

**PGS.MD1**

Copyright © 1995 Soft-Logik Publishing Corporation

**COLLABORATORS**

	<i>TITLE :</i> PGS.MD1		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 23, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>PGS.MD1</b>	<b>1</b>
1.1	addarexxgadget . . . . .	1
1.2	addarexxlist . . . . .	3
1.3	addfontpath . . . . .	4
1.4	addguides . . . . .	4
1.5	addpoint . . . . .	4
1.6	addtodrawing . . . . .	5
1.7	align . . . . .	5
1.8	allocarexxlist . . . . .	6
1.9	allocarexxrequester . . . . .	7
1.10	appendfontprefs . . . . .	8
1.11	appendstyletags . . . . .	8
1.12	begincommandcapture . . . . .	9
1.13	breaktextrouting . . . . .	9
1.14	bringforward . . . . .	10
1.15	bringtofront . . . . .	10
1.16	clearstyletag . . . . .	11
1.17	closebusyrequester . . . . .	11
1.18	closedocument . . . . .	11
1.19	closepath . . . . .	12
1.20	closewindow . . . . .	12
1.21	colorpalette . . . . .	13
1.22	converttopath . . . . .	13
1.23	copyobject . . . . .	14
1.24	copytext . . . . .	14
1.25	cutobject . . . . .	15
1.26	cuttext . . . . .	15
1.27	createcolumns . . . . .	16
1.28	createcompound . . . . .	17
1.29	createdrawing . . . . .	18

---

---

1.30	createpath	18
1.31	currentchapter	19
1.32	currentchapterpath	20
1.33	currentdocument	20
1.34	currentmasterpage	20
1.35	currentmasterpagepath	21
1.36	currentpage	21
1.37	currentpagepath	22
1.38	currentwindow	22
1.39	currentwindowpath	23
1.40	deletechapter	23
1.41	deleteguides	23
1.42	deletemasterpage	24
1.43	deleteobject	24
1.44	deletepage	24
1.45	deletestyletag	25
1.46	deletetext	25
1.47	deletevariable	26
1.48	display	26
1.49	displaydrawings	27
1.50	displayfacingpages	27
1.51	displaygrid	27
1.52	displayguides	28
1.53	displayinvisibles	28
1.54	displaytextlinks	29
1.55	displayoutlines	29
1.56	displaypictures	29
1.57	displayruler	30
1.58	dissolvecompound	31
1.59	dissolvedrawing	31
1.60	dissolveeps	31
1.61	distribute	32
1.62	doarexxrequester	32
1.63	drawbox	33
1.64	drawcolumn	34
1.65	drawellipse	35
1.66	draweps	36
1.67	drawgrid	37
1.68	drawline	38
1.69	drawpicture	38
1.70	drawpolygon	39
1.71	drawtextobj	40
1.72	duplicate	41

---

# Chapter 1

## PGS.MD1

### 1.1 addarexxgadget

ADDAREXXGADGET

External macros only!

**Purpose:** Adds a gadget to a requester and returns the gadget handle to RESULT. The requester must already be allocated.

**Syntax:** addarexxgadget <reqhandle/I type/S left/I top/I width/I [height/I]> [LABEL name/S] [LABELPOS <LEFT|RIGHT|ABOVE|ABOVELEFT|ABOVERIGHT|BELOW|BELOWLEFT|BELOWRIGHT|CENTER|CENTERLEFT|CENTERRIGHT|LEFTABOVE|LEFTBELOW|RIGHTABOVE|RIGHTBELOW>] plus gadget specific parameters...

Type	Max	Gadget Specific Parameters
EXIT	10	no additional parameters
STRING	10	[STRING text/S]
MULTILINE	5	[STRING text/S]
TEXT	10	[STRING text/S] [HIGHDATA <TRUE FALSE>] [BORDER <NONE RAISED RECESSED SHINE SHADOW TEXT>]
CHECKBOX	10	[CHECKED <TRUE FALSE>]
POPUP	10	[LIST handle/I] [CURRENT listnumber/I]
CYCLE	5	[LIST handle/I] [CURRENT listnumber/I]
SLIDER	5	[CURRENT value/D] [TOTAL value/D]
SCROLLIST	5	[LIST handle/I] [CURRENT listnumber/I]

**Format:**

Parameter	Values to enter
reqhandle	is the requester in which to add the gadget.
type	is the type of gadget to add.
left	is the left coordinate of the gadget.
top	is the top coordinate of the gadget.
width	is the horizontal size of the gadget.
height	is the vertical size of multiline and scrollist gadgets. This height MUST be included for these gadgets and must NOT be included for other gadget types. (min height=36)
LABEL	is the label attached to the gadget.
LABELPOS	is the label position relative to the gadget. (Default=left for all but checkbox which is right).

HIGHDATA makes the text the highlight color (white).  
 BORDER is the border type.  
 CHECKED is the default state of a checkbox.  
 STRING is the default string of a text gadget.  
 LIST is the list to display in the gadget.  
 CURRENT is the default list option to set for a popup or scrolling list (first item=0, Second=1...), or the default slider position.  
 TOTAL is the range of a slider.

**Result:** The handle (identification number) of the new gadget is returned to the RESULT variable. The handle is required in order to set the gadget options later.

**Notes:** The width and position of gadgets does not include the label. Labels are independent of the size of the gadget and are always positioned to the left of the gadget. In addition, labels are offset from the gadget by 4 pixels. For example, if you wanted to place a popup with a label of "Name" set 12 pixels from the left edge of the requester, you would need to set the gadget's left coordinate to 48 pixels (12 pixels + 4 pixels separation + 4 characters \* 8 pixels per character). If you just specified 12 pixels, the label would be outside of the requester.

The height of gadgets other than those with Height parameters is always 14 except for checkboxes which are 11.

There is a maximum number of each type of gadget which can be added to a requester. See the 'Max' allowed list above.

The first exit gadget added is always the default gadget, and the second is always the gadget which can be activated by pressing the Esc key. Label position is not applicable to exit gadgets because the label is centered on the gadget.

There are several types of text gadgets. A 'string' gadget is an editable text gadget. A 'multiline' text gadget is one with a scrollbar in which you can type several lines of text. A 'text' gadget is a non-editable text gadget.

Precede the character to underscore as a bound keyboard equivalent in the label name. For example, "\_Ok" would make "O" the keyboard shortcut for the "Ok" gadget.

**Example:** addarexxgadget reqhandle exit 16 114 70 hidden false label 'Cancel'  
 Cancel=RESULT

```
addarexxgadget reqhandle scrolllist 16 8 90 72 hidden false
Fonts=RESULT
```

```
addarexxgadget TestReq cycle 124 52 200 hidden falselabel 'Choices'
Choices=RESULT
```

See also

ADDAREXXLIST  
 Adds an item to a list.

```

ALLOCAREXXLIST
    Allocates a list.

ALLOCAREXXREQUESTER
    Allocates a requester.

DOAREXXREQUESTER
    Opens a requester.
FREEAREXXLIST      Unallocates an arexx list.
FREEAREXXREQUESTER Unallocates a requester.
GETAREXXGADGET     Gets the final status of a gadget.
SETAREXXGADGET     Sets a gadget.

```

Command Format

## 1.2 addarexxlist

```
ADDAREXXLIST
```

External macros only!

Purpose: Adds an item to an allocated list.

Syntax: addarexxlist <listhandle/S item/S>

Format: Parameter Values to enter  
listhandle is the list to which to add the item.  
item is the item to add.

Note: The first item added to a list is item 0, the second is item 1...

There is no limit to the number of items in a list.

Example: addarexxlist choices 'First choice'

See also

```

ADDAREXXGADGET
    Adds a gadget.

ALLOCAREXXLIST
    Allocates a list.

ALLOCAREXXREQUESTER
    Allocates a requester.

DOAREXXREQUESTER
    Opens a requester.
GETAREXXGADGET     Gets the final status of a gadget.
FREEAREXXLIST      Unallocates an arexx list.
FREEAREXXREQUESTER Unallocates a requester.
SETAREXXGADGET     Sets a gadget.

