NAME

getdtablesize − get file descriptor table size

SYNOPSIS

#include <unistd.h>

int getdtablesize(void);

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

getdtablesize():
Since glibc 2.12:
/* Glibc since 2.19: */ _DEFAULT_SOURCE
|| /* Glibc versions <= 2.19: */ _BSD_SOURCE
|| ! (_POSIX_C_SOURCE >= 200112L)
Before glibc 2.12:
_BSD_SOURCE || _XOPEN_SOURCE >= 500

DESCRIPTION

getdtablesize() returns the maximum number of files a process can have open, one more than the largest possible value for a file descriptor.

RETURN VALUE

The current limit on the number of open files per process.

ERRORS

On Linux, **getdtablesize**() can return any of the errors described for **getrlimit**(2); see NOTES below.

ATTRIBUTES

For an explanation of the terms used in this section, see **attributes**(7).

Interface	Attribute	Value
getdtablesize()	Thread safety	MT-Safe

CONFORMING TO

SVr4, 4.4BSD (the **getdtablesize**() function first appeared in 4.2BSD). It is not specified in POSIX.1; portable applications should employ *sysconf(_SC_OPEN_MAX)* instead of this call.

NOTES

getdtablesize() is implemented as a libc library function. The glibc version calls **getrlimit**(2) and returns the current **RLIMIT_NOFILE** limit, or **OPEN_MAX** when that fails.

SEE ALSO

close(2), **dup**(2), **getrlimit**(2), **open**(2)

COLOPHON

This page is part of release 5.05 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at https://www.kernel.org/doc/man−pages/.