

**HMG HPDF**

# HMG\_HPDP Introduction

HMG\_HPDP is a small library to create PDF documents in a simple way using xBase syntax. Output from HMG programs can be exported to PDF with HMG PRINT like commands. Apart from HMG PRINT features this library has some additional features also.

It is based on HPDP in Harbour. HPDP is originally ported from LibHaru to Harbour.

## USAGE:

Just by including the following line into the initial lines of the HMG code, the commands discussed in this document can be used.

```
#include "hmg_hpdp.ch"
```

# General Points

1. The unit of measurement in HMG\_HPDPF is millimeters.
2. The maximum page length/width is 5080 mm.

# SELECT HPDFDOC

This is the command to initialize the HPDFDOC. This is the first command to be used to invoke HMG\_HPDPF. If you call any other command before calling this command, you will get an error that PDF object is not found. All the available papersizes are listed in hmg\_hpdpf.ch. User can specify his own paper length and width also. The maximum paper length/width is 5080 mm.

## Syntax:

```
SELECT HPDFDOC cPDFFile ;  
    [ ORIENTATION nOrientation ] ;  
    [ PAPERSIZE nPaperSize ] ;  
    [ PAPERLENGTH nPaperLength ] ;  
    [ PAPERWIDTH nPaperWidth ]
```

## Examples:

```
SELECT HPDFDOC "sample.pdf"  
SELECT HPDFDOC "sample.pdf" ORIENTATION HPDF_ORIENT_LANDSCAPE  
SELECT HPDFDOC "sample.pdf" PAPERSIZE HPDF_PAPER_LEGAL  
SELECT HPDFDOC "sample.pdf" PAPERLENGTH 300 PAPERWIDTH 300
```

# START HPDFDOC

This command is used to start the HPDFDOC.

## **Syntax:**

```
START HPDFDOC
```

## **Examples:**

```
START HPDFDOC
```

# END HPDFDOC

This command is used to close the HPDFDOC and create the PDF file as mentioned in the SELECT HPDFDOC command. Until this command is executed the PDF file will not be created.

## **Syntax:**

```
END HPDFDOC
```

## **Examples:**

```
END HPDFDOC
```

# ABORT HPDFDOC

This command is used to close the HPDFDOC and abort the PDF file creation process. If this command is executed the PDF file will not be created.

## **Syntax:**

```
ABORT HPDFDOC
```

## **Examples:**

```
ABORT HPDFDOC
```

# SET HPDFINFO

This command is used to insert some document attributes/properties about the author, creation date etc., These informations are meta-data and are not shown inside the PDF document.

## Syntax:

```
SET HPDFINFO AUTHOR TO cValue
SET HPDFINFO CREATOR TO cValue
SET HPDFINFO TITLE TO cValue
SET HPDFINFO SUBJECT TO cValue
SET HPDFINFO KEYWORDS TO cValue
SET HPDFINFO DATECREATED TO dValue [ TIME cTimeString ]
SET HPDFINFO DATEMODIFIED TO dValue [ TIME cTimeString ]
```

## Examples:

```
SET HPDFINFO AUTHOR TO "My Author"
SET HPDFINFO CREATOR TO "My Creator"
SET HPDFINFO TITLE TO "Doc Title"
SET HPDFINFO SUBJECT TO "Doc Subject"
SET HPDFINFO KEYWORDS TO "Keyword1, Keyword2"
SET HPDFINFO DATECREATED TO date()
SET HPDFINFO DATEMODIFIED TO date()
SET HPDFINFO DATECREATED TO date() TIME time()
SET HPDFINFO DATEMODIFIED TO date() TIME time()
```



# GET HPDFINFO

This command is used to retrieve the document attributes/properties about the author, creation date etc., already stored in the PDF document to the memory variable specified.

