

**PGS.MD7**

Copyright © 1995 Soft-Logik Publishing Corporation

---

**COLLABORATORS**

	<i>TITLE :</i> PGS.MD7		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 23, 2022	

**REVISION HISTORY**

<i>NUMBER</i>	<i>DATE</i>	<i>DESCRIPTION</i>	<i>NAME</i>

# Contents

<b>1</b>	<b>PGS.MD7</b>	<b>1</b>
1.1	setscreen . . . . .	1
1.2	setscript . . . . .	2
1.3	setshadow . . . . .	2
1.4	setsmallcaps . . . . .	3
1.5	setsmart . . . . .	4
1.6	setstroke . . . . .	4
1.7	setstrokebegin . . . . .	5
1.8	setstrokecap . . . . .	6
1.9	setstrokecount . . . . .	6
1.10	setstrokeend . . . . .	7
1.11	setstrokejoin . . . . .	8
1.12	setstrokeoffset . . . . .	8
1.13	setstrokestyle . . . . .	9
1.14	setstrokeweight . . . . .	10
1.15	setstyletagname . . . . .	10
1.16	setsubscript . . . . .	11
1.17	setsuperscript . . . . .	11
1.18	settabruler . . . . .	12
1.19	settexteditor . . . . .	13
1.20	settextframe . . . . .	13
1.21	settextpath . . . . .	14
1.22	settextrouting . . . . .	14
1.23	settextwrap . . . . .	15
1.24	settoolmode . . . . .	15
1.25	settracking . . . . .	16
1.26	settrackrange . . . . .	16
1.27	settracktable . . . . .	17
1.28	settypeangle . . . . .	18
1.29	settypesize . . . . .	18

---

---

1.30	settypestyle	19
1.31	setunderline	19
1.32	setundolevels	20
1.33	setvariablename	21
1.34	setvariablevalue	21
1.35	setviewdefaults	21
1.36	setwidoworphan	22
1.37	setwindowname	22
1.38	setwindowdefaults	23
1.39	setwindowpos	23
1.40	simplifypath	24
1.41	skew	25
1.42	smoothpath	25
1.43	snaptogrid	26
1.44	snaptoguides	26
1.45	splitpaths	27
1.46	stroked	27
1.47	stylepalette	28
1.48	textcursor	28
1.49	transform	29
1.50	undo	29
1.51	ungroup	30
1.52	unlock	30
1.53	updatefontpath	31

---

# Chapter 1

## PGS.MD7

### 1.1 setscreen

SETSCREEN

**Purpose:** Sets the screen for an object or text, or for a style tag. If object or text is chosen and nothing is selected, it sets the screen for future objects or text, as applicable.

**Syntax:** setscreen <count/I> <DEFAULT | frequency/D> <DEFAULT | angle/A>  
 <DEFAULT | DOT | LINE | TRIANGLE | PROPELLER | DONUT | RING>  
 [OBJECT | TEXT | SHADOW | REVERSE | UNDERLINE]  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I | STYLETAG name/S]

Format:	Parameter	Values to enter
	count	**
	frequency	is the screen frequency.
	angle	is the screen angle.
	DOT	sets a dot screen type.
	LINE	sets a line screen type.
	TRIANGLE	sets a triangle screen type.
	PROPELLER	sets a propeller screen type.
	DONUT	sets a donut screen type.
	RING	sets a ring screen type.
	ALL	sets the screen type of the fill and all strokes. (Default)
	FILL	sets the screen type of the fill.
	STROKENUMBER	sets the screen type of a specific stroke number. (See note)
	OBJECT	sets the screen type of the object, not text it may contain. (Default)
	TEXT	sets the screen type of the text, not the object.
	SHADOW	sets the screen type of the shadow of text.
	REVERSE	sets the screen type of the reverse of text.
	UNDERLINE	sets the screen type of the underline of text.
	DOCUMENT	is the document name.
	WINDOW	is the window name.
	OBJECTID	is the number of the object. (Default=current)
	STYLETAG	**

**Notes:** STROKENUMBER: Strokes are numbered from 0. This is different than

the program interface which numbers them from 1.

Example: \*\*example not available yet.

Command Format

Object ID numbers

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

## 1.2 setscript

SETSCRIPT

**Purpose:** Sets the subscript or superscript style option at the insertion point, for selected text or for a style tag.

**Syntax:** setscript <OFF | SUB | CUSTOMSUB offset/P scalex/P scaley/P | SUPER | CUSTOMSUPER offset/P scalex/P scaley/P>  
[DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]

<b>Format:</b>	Parameter	Values to enter
	OFF	toggles off the style.
	SUB	is the default subscript style.
	CUSTOMSUB	sets a custom subscriptoffset and scale.
	SUPER	is the default superscript style.
	CUSTOMSUPER	sets a custom superscript offset and scale.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)
	STYLETAG	sets the subscript style for a style tag.

**Notes:** The standard subscript and superscript styles are:

- offset: 40%
- scalex: 60%
- scaley: 60%

**Example:** setscript off  
setscript customsub 30 40 40 document 'project.doc'  
setscript super styletag 'Special Style'

Command Format

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

## 1.3 setshadow

SETSHADOW

**Purpose:** Sets the shadow style option at the insertion point, for selected text or for a style tag.

**Syntax:** setshadow <OFF | STANDARD | CUSTOM offsetx/P offsety/P  
[ROTATE angle/A | SKEW slantangle/A twistangle/A]>  
[DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]

Format: Parameter Values to enter  
 OFF toggles off the style.  
 STANDARD is the default shadow style.  
 CUSTOM sets custom offsets, angle, stroke and color attributes for the shadow.  
 ROTATE is the rotation angle. (Default=current)  
 SKEW is the slant and twist angle. (Default=current)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 STYLETAG sets the shadow style for a style tag.

Notes: The standard shadow style is:

- offsetx: 15%
- offsety: 15%
- slant: 0\textdegree{}
- line: <none>
- fill: 40% tint
- color: Black

Example: setshadow standard  
 setshadow custom 25 30 skew 5 0 document 'project.doc'

Command Format

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

## 1.4 setsmallcaps

SETSMALLCAPS

Purpose: Sets the small cap style option at the insertion point, for selected text or for a style tag.

