**PRINCE GEORGE'S COMMUNITY COLLEGE**

***Introduction to Python INT-2240***

**Project 1: Organizing Folders and Files**

In this project, students will:

* select and use of appropriate Python modules and modules entities to complete specific automation task;
* complete basic file management tasks such as directory and file creation, renaming, and backup;
* archive files and documents using tools such as gzip.

**Motivation:**

As a system or network administrator, you will have to create identical system configurations and load identical documents or files on hundreds or thousands computers. Imagine a classroom with computers that are pre-loaded with certain software and documents to be used by students. This type of task can only be done using a script that will repeat the same actions on all the devices. Python is one of the best and most popular scripting language used today.

**Project Resources:**

1. <http://automatetheboringstuff.com/2e/chapter10/>
2. <https://towardsdatascience.com/10-python-file-system-methods-you-should-know-799f90ef13c2>
3. https://www.journaldev.com/30205/python-os-library-functions
4. https://www.python-course.eu/python3\_file\_management.php
5. <https://www.guru99.com/reading-and-writing-files-in-python.html>

**Instructions:**

You will write a Python module named “**your1stname\_yourlastname\_files.py**” that will contain several functions to perform the following tasks in the prescribed order. Your program should have the statement **“if \_\_name\_\_ == \_\_main\_\_”** at the bottom, which will kick-start the execution of your module. **Note that each task is worth 10 points.**

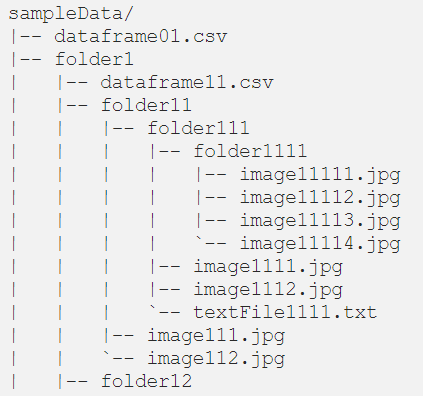


Figure 1

**Phase 0: Tell the user what the code is doing at runtime**

1. For every task, your Python code should display what it is about to do before starting, and inform the user when the task is done. This will also help you with the architecture of your code and debugging.

**Phase 1: Automated creation of directories and files**

1. Create a directory named “**your1stname\_yourlastname\_DIR**” in the “Documents” directory of your computer.
2. Inside the “your1stname\_yourlastname\_DIR” directory, **automate** the creation of directories and files according to the tree structure in Figure 1. You should find patterns and the file/folder names that would allow you to achieve this through a few loops. For instance, you may create variables such as “folder” and “image” and use them to generate names automatically.

**Phase 2: Create a work copy or a directory**

***In real life, it is a good practice to create a duplicate of a folder/file on which you will be working and work on the copy rather than on the original. This will help you preserve the document is things go wrong.***

1. Replicate the content of “your1stname\_yourlastname\_DIR” into a new directory named “**your1stname\_yourlastname\_DIR-CP**” that you will create inside your “Documents” folder.

**Phase 3: Manipulations**

***In real life, you will have to modify files, rename files, delete files, or archive (zip) files.***

1. Go through “your1stname\_yourlastname\_DIR” and delete all “.txt” files that your encounter. The files should be recoverable from the recycle bin.
2. Inside “your1stname\_yourlastname\_DIR-CP” rename all files with names ending in “2.jpg” by replacing this ending with “2.pdf.” This means you will alter those files into .pdf files.
3. Identify all the .pdf files that you created above, and archive them together in a single “.gzip” archive file named “pdf2.gzip.” Move the pdf2.gzip file to “your1stname\_yourlastname\_DIR.”
4. In “your1stname\_yourlastname\_DIR,” create a file named ”listofall.txt” as a record where you list all the files and folders that remain in “your1stname\_yourlastname\_DIR-CP,” and then delete “your1stname\_yourlastname\_DIR-CP” irreversibly, meaning it cannot be recovered. Not more that one file/folder name should be on the same line.

**Phase 4: Backups**

***In real life, you will have to write codes to packup documents.***

1. Create a “Backup” folder in your “Downloads” directory to backup the “pdf2.gzip” and ”listofall.txt” files.

**Phase 5: Visual representation**

***You need to verify that your code performed the right modifications. For directories and files, plotting a tree structure is usually used.***

1. Display a tree structure of “your1stname\_yourlastname\_DIR,” similar to the one in Figure 1.

**Note on submission:**

After you check your code and are satisfy with it, submit your **.py** module file **and** a screenshot of your final tree structure to Blackboard.