NAME

    addwstr, **addnwstr**, **waddwstr**, **waddnwstr**, **mvaddwstr**, **mvaddnwstr**, **mvwaddwstr**, **mvwaddnwstr** - add
    a string of wide characters to a **curses** window and advance cursor

SYNOPSIS

    #include <curses.h>

    int addwstr(const wchar_t **wstr***);**
    int addnwstr(const wchar_t **wstr***, int** *n***);**
    int waddwstr(WINDOW **win***, const wchar_t ****wstr***);**
    int waddnwstr(WINDOW **win***, const wchar_t ****wstr***, int** *n***);**

    int mvaddwstr(int *y***, int** *x***, const wchar_t ****wstr***);**
    int mvaddnwstr(int *y***, int** *x***, const wchar_t ****wstr***, int** *n***);**
    int mvwaddwstr(WINDOW **win***, int** *y***, int** *x***, const wchar_t ****wstr***);**
    int mvwaddnwstr(WINDOW **win***, int** *y***, int** *x***, const wchar_t ****wstr***, int** *n***);**

DESCRIPTION

    These functions write the characters of the (null-terminated) **wchar_t** character string *wstr* on the given
    window. It is similar to constructing a **cchar_t** for each wchar_t in the string, then calling **wadd_wch**
    for the resulting **cchar_t**.

    The *mv* functions perform cursor movement once, before writing any characters. Thereafter, the cursor
    is advanced as a side-effect of writing to the window.

    The four functions with *n* as the last argument write at most *n* **wchar_t** characters, or until a terminating
    null is reached. If *n* is -1, then the entire string will be added.

RETURN VALUE

    All functions return the integer **ERR** upon failure and **OK** on success.

    X/Open does not define any error conditions. This implementation returns an error

    ⊕   if the window pointer is null or

    ⊕   if the string pointer is null or

    ⊕   if the corresponding calls to **wadd_wch** return an error.

    Functions with a "mv" prefix first perform a cursor movement using **wmove**, and return an error if the

position is outside the window, or if the window pointer is null.

NOTES

All of these functions except **waddnwstr** may be macros.

PORTABILITY

These functions are described in the XSI Curses standard, Issue 4.

SEE ALSO

curses(3X), **curs_add_wch**(3X)