Syntax: setsmallcaps <OFF | STANDARD | CUSTOM scalex/P scaley/P>  
 [DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]

Format: Parameter Values to enter  
 OFF toggles off the style.  
 STANDARD is the default small caps style.  
 CUSTOM sets custom scale for the small caps.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 STYLETAG sets the small cap style for a style tag.

Notes: If the STANDARD option is chosen and the font does not have a true small caps or titling style, the normal style will be scaled.

The standard smallcaps style is:

- scalex: 75%
- scaley: 75%

The standard titlingcaps style is:

- scalex: 60%
- scaley: 60%

Example: setsmallcaps standard  
 setsmallcaps custom 30 40 40 document 'project.doc'



## Command Format

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

## 1.5 setsmart

### SETSMART

**Purpose:** Changes the smart typing status.

**Syntax:** setsmart [LIGATURES <ON | OFF | TOGGLE>]  
 [SINGLEQUOTES <ON | OFF | TOGGLE>] [DOUBLEQUOTES <ON | OFF | TOGGLE>]  
 [DASHES <ON | OFF | TOGGLE>] [BULLETS <ON | OFF | TOGGLE>] [SAVE]

**Format:**

Parameter	Values to enter
LIGATURES	sets the smart ligatures status.
SINGLEQUOTES	sets the smart singlequote status.
DOUBLEQUOTES	sets the smart doublequote status.
DASHES	sets the smart dash status.
BULLETS	sets the smart bullet status.
ON	toggles on the option.
OFF	toggles off the option.
TOGGLE	toggles the option on and off.
SAVE	saves the settings to the PageStream3.prefs file.

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

**Example:** setsmart ligatures on dashes off bullets toggle  
 setsmart singlequotes on doublequotes on save

## Command Format

## 1.6 setstroke

### SETSTROKE

**Purpose:** Sets the stroke dash for an object or text, or for a stroke style tag. If object or text is chosen and nothing is selected, it sets the stroke dash for future objects or text, as applicable.

**Syntax:** setstroke <SOLID | PATTERN pattern/S>  
 [ALL | FILL | STROKENUMBER number/I]  
 [OBJECT | TEXT | SHADOW | REVERSE | UNDERLINE | OUTLINE]  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I |  
 STYLETAG name/S]

**Format:**

Parameter	Values to enter
SOLID	sets the dash to solid.
PATTERN	is the dash pattern. (See note)
ALL	sets the dash of the fill and all strokes. (Default)
FILL	sets the dash of the fill.

STROKENUMBER sets the dash of a specific stroke number. (See note)  
 OBJECT sets the dash of the object, not text it may contain.  
 (Default)  
 TEXT sets the dash of the text, not the object.  
 SHADOW sets the dash of the shadow of text.  
 REVERSE sets the dash of the reverse of text.  
 UNDERLINE sets the dash of the underline of text.  
 OUTLINE sets the dash of the outline stroke of text.  
 DOCUMENT is the document name.  
 WINDOW is the window name.  
 OBJECTID is the number of the object. (Default=current)  
 STYLETAG \*\*

Notes: STROKENUMBER: Strokes are numbered from 0. This is different than the program interface which numbers them from 1.

PATTERN: Dashes are comprised of 16 dots which may be toggled on and off. Use a '1' to toggle a dot on, and a '0' to toggle a dot off. For example, a dash pattern with dashes two dots wide with two dots of space between them, would be '1100110011001100'.

Example: setstroke solid  
 setstroke pattern '1010101010101010' strokenumber 1 object  
 setstroke pattern 1111000011110000

Command Format

Object ID numbers

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

## 1.7 setstrokebegin

SETSTROKEBEGIN

Purpose: Sets the beginning arrowhead for an object, or for a stroke style tag. If nothing is selected, it sets the beginning arrowhead for future objects.

Syntax: setstrokebegin <FLAT | ROUND | ARROW> [ALL | STROKENUMBER number/I]  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I |  
 STYLETAG name/S]

Format: Parameter Values to enter  
 FLAT sets a flat line end.  
 ROUND sets a round line end.  
 SQUARE sets a arrow line end.  
 ALL sets the beginning arrowhead for all strokes. (Default)  
 STROKENUMBER sets the beginning arrowhead of a specific stroke  
 number. (See note)  
 DOCUMENT is the document name.  
 WINDOW is the window name.  
 OBJECTID is the number of the object. (Default=current)  
 STYLETAG \*\*

Notes: STROKENUMBER: Strokes are numbered from 0. This is different than the program interface which numbers them from 1.

Example: `setstrokebegin flat`  
`setstrokebegin round strokenumber 1 object`

Command Format

Object ID numbers

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

## 1.8 setstrokecap

SETSTROKECAP

Purpose: Sets the stroke cap for an object, or for a stroke style tag. If nothing is selected, it sets the stroke cap for future objects.

Syntax: `setstrokecap <BUTT | ROUND | SQUARE> [ALL | STROKENUMBER number/I]`  
`[DOCUMENT name/S | WINDOW name/S | OBJECTID number/I |`  
`STYLETAG name/S]`

Format:	Parameter	Values to enter
	BUTT	sets a butt cap.
	ROUND	sets a round cap.
	SQUARE	sets a square cap.
	ALL	sets the cap for all strokes. (Default)
	STROKENUMBER	sets the cap of a specific stroke number. (See note)
	DOCUMENT	is the document name.
	WINDOW	is the window name.
	OBJECTID	is the number of the object. (Default=current)
	STYLETAG	**

Notes: STROKENUMBER: Strokes are numbered from 0. This is different than the program interface which numbers them from 1.

Example: `setstrokecap butt`  
`setstrokecap round strokenumber 1 object`

Command Format

Object ID numbers

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

## 1.9 setstrokecount

SETSTROKECOUNT

Purpose: Sets the number of strokes for the selected object(s) or text. If nothing is selected, it sets the number of strokes for future objects or text, as applicable.

