

This is an example of PDFLib Extension! In the document you can see a header and a footer, formatted text, an array in a table, a Dat-file in a table and some other stuff ...

Here you can see an underlined string!

Below you can see instruments for top and bottom index!

$2^2=4$

$x_i=y$

Now we send the output of example.mod to a PDF file

```

USES ../sysmod/pdflib.mod

PROCEDURE Main
VARDEF pdf, pdf1 : Integer;
VARDEF sss : String;
VARDEF font : STRING;
VARDEF page_width, page_height, left_margin, right_margin,
top_margin, bottom_margin: INTEGER;
VARDEF i : INTEGER;
VARDEF arrCW : REAL[];
VARDEF arrSTR : STRING[.];
VARDEF arrAli : STRING[.];
VARDEF arrHead : STRING[];
VARDEF img_height, img_width : INTEGER;
VARDEF TableWidth : INTEGER;

cgiclosebuffer;
CGIWriteLn('content-type: text/html');
CGIWriteLn("");
CGIWriteLn('<pre>');

page_width := 594;
page_height := 841;
left_margin := right_margin := top_margin := bottom_margin := 10;

CGIWriteLn(TimeStr(Now, 0));

sss := 'This is an example of PDFLib Extension! In the document
you can see a header and a footer, formatted text, an array in a
table, a Dat-file in a table and some other stuff ...';
IF pdf := pdf_new('../export/example.pdf', 'w+') = 0 THEN
cgiwriteln(pdf_error_description);
END

IF pdf_header_init(pdf, 'Helvetica', 10) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_footer_init(pdf, 'Helvetica', 8) <> 0 THEN
cgiwriteln(pdf_error_description);
END
/*
IF pdf_header_add(pdf, "C", "../images/logo_bsd.jpg", 0.1) <> 0
THEN
cgiwriteln(pdf_error_description);
END
*/
IF pdf_header_add(pdf, "R", "&[author]", 0) <> 0 THEN
cgiwriteln(pdf_error_description);
END

```

```

IF pdf_footer_add(pdf, "L", "&[date]", 0) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_footer_add(pdf, "R", "&[page]", 0) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_set_author(pdf, "tdb software service GmbH") <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_page_border(pdf, 1, 0, 0, 1, 1) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_begin_page(pdf, page_width, page_height, left_margin,
right_margin, top_margin, bottom_margin) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF font := pdf_find_font('Courier-Bold') = "" THEN
cgiwriteln(pdf_error_description);
END
IF pdf_set_font(pdf, font, 15) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_set_text_rgbcolor(pdf, 0,0,1) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_str_output(pdf, sss, "C", (page_width - left_margin -
right_margin) / 2 - 150, (page_width - left_margin - right_margin) / 2
+ 150) <> 0 THEN
cgiwriteln(pdf_error_description);
END
/*
img_height := pdf_get_img_height("../images/logo_1.jpg");
img_width := pdf_get_img_width("../images/logo_1.jpg");
IF pdf_image_xy(pdf, '../images/logo_1.jpg', (page_width -
left_margin - right_margin - img_width) / 2, pdf_get_y(pdf),
(page_width - left_margin - right_margin + img_width) / 2,
pdf_get_y(pdf) + img_height, 0) <> 0 THEN
cgiwriteln(pdf_error_description);
END
*/
pdf_goto_xy(pdf, 0, 200);
pdf_writeln(pdf, "")

IF font := pdf_find_font('Courier-BoldOblique') = "" THEN
cgiwriteln(pdf_error_description);
END
IF pdf_set_font(pdf, font, 15) <> 0 THEN
cgiwriteln(pdf_error_description);
END

IF pdf_write_underlined(pdf, "Here you can see an underlined
string!") <> 0 THEN
cgiwriteln(pdf_error_description);
END
pdf_writeln(pdf, "");
pdf_writeln(pdf, "Below you can see instruments for top and bottom
index!");
IF pdf_write(pdf, "2") <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_write_superscript(pdf, "2") <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_writeln(pdf, "=4") <> 0 THEN
cgiwriteln(pdf_error_description);
END

IF pdf_write(pdf, "x") <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_write_subscript(pdf, "i") <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_writeln(pdf, "=y") <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF font := pdf_find_font('Helvetica-Oblique') = "" THEN
cgiwriteln(pdf_error_description);

