Write the command(s) to complete each of the following in a document.

Project 3 – Bash

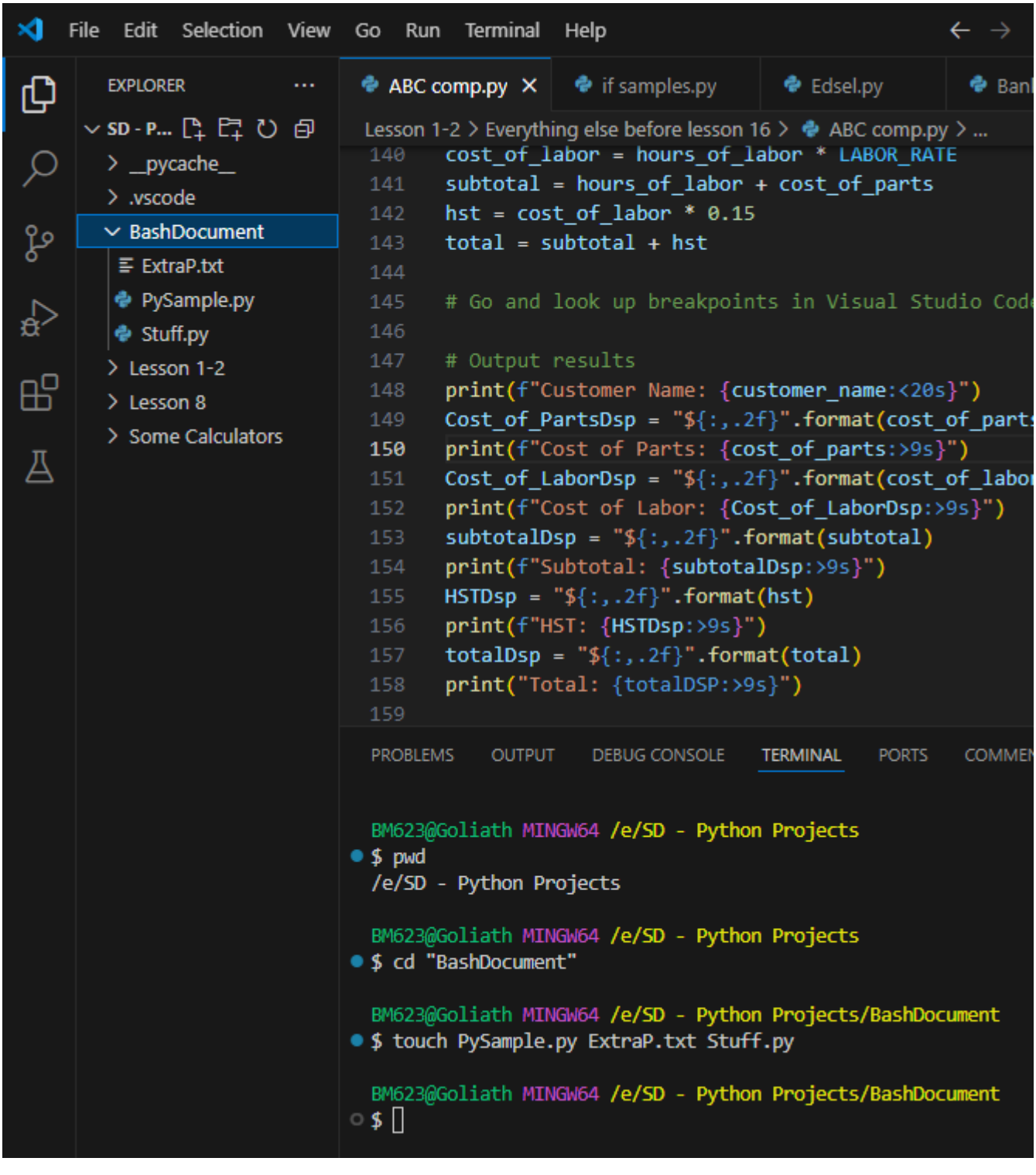
Group 6

Author: Brandon Maloney, Landon Lewis, Cameron Boyer

Project 3 – Bash – Prepare the following in a document.

- Create 3 files – one called **PySample.py**, one called **ExtraP.txt**, and a third called **Stuff.py**. Add code to each python program and add some text to txt file. Add the words **"RetailCost"** and **"getName"** in one or more of these files.