```

Command Format

## 1.3 addfontpath

ADDFONTPATH

**Purpose:** This appends a font directory path to the list of font paths added in the Font Preferences requester.

**Syntax:** addfontpath <filepath/F>

**Format:** Parameter Values to enter  
 filepath is the filepath of the font path to load.

**Example:** addfontpath 'Fonts:\_Bullet\_Outlines'  
 addfontpath 'work:psfonts/AdobeFonts'

Command Format

## 1.4 addguides

ADDGUIDES

**Purpose:** Adds guide(s) to a master page.

**Syntax:** addguides [HORIZONTAL | VERTICAL] <AT {position/D}>  
 [MPG name/S]

**Format:** Parameter Values to enter  
 HORIZONTAL specifies horizontal guides.  
 VERTICAL specifies vertical guides.  
 AT are the position(s) of the guide(s) to add.  
 MPG is the master page name and side.

**Example:** addguides vertical at 2.5 mpg left

See also SETGUIDES and  
 DELETEDGUIDES

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.5 addpoint

ADDPOINT

**Purpose:** Adds a point to a selected path.

**Syntax:** addpoint <[MOVETO pointx/D pointy/D] | [LINETO pointx/D pointy/D] |  
 [CURVETO curvex1/D curvey1/D curvex2/D curvey2/D pointx/D pointy/D] |  
 [ARCTO centerx/D centery/D radiusx/D radiusy/D startangle/A  
 endangle/A slantangle/A twistangle/A] |  
 [ARCNTO centerx/D centery/D radiusx/D radiusy/D startangle/A

```
endangle/A slantangle/A twistangle/A] |
[CLOSEPATH]> [PAGE number/S | WINDOW name/S | OBJECTID number/I]
```

Format: Parameter Values to enter

MOVETO	starts a new subpath within the path.
LINETO	draws a line to the point.
CURVETO	draws a bézier curve to the point.
ARCTO	draws an arc counterclockwise to the point.
ARCNTO	draws an arc clockwise to the point.
CLOSEPATH	closes the open path.
PAGE	is the page number. (Default=current)
WINDOW	is the window name. (Default=current)
OBJECTID	is the number of the object to manipulate.

Example: addpoint lineto 5.1 6.25  
 addpoint moveto 1.6 7.3  
 addpoint arcto 0.5 7.3 1.2 1.65 25 93  
 addpoint closepath page 'project.doc~8'

Command Format  
 Object ID numbers  
 DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.6 addtodrawing

ADDTODRAWING

Purpose: Adds all selected objects to the bottom-most selected drawing.

Syntax: addtodrawing [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter

DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name. (Default=current)

Example: addtodrawing 'Project.doc'

Command Format  
 DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.7 align

ALIGN

Purpose: Aligns objects to each other, to the page or to a point.

Syntax: align <[[LEFT | CENTER | RIGHT] [TOP | MIDDLE | BOTTOM]] |  
 [AT pointx/D pointy/D | AT pointx/D | AT pointy/D | OFFPAGE]>  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter

LEFT	vertically aligns to the left.
------	--------------------------------

CENTER vertically aligns to the center.  
 RIGHT vertically aligns to the right.  
 TOP horizontally aligns to the top.  
 MIDDLE horizontally aligns to the middle.  
 BOTTOM horizontally aligns to the bottom.  
 AT aligns to a point or axis. (Depends on CENTER/MIDDLE)  
 OFFPAGE aligns to the page.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to manipulate.

Example: align center at 5.5 document 'project.doc'  
 align left top  
 align at 5.5 6.5

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.8 allocarexxlist

ALLOCAREXXLIST

External macros only!

Purpose: Allocates a list for a list gadget (scrolling list, popup or cycle). Returns the list handle to RESULT.

Syntax: allocarexxlist (no parameters)

Result: The handle (identification number) of the new list is returned to the RESULT variable. The handle is required in order to add items to the list, or to set the list for a gadget.

Note: Lists are not part of requesters, so they can be allocated before allocating a requester, and can be set to more than one gadget. After a requester has closed, you can continue using the list in other requesters until you unallocate the list.

Warning: You should always unallocate your list before exiting the script!

Example: allocarexxlist  
 listnumber=RESULT

```
allocarexxlist
choices=RESULT
```

See also

ADDAREXXLIST  
 Adds an item to a list.

ADDAREXXGADGET  
 Adds a gadget.

ALLOCAREXXREQUESTER

Allocates a requester.

DOAREXXREQUESTER

Opens a requester.

FREEAREXXLIST	Unallocates an arexx list.
FREEAREXXREQUESTER	Unallocates a requester.
GETAREXXGADGET	Gets the final status of a gadget.
SETAREXXGADGET	Sets a gadget.

Command Format

## 1.9 allocarexxrequester

ALLOCAREXXREQUESTER

External macros only!

**Purpose:** Allocates a new requester which can be opened from an ARExx script. Returns the requester handle to RESULT.

**Syntax:** allocarexxrequester <name/S width/I height/I>

**Format:**

Parameter	Values to enter
name	is the title of the requester.
width	is the width of the requester in pixels.
height	is the height of the requester in pixels.

**Result:** The handle (identification number) of the new requester is returned to the RESULT variable. The handle is required in order to add gadgets or use the requester.

**Notes:** The width and height parameters do not take into account the requester borders and title bar; they are the inside measurements.

This command does not display the requester, but only allocates it. After allocating a requester, you can add gadgets to it and then display it. After you are done with a requester, you must unallocate the requester and any lists used in it.

**Warning:** You should always unallocate your requester before exiting the script!

**Warning:** Requester control commands are passed directly to the application library, as if PageStream itself was creating the requester and adding the gadgets. You are responsible to ensure that you do not add items to non-existent lists, or display unallocated requesters. Attempting to do something not allowed by these commands can cause a non-recoverable system error.

**Example:** allocarexxrequester 'Test Requester' 400 220  
reqhandle=RESULT

```
allocrequester 'Create Shadow' 300 119
ShadowReq=RESULT
```

See also

```

ADDAREXXLIST
  Adds an item to a list.

ALLOCAREXXLIST
  Allocates a list.

ADDAREXXGADGET
  Adds a gadget.

DOAREXXREQUESTER
  Opens a requester.
FREEAREXXLIST      Unallocates an arexx list.
FREEAREXXREQUESTER Unallocates a requester.
GETAREXXGADGET     Gets the final status of a gadget.
SETAREXXGADGET     Sets a gadget.

```

Command Format

## 1.10 appendfontprefs

APPENDFONTPREFS

**Purpose:** This loads a font prefs file and appends the font paths to those added in the Font Preferences requester.

**Syntax:** appendfontprefs <FILE filepath/F | DEFAULT>

**Format:**

Parameter	Values to enter
FILE	is the filename and path of the fontprefs file to load.
DEFAULT	appends the PageStream3.fontprefs file.

**Example:** appendfontprefs file 'NewsletterFonts.fontprefs'  
appendfontprefs default

Command Format

## 1.11 appendstyletags

APPENDSTYLETAGS

**Purpose:** This loads a style tag file and appends its styles to those added in the applicable Style Tag requester.

**Syntax:** appendstyletags <FILE filepath/F | ASK>  
[ALL | TEXT | PARAGRAPH | CHARACTER | OBJECT | COLOR]  
[STATUS | NOSTATUS] [DOCUMENT name/S | WINDOW name/S | CHAPTER name/S]

**Format:**

Parameter	Values to enter
FILE	is the filename and path of the style tag file to load.
ASK	opens a file requester to select the file to load.
ALL	appends all style tags in the file. (Default)

TEXT appends all text style tags in the file.  
 PARAGRAPH appends paragraph style tags in the file.  
 CHARACTER appends character style tags in the file.  
 OBJECT appends object style tags in the file.  
 COLOR appends color style tags (colors) in the file.  
 STATUS displays a status indicator while loading.  
 NOSTATUS does not display a status indicator. (Default)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 CHAPTER is the document/chapter name. (Default=current)

Notes: Style tag files are stored in IFF CTAG format, which is a subset of IFF CTXT, IFF ILUS and IFF DOC. You can load tags out of these files as well as out of CTAG files.