## Syntax:

```
GET HPDFINFO AUTHOR TO cValue
GET HPDFINFO CREATOR TO cValue
GET HPDFINFO TITLE TO cValue
GET HPDFINFO SUBJECT TO cValue
GET HPDFINFO KEYWORDS TO cValue
GET HPDFINFO DATECREATED TO cDateTime
GET HPDFINFO DATEMODIFIED TO cDateTime
```

## Examples:

```
GET HPDFINFO AUTHOR TO cAuthor
GET HPDFINFO CREATOR TO cCreator
GET HPDFINFO TITLE TO cTitle
GET HPDFINFO SUBJECT TO cSubject
GET HPDFINFO KEYWORDS TO cKeywords
GET HPDFINFO DATECREATED TO cDateTime
GET HPDFINFO DATEMODIFIED TO cDateTime
```

# SET HPDFDOC COMPRESS

This command is used to apply compression on the document contents. Level of compression can be specified using various parameters.

## Syntax:

```
SET HPDFDOC COMPRESS NONE | TEXT | IMAGE | METADATA | ALL
```

## Examples:

```
SET HPDFDOC COMPRESS NONE  
SET HPDFDOC COMPRESS TEXT  
SET HPDFDOC COMPRESS IMAGE  
SET HPDFDOC COMPRESS METADATA  
SET HPDFDOC COMPRESS ALL
```

# SET HPDFDOC PERMISSION

This command is used to allow/disallow users to do/deny some operations on the document. This command will be effective only if there is owner/user differentiation using SET PASSWORD command.

## Syntax:

```
SET HPDFDOC PERMISSION TO READ | PRINT | EDIT | COPY | EDIT_ALL
```

## Examples:

```
SET HPDFDOC PERMISSION TO READ  
SET HPDFDOC PERMISSION TO PRINT  
SET HPDFDOC PERMISSION TO EDIT  
SET HPDFDOC PERMISSION TO COPY  
SET HPDFDOC PERMISSION TO EDIT_ALL
```

# SET HPDFDOC PAGEMODE

This command is used to set the page mode to available page opening modes.

## Syntax:

```
SET HPDFDOC PAGEMODE TO NONE | OUTLINE | THUMBS | FULL_SCREEN
```

## Examples:

```
SET HPDFDOC PAGEMODE TO NONE  
SET HPDFDOC PAGEMODE TO OUTLINE  
SET HPDFDOC PAGEMODE TO THUMBS  
SET HPDFDOC PAGEMODE TO FULL_SCREEN
```

# SET HPDFDOC PASSWORD

This command is used to set password for the owner and/or user. The owner will then have more right over the user than specified in SET PERMISSION command. User password is optional in this command.

## Syntax:

```
SET HPDFDOC PASSWORD OWNER cOwnerPass [ USER cUserPass ]
```

## Examples:

```
SET HPDFDOC PASSWORD TO OWNER "Password_1"
```

```
SET HPDFDOC PASSWORD TO OWNER "Password_1" USER "Password_2"
```

# SET HPDFDOC PAGENUMBERING

This command is used to set the page numbering style of the PDF Document and the starting page number to be used for that purpose. Lower/Upper case can be used for Roman/Letters number styles. A prefix can be used to attach before the page numbers in thumbnails.

## Syntax:

```
SET HPDFDOC PAGENUMBERING [ FROM <nPage> ];  
    [ STYLE DECIMAL|ROMAN|LETTERS ];  
    [ UPPER|LOWER ];  
    [ PREFIX cPrefix ]
```

## Examples:

```
SET HPDFDOC PAGENUMBERING  
SET HPDFDOC PAGENUMBERING FROM PAGE 2  
SET HPDFDOC PAGENUMBERING FROM PAGE 2 STYLE ROMAN  
SET HPDFDOC PAGENUMBERING FROM PAGE 2 STYLE LETTERS LOWER  
SET HPDFDOC PAGENUMBERING FROM PAGE 2 STYLE ROMAN PREFIX "Page: "
```

# SET HPDFDOC ENCODING

This command is used to set the character encoding of the PDF document to correctly show the characters of various languages other than the default language (English). Available Character encoding options are StandardEncoding, MacRomanEncoding, WinAnsiEncoding, FontSpecific, ISO8859-2, ISO8859-3, ISO8859-4, ISO8859-5, ISO8859-6, ISO8859-7, ISO8859-8, ISO8859-9, ISO8859-10, ISO8859-11, ISO8859-13, ISO8859-14, ISO8859-15, ISO8859-16, CP1250, CP1251, CP1252, CP1253, CP1254, CP1255, CP1256, CP1257, CP1258, KOI8-R.