- touch PySample.py ExtraP.txt Stuff.py

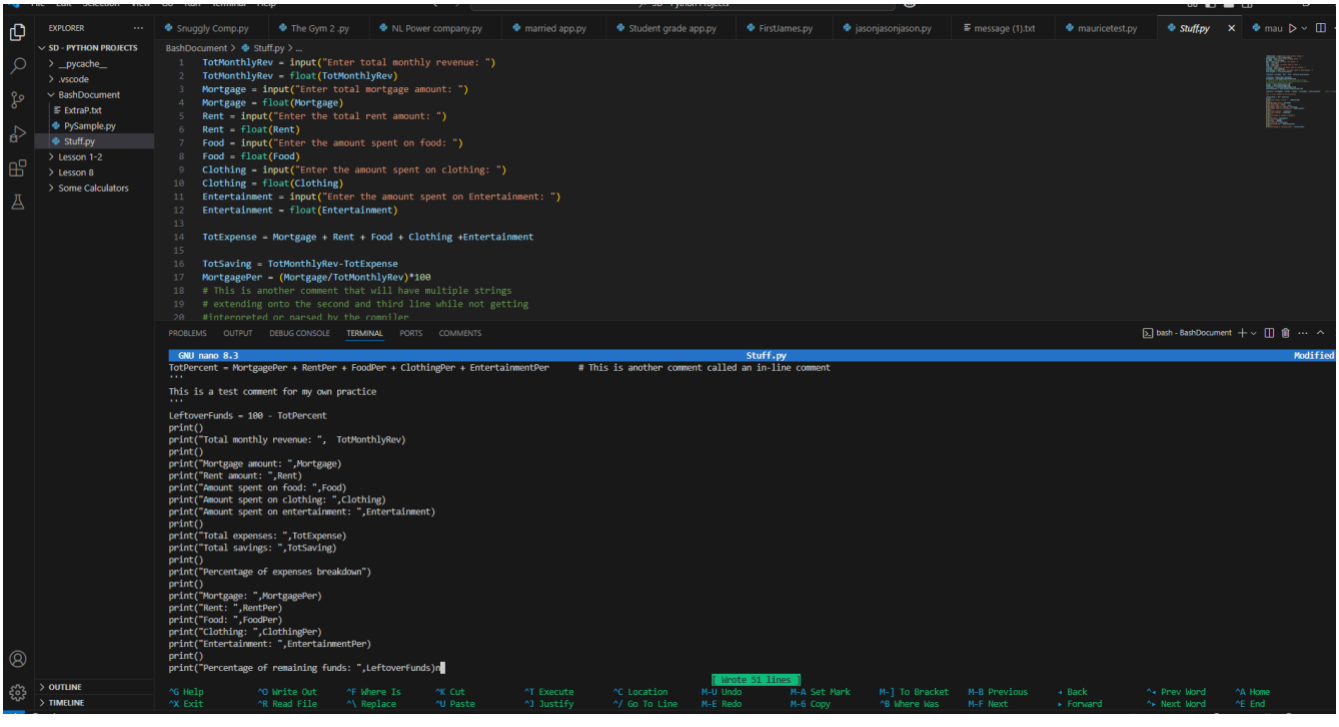


- nano PySample.py AND -nano Stuff.py

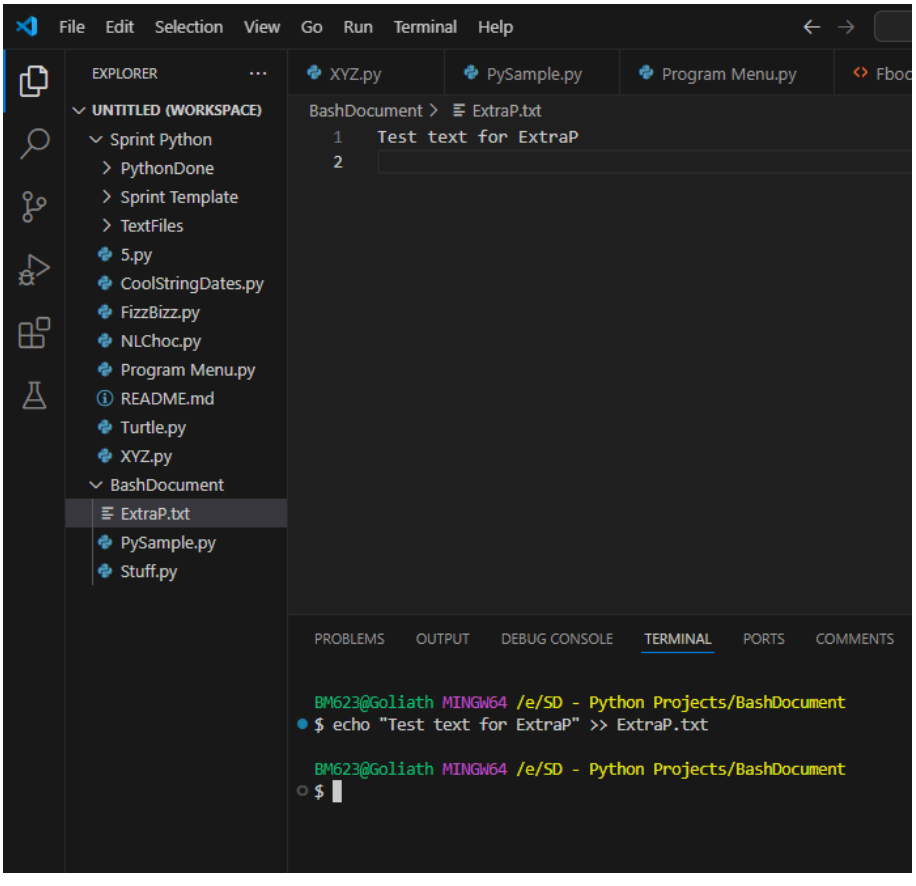
- (insert python code)

- Ctrl + X to save in nano

- Ctrl + O to exit



- echo "Test text for ExtraP" >> ExtraP.txt



- What is the current directory? List the contents of the current directory. List the contents of the current directory with any hidden files. List the files with the permissions displayed.

- pwd

- ls

- ls -la

- ls -l

```
BM623@Goliath MINGW64 /e/SD - Python Projects/BashDocument
$ pwd
/e/SD - Python Projects/BashDocument

BM623@Goliath MINGW64 /e/SD - Python Projects/BashDocument
$ ls
ExtraP.txt  PySample.py  Stuff.py

BM623@Goliath MINGW64 /e/SD - Python Projects/BashDocument
$ ls -la
total 9
drwxr-xr-x 1 BM623 197609    0 Feb 22 14:48 ./
drwxr-xr-x 1 BM623 197609    0 Feb 22 14:48 ../
-rw-r--r-- 1 BM623 197609   21 Feb 22 15:24 ExtraP.txt
-rw-r--r-- 1 BM623 197609    0 Feb 22 14:48 PySample.py
-rw-r--r-- 1 BM623 197609 1800 Feb 22 14:43 Stuff.py

BM623@Goliath MINGW64 /e/SD - Python Projects/BashDocument
$ ls -l
total 5
-rw-r--r-- 1 BM623 197609   21 Feb 22 15:24 ExtraP.txt
-rw-r--r-- 1 BM623 197609    0 Feb 22 14:48 PySample.py
-rw-r--r-- 1 BM623 197609 1800 Feb 22 14:43 Stuff.py

BM623@Goliath MINGW64 /e/SD - Python Projects/BashDocument
$
```