Colors are stored as color style tags, so colors may be appended to the color palette with this command.

Example: `appendstyletags file 'Work:Documents/Newsletter.doc' text`  
`appendstyletags ask`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.12 begincommandcapture

BEGINCOMMANDCAPTURE

Purpose: Start buffering text commands. This is used to avoid a separate redraw for each text attribute application.

Syntax: `begincommandcapture [DOCUMENT name/S | WINDOW name/S]`

Format: Parameter Values to enter

DOCUMENT is the document name. (Default=current)

WINDOW is the window name. (Default=current)

Example: `begincommandcapture`  
`begincommandcapture document 'project.doc'`

See also `endcommandcapture`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.13 breaktextrouting

BREAKTEXTROUTING

Purpose: Breaks the text routing link(s) to and/or from the previous and next frames in a linked series.

Syntax: breakextrouting [TOFRAME] [FROMFRAME]  
 [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Example: breakextrouting toframe

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.14 bringforward

BRINGFORWARD

Purpose: Brings selected object(s) forward one level in the stack.

Syntax: bringforward [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to manipulate.

Example: bringforward  
 bringforward document 'project.doc'

See also

BRINGTOFRONT

.

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.15 bringtofront

BRINGTOFRONT

Purpose: Brings selected object(s) forward to the top of the stack.

Syntax: bringtofront [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to manipulate.

Example: bringtofront  
 bringtofront document 'project.doc'

See also

BRINGFORWARD

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.16 clearstyletag

CLEARSTYLETAG

Purpose: Clears the current style tag.

Syntax: clearstyletag [STYLETAG name/S]

Format: Parameter Values to enter  
 STYLETAG is the current name of the style tag.

Example: clearstyletag styletag 'paragraph text'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.17 closebusyrequester

CLOSEBUSYREQUESTER

External macros only!

Purpose: Closes an open busy requester.

Syntax: closebusyrequester <reqhandle/I>

Format: Parameter Values to enter  
 reqhandle is the handle of the requester to close.

Example: openbusyrequester message 'Thinking hard...' thermometer enabled total ↔  
 100 current 0  
 bh=RESULT  
 do for i = 1 to 100  
 /\* script does something here \*/  
 /\* abort if stop is pressed, else set thermometer \*/  
 getbusyrequester bh  
 if result=1 then break else setbusyrequester bh current i  
 end i  
 closebusyrequester bh

Command Format

## 1.18 closedocument

## CLOSEDOCUMENT

Purpose: Close an open document.

Syntax: closedocument [FORCE | ALERT | QUIET]  
[DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 FORCE ignores the saved status of the document.  
 ALERT opens an alert requester if the document has been changed.  
 QUIET ignores the saved status of the document but returns an error if the document has been changed. (Default)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Notes: This command closes all open views of the document. To close a single view of a document, use the  
 CLOSEWINDOW  
 command.

Example: closedocument force document 'project.doc'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.19 closepath

## CLOSEPATH

Purpose: Closes the selected open polygon paths.

Syntax: closepath [PAGE number/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
 PAGE is the page name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to manipulate.

Example: closepath page 'project.doc~6'

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.20 closewindow

## CLOSEWINDOW

Purpose: Close an open window.

---

Syntax: `closewindow [WINDOW name/S]`

Format: Parameter Values to enter  
 WINDOW is the window name. (Default=current)

Note: This command closes a single view of the document. To close all open views of a document, use the  
 CLOSEDOCUMENT  
 command.

Example: `closewindow force window 'Untitled Document.1~View.1'`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.21 colorpalette

COLORPALETTE

Purpose: Displays the color palette.

Syntax: `colorpalette [SHOW | HIDE | TOGGLE] [AT pointx/I pointy/I]  
 [SIZE width/I height/I] [SAVE]`

Format: Parameter Values to enter  
 SHOW toggles on the palette.  
 HIDE toggles off the palette.  
 TOGGLE toggles the palette on and off.  
 AT sets the palette position.  
 SIZE sets the palette size in pixels. (Min=160x50)  
 SAVE saves the values to PageStream3.prefs.

Note: The TOGGLE parameter may not be used in conjunction with the SAVE parameter.

Example: `colorpalette toggle`  
`colorpalette show at 500 300 size 180 172 save`  
`colorpalette hide`

Command Format

## 1.22 converttopath

CONVERTTOPATH

Purpose: Converts an object to a path.

Syntax: `converttopath [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]`

Format: Parameter Values to enter  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

OBJECTID is the number of the object to manipulate.

Example: converttopath  
 converttopath window 'Untitled Document.1~View.1'

\*\* currently broken

Command Format  
 Object ID numbers  
 DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.23 copyobject

COPYOBJECT

Purpose: Copies the selected object(s) to the clipboard.

Syntax: copyobject [STATUS | NOSTATUS]  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
 STATUS displays a status indicator while copying.  
 NOSTATUS does not display a status indicator. (Default)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to copy.

Example: copyobject  
 copyobject status document 'project.doc'

See also

CUTOBJECT  
 and PASTEOBJECT.

Command Format  
 Object ID numbers  
 DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.24 copytext

COPYTEXT

Purpose: Copies the selected text to the clipboard.

Syntax: copytext [STATUS | NOSTATUS] [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 STATUS displays a status indicator while copying.  
 NOSTATUS does not display a status indicator. (Default)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

---

Example: copytext  
 copytext status document 'project.doc'

See also

CUTTEXT  
 and PASTETEXT.

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.25 cutobject

CUTOBJECT

Purpose: Cuts the selected object(s) to the clipboard.

Syntax: cutobject [STATUS | NOSTATUS]  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
 STATUS displays a status indicator while cutting.  
 NOSTATUS does not display a status indicator. (Default)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to cut.

Example: cutobject  
 cutobject status document 'project.doc'

See also

COPYOBJECT  
 ,  
 DELETEOBJECT  
 and PASTEOBJECT.

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.26 cuttext

CUTTEXT

Purpose: Cuts the selected text to the clipboard.

Syntax: cuttext [STATUS | NOSTATUS] [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 STATUS displays a status indicator while cutting.  
 NOSTATUS does not display a status indicator. (Default)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

---

Example: `cuttext`  
`cuttext status document 'project.doc'`

See also

`COPYTEXT`  
`,`  
`DELETETEXT`  
and `PASTETEXT`.

Command Format

`DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE`

## 1.27 createcolumns

`CREATECOLUMNS`

Purpose: Draws new text column frames on a range of pages with one or more columns in them.

