OS Functions explained

os.system() # Executing a shell command

os.stat() # Get the status of a file

os.environ() # Get the users environment

os.chdir() # Move focus to a different directory

os.getcwd() # Returns the current working directory

os.getgid() # Return the real group id of the current process

os.getuid() # Return the current process’s user id

os.getpid() # Returns the real process ID of the current process

os.getlogin() # Return the name of the user logged

os.access() # Check read permissions

os.chmod() # Change the mode of path to the numeric mode

os.chown() # Change the owner and group id

os.umask(mask) # Set the current numeric umask

os.getsize() # Get the size of a file

os.environ() # Get the users environment

os.uname() # Return information about the current operating system

os.chroot(path) # Change the root directory of the current process to path

os.listdir(path)# List of the entries in the directory given by path

os.getloadavg() # Show queue averaged over the last 1, 5, and 15 minutes

os.path.exists()# Check if a path exists

os.walk() # Print out all directories, sub-directories and files

os.mkdir(path) # Create a directory named path with numeric mode mode

os.remove(path) # Remove (delete) the file path

os.rmdir(path) # Remove (delete) the directory path

os.makedirs(path)# Recursive directory creation function

os.removedirs(path) # Remove directories recursively

os.rename(src, dst) # Rename the file or directory src to dst

OS Functions Examples

Let’s get started to see how we can use these OS functions.

#Get current working directory with os.getcwd()

print os.getcwd()

#Get the status of a file with os.stat()

print "Getting the status of: ", os.stat('/usr/bin/python')

#Execute a shell command with os.system()

os.system('ls -l')

#Return the current process id with os.getpid()

print os.getpid()

os.chmod(path, mode)

#Change the owner and group id of path to the numeric uid and gid with os.chown()

os.chown(path, uid, gid)

#Processes in the system run queue averaged over the last 1, 5, and 15 minutes

print os.getloadavg()

#Check if a path exists with os.path.exists()

if os.path.exists("file.txt"):

#Create a new directory named 'new\_directory' if it doesn't exist already"

os.path.exists("new\_directory") or os.mkdir("new\_directory")

#Check if the path is a directory or a file with os.path.isdir() & os.path.isfile()

path = "/tmp"

if os.path.isdir(path): print "That's a directory"

if os.path.isfile(path): print "That's a file"

#Create a directory with os.makedir()

print os.mkdir('new\_directory', 0666)

#Recursive create directories with os.makedirs()

os.makedirs('dir\_a/dir\_b/dir\_c')

#Remove a directory with os.rmdir()

print os.rmdir('directory')

#Recursively remove empty directories with os.rmdirs()

os.removedirs('dir\_a/dir\_b/dir\_c')

#Rename a file with os.rename()

print os.rename('/path/to/old/file', '/path/to/new/file')

#Rename a file with shutil.move()

print shutil.move('/path/to/old/file', '/path/to/new/file')

#Rename a file with shutil.copy()

print shutil.copy('/path/to/old/file', '/path/to/new/file')

#Get the users home directory

print os.path.expanduser('~')

#Check read permissions with os.access()

path = '/tmp/file.txt'

print os.access(path, os.R\_OK)

#Get the users environment with os.environmen()

home = os.environ['HOME']

print home

#Move focus to a different directory with os.chdir()

print os.chdir('/tmp')

#Print out all directories, sub-directories and files with os.walk()

for root, dirs, files in os.walk("/tmp"):

print root

print dirs

print files

#Get the last time a directory was accessed with os.path.getatime()

os.path.getatime('/tmp')

#Get the last time a directory was modified with os.path.getmtime()

os.path.getmtime('/tmp')

#Get the user ID with os.getuid()

if os.getuid() != 0: print "you are not root"

#Get the group ID with os.getgid()

print os.getgid()

#Return the name of the user logged in with os.getlogin()

print os.getlogin()

#Returns a list of all files in a directory with os.listdir()

for filename in os.listdir("/tmp"):

print "This is inside /tmp", filename

#Get the size of a file with os.path.getsize()

path.getsize("/tmp/file.txt")

[Python sys Module - GeeksforGeeks](https://www.geeksforgeeks.org/python-sys-module/)