```

```

END
IF pdf_set_font(pdf, font, 15) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_str_output(pdf, "Now we send the output of example.mod to
a PDF file", "C", 150, 450) <> 0 THEN
CGIWriteLn(pdf_error_description);
END
IF pdf_set_font(pdf, font, 10) <> 0 THEN
cgiwriteln(pdf_error_description);
END

IF pdf_file_output(pdf, 'example.mod', "L", 100, 400, 1) <> 0 THEN
cgiwriteln(pdf_error_description);
END

sss := "This is example of PDFLib Extension! It is put out in
columns of a table ...";

INITARRAY(arrCW[4]);
arrCW[1] := 25;
arrCW[2] := 50;
arrCW[3] := 100;
arrCW[4] := 50;

INITARRAY(arrSTR[4,1]);
arrSTR[1,0] := "Col #1";
arrSTR[2,0] := "Col #2";
arrSTR[3,0] := "Col #3";
arrSTR[4,0] := "Col #4";

arrSTR[1,1] := sss;
arrSTR[2,1] := sss;
arrSTR[3,1] := sss;
arrSTR[4,1] := sss;

INITARRAY(arrAli[4,1]);
arrAli[1,0] := "c";
arrAli[2,0] := "c";
arrAli[3,0] := "c";
arrAli[4,0] := "c";

arrAli[1,1] := "l";
arrAli[2,1] := "l";
arrAli[3,1] := "l";
arrAli[4,1] := "l";

pdf_goto_xy(pdf, pdf_get_x(pdf), pdf_get_y(pdf) + 1 * 10);
pdf_writeln(pdf, "");
IF pdf_str_output(pdf, "Here we put out to PDF-document a simple
array!", "C", (page_width - left_margin - right_margin) / 2 - 150,
(page_width - left_margin - right_margin) / 2 + 150) <> 0 THEN
cgiwriteln(pdf_error_description);
END
pdf_writeln(pdf, "");
IF pdf_array2table(pdf, arrSTR, arrAli, arrCW, "C", 1, 1) <> 0 THEN
cgiwriteln(pdf_error_description);
END

INITARRAY(arrCW[1]);
TableWidth := 225;
arrCW[1] := TableWidth;
INITARRAY(arrSTR[1,1]);
arrSTR[1,1] := "This row of the table is to show the functions of the
PDF-module";
INITARRAY(arrAli[1,1]);
arrAli[1,1] := "c";

pdf_writeln(pdf, "");
IF pdf_str_output(pdf, "Here we write the table string by string!",
"C", (page_width - left_margin - right_margin) / 2 - 150, (page_width
- left_margin - right_margin) / 2 + 150) <> 0 THEN
cgiwriteln(pdf_error_description);
END
pdf_writeln(pdf, "");
IF pdf_goto_xy(pdf, (page_width - left_margin - right_margin -
TableWidth) / 2, pdf_get_y(pdf)) <> 0 THEN
cgiwriteln(pdf_error_description);