Syntax: `createcolumns [DEFAULT | MARGINS inside/D top/D outside/D bottom/D]`  
`[COLUMNS number/I] [GUTTER space/D] [LINKED | UNLINKED]`  
`[ROTATE angle/A | SKEW slantangle/A twistangle/A | SLANT angle/A |`  
`TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]`  
`[CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]`  
`[PAGE number/s | WINDOW name/S] [TO number/s]`

Format:	Parameter	Values to enter
	<code>DEFAULT</code>	creates frames matching the page guides. (Default)
	<code>MARGINS</code>	is the coordinates of the frames to create.
	<code>COLUMNS</code>	is the number of columns in the frame. (Default=1)
	<code>GUTTER</code>	is the space between columns. (Default=0)
	<code>LINKED</code>	links the frames from page to page.
	<code>UNLINKED</code>	does not link the frames.
	<code>ROTATE</code>	is the rotation angle. (Default=0)
	<code>SKEW</code>	is the slant and twist angle. (Default=0)
	<code>SLANT</code>	is the slant angle. (Default=0)
	<code>TWIST</code>	is the twist angle. (Default=0)
	<code>ABOUT</code>	is the rotation point.
	<code>ABOUTCENTER</code>	rotates around its center. (Default)
	<code>CONSTRAIN</code>	toggles on the resizing constraint.
	<code>FREE</code>	toggles off the resizing constraint. (Default)
	<code>PRINT</code>	toggles on the print flag. (Default)
	<code>NOPRINT</code>	toggles off the print flag.
	<code>INFRONT</code>	creates at the top of the stack. (Default)
	<code>INBACK</code>	creates at the bottom of the stack.
	<code>PAGE</code>	is the page number. (Default=current)
	<code>WINDOW</code>	is the window name. (Default=current)
	<code>TO</code>	is the number of the last page in the range. (Default=same as <code>PAGE</code> )

Notes: Use the `DRAWCOLUMN` command to draw a single column on one page and to draw a fixed size column. Use the `CREATECOLUMNS` command to draw

columns on more than one page, and to draw columns relative to the page size.

Example: `createcolumns default columns 2 gutter 0.25 column to 12`  
`createcolumns margin 1 1.25 0.75 1 page 5 to 15`  
`createcolumns default rotate 45 inback page 'project.doc~8'`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.28 createcompound

CREATECOMPOUND

Purpose: Merges the selected shapes together to create a compound object.

Syntax: `createcompound`  
`[POSITION left/D top/D right/D bottom/D]`  
`[CONTENTOFFSET offsetx/D offsety/D] [CONTENTSCALE scalex/P scaley/P]`  
`[ROTATE angle/A | SKEW slantangle/A twistangle/A |`  
`SLANT angle/A | TWIST angle/A]`  
`[ABOUT pointx/D pointy/D | ABOUTCENTER] [CONSTRAIN | FREE]`  
`[PRINT | NOPRINT] [INFRONT | INBACK | BEST]`  
`[DOCUMENT name/S | WINDOW name/S]`

Format:	Parameter	Values to enter
	POSITION	is the coordinates of the frame. (Default=current)
	CONTENTOFFSET	is the offset in the frame. (Default=0,0)
	CONTENTSCALE	is the scale of the object in the frame. (Default=100%)
	ROTATE	is the rotation angle. (Default=0)
	SKEW	is the slant and twist angle. (Default=0)
	SLANT	is the slant angle. (Default=0)
	TWIST	is the twist angle. (Default=0)
	ABOUT	is the rotation point.
	ABOUTCENTER	rotates around its center. (Default)
	CONSTRAIN	toggles on the resizing constraint.
	FREE	toggles off the resizing constraint. (Default)
	PRINT	toggles on the print flag. (Default)
	NOPRINT	toggles off the print flag.
	INFRONT	creates at the top of the stack.
	INBACK	creates at the bottom of the stack.
	BEST	creates at the optimum stack level. (Default)
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Result: The identification number (handle) of the new object is returned to the RESULT variable.

Example: `createcompound`  
`createcompound position 1.5 2.5 3.25 6.457 contentscale 80.5 constrain`

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.29 createdrawing

CREATEDRAWING

Purpose: Creates a new drawing.

Syntax: `createdrawing`  
 [POSITION left/D top/D right/D bottom/D]  
 [CONTENTOFFSET offsetx/D offsety/D] [CONTENTSCALE scalex/P scaley/P]  
 [ROTATE angle/A | SKEW slantangle/A twistangle/A |  
 SLANT angle/A | TWIST angle/A]  
 [ABOUT pointx/D pointy/D | ABOUTCENTER] [CONSTRAIN | FREE]  
 [PRINT | NOPRINT] [INFRONT | INBACK | BEST]  
 [DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	POSITION	is the coordinates of the frame. (Default=current)
	CONTENTOFFSET	is the offset in the frame. (Default=0,0)
	CONTENTSCALE	is the scale of the object in the frame. (Default=100%)
	ROTATE	is the rotation angle. (Default=0)
	SKEW	is the slant and twist angle. (Default=0)
	SLANT	is the slant angle. (Default=0)
	TWIST	is the twist angle. (Default=0)
	ABOUT	is the rotation point.
	ABOUTCENTER	rotates around its center. (Default)
	CONSTRAIN	toggles on the resizing constraint.
	FREE	toggles off the resizing constraint. (Default)
	PRINT	toggles on the print flag. (Default)
	NOPRINT	toggles off the print flag.
	INFRONT	creates at the top of the stack.
	INBACK	creates at the bottom of the stack.
	BEST	creates at the optimum stack level. (Default)
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Result: The identification number (handle) of the new object is returned to the RESULT variable.

Example: `createdrawing`  
`createdrawing position 1.5 2.5 3.25 6.457 contentscale 80.5 constrain`

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.30 createpath

CREATEPATH

Purpose: Starts a new path.

Syntax: `createpath <left/D top/D right/D bottom/D>`  
 [ROTATE angle/A | SKEW slantangle/A twistangle/A |  
 SLANT angle/A | TWIST angle/A]

```
[ABOUT pointx/D pointy/D | ABOUTCENTER] [CONSTRAIN | FREE]
[PRINT | NOPRINT] [INFRONT | INBACK]
[PAGE number/S | WINDOW name/S]
```

Format: Parameter Values to enter

left...	is the coordinates of the object.
ROTATE	is the rotation angle. (Default=0)
SKEW	is the slant and twist angle. (Default=0)
SLANT	is the slant angle. (Default=0)
TWIST	is the twist angle. (Default=0)
ABOUT	is the rotation point.
ABOUTCENTER	rotates around its center. (Default)
CONSTRAIN	toggles on the resizing constraint.
FREE	toggles off the resizing constraint. (Default)
PRINT	toggles on the print flag. (Default)
NOPRINT	toggles off the print flag.
INFRONT	creates at the top of the stack.
INBACK	creates at the bottom of the stack.
BEST	creates at the optimum stack level. (Default)
DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name. (Default=current)

Result: The identification number (handle) of the new object is returned to the RESULT variable.

Example: `createpath 1.5 2.5 3.25 6.457 scale 80.5 constrain`

Command Format  
Object ID numbers  
DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.31 currentchapter

CURRENTCHAPTER

External macros only!

Purpose: Gets the name of the current chapter.

Syntax: `currentchapter [DOCUMENT name/S | WINDOW name/S]`

Format: Parameter Values to enter

DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name.

Result: The name is returned to RESULT.

Example: `currentchapter /* will return the name of the current chapter */  
chapname=result`

Command Format  
DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.32 currentchapterpath

CURRENTCHAPTERPATH

External macros only!

Purpose: Gets the full path of the current chapter. The path is:  
documentname[:chaptername[:subchaptername]]

Syntax: currentchapterpath [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
DOCUMENT is the document name. (Default=current)  
WINDOW is the window name.

Result: The path is returned to RESULT.

Example: currentchapterpath /\* will return the path of the current chapter \*/  
chappath=result

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.33 currentdocument

CURRENTDOCUMENT

External macros only!

Purpose: Gets the name of the current document.

Syntax: currentdocument [WINDOW name/S]

Format: Parameter Values to enter  
WINDOW is the window name. (Default=current)

Result: The name is returned to RESULT.

Example: currentdocument /\* will return the name of the current document \*/  
docname=result

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.34 currentmasterpage

CURRENTMASTERPAGE

External macros only!

Purpose: Gets the name of the current master page.

---

Syntax: `currentmasterpage [DOCUMENT name/S | WINDOW name/S]`

Format: Parameter Values to enter  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name.

Result: The name is returned to RESULT.

Example: `currentmasterpage /* will return the name of the current master page */  
 pagenum=result`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

### 1.35 currentmasterpagepath

CURRENTMASTERPAGEPATH

External macros only!

