Assignment_8_119

Data Structure, File Handling, and OOP

Program 1:

- **Prompt the user to enter 4 course scores, each out of 10.**
- **Append each score to a list.**
- **Pass the list to a function that:**
  - **Computes the total (out of 40) and percentage.**
  - **Returns a textual grade: Excellent, Very Good, Good, Pass, or Fail.**
- **Print the list, total, percentage, and grade.**

---

program 2

- **Create a program that:**
  1. **Opens file.txt in write mode and writes any content.**
  2. **Opens the same file in append mode and adds more content.**
  3. **Opens the file in read mode and prints its full contents to the screen.**

Program 3: Write this Rectangle Class then create two objects from this class

```
#%%
# User Defined data type
# self.__width  --> private data member (Accessed only inside the class)
# self._width --> protected data member (Accessed inside the class and on the protected)
# self.width   --> public   data member accessed inside and outside the class
```

```python
class Rectangle:
   # Constructor
    def __init__(self, w, l):
        if w> 0 and l>0:
            self.__width =w
            self.__length =l
        else:
            print("You must postive number")
            self.__width =w
            self.__length =0

    # Getters or Accessors
    def getWidth(self):
        return self.__width

    def getLength(self):
        return self.__length
```

```python
# Setters or Modifiers

def setWidth(self, w):
    if w > 0:
        self.__width =w
    else:
        print("You must postive number")
        self.__width =0

def setLength(self, l):
    if l > 0:
        self.__length =l
    else:
        print("You must postive number")
        self.__length =0

def setWidthLength(self, w,l):
    if w> 0 and l>0:
        self.__width =w
        self.__length =l
    else:
        print("You must postive number")
        self.__width =w
        self.__length =0
```

```
def Area(self):
    return self.__width * self.__length

def Perimeter(self):
    return 2 * (self.__width + self.__length)

def Display(self):
    print("--- Rectangle Details ----")
    print("Width = ", self.__width)
    print("Length = ", self.__length)
    print("Area = ", self.Area())
    print("Perimeter = ", self. Perimeter())
    print("========================")
```

```
# Calling
rect1= Rectangle(3,2)

rect1.Display()


w= float(input("Enter width: "))
l= float(input("Enter length: "))
rect2 = Rectangle(w,l)
rect2.Display()
```