Task 6

1. Write a python program for creating class Fraction with members numerator and denominator to represent rational numbers like 1/2 and -3/8. Create a constructor to initialize the objects and write the method for addition and multiplication of two fractions and display the result in proper fraction format.

Ans:

class Fact(object):

    def \_\_init\_\_(self,numerator=1,denominator=1):

        self.denominator=denominator

        self.numerator=numerator

    def add(self,f):

        if self.denominator==f.denominator :

            return (self.numerator + f.numerator),self.denominator

        else:

            n1=(self.numerator \* f.denominator)+(self.denominator \* f.numerator)

            n2=self.denominator \* f.denominator

            return n1,n2

    def mul(self,m):

        if self.denominator==m.denominator:

            return (self.numerator \* m.numerator),self.denominator

        else:

            n1=(self.numerator \* m.denominator)\*(self.denominator \* m.numerator)

            n2=self.denominator \* m.denominator

            return n1,n2

def main():

    obj1=Fact(1,4)

    obj2=Fact(1,4)

    n,d=obj2.add(obj1)

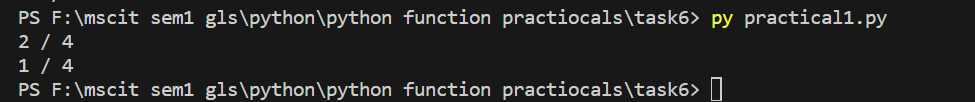
    print(n ,"/", d)

    m,l=obj2.mul(obj1)

    print(m,"/",l)

main()

Output:



1. Write a python program to create a class Room with members length and breadth. Initialize the attributes and find the area of the room and display the result

Ans:

class len(object):

    def \_\_init\_\_(self,length,width):

        self.length=length

        self.width=width

    def area(self):

        return self.length \* self.width

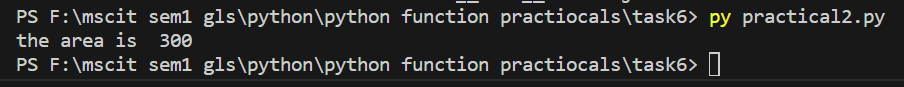
def main():

    obj2=len(30,10)

    print("the area is ",obj2.area())

main()

Output:



1. Create class point with members x and y. Create method for finding the euclidian distance between two points.

Ans:

import math

class point(object):

    def \_\_init\_\_(self,x,y):

        self.x=x

        self.y=y

    def ec(self,p):

        return math.sqrt(((self.x-p.x)\*\*2) +( (self.y-p.y)\*\*2) )

def main():

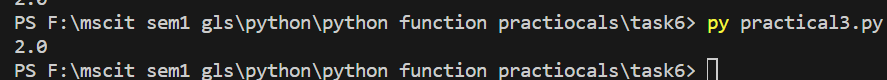
    obj1=point(5,2)

    obj2=point(3,2)

    print(obj2.ec(obj1))

main()

Output:



1. Write a python program to demonstrate the concept of default arguments. Create the employee class: name, department and basic\_salary. Write a function to to calculate total salary of the employee where total salary is sum of basic salary, DA and HRA. DA is 10% of basic salary and HRA is 15% of basic salary. If basic salary is missing, take default argument of basic salary as Rs. 9000

Ans:

class salary(object):

    def \_\_init\_\_(self,name,department,basic\_salary):

        self.name=name

        self.department=department

        self.basic\_salary=basic\_salary

    def total\_salary(self):

        self.da=self.basic\_salary \*0.1

        self.hra=self.basic\_salary \* 0.15

        total\_salary = self.basic\_salary + self.da +self.hra

        return total\_salary

def main():

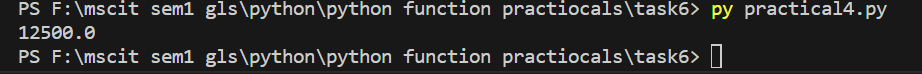
    obj1=salary(name='tejasv modi',department='MscIt',basic\_salary=10000)

    obj2=obj1.total\_salary()

    print(obj2)

main()

Output:



1. Write a python program to create class student with data name, roll no, sem, marks of 5 subject. Create a method to calculate percentage of the student.

Ans:

class mark(object):

    def \_\_init\_\_(self,name,sem,marks):

        self.name=name

        self.sem=sem

        self.marks=marks

    def mar(self):

        sum=0

        for i in self.marks:

            sum = sum+i

        return sum/len(self.marks)

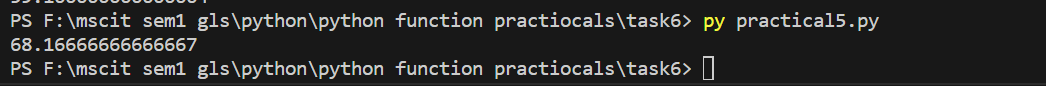
def main():

    obj1=mark(name='tejasv modi',sem=5,marks=[80,40,50,60,80,99])

    print(obj1.mar())

main()

Output:



1. Create class complex with members real and imaginary and create a method for addition and multiplication of two complex numbers.

Ans:

class complex(object):

    def \_\_init\_\_(self,r=0,i=0):

        self.r=r

        self.i=i

    def com(self,x,y):

        self.x = x.r + y.r

        self.y = x.i + y.i

        return complex(self.x,self.y)

    def mul(self,x1,y1):

        self.x = x1.r \* y1.r

        self.y = x1.i \* y1.i

        return complex(self.x,self.y)

def main():

    obj1=complex(10,20)

    obj2=complex(30,50)

    obj3=complex()

    obj4=complex()

    obj3=obj3.com(obj1,obj2)

    obj4=obj4.mul(obj1,obj2)

    print("thesum imaginary number",obj3.i)

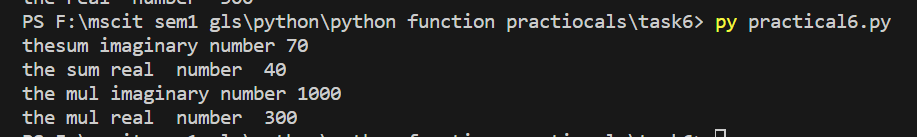
    print("the sum real  number ",obj3.r)

    print("the mul imaginary number",obj4.i)

    print("the mul real  number ",obj4.r)

main()

Output:



1. Create class date with members day, month and year. Create constructor for initialising the values. Write a function for validation of the date. If date is not valid display proper message

Ans:

class dat(object):

    def \_\_init\_\_(self,year,day,month):

        self.year=year

        self.day=day

        self.month=month

    def validate(self):

        if self.isday() and self.ismonth() and self.isYear():

            return "the date is right"

        else:

            return "the date is wrong"

    def ismonth(self):

        if self.month >=1 or self.month <13:

            return True

        else:

            return False

    def isYear(self):

        if self.year >=1 or self.year < 9999:

            return True

        else:

            return False

    def isday(self):

        match(self.month):

            case 1,3,5,7,8,10,12 :

                return self.day >=1 or self.day <32

            case 2:

                return self.day >=1 or self.day < 30

            case default:

                return self.day >=1 or self.day < 31

def main():

    obj1=dat(month=3,day=12,year=2003)

    print(obj1.validate())

main()

Output:

