PROFESSIONAL

AUTOMATE FILE SEGREGATION

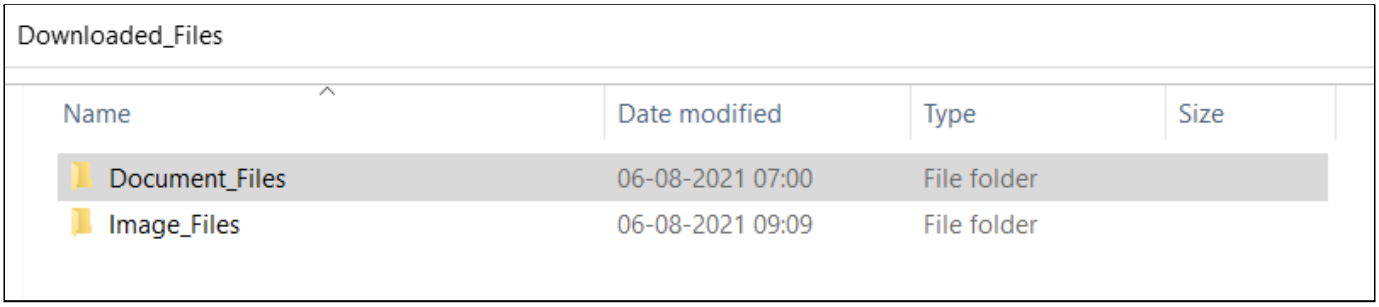
INSTRUCTIONS:

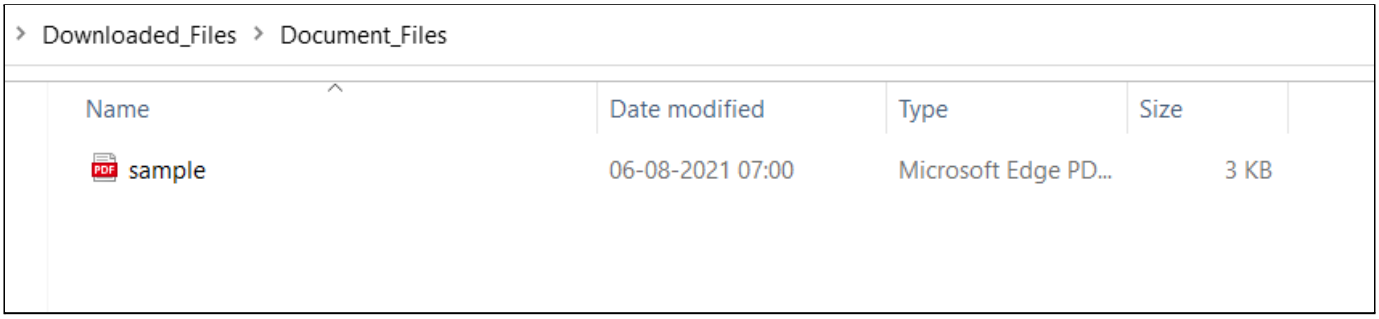
Goal of the Project:

In Class 102, you build a python program to move image files from one folder to another folder using os and shutil modules. In this project, you will write a program to move documents with extension .pdf from one folder to another folder.

Story:

The "**Download**" folder in your friend's computer is loaded with many files, as he keeps downloading his school work in that folder. He asked for your help to organize all the files. You can use your coding knowledge to create a Python program to move all the documents from the "**Downloads**" folder to a separate folder.





***This is just for your reference. We expect you to apply your own creativity to the project.**

PROFESSIONAL

AUTOMATE FILE SEGREGATION

WhiteHat Jr
Live Online Coding for Kids

Getting Started:

1. Create a new folder named **Project102.**
2. Open the folder in VSC.
3. Create a new file named **Move_File.py.**
4. Create a new folder in your system named "**Document_Files**". Make sure to keep that folder out of the **Downloads** folder.

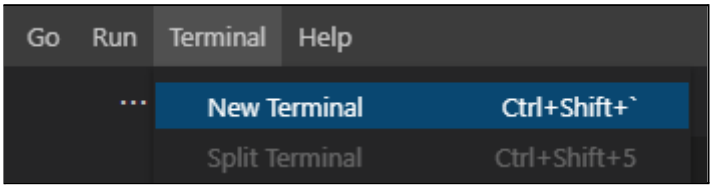
Specific Tasks to complete the Project:

1. Import **os** & **shutil** modules.
2. Create two variables **from_dir** and **to_dir** to store source path and destination path, respectively.
    - Source path will direct to the **Downloads** folder
    - Destination path will direct to "**Document_Files**" folder

    Note: *VSC accepts paths with " / ".*
3. Create a variable, **list_of_files**, to store the names of all the files from the source path using **os.listdir()**
4. Print the **list_of_files** to check in the terminal by running the code.

```
list_of_files = os.listdir(from_dir)
print(list_of_files)
```

5. Run the code using Terminal.



- Run the file using **python Move_File.py / py Move_File.py**
- You will see the names of all the files present at the Source path.

6. Create a **for-in** loop to traverse through the **list_of_files**:
    - Use **os.path.splitext()** on each file name to capture the name & extension of each file.

- Comment on the previous **print()** statement,

7. Write an **if** condition to check if the extension is blank, if the condition is **true** then **continue**.
   - If the **extension** is **blank**; it will jump to the next file and check for the extension of the next file.
   - Else create another **if** condition to check if the extension of the files in the path is one of the extensions in a **list ['.txt', '.doc', '.docx', '.pdf'']**
8. Create 3 variables for the name of the directory paths:
   - Create **path1** as the name of the source path.

     Use string concatenation to merge **from_dir+'/'+file_name**

   - Create **path2** as we want to create a new folder with that extension name and move the files to that folder.

     Use string concatenation to merge **to_dir + '/' + "Document_Files"**

   - Create **path3** to assign the **destination path** with the same file name as the source.

     Use string concatenation to merge **to_dir + '/' + "Document_Files" + '/' + file_name**

9. Check if the folder/directory path exists before moving using an **if** condition: **(See Hint 1)**

   - Create a condition to check if the destination path exists at **path2**.

   - If true, use **print()** method to print a message moving with the file name which is being moved.

   - Use **shutil.move(path1, path3)**.

     In this case, **path1** is the source path and **path3** is the destination path.

10. Else make a new folder/directory then move:

    - Use **os.makedirs()** to create **path2.**

    - Use **print()** method to print a message moving with the file name.

    - Use **shutil.move(path1, path3)**

11. Run the code, check the specified folders for the files.

Submitting the Project:

1. **SAVE** all the changes made to the project.
2. Click on "**Run**" once to check if it is working.
3. Open the GitHub create a repository named **Project102**
4. Upload a file **Move_File.py** and click **Commit Changes**
5. Copy this link and submit it in the Student Dashboard Projects panel against the correct class number.

Hints:

1. Check if the **path2** exists then move the files.
    - Else create the **path2** and move the files.

```
# Check if Folder/Directory Path Exists Before Moving
# Else make a NEW Folder/Directory Then Move
if os.path.exists(path2):
  print("Moving " + file_name + ".....")

  # Move from path1 ---> path3
  shutil.move(path1, path3)

else:
  os.makedirs(path2)
  print("Moving " + file_name + ".....")
  shutil.move(path1, path3)
```

REMEMBER... Try your best, that's more important than being correct.

After submitting your project, your teacher will send you feedback on your work.

———— xxx ———— xxx ———— xxx ———— xxx ———— xxx ————