- Display the full contents of each of the files created above. Display only the first 3 lines of **PySample.py** and the last 3 lines of **Stuff.py**. Display the last 3 lines of all files. If you have a file that is large, what options are available to have it appear one screen at a time?

- cat PySample.py Stuff.py ExtraP.txt

```
BM623@Goliath MINGW64 /e/SD - Python Projects/BashDocument
$ cat PySample.py Stuff.py ExtraP.txt
TotMonthlyRev = input("Enter total monthly revenue: ")
TotMonthlyRev = float(TotMonthlyRev)
Mortgage = input("Enter total mortgage amount: ")
Mortgage = float(Mortgage)
Rent = input("Enter the total rent amount: ")
Rent = float(Rent)
Food = input("Enter the amount spent on food: ")
Food = float(Food)
Clothing = input("Enter the amount spent on clothing: ")
Clothing = float(Clothing)
Entertainment = input("Enter the amount spent on Entertainment: ")
Entertainment = float(Entertainment)

TotExpense = Mortgage + Rent + Food + Clothing +Entertainment

TotSaving = TotMonthlyRev-TotExpense
MortgagePer = (Mortgage/TotMonthlyRev)*100
# This is another comment that will have multiple strings
# extending onto the second and third line while not getting
#interpreted or parsed by the compiler
RentPer = (Rent/TotMonthlyRev)*100
FoodPer = (Food/TotMonthlyRev)*100
ClothingPer = (Clothing/TotMonthlyRev)*100
EntertainmentPer = (Entertainment/TotMonthlyRev)*100

TotPercent = MortgagePer + RentPer + FoodPer + ClothingPer + EntertainmentPer      # This is another comment called
'''
This is a test comment for my own practice
```

