NAME

addseverity – build a list of severity levels for an application for use with fmtmsg

SYNOPSIS

#include <fmtmsg.h>

int addseverity(int *severity*, **const char** **string***);**

MT-LEVEL

Safe

DESCRIPTION

The **addseverity( )** function builds a list of severity levels for an application to be used with the message formatting facility, **fmtmsg( )**. *severity* is an integer value indicating the seriousness of the condition, and *string* is a pointer to a string describing the condition (string is not limited to a specific size).

If **addseverity( )** is called with an integer value that has not been previously defined, the function adds that new severity value and print string to the existing set of standard severity levels.

If **addseverity( )** is called with an integer value that has been previously defined, the function redefines that value with the new print string. Previously defined severity levels may be removed by supplying the **NULL** string. If **addseverity( )** is called with a negative number or an integer value of 0, 1, 2, 3, or 4, the function fails and returns –1. The values 0–4 are reserved for the standard severity levels and cannot be modified. Identifiers for the standard levels of severity are:

MM_HALT Indicates that the application has encountered a severe fault and is halting. Produces the print string **HALT**.

MM_ERROR Indicates that the application has detected a fault. Produces the print string **ERROR**.

MM_WARNING Indicates a condition that is out of the ordinary, that might be a problem, and should be watched. Produces the print string **WARNING**.

MM_INFO Provides information about a condition that is not in error. Produces the print string **INFO**.

MM_NOSEV Indicates that no severity level is supplied for the message.

Severity levels may also be defined at run time using the **SEV_LEVEL** environment variable (see **fmtmsg**(3C)).

EXAMPLES

When the function **addseverity( )** is used as follows:

addseverity(7,"ALERT")

the following call to **fmtmsg( )**:

fmtmsg(MM_PRINT, "UX:cat", 7, "invalid syntax", "refer to manual", "UX:cat:001")

produces:

UX:cat: ALERT: invalid syntax
TO FIX: refer to manual UX:cat:001

RETURN VALUES

addseverity( ) returns **MM_OK** on success or **MM_NOTOK** on failure.

SEE ALSO

fmtmsg(1), **fmtmsg**(3C), **gettxt**(3C), **printf**(3S)