

**NAME**

`fpurge`, `__fpurge` – purge a stream

**SYNOPSIS**

```
/* unsupported */
#include <stdio.h>

int fpurge(FILE *stream);

/* supported */
#include <stdio.h>
#include <stdio_ext.h>

void __fpurge(FILE *stream);
```

**DESCRIPTION**

The function `fpurge()` clears the buffers of the given stream. For output streams this discards any unwritten output. For input streams this discards any input read from the underlying object but not yet obtained via `getc(3)`; this includes any text pushed back via `ungetc(3)`. See also `fflush(3)`.

The function `__fpurge()` does precisely the same, but without returning a value.

**RETURN VALUE**

Upon successful completion `fpurge()` returns 0. On error, it returns `-1` and sets `errno` appropriately.

**ERRORS****EBADF**

`stream` is not an open stream.

**ATTRIBUTES**

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

Interface	Attribute	Value
<code>__fpurge()</code>	Thread safety	MT-Safe race:stream

**CONFORMING TO**

These functions are nonstandard and not portable. The function `fpurge()` was introduced in 4.4BSD and is not available under Linux. The function `__fpurge()` was introduced in Solaris, and is present in glibc 2.1.95 and later.

**NOTES**

Usually it is a mistake to want to discard input buffers.

**SEE ALSO**

`fflush(3)`, `setbuf(3)`, `stdio_ext(3)`

**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/>.