- head -n 3 PySample.py

- tail -n -3 Stuff.py

```
BM623@Goliath MINGW64 /e/SD - Python Projects/BashDocument
$ head -n 3 PySample.py
ItemName = input("Enter the item name")
ItemCost = input("Enter the item cost")
ItemCost =float(ItemCost)

BM623@Goliath MINGW64 /e/SD - Python Projects/BashDocument
$ tail -n -3 Stuff.py
print("Entertainment: ",EntertainmentPer)
print()
print("Percentage of remaining funds: ",LeftoverFunds)
```

- tail -n -3 PySample.py Stuff.py ExtraP.txt

```
BM623@Goliath MINGW64 /e/SD - Python Projects/BashDocument
$ tail -n -3 PySample.py Stuff.py ExtraP.txt
==> PySample.py <==
print(Sale25Off)
print(Sale33Off)
print(Sale50Off)
==> Stuff.py <==
print("Entertainment: ",EntertainmentPer)
print()
print("Percentage of remaining funds: ",LeftoverFunds)

==> ExtraP.txt <==
Test text for ExtraP
```

- you can use more/less and your filename after. Ex. more/less Stuff.py

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS    COMMENTS

TotMonthlyRev = input("Enter total monthly revenue: ")
TotMonthlyRev = float(TotMonthlyRev)
Mortgage = input("Enter total mortgage amount: ")
Mortgage = float(Mortgage)
Rent = input("Enter the total rent amount: ")
Rent = float(Rent)
Food = input("Enter the amount spent on food: ")
Food = float(Food)
Clothing = input("Enter the amount spent on clothing: ")
Clothing = float(Clothing)
Entertainment = input("Enter the amount spent on Entertainment: ")
Entertainment = float(Entertainment)

TotExpense = Mortgage + Rent + Food + Clothing +Entertainment

TotSaving = TotMonthlyRev-TotExpense
MortgagePer = (Mortgage/TotMonthlyRev)*100
# This is another comment that will have multiple strings
# extending onto the second and third line while not getting
#interpreted or parsed by the compiler
RentPer = (Rent/TotMonthlyRev)*100
FoodPer = (Food/TotMonthlyRev)*100
ClothingPer = (Clothing/TotMonthlyRev)*100
EntertainmentPer = (Entertainment/TotMonthlyRev)*100

TotPercent = MortgagePer + RentPer + FoodPer + ClothingPer + EntertainmentPer      # This is another comment called an in-line comment
'''
This is a test comment for my own practice
'''
Stuff.py
```

- Create a directory called **PythonCurrent**, one called **PythonDone** and one called **TextFiles**. In the PythonDone directory create 2 other directories called **2022** and **2023**. In the **TextFiles** directory create 2 other directories called **Python** and **Other**.

- Mkdir PythonCurrent PythonDone TextFiles

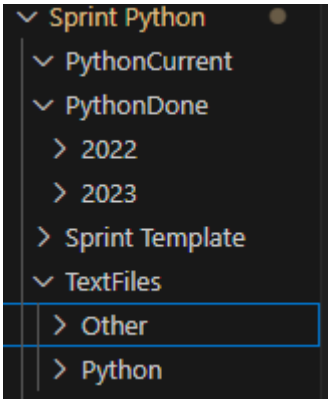
- Cd PythonDone

- mkdir 2022 2023

- Cd ..

- Cd TextFiles

- Mkdir Python Other

- 

- Change the current directory to **PythonDone – notice how the prompt changes to show the current directory**. Use pwd to confirm you are in the PythonDone directory. List the files.

- cd ..

- cd PythonDone

- pwd

- ls

Output: 2022/ 2023/

```
BM623@Goliath MINGW64 ~/Desktop/Sprint Python/TextFiles (main)
$ cd ..