Specified character Encoding shall be applied only to the future text rendering. If you want to apply the character encoding to the whole document, call this command immediately after using START HPDFDOC command.

[Click here for more details about Character Encoding](#)

## **Syntax:**

**SET HPDFDOC ENCODING TO cEncoding**

## **Examples:**

```
SET HPDFDOC ENCODING TO "CP1251"  
SET HPDFDOC ENCODING TO "ISO8859-13"  
SET HPDFDOC ENCODING TO "CP1255"
```

# ROOTOUTLINE

Outlines are like Tree control of the PDF pages. It is easy for navigation and arrangement of the whole doc. This command is used to create a simple ROOT. It will not point/refer any page in the PDF doc. NAME clause should be used to refer the name of the root in the future. If you omit PARENT, the outline will be a ROOT node. Otherwise, it will be a sub-root of a parent root/node.

## Syntax:

```
SET HPDFDOC ROOTOUTLINE TITLE cTitle ;  
    NAME cName ;  
    [ PARENT cParent ]
```

## Examples:

```
SET HPDFDOC ROOTOUTLINE TITLE "Index" NAME "INDEX"  
SET HPDFDOC ROOTOUTLINE TITLE "Sub Index" NAME "SUBINDEX" PARENT "INDEX"
```



# PAGEOUTLINE

Outlines are like Tree control of the PDF pages. It is easy for navigation and arrangement of the whole doc. This command is used to create the outline reference of the current page in the doc. If the PARENT clause is omitted, the current page will be placed as a root. Use PARENT clause to make the current page reference to be inside a root. The root can be a ROOTOUTLINE or a PAGEOUTLINE referenced by its NAME. If you specify NAME clause, the page reference can be used as a ROOT and can be parent of other ROOTOUTLINE or PAGEOUTLINE. The current page can be referred in multiple outlines also. This command can be called only between START HPDFPAGE and END HPDFPAGE.

## Syntax:

```
SET HPDFDOC PAGEOUTLINE TITLE cTitle ;  
    [ NAME cName ];  
    [ PARENT cParent ]
```

## Examples:

```
SET HPDFDOC PAGEOUTLINE TITLE "Index Page" NAME "INDEX"  
SET HPDFDOC PAGEOUTLINE TITLE "Node Page" PARENT "INDEX"  
SET HPDFDOC PAGEOUTLINE TITLE "Intermediate Index Page" NAME "IIP" PARENT "INDEX"
```

# START HPDFPAGE

This command is used to add a new page to the HPDFDOC. The added page will be the current page until END HPDFPAGE is called. Please note that, PAGE once added can not be deleted or removed or moved afterwards. This command can be called only after START HPDFDOC

## Syntax:

```
START HPDFPAGE
```

## Examples:

```
START HPDFPAGE
```

# END HPDFPAGE

This command is used to close the current page of the HPDFDOC.

## **Syntax:**

**END HPDFPAGE**

## **Examples:**

END HPDFPAGE

# INSERT HPDFPAGE

This command is used to insert a new page before the page number specified. The inserted page will be selected as the current page after this command. All the pages will be shifted down. This command can not be called inside a current page selection. Please call END HPDFPAGE to close the current page before calling this command.

## Syntax:

```
INSERT HPDFPAGE BEFORE PAGE nPage
```

## Examples:

```
INSERT HPDFPAGE BEFORE PAGE 3
```

# SELECT HPDFPAGE

This command is used to select an existing page as the current page. This command can not be called inside a current page selection. Please call END HPDFPAGE to close the current page before calling this command.

## Syntax:

```
SELECT HPDFPAGE nPage
```

## Examples:

```
SELECT HPDFPAGE 3
```

# HMG\_HPDPF\_PAGENO()

This function is used to retrieve the current page number. This command shall be called in between the commands START HPDFPAGE and END HPDFPAGE.

## Syntax:

**HMG\_HPDPF\_PAGENO( ) -> nCurrentPage**

## Examples:

nCurrentPage := HMG\_HPDPF\_PageNo()

# HMG\_HPDCF\_PAGECOUNT()

This function is used to retrieve the total pages inside the HPDFDOC at the moment.

