

Test App -- Main

File Edit View Tools Options Help

Load Combo Menu 1 ▾ Menu 2 ▾

1 Example \*\*\* Cstring test \*\*\* --- CString test --- +++ String test +++ ^^^ BString test ^^^  
2 \*\*\* Cstring test \*\*\*  
3 --- CString test ---  
4 +++ String test +++  
5 ^^^ BString test ^^^  
6 Set Font Face to courier New  
7 anInt = <1234567>  
8 Width = 9 < 1234567>  
9 Width = -9 <1234567 >  
10 anSize\_t = <4294967295>  
11 aLong = <-1234567890>  
12 aULong = <4294967294>  
13 aDouble = <12345678.901234>  
14 Prec = 3 <12345678.901>  
15 Width = 14 < 12345678.901>  
16 Width = -14 <12345678.901 >  
17 aDate = <8/1/24 10:30>  
18 Pop Font This should be Arial Font  
19 Font Size = 8 This is a 8pt font.  
20 Pop Font This should be a 12pt font  
21 Bolding Not bold **Some Bold 1234567** Not bold again  
22 Two Pops Bold *Italic Now Pop* Bold again Pop again Not bold italic  
23 nUnderline An Integer 1234567 after underlined segment.  
24 Other Underline Should be underlined 1234567 not underlined  
25 Two Underlines One Underline Two Underlines One Underline No underlines  
26 Center/Right  
At left margin Page1234567 At right margin  
27 Strike Out Hello ~~Sherman Oaks~~ Popped  
28 ? ...

Ready

CAP NUM SCRL

```
// NotePad Examples
```

```
#include "pch.h"
#include "NotePadExamples.h"
#include "NotePad.h"

void NotePadExamples::operator() () {
    Cstring    mumble   = _T(" *** Cstring test *** ");
    CString    foo      = _T(" --- CString test --- ");
    String     goo      = _T(" +++ String test +++ ");
    bstr_t     bs       = _T(" ^^^ BString test ^^^ ");
    int        anInt   = 1234567;
    size_t     anSize_t = 4294967295;
    long       aLong    = -1234567890;
    ulong      aULong   = 4294967294;
    double     aDouble  = 12345678.901234;
    Date       aDate(_T("8/1/24 10:30"));

    notePad << nSetRTab(3) << nSetTab(5) << nSetTab(24); // Set a Right Tab for the line number
                                                               // and a regular tab for the; start of
                                                               // output

    lineNo( 1 );    notePad << _T("Example ") << mumble << foo << goo << bs << nCrlf;
    lineNo( 2 );    notePad << mumble;                                crlf();
    lineNo( 3 );    notePad << foo;                                 crlf();
    lineNo( 4 );    notePad << goo;                                 crlf();
    lineNo( 5 );    notePad << bs;                                 crlf();
    lineNo( 6 );    notePad << _T("Set Font Face to courier New") << nFFace(_T("Courier New")); crlf();

    lineNo( 7, _T("anInt ="));    notePad << _T('<') << anInt << _T(">");           crlf();
    lineNo( 8, _T("Width = 9"));  notePad << _T('<') << nSetWidth(9) << anInt << _T(">");  crlf();
    lineNo( 9, _T("Width = -9")); notePad << _T('<') << nSetWidth(-9) << anInt << _T(">"); crlf();
    lineNo(10, _T("anSize_t =")); notePad << _T('<') << anSize_t << _T(">");           crlf();
    lineNo(11, _T("aLong ="));   notePad << _T('<') << aLong << _T(">");           crlf();
    lineNo(12, _T("aULong ="));  notePad << _T('<') << aULong << _T(">");           crlf();
    lineNo(13, _T("aDouble =")); notePad << _T('<') << aDouble << _T(">");           crlf();
    lineNo(14, _T("Prec = 3"));  notePad << _T('<') << nSetPrec(3) << aDouble << _T(">"); crlf();
    lineNo(15, _T("Width = 14")); notePad << _T('<') << nSetWidth(14) << nSetPrec(3) << aDouble;
                                                notePad << _T(">");                   crlf();
    lineNo(16, _T("Width = -14")); notePad << _T('<') << nSetWidth(-14) << nSetPrec(3) << aDouble;
                                                notePad << _T(">");                   crlf();
    lineNo(17, _T("aDate ="));   notePad << _T('<') << aDate << _T(">");           crlf();
    lineNo(18, _T("Pop Font"));  notePad << nFont << _T("This should be Arial Font");      crlf();
    lineNo(19, _T("Font Size = 8")); notePad << nFontSize(8) << _T("This is a 8pt font.");      crlf();
    lineNo(20, _T("Pop Font"));  notePad << nFont << _T("This should be a 12pt font");      crlf();
    lineNo(21, _T("Bolding"));   notePad << _T("Not bold");
                                notePad << nBold << _T(" Some") << _T(" Bold ") << anInt << nFont;
                                notePad << _T(" Not bold again");                      crlf();
    lineNo(22, _T("Two Pops"));  notePad << _T("Bold ") << nBold << _T("Italic ") << nItalic;
                                notePad << _T("Now Pop ") << nFont << _T("Bold again ");
                                notePad << _T("Pop again ") << nFont << _T("Not bold italic");
                                crlf();
    lineNo(23, _T("nUnderline")); notePad << nUnderLine << _T("An Integer ") << anInt << nFont;
                                notePad << _T(" after underlined segment."); crlf();
    lineNo(24, _T("Other Underline")); notePad << nBeginLine << _T("Should be underlined ");
                                notePad << anInt << nEndLine << _T(" not underlined");      crlf();
    lineNo(25, _T("Two Underlines"));
                                notePad << nUnderLine << _T("One Underline ");
                                notePad << nBeginLine << _T("Two Underlines ");
                                notePad << nFont      << _T("One Underline ");
                                notePad << nEndLine   << _T("No underlines");                  crlf();
    lineNo(26, _T("Center/Right")); crlf();
                                notePad << _T("At left margin") << nCenter << _T("Page") << anInt;
                                notePad << nRight << _T("At") << _T(" right margin");      crlf();
    lineNo(27, _T("Strike Out")); notePad << _T("Hello ") << nStrikeOut << _T("Sherman");
                                notePad << _T(" Oaks") << nFont << _T(" Popped");      crlf();
    lineNo(28, _T("?"));         notePad << _T("...");                      crlf();
}
```

```
void NotePadExamples::lineNo(int n, TCchar* prefix) {
    notePad << nTab << n << nTab;    if (prefix) notePad << prefix << nTab;
}

void NotePadExamples::crlf()          {notePad << nCrlf;}
```