

TextFace Select a style for subsequent text drawing

#include <Quickdraw.h>

Quickdraw

```
void      TextFace(newStyle );
Style     newStyle ;      0=plain text, 1=bold, 4=underline, etc
```

TextFace selects the text-style variation(s) (bold, italic, underline, etc.) for the current GrafPort.

newStyle is an integer value (declared as an enum with a typedef of Style). A value 0 indicates a "plain" unmodified version of the current font. You can use bit-manipulation operations to combine any of the following styles using the constants defined in Quickdraw.h:

<u>bold</u>	1	increased width on vertical strokes
<u>italic</u>	2	<i>slanted toward the right</i>
<u>underline</u>	4	<u>underscored, with breaks on descending letters</u>
<u>outline</u>	8	<u>outlined</u>
<u>shadow</u>	16	Shadowed (outlined, heavier on right bottom)
<u>condense</u>	32	less space between characters
<u>extend</u>	64	more space between characters

Returns: none

Notes: **TextFace** modifies the txFace field of the current GrafPort. It is initially set to 0 (plain text). The variation you select affects all subsequent text drawing and text measuring.

You can read the current style setting by accessing the txFace field of the current GrafPort:

```
curStyle = thePort -> txFace;
```

Here are some examples of usage:

```
TextFace( bold );           /* set to bold */
TextFace( bold | italic );   /* set to bold and italic */
TextFace( thePort->txFace | bold ); /* add bolding */
TextFace( thePort->txFace & ~bold ); /* remove bolding */
TextFace( 0 );              /* set to plain text */
```

The "condense" and "extend" variations change the spacing between characters to an arbitrary value set by the Font Manager. Another way to compress expand text is to call **SetFScaleDisable** and use a smaller or larger font. You may use **SpaceExtra** for spacing control; e.g., as an aid in displaying right-justified text.