Syntax: `setstrokecount <count/I> [OBJECT | TEXT | SHADOW | REVERSE |`  
`UNDERLINE] [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I |`  
`STYLETAG name/S]`

Format: Parameter Values to enter

count	is the number of strokes to set. (1-5)
OBJECT	sets the dash style of the object, not text it may contain. (Default)
TEXT	sets the dash style of the text, not the object.
SHADOW	sets the dash style of the shadow of text.
REVERSE	sets the dash style of the reverse of text.
UNDERLINE	sets the dash style of the underline of text.
DOCUMENT	is the document name.
WINDOW	is the window name.
OBJECTID	is the number of the object. (Default=current)
STYLETAG	**

Note: Strokes are numbered from 0. This is different than the program interface which numbers them from 1. If you create two strokes for an object, they will be numbered 0 and 1. The maximum number of strokes is 5.

Example: setstrokecount '3'  
setstrokecount '2' text

Command Format

Object ID numbers

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

## 1.10 setstrokeend

SETSTROKEEND

Purpose: Sets the ending arrowhead for an object, or for a stroke style tag. If nothing is selected, it sets the beginning arrowhead for future objects.

Syntax: setstrokeend <FLAT | ROUND | ARROW> [ALL | STROKENUMBER number/I]  
[DOCUMENT name/S | WINDOW name/S | OBJECTID number/I |  
STYLETAG name/S]

Format: Parameter Values to enter

FLAT	sets a flat line end.
ROUND	sets a round line end.
SQUARE	sets a arrow line end.
ALL	sets the ending arrowhead for all strokes. (Default)
STROKENUMBER	sets the ending arrowhead of a specific stroke number. (See note)
DOCUMENT	is the document name.
WINDOW	is the window name.
OBJECTID	is the number of the object. (Default=current)
STYLETAG	**

Notes: STROKENUMBER: Strokes are numbered from 0. This is different than the program interface which numbers them from 1.

Example: setstrokeend flat  
setstrokeend round strokenumber 1 object

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

## 1.11 setstrokejoin

SETSTROKEJOIN

**Purpose:** Sets the stroke join for an object, or for a stroke style tag. If nothing is selected, it sets the stroke join for future objects.

**Syntax:** setstrokewidth <MITER <limit/A | DEFAULT> | ROUND | BEVEL>  
 [ALL | STROKENUMBER number/I] [DOCUMENT name/S | WINDOW name/S |  
 OBJECTID number/I | STYLETAG name/S]

<b>Format:</b>	Parameter	Values to enter
	MITER	sets a miter join and specifies the limit.
	DEFAULT	specifies the default miter limit. (11\textdegree{ })
	ROUND	sets a round join.
	BEVEL	sets a bevel join.
	ALL	sets the join for all strokes. (Default)
	STROKENUMBER	sets the join of a specific stroke number. (See note)
	DOCUMENT	is the document name.
	WINDOW	is the window name.
	OBJECTID	is the number of the object. (Default=current)
	STYLETAG	**

**Notes:** STROKENUMBER: Strokes are numbered from 0. This is different than the program interface which numbers them from 1.

**Example:** setstrokejoin bevel  
 setstrokejoin round strokenum 1 object  
 setstrokejoin miter 15

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

## 1.12 setstrokeoffset

SETSTROKEOFFSET

**Purpose:** Sets the stroke offset for an object or text, or for a stroke style tag. If object or text is chosen and nothing is selected, it sets the stroke offset for future objects or text, as applicable.

**Syntax:** setstrokeoffset <offset/D> [ALL | STROKENUMBER number/I]  
 [OBJECT | TEXT | SHADOW | REVERSE | UNDERLINE]  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I |  
 STYLETAG name/S]

**Format:** Parameter Values to enter

width	is the the offset of the stroke.
ALL	sets the offset for all strokes. (Default)
STROKENUMBER	sets the offset of a specific stroke number. (See note)
OBJECT	sets the stroke offset of the object, not text it may contain. (Default)
TEXT	sets the stroke offset of the text, not the object.
SHADOW	sets the stroke offset of the shadow of text.
REVERSE	sets the stroke offset of the reverse of text.
UNDERLINE	sets the stroke offset of the underline of text.
DOCUMENT	is the document name.
WINDOW	is the window name.
OBJECTID	is the number of the object. (Default=current)
STYLETAG	**

Notes: STROKENUMBER: Strokes are numbered from 0. This is different than the program interface which numbers them from 1.

Example: `setstrokeoffset 0.125`  
`setstrokeoffset '3pt' strokenumber 1 object`

Command Format

Object ID numbers

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

## 1.13 setstrokestyle

SETSTROKESTYLE

Purpose: Sets the stroke dash style tag for the selected object(s) or text. If nothing is selected, it sets the dash style for future objects or text, as applicable.

Syntax: `setstrokestyle <dashname/S> [ALL | FILL | STROKENUMBER number/I] [OBJECT | TEXT | SHADOW | REVERSE | UNDERLINE] [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I | STYLETAG name/S]`

Format:	Parameter	Values to enter
	dashname	is the dash style to set.
	ALL	sets the dash style of the fill and all strokes. (Default)
	FILL	sets the dash style of the fill.
	STROKENUMBER	sets the dash style of a specific stroke number. (See note)
	OBJECT	sets the dash style of the object, not text it may contain. (Default)
	TEXT	sets the dash style of the text, not the object.
	SHADOW	sets the dash style of the shadow of text.
	REVERSE	sets the dash style of the reverse of text.
	UNDERLINE	sets the dash style of the underline of text.
	DOCUMENT	is the document name.
	WINDOW	is the window name.
	OBJECTID	is the number of the object. (Default=current)
	STYLETAG	**

Notes: STROKENUMBER: Strokes are numbered from 0. This is different than the program interface which numbers them from 1.