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ cd PythonDone

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonDone (main)
$ pwd
/c/Users/BM623/Desktop/Sprint Python/PythonDone

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonDone (main)
$ ls
2022/  2023/

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonDone (main)
$
```

- Change to the following directories: **PythonCurrent**, **2023**, and **Other**. Prove that you are in the proper directory. Go back to your working directory.

- cd ..

- cd PythonCurrent

- pwd

- cd ../PythonDone/2023

- pwd

- cd ../../TextFiles/Other

- pwd

- cd ../..

```
BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonDone (main)
$ cd ..

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ cd PythonCurrent

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonCurrent (main)
$ pwd
/c/Users/BM623/Desktop/Sprint Python/PythonCurrent

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonCurrent (main)
$ cd ../PythonDone/2023

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonDone/2023 (main)
$ pwd
/c/Users/BM623/Desktop/Sprint Python/PythonDone/2023

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonDone/2023 (main)
$ cd ../..
```

- Move the **Stuff.py** file to the **PythonCurrent** folder and give it the same name. Check and make sure the file has been moved. Copy the **ExtraP.txt** file to the **Python** folder in **TextFiles**.

- mv Stuff.py PythonCurrent

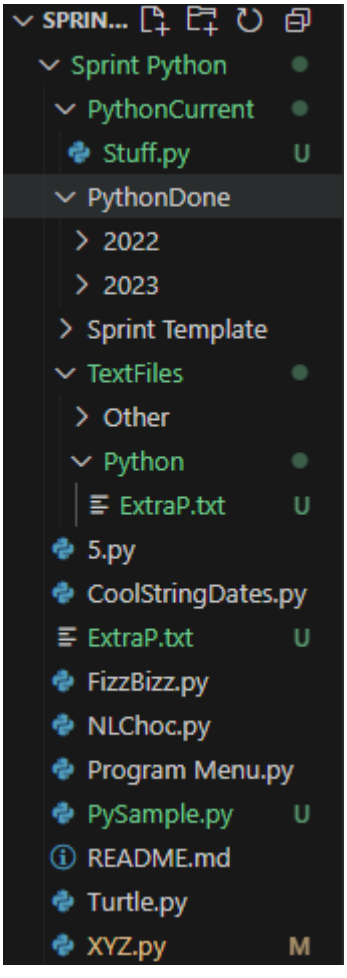
- cd PythonCurrent

- ls

- cd ..

- cp ExtraP.txt TextFiles/Python

```
BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ pwd
/c/Users/BM623/Desktop/Sprint Python

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ mv Stuff.py PythonCurrent

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ ls
 5.py   CoolStringDates.py   ExtraP.txt   FizzBizz.py   NLChoc.py  'Program Menu.py'   PySample.py   PythonCurrent/   PythonDone/   README.md  'Sprint Template'/   TextFiles/   Turtle.py   XYZ.py

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ cd PythonCurrent

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonCurrent (main)
$ ls
Stuff.py

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/PythonCurrent (main)
$ cd ..

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ cp ExtraP.txt TextFiles/Python

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$
```



- Change to the **Python** folder and display the first 6 lines of the file **ExtraP.txt**. Once complete move back to the main folder.

- cd TextFiles/Python

- head -n 6 ExtraP.txt

- cd ../..

```
BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ cd TextFiles/Python

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/TextFiles/Python (main)
$ head -n 6 ExtraP.txt
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP

BM623@Goliath MINGW64 ~/Desktop/Sprint Python/TextFiles/Python (main)
$ cd ../..

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$
```

- Display the contents of the **ExtraP.txt** file from the Python directory in **PythonDone** directory from the current location – you should be in the working directory.

  - cat TextFiles/Python/ExtraP.txt

```
BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ cat TextFiles/Python/ExtraP.txt
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP
Test text for ExtraP

Test text for ExtraP

Test text for ExtraP

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$
```

- From the working directory, find the file called **Stuff.py** searching all subdirectories.

- find . -name Stuff.py

```
Test text for ExtraP

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ find -name Stuff.py
./PythonCurrent/Stuff.py

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$
```

- Change the permissions in the PySample.py so that the owner, group, and everyone else has only read and execute permission.

- chmod 555 PySample.py / chmod u+rx,g+rx,o+rx PySample.py

