Mohona Das

1905756

TT Lab-4

1) WAP in python that uses a class to store the name and marks of the students. Use list to store the marks in three subjects.

```
class students:
    def __init__(self, name):
        self.name = name
        self.marks = []

    def enterMarks(self):
        for i in range(3):
            m = int(input("Enter the marks:"))
            self.marks.append(m)

    def display(self):
        print (self.name, self.marks)

name = input("Enter the name of Student:")
s1 = students(name)
s1.enterMarks()
s1.display()
```

```
Enter the name of Student:Raj
Enter the marks:90
Enter the marks:67
Enter the marks:80
Raj [90, 67, 80]
```

2) WAP in python with class Employee that keeps a track of the number of employees in an organization and also stores their name, designation and salary details.

```
class emp:
    count = 0
    def __init__(self):
        self.name =""
        self.dsgn=""
        self.sal=""
        emp.count = emp.count + 1

    def getdata(self):
       self.name=input("Enter name ")
       self.dsgn=input("Enter designation ")
       self.sal=float(input("Enter salary "))

    def display(self):
        print (self.name,self.dsgn,self.sal)
l=list()
l.append(emp())
l.append(emp())
l.append(emp())
for e in l :
  e.getdata()
  e.display()
  print ("")

print("Count ",emp.count)
```

```
Enter name Mr. Ajay
Enter designation Asst. Professor
Enter salary 120000
Mr. Ajay Asst. Professor 120000.0

Enter name Mr. KL
Enter designation MD
Enter salary 150000
Mr. KL MD 150000.0

Enter name Ms JK
Enter designation CEO
Enter salary 4
Ms JK CEO 4.0

Count  3
```

3) WAP in python that has a class Person storing name and date of birth of a person. The program should subtract the date of birth from today’s date to find out that the person is eligible to vote or not.

```
class vote:
    def __init__(self):
        self.name =""
        self.age=0

    def getdata(self):
       self.name=input("Enter name ")
       self.age=int(input("Enter age "))

    def display(self):
      if self.age >= 18:
        print ("Eligible to vote ",self.name,self.age)
      else:
        print (self.name," is not eligible ",self.age)
```

```
v=vote()
v.getdata()
v.display()
```

```
Enter name Mohona
Enter age 21
Eligible to vote  Mohona  21
```

4) WAP in python that has a class Circle. Use a class variable that defines the value of constant Pie value=3.14. Use this class variable to calculate the area and circumference of the circle with specified radius.

```
class Circle():
    def __init__(self, r):
        self.radius = r

    def area(self):
        return self.radius**2*3.14

    def perimeter(self):
        return 2*self.radius*3.14

obj = Circle(8)
print(obj.area())
print(obj.perimeter())
```

```
200.96
50.24
```

5) WAP in python that has a class Student that stores the name ,roll no, marks(in three subjects). And display the information(name, roll no, total marks) stored about the students.

```
class student:

    def __init__(self):
        self.name =""
        self.roll=0
        self.marks=[]


    def getdata(self):
      self.name=input("Enter name ")
      self.roll=input("Enter roll ")
      for i in range(3):
        m = int(input("Enter the marks "))
        self.marks.append(m)

    def display(self):
        print (self.name,self.roll, self.marks, sum(self.marks))
l=list()
l.append(student())
l.append(student())

for e in l :
  e.getdata()
  e.display()
  print ("")
```

```
Enter name A
Enter roll 1
Enter the marks2
Enter the marks3
Enter the marks4
A 1 [2, 3, 4] 9

Enter name 4
Enter roll 2
Enter the marks4
Enter the marks5
Enter the marks5
4 2 [4, 5, 5] 14
```

6) WAP in python that has a class Rectangle with attribute length and breadth and a method area that returns the area of the circle.

```
class Rectangle:
    def __init__(self, l, w):
        self.length = l
        self.width  = w

    def rectangle_area(self):
        return self.length*self.width

l=int(input("Enter length "))
w=int(input("Enter breadth "))
obj = Rectangle(l,w)
print(obj.rectangle_area())
```

```
Enter length 5
Enter breadth 6
30
```

7) WAP in python that has a class Fraction with attributes numerator and denominator. Enter the values to the attributes and print the function in simplified forms.

```
from math import gcd
class Fraction:
  def __init__(self,n,d):
    self.n=n
    self.d=d

  def display(self):
    d = gcd(self.n, self.d);
    print("Numerator=",(self.n//d)," Denominator=",(self.d//d));

n=int(input("Enter numerator "))
d=int(input("Enter denominator "))
obj=Fraction(n,d)
obj.display()
```

```
Enter numerator4
Enter denominator16
Numerator= 1  Denominator= 4
```

