Here's a structured table of the **`os` module commands**:

Action	Command/Code	Description
Open a file in read-only mode	`python f = os.open("myfile.txt", os.O_RDONLY) contents = os.read(f, 1024) os.close(f)`	Opens a file in read-only mode and reads its contents.
Open a file in write-only mode	`python f = os.open("myfile.txt", os.O_WRONLY) os.write(f, b"Hello, world!") os.close(f)`	Opens a file in write-only mode and writes to it.
List files in a directory	`python files = os.listdir(".") print(files)`	Lists all files and directories in the current directory.
Create a new directory	`python os.mkdir("newdir")`	Creates a new directory named `newdir`.
Remove a file	`python os.remove("myfile.txt")`	Deletes a specified file.
Remove an empty directory	`python os.rmdir("newdir")`	Deletes an empty directory named `newdir`.
Run a system command (`os.system`)	`python os.system("ls")`	Executes a system command (`ls` for Linux/macOS, `dir` for Windows).
Run a system command and capture output	`python f = os.popen("ls") output = f.read() f.close() print(output)`	Runs a command and captures its output in a file-like object.
Check the current working directory	`python cwd = os.getcwd() print(cwd)`	Retrieves the current working directory.
Change the current working directory	`python os.chdir("path/to/directory")`	Changes the current working directory to the specified path.

Check if a path exists	`python exists = os.path.exists("path/to/file_or_directory") print(exists)`	Checks if a specified file or directory exists.
Rename a file or directory	`python os.rename("oldname.txt", "newname.txt")`	Renames a file or directory.
Get file size	`python size = os.path.getsize("myfile.txt") print(size)`	Gets the size of a specified file in bytes.
Get absolute path of a file	`python abs_path = os.path.abspath("myfile.txt") print(abs_path)`	Gets the absolute path of a specified file.
Join paths	`python full_path = os.path.join("folder", "file.txt") print(full_path)`	Combines directory and file names into a single path.