## Syntax:

```
HMG_HPDCF_PAGECOUNT( ) -> nTotPages
```

## Examples:

```
nTotalPages := HMG_HPDCF_PageCount()
```

# @..HPDFPRINT

This command is used to render a single line of text data in the specified font, size, color and alignment at the specified row and column. The text rendered is left aligned unless otherwise specified. Only following Base14 fonts are built-in. Courier, Courier-Bold, Courier-Oblique, Courier-BoldOblique, Helvetica, Helvetica-Bold, Helvetica-Oblique, Helvetica-BoldOblique, Times-Roman, Times-Bold, Times-Italic, Times-BoldItalic, Symbol, ZapfDingbats. The text will be rendered in the current character encoding.

Helvetica is the default font if no fontname is specified. Default fontsize is 12.

True Type Fonts can also be used if the absolute address of the True Type Font file is mentioned. Otherwise the TTF shall be required at the runtime in the same directory as of the executable file.

## Syntax:

```
@ Row , Col HPDFPRINT cText ;  
  [ FONT cFontName ] ;  
  [ SIZE nFontSize ] ;  
  [ COLOR aColor ] ;  
  [ CENTER|RIGHT ]
```

## Examples:

```
@ 205, 10 HPDFPRINT "This is a sample Text in default font."  
This is a sample Text in default font.
```

```
@ 220, 10 HPDFPRINT "This is a sample Text in Times-Roman font." FONT "Times-Roman" SIZE 14  
This is a sample Text in Times-Roman font.
```

```
@ 235, 200 HPDFPRINT "This is right aligned text" FONT "Courier-Bold" SIZE 14 RIGHT  
This is right aligned text
```

```
@ 250, 105 HPDFPRINT "This is center aligned text" COLOR { 255, 0, 0 } CENTER  
This is center aligned text
```

```
@ 270, 105 HPDFPRINT "This is text in bigger font size" SIZE 30 COLOR { 255, 0, 0 }
```

This is text in bigger font size

```
@ 290, 10 HPDFPRINT "The font is Tscparan.TTF" FONT "Tscparan.TTF" size 12
```

The font is Tscparan.TTF



# @..HPDFPRINT..TO

This command is just like the previous command @..HPDFPRINT. However, this command is used to print the text data in multiline. The text will be rendered inside a rectangular area specified in the command. The text can be aligned in JUSTIFY alignment also. All other options are just the same as the previous command.

## Syntax:

```
@ Row , Col HPDFPRINT cText ;  
  TO ToRow, ToCol ;  
  [ FONT cFontName ] ;  
  [ SIZE nFontSize ] ;  
  [ COLOR aColor ] ;  
  [ CENTER|RIGHT ]
```

## Examples:

```
@ 230, 10 HPDFPRINT "This is a small paragraph to be printed inside a rectangular area left aligned." to 250, 60  
@ 230, 80 HPDFPRINT "This is a small paragraph to be printed inside a rectangular area right aligned." to 250, 130 RIGHT  
@ 230, 150 HPDFPRINT "This is a small paragraph to be printed inside a rectangular area justified." to 250, 200 JUSTIFY  
@ 255, 10 HPDFPRINT "This is a small paragraph to be printed inside a rectangular area justified." to 275, 60 JUSTIFY
```

This is a small paragraph  
to be printed inside a  
rectangular area left  
aligned.

This is a small paragraph  
to be printed inside a  
rectangular area right  
aligned.

This is a small paragraph  
to be printed inside a  
rectangular area justified.

This is a small paragraph  
to be printed inside a  
rectangular area justified.

# SET HPDFPAGE LINE SPACING

This command is used to change the default line spacing in @..HPDFPRINT..TO command.

## Syntax:

**SET HPDFPAGE LINE SPACING TO nSpacing**

## Examples:

@ 230, 10 HPDFPRINT "This is a small paragraph to be printed inside a rectangular area left aligned." to 250, 60

SET HPDFPAGE LINE SPACING TO 6

@ 230, 70 HPDFPRINT "This is a small paragraph to be printed inside a rectangular area left aligned with a different linespacing." to 260, 120

SET HPDFPAGE LINE SPACING TO 0

@ 230, 130 HPDFPRINT "This is a small paragraph to be printed inside a rectangular area left aligned line spacing back to normal." to 230, 180

This is a small paragraph  
to be printed inside a  
rectangular area left  
aligned.