Purpose: Gets the full path of the current master page. The path is:  
 [documentname[:chaptername[:subchaptername]~]masterpagename

Syntax: `currentmasterpagepath [DOCUMENT name/S | WINDOW name/S]`

Format: Parameter Values to enter  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name.

Result: The path is returned to RESULT.

Example: `currentmasterpagepath /* will return the path of the current master ↵  
 page */  
 mpagepath=result`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

### 1.36 currentpage

CURRENTPAGE

External macros only!

Purpose: Gets the number of the current page.

Syntax: `currentpage [WINDOW name/S]`

Format: Parameter Values to enter  
 PAGE is the page number. (Default=current)  
 WINDOW is the window name.

Result: The number is returned to RESULT.

Example: `currentpage /* will return the number of the current page */  
pagenum=result`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.37 currentpagepath

CURRENTPAGEPATH

External macros only!

Purpose: Gets the full path of the current page. The path is:  
[documentname[:chaptername[:subchaptername]~]pagenumber

Syntax: `currentpagepath [WINDOW name/S]`

Format: Parameter Values to enter  
WINDOW is the window name. (Default=current)

Result: The path is returned to RESULT.

Example: `currentpagepath /* will return the path of the current page */  
pagepath=result`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.38 currentwindow

CURRENTWINDOW

External macros only!

Purpose: Gets the name of the current window.

Syntax: `currentwindow`

Format: Parameter Values to enter  
WINDOW is the window name. (Default=current)

Result: The name is returned to RESULT.

Example: `currentwindow /* will return the name of the current window */  
winname=result`

Command Format

---

## 1.39 currentwindowpath

CURRENTWINDOWPATH

External macros only!

Purpose: Gets the full path of the current window. The path is:  
[documentname~]windowname

Syntax: currentwindowpath [WINDOW name/S]

Format: Parameter Values to enter  
WINDOW is the window name. (Default=current)

Result: The path is returned to RESULT.

Example: currentwinpath /\* will return the path of the current page \*/  
winpath=result

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.40 deletechapter

DELETECHAPTER

Purpose: Deletes a chapter in a document.

Syntax: deletechapter [CHAPTER name/S | WINDOW name/S] [ALERT]

Format: Parameter Values to enter  
CHAPTER is the document/chapter name. (Default=current)  
WINDOW is the window name. (Default=current)  
ALERT opens an alert requester. (Default=off)

Example: deletechapter chapter 'DocumentName:Chapter 3 alert'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.41 deleteguides

DELETEDGUIDES

Purpose: Deletes existing guide(s) from a master page.

Syntax: deleteguides [HORIZONTAL | VERTICAL]  
<AT position/D MPG name//S>

Format: Parameter Values to enter  
HORIZONTAL specifies horizontal guides  
VERTICAL specifies vertical guides

AT            are the position(s) of the guide(s) to delete.  
 MPG          is the master page name and side.

Example: deleteguides vertical at 2.5 mpg left

See also

ADDGUIDES  
 and SETGUIDES.

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.42 deletemasterpage

DELETEMASTERPAGE

Purpose: Delete an existing master page in a document or chapter.

Syntax: deletemasterpage [MASTERPAGE name/S]

Format: Parameter Values to enter

MASTERPAGE is the master page name. (Default=current)

Example: deletemasterpage 'Two Column Master Page'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.43 deleteobject

DELETEOBJECT

Purpose: Deletes selected object(s).

Syntax: deleteobject [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter

DOCUMENT is the document name. (Default=current)

WINDOW is the window name. (Default=current)

OBJECTID is the number of the object to manipulate.

Example: deleteobject document 'project.doc'

See also

CUTOBJECT

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.44 deletepage

**DELETEPAGE**

Purpose: Deletes page(s) in a document.

Syntax: deletepage [PAGE number/S] [TO number/S]  
[FORCE | ALERT | QUIET]

Format: Parameter Values to enter  
 PAGE is the number of the first page to delete.  
 (Default=current)  
 TO is the number of the last page to delete.  
 FORCE ignores whether there are objects on the pages to be  
 deleted.  
 ALERT opens an alert requester.  
 QUIET ignores whether there are objects on the pages to be  
 deleted, but returns if there are objects. (Default)

Example: deletepage page 5 to 7 force  
 deletepage page 'Project.doc~5' to 7

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

**1.45 deletestyletag****DELETESTYLETAG**

Purpose: Deletes a style tag.

Syntax: deletestyle <STYLETAG name/S>

Format: Parameter Values to enter  
 STYLETAG is the name of the style to delete.

Example: deletestyletag styletag 'Body Text'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

**1.46 deletetext****DELETETEXT**

Purpose: Deletes selected text.

Syntax: deletetext [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Example: deletetext document 'Project.Doc~9'

See also

CUTTEXT

.

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.47 deletevariable

DELETEVARIABLE

Purpose: Deletes a user variable.

Syntax: deletevariable <STYLEVAR name/S>

Format: Parameter Values to enter  
 STYLEVAR is the document/chapter/variable name. (Default=current)

Example: deletevariable stylevar MyBook.doc~author

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.48 display

DISPLAY

Purpose: Changes the page and scale displayed in an open window.

Syntax: display [PAGE [number/s | PREVIOUS] | MPG name/S]  
 [SCALE <custom/P | FULLPAGE | FULLBLEED | FULLPAGEWIDTH |  
 FULLBLEEDWIDTH | FULLPAGEHEIGHT | FULLBLEEDHEIGHT | PREVIOUS>]  
 [OFFSET offsetx/D offsety/D] [WINDOW name/S]

Format: Parameter Values to enter  
 PAGE is the page number to display.  
 PREVIOUS displays the last page edited.  
 MPG is the master page name and side to display.  
 SCALE is the a predefined or custom view scale. (Default=current)  
 PREVIOUS displays the last custom view scale.  
 OFFSET is the page offset from the top left of the window.  
 (Default=current)  
 WINDOW is the window name. (Default=current)

Example: display page 7 scale fullpage  
 display page 'Project.doc~7' scale 117 offset 5.5 '-3'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.49 displaydrawings

DISPLAYDRAWINGS

Purpose: Changes the drawing display status.

Syntax: displaydrawings <SHOW | HIDE | TOGGLE> [WINDOW name/S]

Format: Parameter Values to enter  
SHOW displays drawings.  
HIDE displays a placement box for drawings.  
TOGGLE toggles the display of drawing on and off.  
WINDOW is the window name. (Default=current)

Example: displaydrawings hide

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.50 displayfacingpages

DISPLAYFACINGPAGES

Purpose: Toggles the display of facing pages.

Syntax: displayfacingpages <ON | OFF | TOGGLE> [WINDOW name/S]

Format: Parameter Values to enter  
ON toggles facing pages on.  
OFF toggles facing pages off.  
TOGGLE toggles facing pages on and off.  
WINDOW is the window name. (Default=current)

Example: displayfacingpages on

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.51 displaygrid

DISPLAYGRID

Purpose: Changes the grid display status.

Syntax: displaygrid <SHOW | HIDE | TOGGLE> [INFRONT | INBACK]  
[COLOR number/I] [WINDOW name/S]

Format: Parameter Values to enter  
SHOW displays the drawing grid.  
HIDE hides the drawing grid.  
TOGGLE toggles the drawing grid on and off.  
INFRONT draws the grid on top of objects.

---

INBACK        draws the grid behind objects.  
 COLOR        is the grid color (0, 1, 2, or 3).  
 WINDOW       is the window name. (Default=current)

Example: `displaygrid show infront color 2`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.52 displayguides

DISPLAYGUIDES

Purpose: Changes the guide display status.

