ASSIGNMENT.

Q1. 1. Create the following files using VI editor

- Index.html -> this should display the header only using H1 tag.

52.144.46.236 - PuTTY

```
u95@evb1:~$ vi Index.html
```

52.144.46.236 - PuTTY

```
<!DOCTYPE html>
<html>
        <body>
                <h1>Hi this is header 1</h1>
                <h2>Hi this is header 2</h2>
                <h1>Hi this is header 1 again !!</h1>
                <h2>Hi this is header 2 again !!</h2>
        </body>
</html>
~
~
```

52.144.46.236 - PuTTY

```
u95@evb1:~$ grep h1 Index.html
                <h1>Hi this is header 1</h1>
                <h1>Hi this is header 1 again !!</h1>
u95@evb1:~$
```

- Create a new empty file called Index.css and Index.js

```
u95@evb1:~$ vi Index.js
u95@evb1:~$ vi Index.css
u95@evb1:~$
```

- Create a personclass.py file which creates a class in python.

```
u95@evb1:~$ vi pythonclass.py
```

- the python class Person should have data members PersonId, PersonName, Location.

52.144.46.236 - PuTTY

```
class person:
    def __init__(self) ->None:
        self.PersonId = 0
        self.PersonName = ""
        self.Location = ""
~
~
```

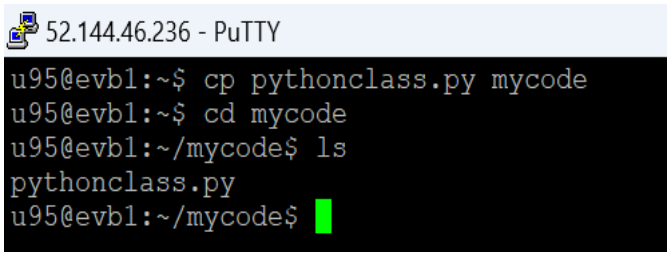
- create getter and setter methods for each of the fields.

```
@property
def PId(self):
    return self.PersonId
@PId.setter
def PId(self,value):
    self.PersonId = value

@property
def PName(self):
    return self.PersonName
@PName.setter
def PName(self,value):
    self.PersonName = value

@property
def location(self):
    return self.Location
@location.setter
def location(self,value):
    self.Location = value
```

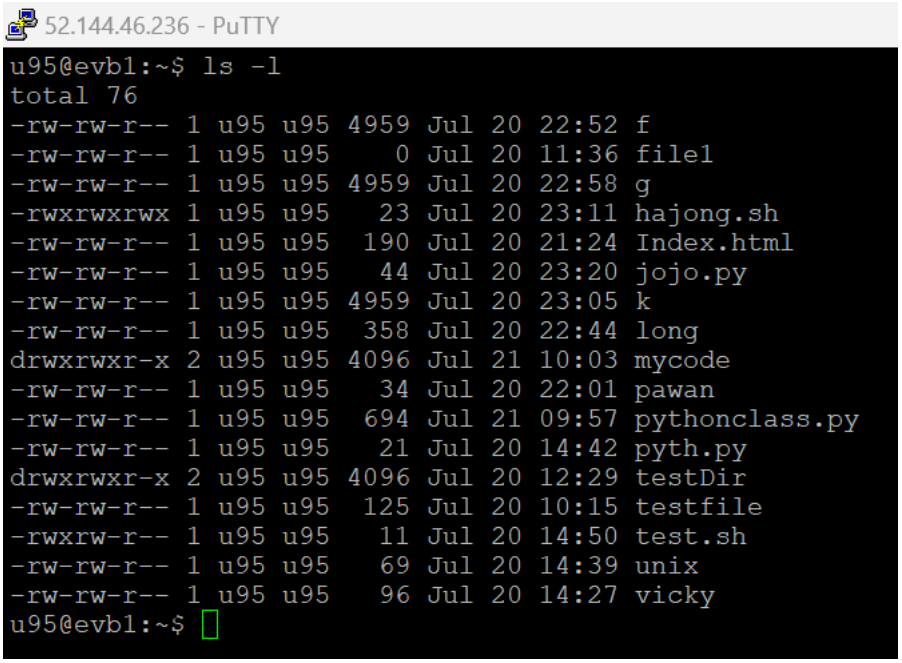
- write a method called show to print all the details.

```
def show(self):
   print("Person_Id : {1} Person_Name : {2} Location : {3}".format(self.PersonId,self.PersonName,self.location))
```

