

Q: How do I programmatically create a window 80 characters wide for any given font?

A: The following code snippet works for a fixed width font such as Ohlfs or Courier. A non-fixed width font can only be approximated by choosing either an "average-width" character, or the widest character in the font.

```
#import <math.h>
#import <appkit/Window.h>
#import <appkit/Font.h>

float          maxcharwidth, screenwidth;
id             screenFont, myWindow;
NXCoordleftMargin, rightMargin, topMargin, bottomMargin;
NXRect         aRect;

[myText getMarginLeft: &leftMargin
               right: &rightMargin
               top: &topMargin
               bottom: &bottomMargin];
screenFont = [myFont screenFont];
```

```

maxcharwidth = MAX([myFont getWidthOf: "0"],
                    [screenFont getWidthOf: "0"]);
screenwidth = ceil(maxcharwidth * 80.0) + leftMargin + rightMargin + 1;

/* Now use that width when creating the window */
NXSetRect(&aRect, 0.0, 0.0, screenwidth, height);
#ifdef 1.0
    myWindow = [Window newContent: &aRect
                      style: NX_TITLEDSTYLE
                      backing: NX_BUFFERED
                      buttonMask: NX_MINIATURIZEBUTTONMASK
                      defer: NO];
#else /* 2.0 or 3.0 */
    myWindow = [[Window alloc] initContent: &aRect
                      style: NX_TITLEDSTYLE
                      backing: NX_BUFFERED
                      buttonMask: NX_MINIATURIZEBUTTONMASK
                      defer: NO];
#endif

```

**Note that the ifdef for 1.0 is for illustration purposes only and is NOT defined in any NEXTSTEP**

**include file.**

The key is to get the width of any character (since all have the same width) of both the screen font and the printer font. Take the maximum width and do a ceil of the calculation, since the printer width may be fractional.

Note that a fudge factor of +1 is added to the calculation. This is a known bug.

Note also that this window does not contain a vertical scrollbar. If it did, the width of the scrollbar would need to be accounted for.

QA613

Valid for 1.0

Valid for 2.0, 3.0 (with the differences noted)