Certainly, here are the answers to your questions:

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

- `shutil.copy(src, dst)` is used to copy a single file from the source path (`src`) to the destination path (`dst`).

- `shutil.copytree(src, dst)` is used to recursively copy an entire directory tree from the source directory (`src`) to the destination directory (`dst`), including all subdirectories and files.

2. What function is used to rename files?

The `os.rename()` function is used to rename files. It takes two arguments: the current file name (path) and the new file name (path).

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

- The `send2trash` module provides a `send2trash()` function that sends a file or directory to the trash or recycle bin, allowing for potential recovery.

- The `shutil` module provides various functions for working with files and directories, including `shutil.rmtree()` for deleting an entire directory tree (including all files and subdirectories) and `os.remove()` for deleting a single file. These deletions are permanent.

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

The `ZipFile` method equivalent to `File` objects' `open()` method is `ZipFile.open()`. This method is used to extract a member (file) from the archive and open it for reading.

5. Here's a simple Python program that searches a folder tree for files with a certain file extension and copies them to a new folder:

```python

import os

import shutil

def copy\_files\_by\_extension(source\_folder, destination\_folder, extension):

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

for filename in filenames:

if filename.endswith(extension):

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

destination\_path = os.path.join(destination\_folder, filename)

shutil.copy(source\_path, destination\_path)

print(f"Copying {source\_path} to {destination\_path}")

source\_folder = "source\_folder" # Specify the source folder path

destination\_folder = "destination\_folder" # Specify the destination folder path

extension = ".pdf" # Specify the file extension you're looking for

if not os.path.exists(destination\_folder):

os.makedirs(destination\_folder)

copy\_files\_by\_extension(source\_folder, destination\_folder, extension)

Replace `"source\_folder"` and `"destination\_folder"` with the actual paths of your source and destination folders, and update `extension` to the desired file extension you want to copy. This program uses the `os.walk()` function to traverse the folder tree and `shutil.copy()` to copy the files to the new folder.