Example: `setstrokestyle '15'`  
`setstrokestyle 'Dots' strokenumber 1 object`

Command Format

Object ID numbers

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

## 1.14 setstrokeweight

SETSTROKEWEIGHT

Purpose: Sets the stroke weight for an object or text, or for a stroke style tag. If object or text is chosen and nothing is selected, it sets the stroke weight for future objects or text, as applicable.

Syntax: `setstrokeweight <weight/D> [ALL | STROKENUMBER number/I]`  
`[OBJECT | TEXT | SHADOW | REVERSE | UNDERLINE] [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I | STYLETAG name/S]`

Format:	Parameter	Values to enter
	weight	is the the weight of the stroke.
	ALL	sets the weight for all strokes. (Default)
	STROKENUMBER	sets the weight of a specific stroke number. (See note)
	OBJECT	sets the stroke weight of the object, not text it may contain. (Default)
	TEXT	sets the stroke weight of the text, not the object.
	SHADOW	sets the stroke weight of the shadow of text.
	REVERSE	sets the stroke weight of the reverse of text.
	UNDERLINE	sets the stroke weight of the underline of text.
	DOCUMENT	is the document name.
	WINDOW	is the window name.
	OBJECTID	is the number of the object. (Default=current)
	STYLETAG	**

Notes: STROKENUMBER: Strokes are numbered from 0. This is different than the program interface which numbers them from 1.

Example: `setstrokeweight 2`  
`setstrokeweight 1.5 strokenumber 1 object`  
`setstrokeweight 5 text`

Command Format

Object ID numbers

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

## 1.15 setstyletagname

SETSTYLETAGNAME

Purpose: Changes the name of a style tag.

Syntax: `setstyletagname <newstylename/S> [STYLETAG name/S]`

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

Example: `setstyletagname 'Body Text' styletag 'paragraph text'`

Command Format

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

## 1.16 setsubscript

SETSUBSCRIPT

Purpose: Sets the subscript style option at the insertion point, for selected text or for a style tag.

Syntax: `setsubscript <OFF | STANDARD | CUSTOM offset/P scalex/P scaley/P> [DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]`

Format: Parameter Values to enter  
 OFF toggles off the style.  
 STANDARD is the default subscript style.  
 CUSTOM sets custom offset and scale for the subscript.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 STYLETAG sets the subscript style for a style tag.

Notes: The standard subscript style is:

- offset: 40%
- scalex: 60%
- scaley: 60%

Example: `setsubscript off`  
`setsubscript custom 30 40 40 document 'project.doc'`  
`setsubscript standard styletag 'Special Style'`

Command Format

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

## 1.17 setsuperscript

SETSUPERSCRIP

Purpose: Sets the superscript style option at the insertion point, for selected text or for a style tag.

Syntax: `setsuperscript <STANDARD | CUSTOM offset/P scalex/P scaley/P> [DOCUMENT name/S | WINDOW name/S | STYLETAG/S]`



Format: Parameter Values to enter  
 OFF toggles off the style.  
 STANDARD is the default superscript style.  
 CUSTOM sets custom offset and scale for the superscript.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 STYLETAG sets the superscript style for a style tag.

Notes: The standard superscript style is:  
 - offset: 40%  
 - scalex: 60%  
 - scaley: 60%

Example: setsuperscript off  
 setsuperscript custom 30 40 40 document 'project.doc'  
 setsuperscript standard styletag 'Special Style'

Command Format

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

## 1.18 settabruler

SETTABRULER

Purpose: Sets the tabs for the paragraph at the insertion point, for selected paragraphs or for a style tag.

Syntax: settabruler <<LEFT | CENTER | RIGHT | USDECIMAL | EURODECIMAL | CUSTOMDECIMAL unicode/I> position/D> [LEADER unicode/I]> [DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]

Format: Parameter Values to enter  
 LEFT sets a new left aligned tab.  
 CENTER sets a new center aligned tab.  
 RIGHT sets a new right aligned tab.  
 USDECIMAL sets a new period aligned tab.  
 EURODECIMAL sets a new comma aligned tab.  
 CUSTOMDECIMAL sets a new character aligned tab and the character on which to align.  
 position is the position of the new tabs.  
 LEADER sets a character with which to fill the tab. (Default=blank ← )  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 STYLETAG sets the tabs for a style tag.

Notes: To delete or move tabs from a macro, you should get the existing tab positions and then reset them as required.

You can enter a unicode value if the character for the leader or custom decimal is not part of the AmigaDOS character set. Type a backslash followed by the unicode value for a unicode character.

Example: settabruler left 0.5 left 1.25 right 2.25

```

settabruler center 1.5
settabruler usdecimal 6 document 'project.doc'
settabruler customdecimal \8226
settabruler customdecimal '..'

```

Command Format

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

## 1.19 settexteditor

SETTEXTEDITOR

**Purpose:** Set the default text editor macro for the Send to Editor command.

**Syntax:** settexteditor <macro/F> [SAVE]

**Format:** Parameter Values to enter  
macro is the filename and path of the macro to use.  
SAVE saves the setting to PageStream3.prefs.

**Example:** settexteditor 'Macros/SendtoPageLiner.rexx'

Command Format

## 1.20 settextframe

SETTEXTFRAME

**Purpose:** Toggles the ability of a shape to hold text.

**Syntax:** settextframe [ON | OFF | TOGGLE] [COLUMNS number/I]  
[GUTTER space/D] [FORCE | ALERT | QUIET]  
[DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

**Format:** Parameter Values to enter  
ON toggles on the ability of a shape to hold text.  
OFF toggles off the ability to a shape to hold text.  
TOGGLE toggles the ability of a shape to hold text.  
COLUMNS is the number of columns in the frame.  
GUTTER is the space between columns.  
FORCE when OFF is used, ignores text is in frame.  
ALERT when OFF is used, opens an alert requester if text is in the frame.  
QUIET when OFF is used, ignores text in the frame but returns an error if there is text in the frame. (Default)  
DOCUMENT is the document name. (Default=current)  
WINDOW is the window name. (Default=current)  
OBJECTID is the number of the object. (Default=current)

**Note:** Contrary to the manual, text frames created from shapes and paths can contain multiple columns. Use this macro command to change the

number of columns in a text shape frame or any text frame.