This is a small paragraph  
to be printed inside a  
rectangular area left  
aligned with a different  
linespacing.

This is a small paragraph  
to be printed inside a  
rectangular area left  
aligned line spacing back  
to normal.

# @..HPDFPRINT LINE TO

This command is used to draw a continuous line from the specified location to a specified another location. Color and width of the line can be specified optionally. The default color is black.

## Syntax:

```
@ Row , Col HPDFPRINT LINE ;  
  TO ToRow, ToCol ;  
  [ PENWIDTH nWidth ] ;  
  [ COLOR aColor ] ;
```

## Examples:

```
@ 230, 10 HPDFPRINT LINE TO 230, 50  
@ 235, 10 HPDFPRINT LINE TO 235, 50 PENWIDTH 1 COLOR { 255, 0, 0 }  
@ 230, 70 HPDFPRINT LINE TO 250, 70 PENWIDTH 1 COLOR { 255, 0, 0 }  
@ 230, 100 HPDFPRINT LINE TO 250, 150 PENWIDTH 2 COLOR { 255, 0, 255 }
```



# @..HPDFPRINT RECTANGLE TO

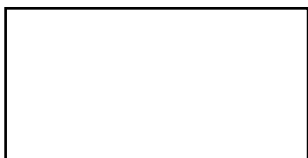
This command is used to draw a rectangle from the specified location to a specified another location. Color and width of the line can be specified optionally. The default color is black. The rectangle can be filled. Rounded clause will draw a rounded rectangle. Optionally Curve length can also be specified for Rounded rectangle.

## Syntax:

```
@ Row , Col HPDFPRINT RECTANGLE ;  
  TO ToRow, ToCol ;  
  [ PENWIDTH nWidth ] ;  
  [ COLOR aColor ] ;  
  [ FILLED ] ;  
  ROUNDED ;  
  [ CURVE nCurve ] ;
```

## Examples:

```
@ 230, 10 HPDFPRINT RECTANGLE TO 250, 50  
@ 230, 60 HPDFPRINT RECTANGLE TO 250, 100 PENWIDTH 1 COLOR { 255, 0, 0 }  
@ 230, 110 HPDFPRINT RECTANGLE TO 250, 150 PENWIDTH 1 COLOR { 255, 0, 0 } FILLED  
@ 260, 10 HPDFPRINT RECTANGLE TO 280, 50 PENWIDTH 2 COLOR { 255, 0, 255 } ROUNDED  
@ 260, 60 HPDFPRINT RECTANGLE TO 280, 150 PENWIDTH 2 COLOR { 255, 0, 255 } ROUNDED CURVE 10  
@ 260, 160 HPDFPRINT RECTANGLE TO 280, 200 PENWIDTH 2 COLOR { 255, 255, 0 } FILLED ROUNDED CURVE 10
```



# @..HPDFPRINT CIRCLE

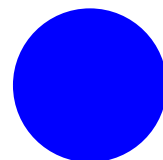
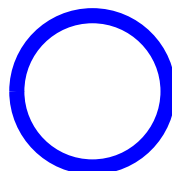
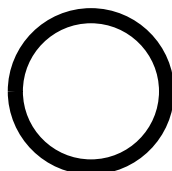
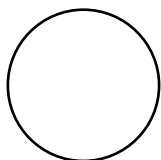
This command is used to draw a circle with the given radius having row and col as the center. Color and width of the line can be specified optionally. The default color is black. The circle can be filled.

