



Linux Ubuntu 26.04 Manual Pages on command 'basename.3'

\$ man basename.3

basename(3) Library Functions Manual basename(3)

NAME

basename, dirname - parse pathname components

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

```
#include <libgen.h>

char *dirname(char *path);

char *basename(char *path);
```

DESCRIPTION

Warning: there are two different functions `basename()`; see below.

The functions `dirname()` and `basename()` break a null-terminated pathname string into directory and filename components. In the usual case, `dirname()` returns the string up to, but not including, the final '/', and `basename()` returns the component following the final '/'. Trailing '/' characters are not counted as part of the pathname.

If path does not contain a slash, `dirname()` returns the string "." while `basename()` returns a copy of path. If path is the string "/", then both `dirname()` and `basename()` return the string "/". If path is a null pointer or points to an empty string, then both `dirname()` and `basename()` return the string ".".

Concatenating the string returned by `dirname()`, a "/", and the string returned by `basename()` yields a complete pathname.

Both `dirname()` and `basename()` may modify the contents of path, so it may be desirable to pass a copy when calling one of these functions.

The following list of examples (taken from SUSv2) shows the strings returned by `dirname()` and `basename()` for different paths:

RETURN VALUE

ATTRIBUTES

VERSIONS

With glibc, one gets the POSIX version of `basename()` when `<libgen.h>` is included, and the GNU version otherwise.

STANDARDS

POSIX.1-2008.

HISTORY

POSIX.1-2001.

BUGS

In the glibc implementation, the POSIX versions of these functions modify the path argument, and segfault when called with a static string such as "/usr".

Before glibc 2.2.1, the glibc version of `dirname()` did not correctly handle pathnames with trailing '/' characters, and generated a segfault if given a NULL argument.

EXAMPLES

The following code snippet demonstrates the use of `basename()` and `dirname()`:

```
char *dirc, *basec, *bname, *dname;

char *path = "/etc/passwd";

dirc = strdup(path);

basec = strdup(path);

dname = dirname(dirc);

bname = basename(basec);

printf("dirname=%s, basename=%s\n", dname, bname);
```

SEE ALSO

`basename(1)`, `dirname(1)`

Linux man-pages 6.17

2026-02-08

`basename(3)`