Example: `setttextframe on columns 3`  
`setttextframe off quiet`

Command Format

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

## 1.21 setttextpath

SETTEXTPATH

Purpose: To set the default file path for inserting and exporting text.

Syntax: `setttextpath <filepath/F> [SAVE]`

Format: Parameter Values to enter  
 filepath is the path for the files.  
 SAVE saves the filepath to PageStream3.prefs.

Example: `setttextpath 'Work:FinalWriter/FWDocs'`

Command Format

## 1.22 settextrouting

SETTEXTROUTING

Purpose: Links frames together to flow text from one to another.

Syntax: `settextrouting <FROM <OBJECTID number/I | pointx/D pointy/D [FRONTMOST | ←  
 BACKMOST]  
 [PAGE number/S | MPG name/S | DOCUMENT name/S | WINDOW name/S]>>  
 <<TO <OBJECTID number/I | pointx/D pointy/D [FRONTMOST | BACKMOST]  
 [PAGE number/S | MPG name/S | DOCUMENT name/S | WINDOW name/S]>>`

Format: Parameter Values to enter  
 FROM is the position on a page of the text frame to link from.  
 TO is the position on a page of the text frame to link to.  
 OBJECTID is the identification number of the text frame.  
 FRONTMOST specifies the frontmost frame at the point.  
 BACKMOST specifies the backmost frame at the point.  
 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)

Note: You cannot link a text frame to a frame in another document. If the DOCUMENT or WINDOW parameters are used, they must specify the same document in both the FROM and TO parameters.

Example: `settextrouting from 1.25 2.25 page 5 to 1.25 2.25 page 6`

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

## 1.23 settextrwrap

SETTEXTWRAP

Purpose: Sets the text wrap options for the selected object(s).

Syntax: settextrwrap [SHAPE | BOUNDINGBOX | FENCE]  
 [STANDOFF offsetx/D offsety/D]  
 [NOWRAP | WRAPLEFT | WRAPRIGHT | WRAPJUMP | WRAPAROUND |  
 WRAPINSIDE] [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format:	Parameter	Values to enter
	SHAPE	wraps text around the path of the object(s).
	BOUNDINGBOX	wraps text around the bounding box of the object(s).
	FENCE	wraps text around the fence of the object(s).
	STANDOFF	sets the offset of the text from the object(s).
	NOWRAP	flows text over the objects.
	WRAPLEFT	flows text to the left of the objects.
	WRAPRIGHT	flows text to the right of the objects.
	WRAPJUMP	does not flow text on either side of the objects.
	WRAPAROUND	flows text on both sides of the objects.
	WRAPINSIDE	flows text inside the objects, but not outside.
	DOCUMENT	is the document name.
	WINDOW	is the window name.
	OBJECTID	is the number of the object. (Default=current)

Example: settextrwrap wrapjump standoff 0.125 0.125

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

## 1.24 settoolmode

SETTOOLMODE

Purpose: Sets the toolbox mode for a document window.

Syntax: settoolmode <OBJECT | RESHAPE | TEXT | MAGNIFY | COLUMN | LINE |  
 BOX | ELLIPSE | ARC | PEN | FREEHAND | GRID | PREVIOUS>  
 [WINDOW name/S]

Format:	Parameter	Values to enter
	OBJECT	is the Object tool.
	RESHAPE	is the Reshape tool.
	TEXT	is the Text tool.
	MAGNIFY	is the Magnify tool.

COLUMN is the Column tool.  
 LINE is the Line tool.  
 BOX is the Box tool.  
 ELLIPSE is the Ellipse tool.  
 ARC is the Arc tool.  
 PEN is the Pen tool.  
 FREEHAND is the Freehand tool.  
 GRID is the Grid tool. There is no grid tool in the toolbox,  
 but the toolmode may be set to draw grids.  
 PREVIOUS sets the previously set tool.  
 WINDOW is the window name. (Default=current)

Notes: You do not have to set the toolbox mode in order to draw an object with a macro command. This command is only used when you wish to set the interface's toolbox mode.

Example: `settoolmode text window 'Untitled Document.1~View.1'`

Command Format

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

## 1.25 settracking

### SETTRACKING

Purpose: Sets the tracking value at the insertion point, for selected text or for a style tag. Tracking can be applied on top of a tracking table's preset tracking.

Syntax: `settracking <track/D>`  
`[DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]`

Format: Parameter Values to enter  
 track is the tracking value. (-327 to 327)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 STYLETAG sets the tracking for a style tag.

Notes: Tracking units are 1/100th of an em space.

Example: `settracking '-10'`  
`settracking 15 document 'project.doc'`  
`settracking 5 styletag 'Body Text'`

See also

### SETTRACKTABLE

Command Format

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

## 1.26 settrackrange

## SETTRACKRANGE

**Purpose:** Sets the tracking range options at the insertion point, for selected text or for a style tag.

**Syntax:** `settrackrange <minc/P optc/P maxc/P minw/P optw/P maxw/P>`  
`[DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]`

**Format:**

Parameter	Values to enter
<code>minc</code>	is the minimum character spacing.
<code>optc</code>	is the optimum character spacing.
<code>maxc</code>	is the maximum character spacing.
<code>minw</code>	is the minimum word spacing.
<code>optw</code>	is the optimum word spacing.
<code>maxw</code>	is the maximum word spacing.
<code>DOCUMENT</code>	is the document name. (Default=current)
<code>WINDOW</code>	is the window name. (Default=current)
<code>STYLETAG</code>	sets the tracking for a style tag.

**Example:** `settrackrange '-5 0 25 75 100 150'`  
`settrackrange '-5 0 25 75 100 150' styletag 'Body Text'`

See also

## SETTRACKING

Command Format

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

## 1.27 settracktable

## SETTRACKTABLE

**Purpose:** Sets the tracking table at the insertion point, for selected text or for a style tag. A tracking table is a preset formula which tracks tighter at large sizes. Tracking tables are non-editable. Tracking can be applied on top of a tracking table's preset tracking.

