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

The shutil module in Python provides two functions for copying files and directories: shutil.copy() and shutil.copytree(). The main difference between the two functions is that shutil.copy() copies a single file, while shutil.copytree() copies an entire directory tree, including all of its files and subdirectories.

Here are some more details on each function:

shutil.copy(src, dst, *, follow_symlinks=True): This function copies the file at the src path to the dst path. If the dst path specifies a directory, the file is copied into that directory with the same name. If the dst path specifies a file name, the file is copied to that file name. If the dst file already exists, it is overwritten. The follow_symlinks parameter specifies whether to follow symbolic links or not (the default is True).

2. What function is used to rename files??

The os.rename() function is used to rename files in Python. It is part of the built-in os module and allows you to rename a file by providing the current file name and the new file name.

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

Both the send2trash and shutil modules in Python provide functions to delete files and directories, but there are some differences between the two.

The main difference is that the send2trash module moves the file or directory to the operating system's trash or recycle bin instead of deleting it permanently, while the shutil module deletes the file or directory permanently. This means that if you use send2trash to delete a file, you can potentially recover it later from the trash or recycle bin, whereas if you use shutil to delete a file, it will be gone forever.

4.ZipFile objects have a close() method just like File objects’ close() method. What ZipFile method is equivalent to File objects’ open() method?

The equivalent method of the ZipFile object to the open() method of a File object is the ZipFile() method. It is used to create a new ZipFile object that represents a ZIP archive, which can then be used to read, write, or modify the contents of the archive.

The syntax of the ZipFile() method is as follows:

python Copy code zipfile.ZipFile(file, mode='r', compression=ZIP_STORED, allowZip64=True, *, compresslevel=None)

Here, file is the path to the ZIP archive file that you want to open. The mode parameter specifies the mode in which the archive should be opened, which can be 'r' (read), 'w' (write), 'x' (exclusive write), or 'a' (append). The compression parameter specifies the compression method to use when writing files to the archive. The allowZip64 parameter specifies whether to allow ZIP files with sizes greater than 2 GB. The compresslevel parameter specifies the level of compression to use when writing files to the archive.

5. Create a programme that searches a folder tree for files with a certain file extension (such as .pdf or .jpg). Copy these files from whatever location they are in to a new folder

In [1]:

```python
import os
import shutil

# Set the source and destination directories
src_dir = '/path/to/source/directory'
dest_dir = '/path/to/destination/directory'

# Set the file extension to search for
file_ext = '.pdf'

# Walk the directory tree and copy matching files
for root, dirs, files in os.walk(src_dir):
    for file in files:
        if file.endswith(file_ext):
            src_path = os.path.join(root, file)
            dest_path = os.path.join(dest_dir, file)
            shutil.copy2(src_path, dest_path)
            print(f"Copied {src_path} to {dest_path}")
```

In [ ]: