 19, CurrY: 554	This is a text line that is centre justified with wrap 1234 5678 aaaa bbbb ccc, CurrLine: 19, CurrY: 552	This is a text line that is right justified with wrap 1234 5678 aaaa bbbb ccc, CurrLine: 19, CurrY: 552