**Syntax:** `settracktable <NONE | MONSPACED | VERYLOOSE | LOOSE | NORMAL | TIGHT | VERYTIGHT>` `[DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]`

**Format:**

Parameter	Values to enter
<code>NONE</code>	does not set a tracking table.
<code>VERYLOOSE</code>	is very loose tracking.
<code>LOOSE</code>	is loose tracking.
<code>NORMAL</code>	is normal tracking.
<code>TIGHT</code>	is tight tracking.
<code>VERYTIGHT</code>	is very tight tracking.
<code>DOCUMENT</code>	is the document name. (Default=current)
<code>WINDOW</code>	is the window name. (Default=current)
<code>STYLETAG</code>	sets the tracking for a style tag.

**Example:** `settracktable loose`  
`settracktable normal document 'project.doc'`  
`settracktable tight styletag 'Body Text'`

See also

SETTRACKING

Command Format

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

## 1.28 settypeangle

SETTYPEANGLE

**Purpose:** Sets the type angle at the insertion point, for selected text or for a style tag.

**Syntax:** settypeangle [ROTATE angle/A] [SKEW slantangle/A twistangle/A]  
[DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]

**Format:**

Parameter	Values to enter
ROTATE	is the rotation angle. (Default=current)
SKEW	is the slant and twist angle. (Default=current)
DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name. (Default=current)
STYLETAG	sets the type angle for a style tag.

**Notes:** This sets the italic slant angle for fonts that lack a true italic style, and also allows you to twist and rotate blocks of text. To set only the italic slant angle, set the slantangle to the desired value and set the twistangle to 0.

**Example:** settypeangle rotate 25  
settypeangle skew 15 30  
settypeangle rotate 14 styletag 'Rotated Text'

Command Format

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

## 1.29 settypesize

SETTYPESIZE

**Purpose:** Sets the type size at the insertion point, for selected text or for a style tag.

**Syntax:** settypesize <size/D> [WIDTH scale/P]  
[DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]

**Format:**

Parameter	Values to enter
size	is the type size. (1 to 50,000pt, 0.001pt increments)
WIDTH	is the horizontal scale. (1 to 655%, 0.01% increments) (Default=current)
DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name. (Default=current)
STYLETAG	sets the type size for a style tag.

---

```
Example: settypesize 15
         settypesize '0.5"'
         settypesize 12 width 80
         settypesize 14 styletag 'Body Text'
```

Command Format

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

## 1.30 settypestyle

SETTYPESTYLE

Purpose: Sets the type style at the insertion point, for selected text or for a style tag.

```
Syntax: settypestyle <NORMAL | [[BOLD] [ITALIC] [REVERSE] [OUTLINE]
         [SHADOW] [UNDERLINE] [SMALLCAPS] [SUPERSCRIP] [SUBSCRIPT]]>
         [DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]
```

Format:	Parameter	Values to enter
	NORMAL	turns off all type styles and sets the Normal style.
	BOLD	sets the Bold type style.
	ITALIC	sets the Italic type style.
	REVERSE	sets the Reverse type style.
	OUTLINE	sets the Outline type style.
	SHADOW	sets the Shadow type style.
	UNDERLINE	sets the Underline type style.
	SMALLCAPS	sets the Small Caps type style.
	SUPERSCRIP	sets the Superscript type style.
	SUBSCRIPT	sets the Subscript type style.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)
	STYLETAG	sets the type style for a style tag.

```
Example: settypestyle normal
         settypestyle bold italic shadow
         settypestyle bold styletag 'Headline'
```

Command Format

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

## 1.31 setunderline

SETUNDERLINE

Purpose: Sets the underline style option at the insertion point, for selected text or for a style tag.

```
Syntax: setunderline <OFF | SINGLE | DOUBLE | STRIKETHROUGH |
         CUSTOM <CONTINUOUS | WORD | CHARACTER> offset/P>
         [DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]
```



Format: Parameter Values to enter

OFF toggles off the style.

SINGLE is standard underlining.

DOUBLE is double underlining.

STRIKETHROUGH is standard strikethrough.

CUSTOM sets custom format, offset, stroke and color attributes for the underline.

DOCUMENT is the document name. (Default=current)

WINDOW is the window name. (Default=current)

STYLETAG sets the underline style for a style tag.

Notes: The standard single underline style is:

- format: continuous
- offset: 10%
- line: 0.5pt
- color: text color

The standard double underline style is:

- format: continuous
- offset: 10%
- stroke offset: -0.5 pt & 0.5 pt
- line: 0.5pt
- color: text color

The standard strikethrough style is:

- format: continuous
- offset: -40%
- line: 0.5pt
- color: text color

Example: setunderline off  
 setunderline custom word 20 document 'project.doc'  
 setunderline double styletag 'Table Text 2'

Command Format

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

## 1.32 setundolevels

SETUNDOLEVELS

Purpose: Changes the number of undo levels available.

Syntax: setundolevels <levels/I> [SAVE]

Format: Parameter Values to enter

levels is the number of undo levels to permit.

SAVE saves the value to the PageStream3.prefs file.

Notes: The number of undo levels is limited only by memory. For unlimited levels of undo, set levels to -1. It is recommended that you restrict the number of levels of undo to conserve memory.

Example: setundolevels 16 save

Command Format

### 1.33 setvariablename

SETVARIABLENAME

Purpose: Sets the name of a user variable.

Syntax: setvariablename <varname/S VARIABLE name/S>

Format: Parameter Values to enter  
 varname is the new name of the variable.  
 VARIABLE is the document/chapter/variable name. (Default=current)

Example: setvariablename author variable writer  
 setvariablename author variable MyBook.doc~writer

Command Format

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

### 1.34 setvariablevalue

SETVARIABLEVALUE

Purpose: Sets the string of a user variable.

