



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'duplocale.3p' command

\$ man duplocale.3p

DUPLOCALE(3P) POSIX Programmer's Manual DUPLOCALE(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

duplocale ? duplicate a locale object

SYNOPSIS

```
#include <locale.h>

locale_t duplocale(locale_t locobj);
```

DESCRIPTION

The duplocale() function shall create a duplicate copy of the locale object referenced by the locobj argument.

If the locobj argument is LC_GLOBAL_LOCALE, duplocale() shall create a new locale object containing a copy of the global locale determined by the setlocale() function.

The behavior is undefined if the locobj argument is not a valid locale object handle.

RETURN VALUE

Upon successful completion, the duplocale() function shall return a handle for a new locale object. Otherwise, duplocale() shall return (locale_t)0 and set errno to indicate the error.

ERRORS

The `duplocale()` function shall fail if:

ENOMEM There is not enough memory available to create the locale object or load the locale data.

The following sections are informative.

EXAMPLES

Constructing an Altered Version of an Existing Locale Object

The following example shows a code fragment to create a slightly altered version of an existing locale object. The function takes a locale object and a locale name and it replaces the `LC_TIME` category data in the locale object with that from the named locale.

```
#include <locale.h>

...

locale_t
with_changed_lc_time (locale_t obj, const char *name)
{
    locale_t retval = duplocale (obj);
    if (retval != (locale_t) 0)
    {
        locale_t changed = newlocale (LC_TIME_MASK, name, retval);
        if (changed == (locale_t) 0)
            /* An error occurred. Free all allocated resources. */
            freelocale (retval);
        retval = changed;
    }
    return retval;
}
```

APPLICATION USAGE

The use of the `duplocale()` function is recommended for situations where a locale object is being used in multiple places, and it is possible that the lifetime of the locale object might end before all uses are finished. Another reason to duplicate a locale object is if a slightly modified form is needed. This can be achieved by a call to `newlocale()`

following the `duplocale()` call.

As with the `newlocale()` function, handles for locale objects created by the `duplocale()` function should be released by a corresponding call to `freelocale()`.

The `duplocale()` function can also be used in conjunction with `uselocale((locale_t)0)`. This returns the locale in effect for the calling thread, but can have the value `LC_GLOBAL_LOCALE`. Passing `LC_GLOBAL_LOCALE` to functions such as `isalnum_l()` results in undefined behavior, but applications can convert it into a usable locale object by using `duplocale()`.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

`freelocale()`, `newlocale()`, `uselocale()`

The Base Definitions volume of POSIX.1-2017, `<locale.h>`

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html>.

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html.