## Syntax:

```
@ Row , Col HPDFPRINT CIRCLE RADIUS nRadius ;  
  [ PENWIDTH nWidth ] ;  
  [ COLOR aColor ] ;  
  [ FILLED ] ;
```

## Examples:

```
@ 230, 40 HPDFPRINT CIRCLE RADIUS 10  
@ 230, 80 HPDFPRINT CIRCLE RADIUS 10 PENWIDTH 2  
@ 230, 120 HPDFPRINT CIRCLE RADIUS 10 PENWIDTH 2 COLOR { 0, 0, 255 }  
@ 230, 160 HPDFPRINT CIRCLE RADIUS 10 COLOR { 0, 0, 255 } FILLED
```



# @..HPDFPRINT ELLIPSE

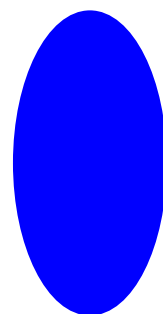
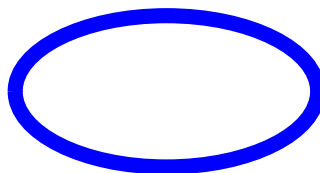
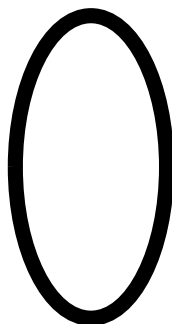
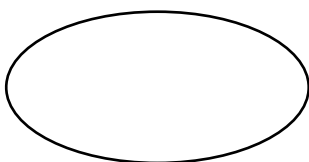
This command is used to draw an ellipse with the given horizontal radius and vertical radius having row and col as the center. Color and width of the line can be specified optionally. The default color is black. The ellipse can be filled.

## Syntax:

```
@ Row , Col HPDFPRINT ELLIPSE HORIZONTAL RADIUS nHRadius ;  
  VERTICAL RADIUS nVRadius ;  
  [ PENWIDTH nWidth ] ;  
  [ COLOR aColor ] ;  
  [ FILLED ];
```

## Examples:

```
@ 230, 40 HPDFPRINT ELLIPSE HORIZONTAL RADIUS 20 VERTICAL RADIUS 10  
@ 240, 80 HPDFPRINT ELLIPSE HORIZONTAL RADIUS 10 VERTICAL RADIUS 20 PENWIDTH 2  
@ 230, 120 HPDFPRINT ELLIPSE HORIZONTAL RADIUS 20 VERTICAL RADIUS 10 PENWIDTH 2 COLOR { 0, 0, 255 }  
@ 240, 160 HPDFPRINT ELLIPSE HORIZONTAL RADIUS 10 VERTICAL RADIUS 20 COLOR { 0, 0, 255 } FILLED
```



# @..HPDFPRINT ARC

This command is used to draw an arc with given radius from given angle to given another angle. Color and width of the line can be specified optionally. The default color is black.

## Syntax:

```
@ Row , Col HPDFPRINT ARC RADIUS nRadius ;  
  ANGLE FROM nFromAngle ;  
  TO nToAngle ;  
  [ PENWIDTH nWidth ] ;  
  [ COLOR aColor ] ;
```

## Examples:

```
@ 230, 40 HPDFPRINT ARC RADIUS 10 ANGLE FROM 0 TO 180
```

```
@ 230, 80 HPDFPRINT ARC RADIUS 10 ANGLE FROM 90 TO 270 PENWIDTH 2
```

```
@ 230, 120 HPDFPRINT ARC RADIUS 10 ANGLE FROM 30 TO 200 PENWIDTH 2 COLOR { 0, 100, 100 }
```



# @..HPDFPRINT CURVE

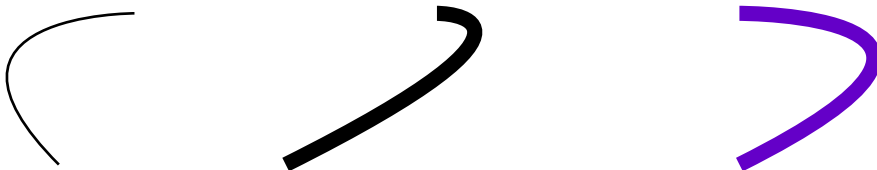
This command is used to draw a Bézier curve from a point taking second point as curvature to the third point. Color and width of the line can be specified optionally. The default color is black.