Syntax: setvariablevalue <varvalue/S VARIABLE name/S>

Format: Parameter Values to enter  
 varvalue is the new string for the variable.  
 VARIABLE is the document/chapter/variable name. (Default=current)

Example: 'setvariablevalue "Bill Watson" '  
 'setvariablevalue "Bill Watson" variable MyBook.doc~author'

Command Format

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

### 1.35 setviewdefaults

SETVIEWDEFAULTS

Purpose: Changes the default view options.

Syntax: setviewdefaults [SCALE <custom/P | FULLPAGE | FULLBLEED |  
 FULLPAGEWIDTH | FULLBLEEDWIDTH | FULLPAGEHEIGHT | FULLBLEEDHEIGHT>]  
 [SCALERANGE minscale/P maxscale/P] [FACINGPAGES <ON|OFF|TOGGLE>]  
 [GRID <SHOW|HIDE|TOGGLE>] [SNAPTOGRID <ON|OFF|TOGGLE>]  
 [GUIDES <SHOW|HIDE|TOGGLE>] [SNAPTOGUIDES <ON|OFF|TOGGLE>]  
 [DRAWINGS <SHOW|HIDE|TOGGLE>] [OUTLINE <SHOW|HIDE|TOGGLE>]

```
[TEXTLINKS <SHOW|HIDE|TOGGLE>] [INVISIBLES <SHOW|HIDE|TOGGLE>]
[RULER <SHOW|HIDE|TOGGLE>] [PICTURES <SHOW|HIDE|TOGGLE>] [SAVE]
```

Format: Parameter Values to enter

SCALE	is the a predefined or custom view scale.
SCALERANGE	is the minimum and maximum view scale.
GRID	is the show grid toggle.
SNAPTOGRID	is the snap-to-grid toggle.
GUIDES	is the show guides toggle.
SNAPTOGUIDES	is the snap-to-guides toggle.
DRAWINGS	is the show drawings toggle.
OUTLINE	is the show column outlines toggle.
TEXTLINKS	is the show text links toggle.
INVISIBLES	is the show invisibles toggle.
RULER	is the show rulers toggle.
PICTURES	is the show pictures toggle.
SAVE	saves the values to the PageStream3.prefs file.

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

Example: setviewdefaults scale 50 grid hide pictures show save

Command Format

## 1.36 setwidoworphan

SETWIDOWORPHAN

Purpose: Sets the widow and orphan control options for the paragraph at the insertion point, for selected paragraphs or for a style tag.

Syntax: setwidoworphan <OFF | KEEPPARAGRAPH | KEEPLINES start/I end/I>  
[DOCUMENT name/S | WINDOW name/S | STYLETAG name/S]

Format: Parameter Values to enter

OFF	sets no widor/orphan control.
KEEPPARAGRAPH	keeps the whole paragraph together.
KEEPLINES	keeps a number of lines together at the start and/or end of a paragraph.
DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name. (Default=current)
STYLETAG	sets widow and orphan options for a style tag.

Example: setwidoworphan keeplines 3 3  
setwidoworphan keepparagraph document 'project.doc'  
setwidoworphan keeplines 2 5 styletag 'Body Text'

Command Format

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

## 1.37 setwindowname

**SETWINDOWNAME**

Purpose: Changes the view name of a window.

Syntax: `setwindowname <windowname/S> [WINDOW name/S]`

Format: Parameter Values to enter  
 windowname is the new view name of a window.  
 WINDOW is the old window name. (Default=current)

Notes: windowname is just the new view name and cannot include the document name. The document name can be changed separately with the SETDOCUMENTNAME command.

Example: `setwindowname 'My View' window 'Untitled Document.1~View.1'`

Command Format

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

**1.38 setwindowdefaults****SETWINDOWDEFAULTS**

Purpose: Sets the size and position of the default document window.

Syntax: `setwindowdefaults [AT pointx/I pointy/I] [SIZE width/I height/I] [SAVE]`

Format: Parameter Values to enter  
 AT sets the window position.  
 SIZE sets the window size in pixels.  
 SAVE saves the window size and position to PageStream3.prefs.

Notes: If the AT pointy coordinate is set to -1, the window will open immediately below the screen title bar, regardless of the screen font height. If either SIZE coordinate is set to 10000, the window will open at the maximum size of the screen in that direction.

This command does not affect open document windows. It sets the position and size of future document windows. Use the

**SETWINDOWPOS**

command to set the position and size of open document windows.

Example: `setwindowdefaults at 0 -1 size 640 10000`  
`setwindowdefaults at 0 -1 save`

Command Format

**1.39 setwindowpos**

## SETWINDOWPOS

Purpose: Sets the size and position of an open document window.

Syntax: setwindowpos [AT pointx/I pointy/I] [SIZE width/I height/I]  
[WINDOW name/S]

Format: Parameter Values to enter  
AT sets the window position.  
SIZE sets the window size in pixels.  
WINDOW is the window name. (Default=current)

Notes: If the AT pointy coordinate is set to -1, the window will open immediately below the screen title bar, regardless of the screen font height. If either SIZE coordinate is set to 10000, the window will open at the maximum size of the screen in that direction.

This command does not affect future document windows. It sets the position and size of an open document window only. Use the

## SETWINDOWDEFAULTS

command to set the position and size of future document windows.

Example: setwindowpos at 0 -1 size 640 10000

Command Format

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

## 1.40 simplifypath

## SIMPLIFYPATH

Purpose: Simplifies the path by replacing lines and curves with a minimum number of lines and curves. There is no loss of accuracy. Use this command to simplify an overly complex path.

Syntax: simplifypath [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: simplifypath  
simplifypath window 'Untitled Document.1~View.1'

See also:

## SMOOTHPATH

Command Format

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

## 1.41 skew

SKEW

Purpose: Skew selected objects horizontally (slant) and/or vertically (twist).

Syntax: skew <slantangle/A twistangle/A>  
 [ABOUT pointx/D pointy/D | ABOUTCENTER]  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
 slantangle is the slant angle.  
 twistangle is the twist angle.  
 ABOUT is the rotation point.  
 ABOUTCENTER rotates around its center. (Default)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to manipulate.

