

NAME

`fclose` – close a stream

SYNOPSIS

```
#include <stdio.h>
```

```
int fclose(FILE *stream);
```

DESCRIPTION

The `fclose()` function flushes the stream pointed to by *stream* (writing any buffered output data using `fflush(3)`) and closes the underlying file descriptor.

The behaviour of `fclose()` is undefined if the *stream* parameter is an illegal pointer, or is a descriptor already passed to a previous invocation of `fclose()`.

RETURN VALUE

Upon successful completion, 0 is returned. Otherwise, **EOF** is returned and *errno* is set to indicate the error. In either case, any further access (including another call to `fclose()`) to the stream results in undefined behavior.

ERRORS**EBADF**

The file descriptor underlying *stream* is not valid.

The `fclose()` function may also fail and set *errno* for any of the errors specified for the routines `close(2)`, `write(2)`, or `fflush(3)`.

ATTRIBUTES

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

Interface	Attribute	Value
<code>fclose()</code>	Thread safety	MT-Safe

CONFORMING TO

POSIX.1-2001, POSIX.1-2008, C89, C99.

NOTES

Note that `fclose()` flushes only the user-space buffers provided by the C library. To ensure that the data is physically stored on disk the kernel buffers must be flushed too, for example, with `sync(2)` or `fsync(2)`.

SEE ALSO

`close(2)`, `fcloseall(3)`, `fflush(3)`, `fileno(3)`, `fopen(3)`, `setbuf(3)`

COLOPHON

This page is part of release 5.10 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/>.