```
BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ chmod 555 PySample.py
```

-I tried this but I could not get the executable permission to stick no matter what. Apparently it's due to how windows handles .exe files

- Find the text **getName** in a file and indicate which file(s) it is located –search all subdirectories. Do the same with the word **RetailCost** – in this case allow the search by ignoring case. Direct the output for one of these to a file called **FindResults.txt**. How could you send both results to the same file without overwriting it?

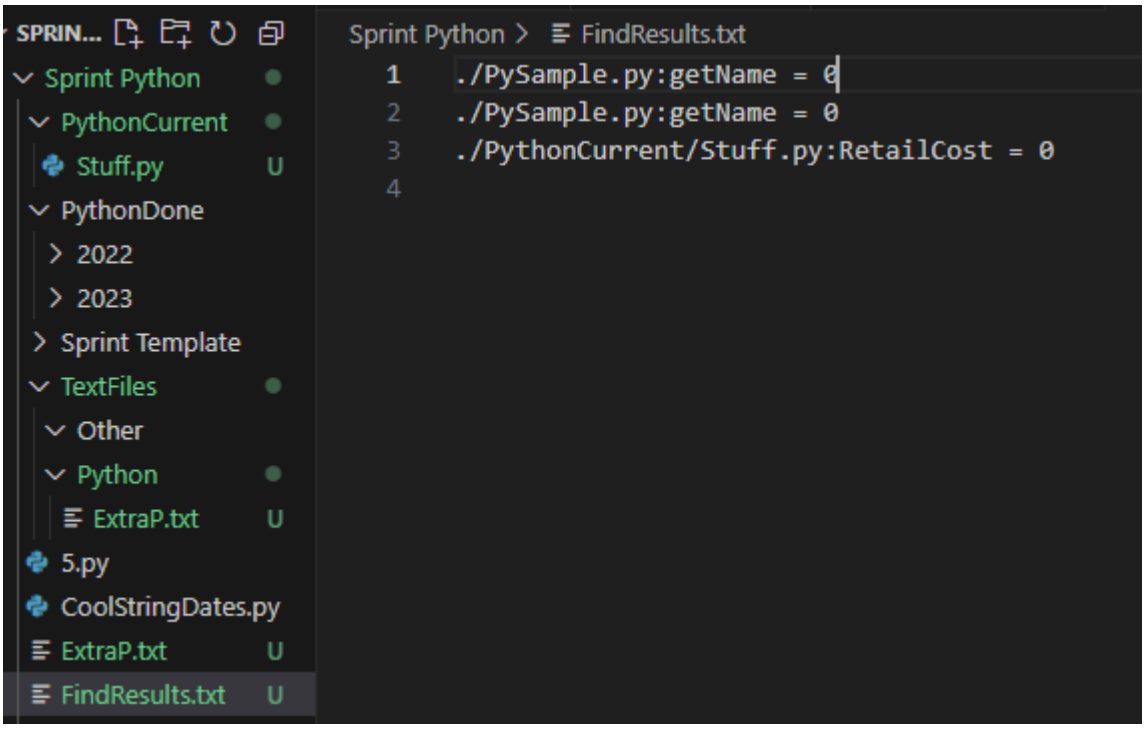
- grep -r "getName"

- grep -i -r "RetailCost"

- grep -r "getName" . > FindResults.txt

- grep -r "getName" . >> FindResults.txt

- grep -i -r "RetailCost" . >> FindResults.txt



- Perform one other command that you feel would be useful. Write it down with an explanation indicating what it is doing and why you feel it would be beneficial.

- du -sh

(du = Disk Usage)

(-s=summarize total space used by the specific directory)

(-h=makes it readable and more digestible to humans.)

```
BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ du -s
61651   .

BM623@Goliath MINGW64 ~/Desktop/Sprint Python (main)
$ du -sh
61M     .
```

Here we can see the Sprint Python domain I am in is 61651 bytes, but so we can understand it, the value is converted into a value we are familiar with, turning the result into 61 megabytes.

The “du -sh” command allows the user to see the estimated file space usage. When “-s” is used with “du” it summarizes the total space used by the directory you are currently on. When -h is added to -s, it makes the format presented more readable to humans. This is useful for quickly checking the size of a directory, or specific folder, and making it easy to understand so you can ensure you do not run out of storage. It is also a very helpful tool when trying to identify un-optimized files that are too large for the product.

Bash Part Evaluation Criteria:

- Commands: Use accurate Bash commands with screenshots of the outputs.
- Formatting: **Organize commands in a well-structured Word document and save it as a PDF before submission.**