Syntax: `displayguides <SHOW | HIDE | TOGGLE> [INFRONT | INBACK]  
 [COLOR number/I] [RULER <ON | OFF | TOGGLE>]  
 [PAGE <ON | OFF | TOGGLE>] [WINDOW name/S]`

Format: Parameter Values to enter  
 SHOW        displays guides.  
 HIDE        hides guides.  
 TOGGLE     toggles the guides on and off.  
 INFRONT    draws guides on top of objects.  
 INBACK     draws guides behind objects.  
 COLOR       is the guide color (0, 1, 2, or 3).  
 RULER       changes the ruler guide state.  
 PAGE        changes the page guide state.  
 ON          toggles the option on.  
 OFF         toggles the option off.  
 TOGGLE     toggles the option on and off.  
 WINDOW     is the window name. (Default=current)

Example: `displayguides show infront color 2 ruler on page toggle`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.53 displayinvisibles

DISPLAYINVISIBLES

Purpose: Changes the invisible symbol display status.

Syntax: `displayinvisibles [SHOW | HIDE | TOGGLE] [WINDOW name/S]`

Format: Parameter Values to enter  
 SHOW        displays invisible symbols.  
 HIDE        hides invisible symbols.  
 TOGGLE     toggles the display of invisible symbols on and off.  
 WINDOW     is the window name. (Default=current)

Example: displayinvisibles toggle

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.54 displaytextlinks

DISPLAYTEXTLINKS

Purpose: Changes the text routing links display status

Syntax: displaylinks <SHOW | HIDE | TOGGLE> [WINDOW name/S]

Format: Parameter Values to enter  
SHOW displays text links.  
HIDE hides text links.  
TOGGLE toggles the display of text links on and off.  
WINDOW is the window name. (Default=current)

Example: displaytextlinks toggle

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.55 displayoutlines

DISPLAYOUTLINES

Purpose: Changes the frame and column outline display status.

Syntax: displayoutlines <SHOW | HIDE | TOGGLE> [WINDOW name/S]

Format: Parameter Values to enter  
SHOW displays the frame and column outlines.  
HIDE hides the frame and column outlines.  
TOGGLE toggles the outlines on and off.  
WINDOW is the window name. (Default=current)

Example: displayoutlines toggle

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.56 displaypictures

DISPLAYPICTURES

Purpose: Changes the picture display status.

Syntax: displaypictures <SHOW | HIDE | TOGGLE> [WINDOW name/S]

---

Format: Parameter Values to enter  
 SHOW displays pictures.  
 HIDE displays a placement box for pictures.  
 TOGGLE toggles the display of pictures on and off.  
 WINDOW is the window name. (Default=current)

Example: displaypictures hide

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.57 displayruler

DISPLAYRULER

Purpose: Changes the ruler display status.

Syntax: displayruler <SHOW | HIDE | TOGGLE> [OFFSET offsetx/I offsety/I]  
 [ZERO zerox/D zeroy/D] [MSYS horzunits/S vertunits/S]  
 [DIRECTION <LEFT | RIGHT> <UP | DOWN>] [WINDOW name/S]

Format: Parameter Values to enter  
 SHOW displays the rulers.  
 HIDE hides the rulers.  
 TOGGLE toggles the rulers on and off.  
 OFFSET is the ruler offset in pixels from the top left corner of the window.  
 ZERO is the ruler zero point.  
 MSYS is the measurement system units for the rulers.  
 DIRECTION is the direction in which the rulers are measured.  
 WINDOW is the window name. (Default=current)

Notes: Allowed strings for MSYS are:

- INCHES
- CENTIMETERS
- MILLIMETERS
- PICAS
- POINTS
- PRINTERPICAS
- PRINTERPOINTS
- CICEROS
- DIDOTPOINTS
- FEET
- METERS
- SAMEAS: sets to the same as the applicable measurement system.

Example: displayruler show offset 30 120 msys inches inches  
 displayruler direction left up  
 displayruler zero 1.5 4.25

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.58 dissolvecompound

DISSOLVECOMPOUND

Purpose: Dissolves a selected compound object.

Syntax: dissolvecompound [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
DOCUMENT is the document name. (Default=current)  
WINDOW is the window name. (Default=current)  
OBJECTID is the number of the object to manipulate.

Example: dissolvecompound document 'project.doc'

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.59 dissolvedrawing

DISSOLVEDRAWING

Purpose: Dissolves a selected drawing.

Syntax: dissolvedrawing [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
DOCUMENT is the document name. (Default=current)  
WINDOW is the window name. (Default=current)  
OBJECTID is the number of the object to manipulate.

Example: dissolvedrawing document 'project.doc'

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.60 dissolveeps

DISSOLVEEPS

Purpose: Separates an EPS object and its bitmap preview into two separate objects. The bitmap preview will become a separate picture object and the EPS object will no longer have a preview. This command is only applicable to EPS objects with bitmap previews.

Syntax: dissolveeps [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
DOCUMENT is the document name.  
WINDOW is the window name.

---

OBJECTID is the number of the object. (Default=current)

Example: dissolve object  
 dissolve object document 'project.doc'

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.61 distribute

DISTRIBUTE

Purpose: Distribute objects over an area.

Syntax: distribute [HORIZONTAL <LEFT | CENTER | RIGHT | GAP>  
 [BOUNDINGBOX | SPACING space/D]]  
 [VERTICAL <TOP | MIDDLE | BOTTOM | GAP>  
 [BOUNDINGBOX | SPACING space/D]]  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format:	Parameter	Values to enter
	HORIZONTAL	is horizontal distribution.
	LEFT	is left edge alignment.
	CENTER	is center axis alignment.
	RIGHT	is right edge alignment.
	GAP	is gap alignment.
	VERTICAL	is vertical distribution.
	TOP	is top edge alignment.
	MIDDLE	is middle axis alignment.
	BOTTOM	is bottom edge alignment.
	BOUNDING BOX	distributes objects in their bounding box. (Default)
	SPACE	distributes objects with a set spacing.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)
	OBJECTID	is the number of the object to manipulate.

Example: distribute horizontal left boundingbox

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.62 doarexxrequester

DOAREXXREQUESTER

Purpose: Displays an allocated requester.

Syntax: doarexxrequester <reqhandle/I>

Format: Parameter Values to enter

reqhandle is the requester to display.

**Result:** The handle of the exit gadget selected will be returned to the RESULT variable.

**Notes:** This command displays a requester, but the requester and its gadgets must be allocated first.

**Warning:** You should always unallocate your requester before exiting the script!

**Example:** doarexxrequester TestRequester  
action=RESULT

See also

ADDAREXXLIST  
Adds an item to a list.

ALLOCAREXXLIST  
Allocates a list.

ADDAREXXGADGET  
Adds a gadget.

ALLOCAREXXREQUESTER  
Allocates a requester.

FREEAREXXLIST Unallocates an arexx list.

FREEAREXXREQUESTER Unallocates a requester.

GETAREXXGADGET Gets the final status of a gadget.

SETAREXXGADGET Sets a gadget.

Command Format

## 1.63 drawbox

DRAWBOX

**Purpose:** Draws a new box.

**Syntax:** drawbox <left/D top/D right/D bottom/D>  
[NORMAL | ROUND | SCALLOP | BEVEL | INSET]  
[CORNER radius/D | ECORNER radiusx/D radiusy/D]  
[ROTATE angle/A | SKEW slantangle/A twistangle/A | SLANT angle/A |  
TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]  
[CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]  
[PAGE number/s | MPG name/S | DOCUMENT name/S | WINDOW name/S]

**Format:**

Parameter	Values to enter
left...	is the coordinates of the box.
NORMAL	sets square corners. (Corner radii=0)
ROUND	sets outward rounded corners. (Corner radii>0)
SCALLOP	sets inward rounded corners. (Corner radii>0)
BEVEL	sets angled corners. (Corner radii>0)
INSET	sets square insert corners. (Corner radii>0)
CORNER	is the corner radius. (Default=0)