```

```

END
IF pdf_write_table_string(pdf, arrSTR, arrAli, arrCW, 2, 1) <> 0
THEN
cgiwriteln(pdf_error_description);
END

INITARRAY(arrCW[4]);
arrCW[1] := 75;
arrCW[2] := 50;
arrCW[3] := 50;
arrCW[4] := 50;
INITARRAY(arrSTR[4,1]);
arrSTR[1,1] := "Column #1. Adjusted to left.";
arrSTR[2,1] := "Column #2. Adjusted to center.";
arrSTR[3,1] := "Column #3. Adjusted to right.";
arrSTR[4,1] := "Column #4. Adjusted by width.";
INITARRAY(arrAli[4,1]);
arrAli[1,1] := "l";
arrAli[2,1] := "c";
arrAli[3,1] := "r";
arrAli[4,1] := "w";
IF pdf_goto_xy(pdf, (page_width - left_margin - right_margin -
TableWidth) / 2, pdf_get_y(pdf)) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_write_table_string(pdf, arrSTR, arrAli, arrCW, 2, 0) <> 0
THEN
cgiwriteln(pdf_error_description);
END

INITARRAY(arrCW[10]);
arrCW[1] := 50;
arrCW[2] := 50;
arrCW[3] := 50;
arrCW[4] := 50;
arrCW[5] := 50;
arrCW[6] := 50;
arrCW[7] := 50;
arrCW[8] := 50;
arrCW[9] := 50;
arrCW[10] := 50;

INITARRAY(arrHead[4]);
arrHead[1] := "Col #1";
arrHead[2] := "Col #2";
arrHead[3] := "Col #3";
arrHead[4] := "Col #4";

pdf_writeln(pdf, "");
IF pdf_str_output(pdf, "Here we put out the './database/adr.dat' to
the PDF-file!", "C", (page_width - left_margin - right_margin) / 2 -
150, (page_width - left_margin - right_margin) / 2 + 150) <> 0
THEN
cgiwriteln(pdf_error_description);
END
pdf_writeln(pdf, "")
IF pdf_dat2table(pdf, 0, arrCW, arrHead, "./database/adr.dat", "", 0)
<> 0 THEN
cgiwriteln(pdf_error_description);
END

IF pdf1 := pdf_new('./export/double.pdf', 'w+') = 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_begin_page(pdf1, page_width, page_height, left_margin,
right_margin, top_margin, bottom_margin) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF font := pdf_find_font('Courier-Bold') = "" THEN
cgiwriteln(pdf_error_description);
END
IF pdf_set_font(pdf1, font, 15) <> 0 THEN
cgiwriteln(pdf_error_description);
END
sss := "This PDF-document is generated while the first
PDF-document is not finished yet!";
IF pdf_str_output(pdf1, sss, "C", (page_width - left_margin -
right_margin) / 2 - 150, (page_width - left_margin - right_margin) / 2

```

```

+ 150) <> 0 THEN
cgiwriteln(pdf_error_description);
END
IF pdf_close(pdf1) <> 0 THEN
cgiwriteln(pdf_error_description);
END

IF pdf_close(pdf) <> 0 THEN
cgiwriteln(pdf_error_description);
END

CGIWriteLn(TimeStr(Now, 0));

CGIWriteLn('</pre>');
ENDPROC

```

Here we put out to PDF-document a simple array!

Col #1	Col #2	Col #3	Col #4
This is example of PDFLib Extension! It is put out in columns of a table ...	This is example of PDFLib Extension! It is put out in columns of a table ...	This is example of PDFLib Extension! It is put out in columns of a table ...	This is example of PDFLib Extension! It is put out in columns of a table ...

Here we write the table string by string!

This row of the table is to show the functions of the PDF-module			
Column #1.	Column #2.	Column #3.	Column # 4 .
Adjusted to left.	Adjusted to center.	Adjusted to right.	Adjusted by width.

Here we put out the './database/adr.dat' to the PDF-file!

Col #1	Col #2	Col #3	Col #4
Record 1	1	12.12.1900	This is record number 1. Here is output memo-field of database ADR.DAT. It should be output into column of table of PDF-document. This PDF-document generated by PDFLib.mod which based on using memory's arrays. The length of this string is more than 256 characters. So that is end of string!
Record 2	2	11.11.2003	This is record number 2.
Record 3	3	11.11.2003	This is record number 3.