## Syntax:

```
@ Row , Col HPDFPRINT CURVE ;  
  FROM FromRow, FromCol ;  
  TO ToRow, ToCol ;  
  [ PENWIDTH nWidth ] ;  
  [ COLOR aColor ] ;
```

## Examples:

```
@ 230, 40 HPDFPRINT CURVE FROM 230, 10 TO 250, 30  
@ 230, 80 HPDFPRINT CURVE FROM 230, 100 TO 250, 60 PENWIDTH 2  
@ 230, 120 HPDFPRINT CURVE FROM 230, 160 TO 250, 120 PENWIDTH 2 COLOR { 100, 0, 200 }
```





# @..HPDFPRINT IMAGE

This command is used to insert a PNG/JPG format image file from the resource or the given path inside the document at the current location. The image will be stretched according to the given width and length. Specify image type 1 for PNG and 2 for JPG.

## Syntax:

```
@ Row , Col HPDFPRINT IMAGE cImage ;  
    WIDTH nWidth ;  
    LENGTH nLength  
    TYPE JPG|PNG ;
```

## Examples:

```
@ 230, 10 HPDFPRINT IMAGE "hmghpdf.png" TYPE PNG width 100 height 30
```

```
@ 230, 110 HPDFPRINT IMAGE "imgpng" TYPE PNG width 40 height 40
```

```
@ 230, 160 HPDFPRINT IMAGE "imgjpg" TYPE JPG width 40 height 40
```



# @..HPDFTOOLTIP

This command is used to insert a tooltip at the specified place with the specified icon. When the user hovers the mouse over this icon, the tooltip will be displayed.

## Syntax:

```
@ Row , Col HPDFPRINT TOOLTIP cToolTip ;  
    ICON COMMENT | KEY | NOTE | HELP |  
    NEW_PARAGRAPH | PARAGRAPH | INSERT
```

## Examples:

```
@ 240, 10 HPDFTOOLTIP "Comment Tooltip" ICON COMMENT  
@ 250, 10 HPDFTOOLTIP "Key Tooltip" ICON KEY  
@ 260, 10 HPDFTOOLTIP "Note Tooltip" ICON NOTE  
@ 240, 50 HPDFTOOLTIP "Help Tooltip" ICON HELP  
@ 250, 50 HPDFTOOLTIP "New_Paragraph Tooltip" ICON NEW_PARAGRAPH  
@ 260, 50 HPDFTOOLTIP "Paragraph Tooltip" ICON PARAGRAPH  
@ 240, 90 HPDFTOOLTIP "Insert Tooltip" ICON INSERT
```



# @..HPDFPAGELINK

This command is used to link a particular page to the specified text. Font, size, color, alignment options are the same as text data. A BORDER around the link can also be drawn with the optional width parameter.

## Syntax:

```
@ Row, Col HPDFPAGELINK cLink TO nPage ;  
  [ FONT cFontName ] ;  
  [ SIZE nFontSize ] ;  
  [ COLOR aColor ] ;  
  [ CENTER|RIGHT ] ;  
  [ BORDER ];  
  [ WIDTH nwidth ]
```

## Examples:

```
@ 230, 10 HPDFPAGELINK "Go to Page 5" TO 5  
@ 230, 70 HPDFPAGELINK "Go to Page 8" TO 8 FONT "Courier" size 18 color { 255, 0, 0 }  
@ 230, 130 HPDFPAGELINK "Go to Page 10" TO 10 color { 255, 0, 0 } BORDER  
@ 250, 10 HPDFPAGELINK "Go to Page 12" TO 12 color { 0, 0, 255 } BORDER WIDTH 1
```

Go to Page 5

Go to Page 8

Go to Page 10

Go to Page 12

# @..HPDFURLLINK

This command is used to link a particular Web URL/E-Mail ID to the specified text. Font, size, color, alignment options are the same as text data.

## Syntax:

```
@ Row, Col HPDFURLLINK cLink TO nPage ;  
  [ FONT cFontName ] ;  
  [ SIZE nFontSize ] ;  
  [ COLOR aColor ] ;  
  [ CENTER|RIGHT ] ;
```

## Examples:

```
@ 230, 10 HPDFURLLINK "HMGFORUM" TO "http://www.hmgforum.com" color { 0, 0, 255 }  
@ 235, 10 HPDFURLLINK "Mail me" TO "mailto:srgiri@dataone.in" color { 0, 0, 255 }
```

HMGFORUM  
Mail me

# Links

1. <http://www.hmgforum.com>
2. <http://harbour-project.org>
3. <http://libharu.org>