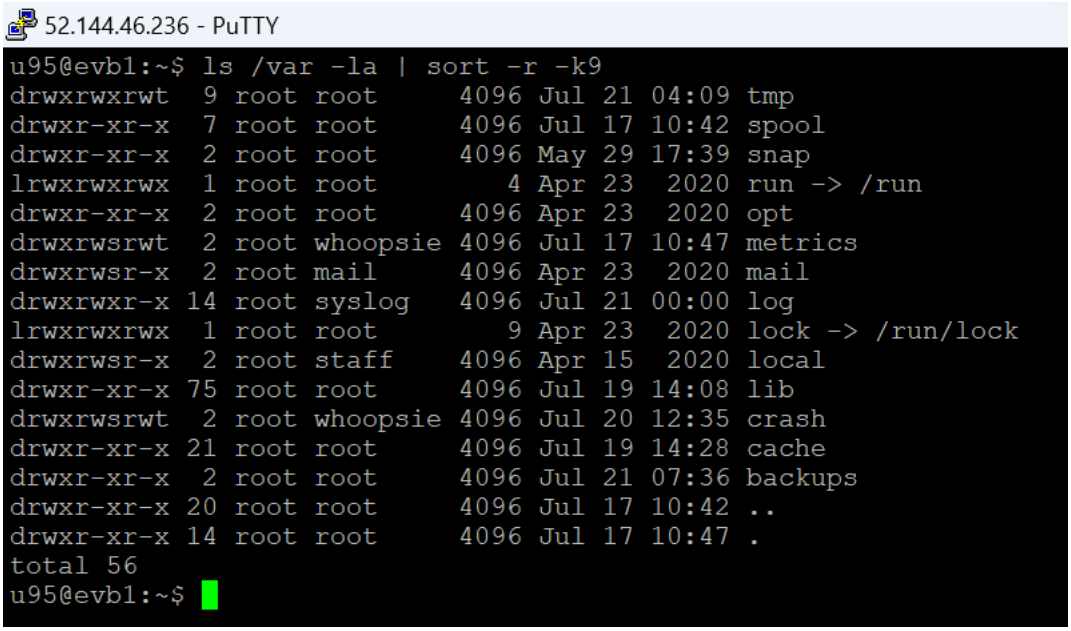
Q2. Copy the file personclass.py from the current directory to the mycode subdirectory.



Q3. List the files in the current directory, in “long listing format”.



Q4. List all files, including hidden files, in the /var directory, in reverse alphabetical order and long listing format. (Notice the slash in the directory!)



Q5. Rename the file personclass.py to person.py.

```
u95@evb1:~$ mv pythonclass.py person.py
```

Q6. Delete the files Index.html, Index.css and Index.js in a single command.

```
52.144.46.236 - PuTTY
u95@evb1:~$ rm Index.html Index.css Index.js
```

Q7. The cat command outputs the contents of a file to the terminal. The less command outputs the contents of a file to the terminal, page by page, pausing for you to press a key.

Use whichever command is best suited to display the contents of the file person.py.

```
u95@evb1:~$ cat person.py
class person:
    def __init__(self) ->None:
        self.PersonId = 0
        self.PersonName = ""
        self.Location = ""

    @property
    def PId(self):
        return self.PersonId
    @PId.setter
    def PId(self,value):
        self.PersonId = value

    @property
    def PName(self):
        return self.PersonName
    @PName.setter
    def PName(self,value):
        self.PersonName = value

    @property
    def location(self):
        return self.Location
    @location.setter
    def location(self,value):
        self.Location = value

    def show(self):
        print("Person_Id : {1} Person_Name : {2} Location : {3}".format(self.PersonId,self.PersonName,self.location))
u95@evb1:~$
```

The head and tail commands output only the first or last few lines (respectively) of a file to the terminal.

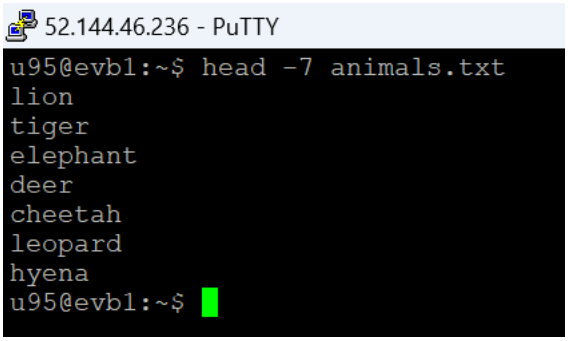
Head:

```
52.144.46.236 - PuTTY
u95@evb1:~$ head -5 person.py
class person:
    def __init__(self) ->None:
        self.PersonId = 0
        self.PersonName = ""
        self.Location = ""
u95@evb1:~$
```

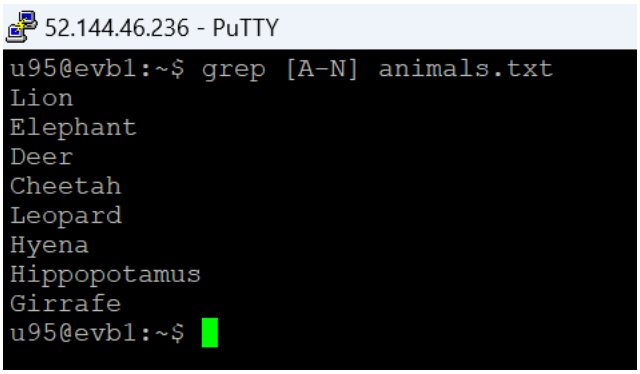
Tail:

```
u95@evb1:~$ tail -5 person.py
    def location(self,value):
        self.Location = value

    def show(self):
        print("Person_Id : {1} Person_Name : {2} Location : {3}".format(self.PersonId,self.PersonName,self.location))
```