8) WAP in python that has an abstract class Polygon. Derive two classes Rectangle and Triangle from polygon and write methods to get the details of their dimensions and hence calculate the area.

```
from abc import ABC, abstractmethod
class Polygon(ABC):
    @abstractmethod
    def get_area(self):
        pass
class Rectangle(Polygon):

    def __init__(self,l):
        self.x=l[0]
        self.y=l[1]

    def get_area(self):
        return self.x * self.y

class Triangle(Polygon):

    def __init__(self, l):
      self.a=l[0]
      self.b=l[1]
      self.c=l[2]

    def get_area(self):
        s = (self.a + self.b + self.c) / 2
        return (s*(s-self.a)*(s-self.b)*(s-self.c)) ** 0.5

t=Triangle([3.0, 4.0, 5.0])
print("Area of triangle=",t.get_area())

t=Rectangle([3.0, 4.0, 5.0])
print("Area of Rectangle=",t.get_area())
```

```
Area of triangle= 6.0
Area of Rectangle= 12.0
```

9) WAP in python with class Bill. The users have the option to pay the bill by card or by cash. Use the inheritance to model this situation

```
class Bill:
  def __init__(self,amt):
    self.amt=amt
  def func1(self):
    print("Amount is ::",self.amt)


class Cash(Bill):
  def __init__(self,amt):
    super().__init__(amt)
  def func2(self):
    print("Paid by cash")

class Card(Bill):
  def __init__(self,amt):
    super().__init__(amt)
  def func3(self):
    print("Paid by card")

# Driver's code
obj1 = Cash(1000)
obj1.func1()
obj1.func2()

obj1 = Card(10000)
obj1.func1()
obj1.func3()
```

```
Amount is :: 1000
Paid by cash
Amount is :: 10000
Paid by card
```

10) WAP in python has a class Person. Inherit a class Faculty from person which also have class publications.

```
class Person:
  def __init__(self,name):
    self.name=name

class Faculty(Person):
  def __init__(self,name,id,pname):
    super().__init__(name)
    self.id = id
    self.pub = self.Publication(pname)

  def show(self):
    print("Name:", self.name,"\nID : ",self.id)

  class Publication:
      def __init__(self,pname):
          self.name = pname


      def display(self):
          print("Publication Name:", self.name)


obj= Faculty("S. Arora",12345,"DN Publication")
obj.show()
g = obj.pub
g.display()
```

```
Name: S. Arora
ID :  12345
Publication Name: DN Publication
```

11) Write a Python program that has a class Store that keeps a record of code and price of each product. Display a menu of all products to the user and prompt him to enter the quantity of each item required. Generate a bill and display the total amount.

```
class Store:
  def getdata(self):
    self.name=input("Enter name ")
    self.q=int(input("Enter quantity "))
    self.code=(input("Enter Code "))
    self.price=float(input("Enter price "))

  def display(self):
    print(self.name," ",self.q," ",self.code," ",self.price)


l=list()
l.append(Store())
l.append(Store())
for e in l:
  e.getdata();

for e in l:
  e.display();

name=input("Enter name ")
quant=int(input("Enter quantity "))
for e in l:
  if name==e.name and quant<=e.q:
    print("Bill : " ,quant*e.price)
    e.q=e.q-quant
    break;
```

```
Enter name A
Enter quantity 12
Enter Code 12A
Enter price 12
Enter name B
Enter quantity 19
Enter Code 45C
Enter price 15
A   12   12A   12.0
B   19   45C   15.0
Enter name A
Enter quantity 3
Bill :  36.0
```

12) Write a Python program that has a class Numbers with values stored in a list. Write a class method to find the largest value.

```
class Numbers:
  def __init__(self):
    self.l=[1,2,3,4,5,6,8,33,-7465,123]
  def maxi(self):
    return max(self.l)

obj=Numbers();
print(obj.maxi())
```

```
123
```

13) Write a Python Program to create a class that stores a string and all its status details such as the number of uppercase characters, vowels and consonants, spaces etc.

```
class Count:
  def __init__ (self):
    self.s=""
    self.v = self.c = self.u = self.l= 0
  def getdata(self,s):
    self.s=s;
  def counts(self):
    for i in self.s:
      if(i == 'a' or i == 'e' or i == 'i' or i == 'o' or i == 'u'or i == 'A' or i == 'E' or i
         self.v = self.v +1
      else:
         self.c = self.c + 1
      if i.isupper() :
        self.u = self.u + 1

      if i.islower():
        self.l = self.l + 1

  def display(self):
    print(self.s," Vowels ",self.v," consonants ",self.c," lower case ",self.l," upper case "


obj=Count()
obj.getdata("World")
obj.counts()
obj.display()
```

```
World  Vowels  1  consonants  4  lower case  4  upper case  1
```

