1. **shutil.copy(src, dst)** is used to copy a single file from source **src** to destination **dst**. On the other hand, **shutil.copytree(src, dst)** is used to recursively copy an entire directory tree, including all of its files and subdirectories, from source **src** to destination **dst**.

Example:

pythonCopy code

import shutil # Copy a file shutil.copy('source\_file.txt', 'destination\_folder/') # Copy an entire directory tree shutil.copytree('source\_directory/', 'destination\_folder/')

1. The **os.rename(src, dst)** function is commonly used to rename files. It takes the current name (**src**) and the desired new name (**dst**) as arguments.

Example:

pythonCopy code

import os # Rename a file os.rename('old\_name.txt', 'new\_name.txt')

1. The **send2trash** module and **shutil** module provide different approaches for deleting files:
   * **send2trash** (**send2trash.send2trash(path)**) moves the file or directory to the system's trash or recycle bin instead of permanently deleting it. This provides a safety net in case you need to recover the file.
   * **shutil** (**shutil.rmtree(path)** for directories or **os.remove(path)** for files) is more direct and permanently deletes the file or directory without moving it to the trash. Be cautious when using these functions, as the data cannot be easily recovered.
2. The equivalent of the **open()** method for **ZipFile** objects is **ZipFile(file, mode)**, where **file** is the name of the ZIP archive file, and **mode** specifies the mode in which the file is opened (e.g., 'r' for reading, 'w' for writing, 'a' for appending).

Example:

pythonCopy code

import zipfile # Open a ZIP file for reading with zipfile.ZipFile('example.zip', 'r') as zip\_file: # Do something with the ZIP file

1. Here's a simple Python program that searches a folder tree for files with a specific file extension (e.g., **.pdf** or **.jpg**) and copies them to a new folder:

pythonCopy code

import os import shutil def copy\_files\_with\_extension(source\_folder, destination\_folder, extension): for root, dirs, files in os.walk(source\_folder): for file in files: if file.endswith(extension): source\_path = os.path.join(root, file) destination\_path = os.path.join(destination\_folder, file) shutil.copy(source\_path, destination\_path) print(f'Copied: {source\_path} to {destination\_path}') # Example usage source\_folder = '/path/to/source/folder' destination\_folder = '/path/to/destination/folder' file\_extension = '.pdf' copy\_files\_with\_extension(source\_folder, destination\_folder, file\_extension)

Replace **'/path/to/source/folder'**, **'/path/to/destination/folder'**, and **'.pdf'** with the appropriate values for your case.