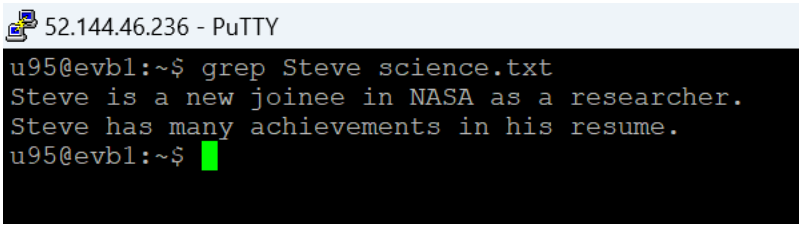
Display only the first 7 lines of the file animals.txt from the current directory on the terminal. Create the file if it does not exist and add some 10 animal names, one on each line.



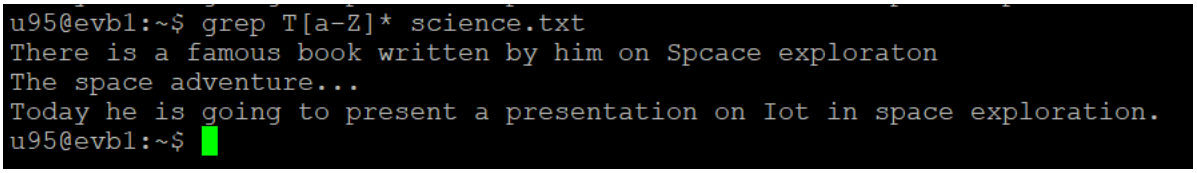
Q8. List all animals names that begin between [A-N].



Q9. List all the lines which contains "Steve" in the file science.txt.



Q10. List all the lines that begin with T in the file science.txt.



Q11. Make a copy of the file science.txt and move the copy into a new directory "Science" . Create the directory, if required.

```
u95@evbl:~$ cp science.txt Sci.txt
u95@evbl:~$ mkdir Science
u95@evbl:~$ mv Sci.txt ./Science
u95@evbl:~$
```

Q12. Show the first 15 lines from science.txt.

52.144.46.236 - PuTTY

```
u95@evbl:~$ head -15 science.txt
Steve is a new joinee in NASA as a researcher.
He has completed his B.tech from stanford University California.
Steve has many achievements in his resume.
From First ranker as a Coder in hackerrank to giving
contribution is spacex rocket program.
There is a famous book written by him on Spcace exploraton
The space adventure...
Today he is going to present a presentation on Iot in space exploration
Space exploration is the use of astronomy and space technology to explore
outer space.While the exploration space is carried out mainly
by astronomers with telescope,its physical exploration is conducted
both by uncrewed robotic space probes and human spaceflight.
pcae exploration, like its classical form astronomy, is
one of the main sources for space science.
while the observation of objects in space , known as
u95@evbl:~$
```

Q13. Write a python script to accept 4 numbers from the user, store the items into a list. Collect another input from the user for the multiples value. Based on the input multiples value, extract all elements that are a multiple of the input multiples value.

Input Numbers

1. 100

2. 56

3. 90

4. 77

Multiples value: 10

Output: 100, 90

Python Code Snippet:

52.144.46.236 - PuTTY

```
print("Input Numbers:")
li = []
for i in range (1,5,1):
    a = int(input("{0} : ".format(i)))
    li.append(a)

mul = int(input("Multiples value: "))
a = ""
for j in li:
    if(j%mul == 0):
        a+= str(j)+", "
print("Output : ",a)
~
```

Output:

52.144.46.236 - PuTTY

```
u95@evb1:~$ python3 q13.py
Input Numbers:
1 : 100
2 : 56
3 : 90
4 : 77
Multiples value: 10
Output :  100, 90,
u95@evb1:~$
```

SELF DISCOVERY:

1. Set the file MyProgram.java to have a last-modified date of January 1, 2020, 4:15am.

```
u95@evb1:~$ touch MyProgram.java -t 202001010415
u95@evb1:~$ ls -l MyProgram.java
-rw-rw-r-- 1 u95 u95 0 Jan  1  2020 MyProgram.java
u95@evb1:~$
```

2. You can use a * (asterisk) as a “wild-card” character to specify a group of files. For example, *foo means all files whose names end with foo, and foo* means all files whose names begin with foo. You can use a wildcard in the middle of a file name, such as foo*bar for all files that start with foo and end with bar.
   List all web page files (files whose names end with the extension .html or .css) in the current directory. Note that the ls command can accept more than one parameter for what files you want it to list (e.g. ls website/ python/).

```
u95@evb1:~$ ls *.html *.css
girrafe.css  hello.html  index.css  index.html  sol.html  solve.html  style.css
u95@evb1:~$
```

   Copy all the text files (files whose names end with .txt) from the current folder to the python subdirectory.

```
u95@evb1:~$ cp *.txt python
u95@evb1:~$ cd pyhton
```

```
u95@evb1:~/python$ ls
animals.txt  science.txt  ss.txt
u95@evb1:~/python$
```

----------END----------