# Absolute vs Relative Path in Python – File Operations

## 1. Definitions

• Absolute Path: The full path from the root directory (e.g., C:/Users/Geetha/Documents/file.txt).

• Relative Path: A path relative to the current working directory (e.g., ../backup/file.txt).

The shutil module in Python is part of the standard library and is used for **high-level file operations**, such as copying, moving, deleting, and archiving files and directories. It's especially useful when you're working with file system tasks that go beyond simple reading and writing.

**📌 Key Uses of shutil**

| **Function** | **Description** | **Example** |
| --- | --- | --- |
| shutil.copy(src, dst) | Copies a file from src to dst | shutil.copy('file1.txt', 'backup/file1.txt') |
| shutil.copy2(src, dst) | Like copy() but also copies metadata (timestamps) | shutil.copy2('file1.txt', 'backup/file1.txt') |
| shutil.copytree(src, dst) | Recursively copies an entire directory tree | shutil.copytree('my\_folder', 'backup\_folder') |
| shutil.move(src, dst) | Moves a file or directory to a new location | shutil.move('file.txt', 'archive/file.txt') |
| shutil.rmtree(path) | Deletes an entire directory tree | shutil.rmtree('old\_folder') |
| shutil.disk\_usage(path) | Returns disk usage statistics (total, used, free) | shutil.disk\_usage("/") |
| shutil.make\_archive(base\_name, format, root\_dir) | Creates a compressed archive (e.g., zip, tar) | shutil.make\_archive("backup", "zip", "my\_folder") |
| shutil.unpack\_archive(filename, extract\_dir) | Extracts an archive | shutil.unpack\_archive('backup.zip', 'extracted') |

**✅ Real-world Example**

import shutil

import os

# Create a backup of a text file

source = "report.txt"

destination = "backup/report\_backup.txt"

# Ensure destination folder exists

os.makedirs("backup", exist\_ok=True)

# Copy file

shutil.copy2(source, destination)

print("Backup completed.")

**✅ Why use shutil?**

* **Cross-platform**: Works on all platforms (Windows, Linux, Mac).
* **Safe and simple**: Handles file metadata and directory trees efficiently.
* **Useful for automation**: Ideal for backup scripts, installers, or log file management.

## 2. File Copy Example using Absolute and Relative Path

Python code example:

import os  
import shutil  
  
# Absolute Paths  
source\_abs = r" C:\Personal\BVRIT Python\Notes Repo\BVRIT\_Python\_notes\Programs\File Handling \ The-zen-of-python.txt"  
destination\_abs = r" C:\Personal\Backup \data.txt"  
  
# Relative Paths (assuming you're in "project" folder)  
source\_rel = " The-zen-of-python.txt"  
destination\_rel = "..\\..\\backup\data.txt"  
  
# Copy using absolute path  
shutil.copy(source\_abs, destination\_abs)  
print("File copied using absolute path.")  
  
# Copy using relative path  
shutil.copy(source\_rel, destination\_rel)  
print("File copied using relative path.")

## 3. Get File Size

Python code to get file size:

file\_path = "backup/data.txt"  
if os.path.exists(file\_path):  
 size = os.path.getsize(file\_path)  
 print(f"File Size: {size} bytes")

## 4. Common File-Related Functions in Python

|  |  |  |  |
| --- | --- | --- | --- |
| Function | Module | Description | Example |
| os.path.exists() | os | Check if file/folder exists | os.path.exists('file.txt') |
| os.remove() | os | Delete a file | os.remove('file.txt') |
| os.rename() | os | Rename a file | os.rename('old.txt', 'new.txt') |
| os.listdir() | os | List files in a directory | os.listdir('.') |
| os.makedirs() | os | Create a directory | os.makedirs('backup/folder') |
| shutil.copy() | shutil | Copy a file | shutil.copy('a.txt', 'b.txt') |
| shutil.copytree() | shutil | Copy an entire directory | shutil.copytree('src', 'dst') |
| shutil.move() | shutil | Move a file/folder | shutil.move('a.txt', 'new/') |
| os.path.getsize() | os | Get file size | os.path.getsize('file.txt') |
| os.path.abspath() | os | Get absolute path | os.path.abspath('file.txt') |
| os.getcwd() | os | Get current directory | os.getcwd() |
| os.chdir() | os | Change current directory | os.chdir('..') |

## 5. Copying Entire Folder Example

src\_folder = "project"  
dst\_folder = "backup\_project"  
  
# Copy full folder tree  
shutil.copytree(src\_folder, dst\_folder)  
print("Folder copied successfully.")