1. How do you distinguish between shutil.copy() and shutil.copytree()?

* shutil.copy(src, dst): This function is used to copy a single file from the source (src) to the destination (dst). It preserves the metadata of the file but does not copy directories.
* shutil.copytree(src, dst): This function is used to recursively copy an entire directory tree from the source directory (src) to the destination directory (dst). It copies all files and subdirectories along with their content and metadata.

2. What function is used to rename files?

The function used to rename files is os.rename(src, dst) from the os module. It renames the file or directory named src to dst.

3. What is the difference between the delete functions in the send2trash and shutil modules?

* send2trash: The send2trash module provides a function called send2trash() that moves files and directories to the system's trash or recycle bin instead of immediately deleting them. This allows for a safer deletion process, as items can be recovered from the trash if needed.
* shutil: The shutil module provides functions like shutil.rmtree() to remove entire directories and their contents or shutil.remove() to delete individual files. These functions permanently delete the files and directories, and they cannot be easily recovered without backups.

4. What ZipFile method is equivalent to File objects’ open() method?

The ZipFile method equivalent to File objects' open() method is ZipFile.open(filename, mode). This method is used to open a file within a ZIP archive and returns a file-like object that can be used to read or write data within that file.

5. Create a program that searches a folder tree for files with a certain file extension and copies them to a new folder:

import os

import shutil

def copy\_files\_with\_extension(source\_dir, target\_dir, extension):

if not os.path.exists(target\_dir):

os.makedirs(target\_dir)

for foldername, subfolders, filenames in os.walk(source\_dir):

for filename in filenames:

if filename.endswith(extension):

source\_path = os.path.join(foldername, filename)

target\_path = os.path.join(target\_dir, filename)

shutil.copy2(source\_path, target\_path) # Preserves metadata