Example: skew 90 131 about 2.2 3.125 document 'project.doc'

Command Format

Object ID numbers

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

## 1.42 smoothpath

SMOOTHPATH

Purpose: Smooths the path by replacing lines and curves with a minimum number of curves. There is a loss of accuracy because rough spots will be rounded out. Use this command to smooth a freehand path.

Syntax: smoothpath [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: smoothpath  
 smoothpath window 'Untitled Document.1~View.1'

See also:

SIMPLIFYPATH

Command Format

Object ID numbers

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

## 1.43 snaptogrid

SNAPTOGRID

Purpose: Changes the snap-to-grid status.

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

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

Notes: Grid spacing and strength are part of the master page description.  
The snap-to-grid toggle is part of the window settings.

Snap-to-grid does not affect drawing with macro commands; it affects drawing and editing from the interface only.

Example: `snaptogrid on`  
`snaptogrid off window 'Untitled Document.1~View.1'`

See also SETGRID.

Command Format

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

## 1.44 snaptoguides

SNAPTOGUIDES

Purpose: Changes the snap-to-guide status.

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

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

Notes: Guide position and strength are part of the master page description. The snap-to-guide toggle is part of the window settings.

Snap-to-guides does not affect drawing with macro commands; it affects drawing and editing from the interface only.

Example: `snaptoguides on`  
`snaptoguides off window 'Untitled Document.1~View.1'`

See also SETGUIDES, ADDGUIDES and DELETEGUIDES,  
SNAPTOGUIDESRID

---

Command Format

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

## 1.45 splitpaths

SPLITPATHS

Purpose: Splits the paths of selected path(s) into their subpath(s).

Syntax: splitpaths [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: splitpaths

splitpaths document 'project.doc'

Command Format

Object ID numbers

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

## 1.46 stroked

STROKED

Purpose: Toggles the stroke for an object or text, or for a style tag. If object or text is chosen and nothing is selected, it toggles the stroke for future objects or text, as applicable.

Syntax: stroked <ON | OFF | TOGGLE>

[DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter

ON toggles on the stroke.

OFF toggles off the stroke.

TOGGLE toggles the stroke.

DOCUMENT is the document name.

WINDOW is the window name.

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

Example: stroked on

stroked toggle document 'project.doc'

See also

SETSTROKE

Command Format

Object ID numbers

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



## 1.47 stylepalette

STYLEPALETTE

Purpose: Displays the style tag palette.

Syntax: stylepalette [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: stylepalette toggle  
stylepalette show at 500 300 size 180 172 save  
stylepalette hide

Command Format

## 1.48 textcursor

TEXTCURSOR

Purpose: Moves the text insertion point.

Syntax: textcursor <LEFT | RIGHT | UP | DOWN | WORDLEFT | WORDRIGHT |  
BOL | EOL | BOP | EOP | BOC | EOC | BOF | EOF | BOA | EOA>  
[DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	LEFT	moves 1 character left.
	RIGHT	moves 1 character right.
	UP	moves 1 line up.
	DOWN	moves 1 line down.
	WORDLEFT	moves 1 word left.
	WORDRIGHT	moves 1 word right.
	BOL	moves to the beginning of the line.
	EOL	moves to the end of the line.
	BOP	moves to the beginning of the paragraph.
	EOP	moves to the end of the paragraph.
	BOC	moves to the beginning of the column.
	EOC	moves to the end of the column.
	BOF	moves to the top of the frame's first column.
	EOF	moves to the bottom of the frame's last column.
	BOA	moves to the top of the article's first column.
	EOA	moves to the bottom of the article's last column.
	DOCUMENT	is the document name. (Default=current)

WINDOW is the window name. (Default=current)

```
Example: textcursor left
textcursor wordright
textcursor boa document 'Project.Doc'
textcursor eop
```

Command Format

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

## 1.49 transform

TRANSFORM

Purpose: Copies, scales, offsets, rotates, slants and twists selected object(s).

```
Syntax: transform <count/I> [OFFSET offsetx/D offsety/D]
[SCALE scalex/P scaley/P] [ROTATE angle/A |
SKEW slantangle/A twistangle/A | SLANT angle/A | TWIST angle/A]
[ABOUT pointx/D pointy/D | ABOUTCENTER]
[DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]
```

Format:	Parameter	Values to enter
	count	is the number of copies to make of the original object.
	OFFSET	is the offset from the original or last copy. (Default=0)
	ROTATE	is the rotation angle.
	SKEW	is the slant and twist angle.
	SLANT	is the slant angle.
	TWIST	is the twist angle.
	ABOUT	is the rotation point.
	ABOUTCENTER	rotates around its center.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)
	OBJECTID	is the number of the object to manipulate.

Note: If count=0, the the original object will be transformed. If count>0, then each successive object will be transformed by these values.

```
Example: transform 5 offset 0.25 0 scale 100 83
transform 3 skew 90 131 about 2.2 3.125 document 'project.doc'
```

Command Format

Object ID numbers

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

## 1.50 undo

UNDO

---

Purpose: Undoes the last command or action.

Syntax: undo [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: undo document 'project.doc'

See also REDO.

Command Format

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

## 1.51 ungroup

UNGROUP

Purpose: Ungroups selected logical groups into their component objects and/or groups.

Syntax: ungroup [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: ungroup  
ungroup document 'project.doc'

See also GROUP.

Command Format

Object ID numbers

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

## 1.52 unlock

UNLOCK

Purpose: Unlocks selected locked objects so that they can be edited.

Syntax: unlock [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: unlock

---

```
unlock document 'project.doc'
```

See also LOCK

Command Format

Object ID numbers

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

## 1.53 updatefontpath

UPDATEFONTPATH

Purpose: This updates an existing font directory path.

Syntax: updatefontpath <filepath/F>

Format: Parameter Values to enter

filepath is the filepath of the font path to update.

Example: updatefontpath 'Fonts:\_Bullet\_Outlines'  
updatefontpath 'work:psfonts/AdobeFonts'

Command Format

---