ECORNER is the elliptical corner radius. (Default=0)  
 ROTATE is the rotation angle. (Default=0)  
 SKEW is the slant and twist angle. (Default=0)  
 SLANT is the slant angle. (Default=0)  
 TWIST is the twist angle. (Default=0)  
 ABOUT is the rotation point.  
 ABOUTCENTER rotates around its center. (Default)  
 CONSTRAIN toggles on the resizing constraint.  
 FREE toggles off the resizing constraint. (Default)  
 PRINT toggles on the print flag. (Default)  
 NOPRINT toggles off the print flag.  
 INFRONT creates at the top of the stack. (Default)  
 INBACK creates at the bottom of the stack.  
 PAGE is the page number. (Default=current)  
 MPG is the master page name and side. (Default=current)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

**Result:** The identification number (handle) of the new object is returned to the RESULT variable.

**Example:** drawbox 1.75 1.5 2.75 2.5  
 drawbox 3 3 6 6 inback page 'project.doc~8'  
 drawbox 6 7 8 9 ecorner 0.1 0.2 skew 0 45

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.64 drawcolumn

### DRAWCOLUMN

**Purpose:** Draws a new text column frame with one or more columns in it.

**Syntax:** drawcolumn <left/D top/D right/D bottom/D>  
 [COLUMNS number/I] [GUTTER space/D]  
 [ROTATE angle/A | SKEW slantangle/A twistangle/A | SLANT angle/A |  
 TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]  
 [CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]  
 [PAGE number/s | MPG name/S | DOCUMENT name/S | WINDOW name/S]

**Format:**

Parameter	Values to enter
left...	is the coordinates of the frame or frameless text object.
COLUMNS	is the number of columns in a text frame.
GUTTER	is the space between columns.
ROTATE	is the rotation angle. (Default=0)
SKEW	is the slant and twist angle. (Default=0)
SLANT	is the slant angle. (Default=0)
TWIST	is the twist angle. (Default=0)
ABOUT	is the rotation point.
ABOUTCENTER	rotates around its center. (Default)
CONSTRAIN	toggles on the resizing constraint.
FREE	toggles off the resizing constraint. (Default)
PRINT	toggles on the print flag. (Default)

NOPRINT toggles off the print flag.  
 INFRONT creates at the top of the stack. (Default)  
 INBACK creates at the bottom of the stack.  
 PAGE is the page number. (Default=current)  
 MPG is the master page name and side. (Default=current)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

**Result:** The identification number (handle) of the new object is returned to the RESULT variable.

**Notes:** Use the DRAWCOLUMN command to draw a single column on one page and to draw a fixed size column. Use the  
           CREATECOLUMNS  
           command to draw  
 columns on more than one page, and to draw columns relative to the page size.

**Example:** drawcolumn 1 1 7.5 10 columns 2 gutter 0.25  
 drawcolumn 1 1 5 6 rotate 45 inback page 'project.doc~8'

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.65 drawellipse

DRAWELLIPSE

**Purpose:** Draws a new ellipse.

**Syntax:** drawellipse <centerx/D centery/D radiusx/D radiusy/D>  
 [PIE | ARC | ELLIPSE] [ANGLES startangle/A endangle/A]  
 [ROTATE angle/A | SKEW slantangle/A twistangle/A | SLANT angle/A |  
 TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]  
 [CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]  
 [PAGE number/s | MPG name/S | DOCUMENT name/S | WINDOW name/S]

**Format:**

Parameter	Values to enter
center...	is the coordinates of the center point.
RADIUS...	is the radius of the ellipse.
PIE	is a closed arc of an ellipse.
ARC	is an open arc of an ellipse
ELLIPSE	is a closed ellipse. (Default)
ANGLES	are the starting and ending angles. (Default=90\textdegree ↔ {},360\textdegree{})
ROTATE	is the rotation angle. (Default=0)
SKEW	is the slant and twist angle. (Default=0)
SLANT	is the slant angle. (Default=0)
TWIST	is the twist angle. (Default=0)
ABOUT	is the rotation point.
ABOUTCENTER	rotates around its center. (Default)
CONSTRAIN	toggles on the resizing constraint.
FREE	toggles off the resizing constraint. (Default)
PRINT	toggles on the print flag. (Default)

NOPRINT toggles off the print flag.  
 INFRONT creates at the top of the stack. (Default)  
 INBACK creates at the bottom of the stack.  
 PAGE is the page number. (Default=current)  
 MPG is the master page name and side. (Default=current)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

**Result:** The identification number (handle) of the new object is returned to the RESULT variable.

**Example:** drawellipse 3 3 1.5 1.5  
 drawellipse 3 3 1.5 1.5 page 'project.doc~6' constrain  
 drawellipse 5.25 6.125 1.5 2.5 pie angles 45 90 print

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.66 draweps

DRAWEPS

**Purpose:** Places an Encapsulated PostScript illustration frame.

**Syntax:** draweps <left/D top/D right/D bottom/D>  
 [CONTENTOFFSET offsetx/D offsety/D] [CONTENTSCALE scalex/P scaley/P]  
 [ROTATE angle/A | SKEW slantangle/A twistangle/A | SLANT angle/A |  
 TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]  
 [CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]  
 [FRAMELESS | FRAMED] [INTERNAL | EXTERNAL] [FILE filepath/F]  
 [PAGE number/s | MPG name/S | DOCUMENT name/S | WINDOW name/S]

**Format:**

Parameter	Values to enter
left...	is the coordinates of the frame.
CONTENTOFFSET	is the offset in the frame. (Default=0,0)
CONTENTSCALE	is the scale of the object in the frame. (Default=100%)
ROTATE	is the rotation angle. (Default=0)
SKEW	is the slant and twist angle. (Default=0)
SLANT	is the slant angle. (Default=0)
TWIST	is the twist angle. (Default=0)
ABOUT	is the rotation point.
ABOUTCENTER	rotates around its center. (Default)
CONSTRAIN	toggles on the resizing constraint.
FREE	toggles off the resizing constraint. (Default)
PRINT	toggles on the print flag. (Default)
NOPRINT	toggles off the print flag.
FRAMELESS	makes the object frameless.
FRAMED	makes the object framed.
INTERNAL	stores the EPS object in the document. (Default)
EXTERNAL	does not store the EPS object in the document.
FILE	is the filepath and name of the EPS object.
INFRONT	creates at the top of the stack. (Default)
INBACK	creates at the bottom of the stack.
PAGE	is the page number. (Default=current)

MPG is the master page name and side. (Default=current)  
DOCUMENT is the document name. (Default=current)  
WINDOW is the window name. (Default=current)

Result: The identification number (handle) of the new object is returned to the RESULT variable.

Example: draweps 1.75 1.5 2.75 2.5  
draweps 3 3 6 6 offset -2 -2 contentscale 80 constrain  
draweps 2 2.5 3.125 5.625 page 'project.doc~6'

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.67 drawgrid

DRAWGRID

Purpose: Draws a new grid object.

Syntax: drawgrid <pointx1/D pointy1/D pointx2/D pointy2/D |  
POINTS pointx1/D pointy1/D pointx2/D pointy2/D pointx3/D  
pointy3/D pointx4/D pointy4/D> [DIVISIONS numberx/I numbery/I]  
[ROTATE angle/A | SKEW slantangle/A twistangle/A | SLANT angle/A |  
TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]  
[CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]  
[PAGE number/s | MPG name/S | DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter

pointx1... is the coordinates of a rectangular grid.

POINTS is the coordinates of the vertices of a non-rectangular grid. (Numbered counterclockwise)

DIVISIONS are the number of grid cells in each direction. (Default=4x4)

ROTATE is the rotation angle. (Default=0)

SKEW is the slant and twist angle. (Default=0)

SLANT is the slant angle. (Default=0)

TWIST is the twist angle. (Default=0)

ABOUT is the rotation point.

ABOUTCENTER rotates around its center. (Default)

CONSTRAIN toggles on the resizing constraint.

FREE toggles off the resizing constraint. (Default)

PRINT toggles on the print flag. (Default)

NOPRINT toggles off the print flag.

INFRONT creates at the top of the stack. (Default)

INBACK creates at the bottom of the stack.

