

explain\_openat\_or\_die(3)

explain\_openat\_or\_die(3)

**NAME**

explain\_openat\_or\_die – open a file relative to a dir fd and report errors

**SYNOPSIS**

```
#include <libexplain/openat.h>
```

```
int explain_openat_or_die(int fildes, const char *pathname, int flags, mode_t mode);
```

```
int explain_openat_on_error(int fildes, const char *pathname, int flags, mode_t mode);
```

**DESCRIPTION**

The **explain\_openat\_or\_die** function is used to call the *openat(2)* system call. On failure an explanation will be printed to *stderr*, obtained from the *explain\_openat(3)* function, and then the process terminates by calling `exit(EXIT_FAILURE)`.

The **explain\_openat\_on\_error** function is used to call the *openat(2)* system call. On failure an explanation will be printed to *stderr*, obtained from the *explain\_openat(3)* function, but still returns to the caller.

*fildes*     The fildes, exactly as to be passed to the *openat(2)* system call.

*pathname*  
           The pathname, exactly as to be passed to the *openat(2)* system call.

*flags*     The flags, exactly as to be passed to the *openat(2)* system call.

*mode*     The mode, exactly as to be passed to the *openat(2)* system call.

**RETURN VALUE**

The **explain\_openat\_or\_die** function only returns on success, see *openat(2)* for more information. On failure, prints an explanation and exits, it does not return.

The **explain\_openat\_on\_error** function always returns the value return by the wrapped *openat(2)* system call.

**EXAMPLE**

The **explain\_openat\_or\_die** function is intended to be used in a fashion similar to the following example:

```
int result = explain_openat_or_die(fildes, pathname, flags, mode);
```

**SEE ALSO**

*openat(2)*  
           open a file relative to a directory file descriptor

*explain\_openat(3)*  
           explain *openat(2)* errors

*exit(2)*   terminate the calling process

**COPYRIGHT**

libexplain version 1.4

Copyright © 2013 Peter Miller

