

NAME

uselocale – set/get the locale for the calling thread

LIBRARY

Standard C library (*libc*, *-lc*)

SYNOPSIS

```
#include <locale.h>
```

```
locale_t uselocale(locale_t newloc);
```

Feature Test Macro Requirements for glibc (see **feature_test_macros(7)**):

uselocale():

Since glibc 2.10:

 _XOPEN_SOURCE >= 700

Before glibc 2.10:

 _GNU_SOURCE

DESCRIPTION

The **uselocale()** function sets the current locale for the calling thread, and returns the thread's previously current locale. After a successful call to **uselocale()**, any calls by this thread to functions that depend on the locale will operate as though the locale has been set to *newloc*.

The *newloc* argument can have one of the following values:

A handle returned by a call to **newlocale(3)** or **duplocale(3)**

The calling thread's current locale is set to the specified locale.

The special locale object handle **LC_GLOBAL_LOCALE**

The calling thread's current locale is set to the global locale determined by **setlocale(3)**.

(*locale_t*) 0

The calling thread's current locale is left unchanged (and the current locale is returned as the function result).

RETURN VALUE

On success, **uselocale()** returns the locale handle that was set by the previous call to **uselocale()** in this thread, or **LC_GLOBAL_LOCALE** if there was no such previous call. On error, it returns (*locale_t*) 0, and sets *errno* to indicate the error.

ERRORS

EINVAL

newloc does not refer to a valid locale object.

VERSIONS

The **uselocale()** function first appeared in glibc 2.3.

STANDARDS

POSIX.1-2008.

NOTES

Unlike **setlocale(3)**, **uselocale()** does not allow selective replacement of individual locale categories. To employ a locale that differs in only a few categories from the current locale, use calls to **duplocale(3)** and **newlocale(3)** to obtain a locale object equivalent to the current locale and modify the desired categories in that object.

EXAMPLES

See **newlocale(3)** and **duplocale(3)**.

SEE ALSO

locale(1), **duplocale(3)**, **freelocale(3)**, **newlocale(3)**, **setlocale(3)**, **locale(5)**, **locale(7)**