PAGE is the page number. (Default=current)

MPG is the master page name and side. (Default=current)

DOCUMENT is the document name. (Default=current)

WINDOW is the window name. (Default=current)

Result: The identification number (handle) of the new object is returned to the RESULT variable.

```
Example: drawgrid 2 2 6 6 divisions 5 10
         drawgrid 2 2 6 6 divisions 5 10 constrain noprint
         drawgrid points 1 1 1 3 5 6 2 3 page 'project.doc~6'
```

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.68 drawline

DRAWLINE

Purpose: Draws a new line.

```
Syntax: drawline <left/D top/D right/D bottom/D>
        [ROTATE angle/A | SKEW slantangle/A twistangle/A | SLANT angle/A |
        TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]
        [CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]
        [PAGE number/s | MPG name/S | DOCUMENT name/S | WINDOW name/S]
```

Format:	Parameter	Values to enter
	left...	is the coordinates of the line.
	ROTATE	is the rotation angle. (Default=0)
	SKEW	is the slant and twist angle. (Default=0)
	SLANT	is the slant angle. (Default=0)
	TWIST	is the twist angle. (Default=0)
	ABOUT	is the rotation point.
	ABOUTCENTER	rotates around its center. (Default)
	CONSTRAIN	toggles on the resizing constraint.
	FREE	toggles off the resizing constraint. (Default)
	PRINT	toggles on the print flag. (Default)
	NOPRINT	toggles off the print flag.
	INFRONT	creates at the top of the stack. (Default)
	INBACK	creates at the bottom of the stack.
	PAGE	is the page number. (Default=current)
	MPG	is the master page name and side. (Default=current)
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Result: The identification number (handle) of the new object is returned to the RESULT variable.

```
Example: drawline 1 1 5 5
         drawline 2 1.25 8.5 11 page 'project.doc~6' noprint
```

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.69 drawpicture

## DRAWPICTURE

Purpose: Places a picture frame.

Syntax: drawpicture <left/D top/D right/D bottom/D>  
 [CONTENTOFFSET offsetx/D offsety/D] [CONTENTSCALE scalex/P scaley/P]  
 [ROTATE angle/A | SKEW slantangle/A twistangle/A | SLANT angle/A |  
 TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]  
 [CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]  
 [DPI xdpi/I ydpi/I] [FRAMELESS | FRAMED] [INTERNAL | EXTERNAL]  
 [FILE filepath/F]  
 [PAGE number/s | MPG name/S | DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	left...	is the coordinates of the frame.
	CONTENTOFFSET	is the offset in the frame. (Default=0,0)
	CONTENTSCALE	is the scale of the object in the frame. (Default=100%)
	ROTATE	is the rotation angle. (Default=0)
	SKEW	is the slant and twist angle. (Default=0)
	SLANT	is the slant angle. (Default=0)
	TWIST	is the twist angle. (Default=0)
	ABOUT	is the rotation point.
	ABOUTCENTER	rotates around its center. (Default)
	CONSTRAIN	toggles on the resizing constraint.
	FREE	toggles off the resizing constraint. (Default)
	PRINT	toggles on the print flag. (Default)
	NOPRINT	toggles off the print flag.
	FRAMELESS	makes the object frameless.
	FRAMED	makes the object framed.
	INTERNAL	stores the picture in the document. (Default)
	EXTERNAL	does not store the picture in the document.
	FILE	is the filepath and name of the picture.
	DPI	is the resolution of the picture. (Default=72x72)
	INFRONT	creates at the top of the stack. (Default)
	INBACK	creates at the bottom of the stack.
	PAGE	is the page number. (Default=current)
	MPG	is the master page name and side. (Default=current)
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Result: The identification number (handle) of the new object is returned to the RESULT variable.

Example: drawpicture 0.5 0.5 2.5 3 dpi 150 150 external

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.70 drawpolygon

## DRAWPOLYGON

Purpose: Draws a new regular polygon.

Syntax: drawpolygon <centerx/D centery/D radiusx/D radiusy/D>  
 [SIDES number/I] [OFFSETANGLE angle/A] [DEFLECTION amount/P]  
 [ANGLEDEFLECTION amount/P] [NORMAL | STAR | PUFFY | SCALLOP | WAVY]  
 [ROTATE angle/A | SKEW slantangle/A twistangle/a | SLANT angle/A |  
 TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]  
 [CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]  
 [PAGE number/s | MPG name/S | DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	center...	is the coordinates of the center point.
	RADIUS...	is the radius of the polygon.
	SIDES	is the number of sides. (Default=5) (>2)
	OFFSETANGLE	is pre-rotation angle. (Default=0\textdegree{ })
	DEFLECTION	is the alternate point radius. (Default=40%)
	ANGLEDEFLECTION	is the alternate point angle. (Default=0%)
	NORMAL	sets straight sides. (Default)
	STAR	sets straight sides with with alternate points.
	PUFFY	sets outward curved sides with alternate points.
	SCALLOP	sets inward curved sides with alternate points.
	WAVY	sets alternating in/outward curved sides with alternate points.
	ROTATE	is the rotation angle. (Default=0)
	SKEW	is the slant and twist angle. (Default=0)
	SLANT	is the slant angle. (Default=0)
	TWIST	is the twist angle. (Default=0)
	ABOUT	is the rotation point.
	ABOUTCENTER	rotates around its center. (Default)
	CONSTRAIN	toggles on the resizing constraint.
	FREE	toggles off the resizing constraint. (Default)
	PRINT	toggles on the print flag. (Default)
	NOPRINT	toggles off the print flag.
	INFRONT	creates at the top of the stack. (Default)
	INBACK	creates at the bottom of the stack.
	PAGE	is the page number. (Default=current)
	MPG	is the master page name and side. (Default=current)
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Result: The identification number (handle) of the new object is returned to the RESULT variable.

Example: drawpolygon 0.3 1.8 1.25 1.5 sides 8 puffy

Object ID numbers

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.71 drawtextobj

DRAWTEXTOBJ

Purpose: Starts a new frameless text object.

Syntax: drawtextobj <left/D top/D>

```
[ROTATE angle/A | SKEW slantangle/A twistangle/A | SLANT angle/A |
TWIST angle/A] [ABOUT pointx/D pointy/D | ABOUTCENTER]
[CONSTRAIN | FREE] [PRINT | NOPRINT] [INFRONT | INBACK]
[PAGE number/s | MPG name/S | DOCUMENT name/S | WINDOW name/S]
```

Format: Parameter Values to enter

left/top	is the top left corner of the frameless text object.
ROTATE	is the rotation angle. (Default=0)
SKEW	is the slant and twist angle. (Default=0)
SLANT	is the slant angle. (Default=0)
TWIST	is the twist angle. (Default=0)
ABOUT	is the rotation point.
ABOUTCENTER	rotates around its center. (Default)
CONSTRAIN	toggles on the resizing constraint.
FREE	toggles off the resizing constraint. (Default)
PRINT	toggles on the print flag. (Default)
NOPRINT	toggles off the print flag.
INFRONT	creates at the top of the stack. (Default)
INBACK	creates at the bottom of the stack.
PAGE	is the page number. (Default=current)
MPG	is the master page name and side. (Default=current)
DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name. (Default=current)

Result: The identification number (handle) of the new object is returned to the RESULT variable.

Example: drawtextobj 1 1  
drawtextobj 1 1 rotate 45 inback page 'project.doc~8'

Command Format  
Object ID numbers  
DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.72 duplicate

DUPLICATE

Purpose: Duplicates the selected object(s) once with a small offset.

Syntax: duplicate [OFFSET offsetx/D offsety/D]  
[DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter

OFFSET	overrides the default duplicate offsets.
DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name. (Default=current)
OBJECTID	is the number of the object to manipulate.

Example: duplicate offset 1.125 -2.5  
duplicate document 'project.doc'

See also SETDUPLICATEDEFAULTS.

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

---