14) Write a Python Program using class and object to deposit or withdraw money in a bank account. If the balance is 0 -> Display as ”New Account Created” If the balance is credited with some amount-> Show the New Balance

```
class Bank_Account:
  def __init__(self,b=0):
    self.balance=b
    print("Hello!!! Welcome to the Deposit & Withdrawal Machine")

  def deposit(self):
    amount=float(input("Enter amount to be Deposited: "))
    self.balance += amount
    print("\n Amount Deposited:",amount," Total Balance :",self.balance)
  def withdraw(self):
    amount = float(input("Enter amount to be Withdrawn: "))
    if self.balance>=amount:
      self.balance-=amount
      print("\n You Withdrew:", amount," Total Balance:",self.balance)
    else:
      print("\n Insufficient balance ")

  def display(self):
    print("\n Net Available Balance=",self.balance)

s=Bank_Account()
s.display()
if s.balance == 0:
  print ("Account created")

s.deposit()
s.withdraw()

s1=Bank_Account(1000)
s1.display()
if s.balance == 0:
  print ("Account created")
s1.deposit()
s1.withdraw()
```

```
Hello!!! Welcome to the Deposit & Withdrawal Machine

 Net Available Balance= 0
Account created
Enter amount to be Deposited: 1000

 Amount Deposited: 1000.0  Total Balance : 1000.0
Enter amount to be Withdrawn: 1

 You Withdrew: 1.0  Total Balance: 999.0
Hello!!! Welcome to the Deposit & Withdrawal Machine

 Net Available Balance= 1000
Enter amount to be Deposited: 9000

 Amount Deposited: 9000.0  Total Balance : 10000.0
Enter amount to be Withdrawn: 12000

 Insufficient balance
```

15) Write a Program to print an index at which a particular value exists. If the value exists at multiple locations in the list, then prints all the indices. Also, count the number of times that value is

repeated in the list.

```
class Code:
  def __init__ (self,l):
    self.l=l.copy()

  def cal(self,n):
    c=0
    for i in self.l:
      if i==n:
        c+=1
        print(self.l.index(i))


    if c==0:
      print("Number doesn't exist in the list")

    else :
      print("The no of times ",n," occurs is ",c)

l=[2,43,21,5654,231,6545,213,11,456,65]
obj=Code(l.copy())
n=int(input("Enter  number to search "))
obj.cal(n)
```

```
Enter  number to search 2
0
The no of times  2  occurs is  1
```

16) Write a program that forms a list of the first character of every word in another list.

```
class Code:
  def __init__ (self,l):
    self.l=l.copy()

  def cal(self):
    li=list();
    for i in self.l:
      li.append(i[0])

    return li.copy()

l = ['cat', 'dog', 'rabbit', 'horse','dog']
obj=Code(l.copy())
li=obj.cal()
print(li)
```

```
['c', 'd', 'r', 'h', 'd']
```

17) Write a Python program that reads a file line by line. Each line read from the file is copied to another file with line numbers specified at the beginning of the line.

```
with open("/hi.txt") as f:
    with open("out.txt", "w") as f1:
        i=1
        for line in f:
            line=str(i)+":" +line
            f1.write(line)
            i+=1

#Original File Content
print("Original File Content")
with open("/hi.txt","r") as f:
    for l in f:
        print(l)


#Output File Content
print("Output File")
with open("out.txt","r") as f:
    for l in f:
        print(l)
```

```
Original File Content
Lorem ipsum dolor sit amet. Ea obcaecati eligendi et reprehenderit autem quo porro praes

sit quam consectetur? Ut repellat et commodi maxime eum dicta laboriosam 33 facilis odic

maxime ut porro autem sed molestiae delectus aut velit nobis et perferendis porro et eni

Est rerum reprehenderit eos nesciunt deleniti a dolores deserunt qui facere sunt qui lau

et consequuntur galisum. Et culpa quaerat et officia veritatis eos laborum minus cum rer
Output File
1:Lorem ipsum dolor sit amet. Ea obcaecati eligendi et reprehenderit autem quo porro pra

2:sit quam consectetur? Ut repellat et commodi maxime eum dicta laboriosam 33 facilis od

3:maxime ut porro autem sed molestiae delectus aut velit nobis et perferendis porro et e

4:Est rerum reprehenderit eos nesciunt deleniti a dolores deserunt qui facere sunt qui l

5:et consequuntur galisum. Et culpa quaerat et officia veritatis eos laborum minus cum r
```