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

**Ans:** The shutil.copy() function will copy a single file, while shutil.copytree() will copy an entire folder, along with all its contents.

**2. What function is used to rename files??**

**Ans:** The shutil.move() function is used for renaming files, as well as moving them.

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

**Ans:** The send2trash functions will move a file or folder to the recycle bin, while shutil functions will permanently delete files and folders.

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

**Ans:** The zipfile.ZipFile() function is equivalent to the open() function; the first argument is the filename, and the second argument is the mode to open the ZIP file in (read, write, or append).

**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.**

**Ans:** The programme is:

|  |
| --- |
|  |
| #!/usr/bin/envpython3 |
| # Write a program that walks through a folder tree  # and searches 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.  import os, shutil  def selectiveCopy(folder, extensions, destFolder):  folder = os.path.abspath(folder)  destFolder = os.path.abspath(destFolder)  print('Looking in', folder, 'for files with extensions of', ', '.join(extensions))  for foldername, subfolders, filenames in os.walk(folder):  for filename in filenames:  name, extension = os.path.splitext(filename)  if extension in extensions:  fileAbsPath = foldername + os.path.sep + filename  print('Coping', fileAbsPath, 'to', destFolder)  shutil.copy(fileAbsPath, destFolder)  extensions = ['.php', '.py']  folder = 'randomFolder'  destFolder = 'selectiveFolder'  selectiveCopy(folder, extensions, destFolder) |
|  |