

# PYTHON OS FILE/DIRECTORY METHODS

[http://www.tutorialspoint.com/python/os\\_file\\_methods.htm](http://www.tutorialspoint.com/python/os_file_methods.htm)

Copyright © tutorialspoint.com

The **os** module provides a big range of useful methods to manipulate files and directories. Most of the useful methods are listed here:

## SN Methods with Description

- 1  
[os.accesspath, mode](#)  
Use the real uid/gid to test for access to path.
- 2  
[os.chdirpath](#)  
Change the current working directory to path
- 3  
[os.chflagspath, flags](#)  
Set the flags of path to the numeric flags.
- 4  
[os.chmodpath, mode](#)  
Change the mode of path to the numeric mode.
- 5  
[os.chownpath, uid, gid](#)  
Change the owner and group id of path to the numeric uid and gid.
- 6  
[os.chrootpath](#)  
Change the root directory of the current process to path.
- 7  
[os.closefd](#)  
Close file descriptor fd.
- 8  
[os.closerangefd\\_low, fd\\_high](#)  
Close all file descriptors from fd\_low *inclusive* to fd\_high *exclusive*, ignoring errors.
- 9  
[os.dupfd](#)  
Return a duplicate of file descriptor fd.
- 10  
[os.dup2fd, fd2](#)

Duplicate file descriptor fd to fd2, closing the latter first if necessary.

11

[os.fchdirfd](#)

Change the current working directory to the directory represented by the file descriptor fd.

12

[os.fchmodfd, mode](#)

Change the mode of the file given by fd to the numeric mode.

13

[os.fchownfd, uid, gid](#)

Change the owner and group id of the file given by fd to the numeric uid and gid.

14

[os.fdatasyncfd](#)

Force write of file with filedescriptor fd to disk.

15

[os.fdopenfd\[, mode\[, bufsize\]\]](#)

Return an open file object connected to the file descriptor fd.

16

[os.fpathconffd, name](#)

Return system configuration information relevant to an open file. name specifies the configuration value to retrieve.

17

[os.fstatfd](#)

Return status for file descriptor fd, like stat.

18

[os.fstatvfsfd](#)

Return information about the filesystem containing the file associated with file descriptor fd, like statvfs.

19

[os.fsyncfd](#)

Force write of file with filedescriptor fd to disk.

20

[os.ftruncatefd, length](#)

Truncate the file corresponding to file descriptor fd, so that it is at most length bytes in size.

21

[os.getcwd](#)

Return a string representing the current working directory.

22

[os.getcwd](#)

Return a Unicode object representing the current working directory.

23

[os.isattyfd](#)

Return True if the file descriptor `fd` is open and connected to a *tty-like* device, else False.

24

[os.lchflagspath, flags](#)

Set the flags of `path` to the numeric flags, like `chflags`, but do not follow symbolic links.

25

[os.lchmodpath, mode](#)

Change the mode of `path` to the numeric mode.

26

[os.lchownpath, uid, gid](#)

Change the owner and group id of `path` to the numeric `uid` and `gid`. This function will not follow symbolic links.

27

[os.linksrc, dst](#)

Create a hard link pointing to `src` named `dst`.

28

[os.listdirpath](#)

Return a list containing the names of the entries in the directory given by `path`.

29

[os.lseekfd, pos, how](#)

Set the current position of file descriptor `fd` to position `pos`, modified by `how`.

30

[os.lstatpath](#)

Like `stat`, but do not follow symbolic links.

31

[os.majordevice](#)

Extract the device major number from a raw device number.

32

[os.makedevmajor, minor](#)

Compose a raw device number from the major and minor device numbers.

- 33 [os.makedirpath\[, mode\]](#)  
Recursive directory creation function.
- 34 [os.minordevice](#)  
Extract the device minor number from a raw device number .
- 35 [os.mkdirpath\[, mode\]](#)  
Create a directory named path with numeric mode mode.
- 36 [os.mkfifopath\[, mode\]](#)  
Create a FIFO *anamedpipe* named path with numeric mode mode. The default mode is 0666 *octal*.
- 37 [os.mknodfilename\[, mode = 0600, device\]](#)  
Create a filesystem node *file, devicespecialfileornamedpipe* named filename.
- 38 [os.openfile, flags\[, mode\]](#)  
Open the file file and set various flags according to flags and possibly its mode according to mode.
- 39 [os.openpty](#)  
Open a new pseudo-terminal pair. Return a pair of file descriptors *master, slave* for the pty and the tty, respectively.
- 40 [os.pathconfpath, name](#)  
Return system configuration information relevant to a named file.
- 41 [os.pipe](#)  
Create a pipe. Return a pair of file descriptors *r, w* usable for reading and writing, respectively.
- 42 [os.popencommand\[, mode\[, bufsize\]\]](#)  
Open a pipe to or from command.
- 43 [os.readfd, n](#)  
Read at most n bytes from file descriptor fd. Return a string containing the bytes read. If the end of the file referred to by fd has been reached, an empty string is returned.

44

[os.readlinkpath](#)

Return a string representing the path to which the symbolic link points.

45

[os.removepath](#)

Remove the file path.

46

[os.removedirspath](#)

Remove directories recursively.

47

[os.rename](#)*src, dst*

Rename the file or directory *src* to *dst*.

48

[os.rename](#)*old, new*

Recursive directory or file renaming function.

49

[os.rmdir](#)*path*

Remove the directory path

50

[os.stat](#)*path*

Perform a stat system call on the given path.

51

[os.stat](#) *float* *times**[newvalue]*

Determine whether *stat\_result* represents time stamps as float objects.

52

[os.stat](#)*vfs**path*

Perform a statvfs system call on the given path.

53

[os.symlink](#)*src, dst*

Create a symbolic link pointing to *src* named *dst*.

54

[os.tcgetpgrp](#)*fd*

Return the process group associated with the terminal given by *fd*  
*anopenfiledescriptorasreturnedbyopen()*.

55

[os.tcsetpgrp](#)*fd, pg*

Set the process group associated with the terminal given by `fd` *anopenfiledescriptorasreturnedbyopen()* to `pg`.

- 56 [`os.tempnam\[dir\[, prefix\]\]`](#)
- Return a unique path name that is reasonable for creating a temporary file.
- 57 [`os.tmpfile`](#)
- Return a new file object opened in update mode `w + b`.
- 58 [`os.tmpnam`](#)
- Return a unique path name that is reasonable for creating a temporary file.
- 59 [`os.ttynamefd`](#)
- Return a string which specifies the terminal device associated with file descriptor `fd`. If `fd` is not associated with a terminal device, an exception is raised.
- 60 [`os.unlinkpath`](#)
- Remove the file path.
- 61 [`os.utimepath, times`](#)
- Set the access and modified times of the file specified by `path`.
- 62 [`os.walktop\[, topdown = True\[, onerror = None\[, followlinks = False\]\]\]`](#)
- Generate the file names in a directory tree by walking the tree either top-down or bottom-up.
- 63 [`os.writefd, str`](#)
- Write the string `str` to file descriptor `fd`. Return the number of bytes actually written.