

```
In [19]: #1.Create a class cal1 that will calculate sum of three numbers. Create
#setdata() method which has three parameters that contain numbers.
#Create display() method that will calculate sum and display sum.

class cal1:
    def setdata(self,a,b,c):
        self.a=a
        self.b=b
        self.c=c

    def display(self):
        sum=self.a+self.b+self.c
        print(sum)

a=cal1()
a.setdata(5,3,6)
a.display()
```

14

```
In [20]: #2. Create a class cal2 that will calculate area of a circle. Create set
data()
#method that should take radius from the user. Create area() method
#that will calculate area . Create display() method that will display ar
ea .

class cal2:
    def setdata(self,r):
        self.r=float(input("Enter the radias"))

    def area(self):
        self.area=3.14*3.14*self.r

    def display(self):
        print("Area of the circle is: ",self.area)

a=cal2()
a.setdata(4)
a.area()
a.display()
```

Enter the radias5
Area of the circle is: 49.298

```
In [23]: #3. Create a class cal3 that will calculate simple interest. Create
#constructor method which has three parameters .Create calInterest()
#method that will calculate Interest . Create display() method that will
#display Interest.

class cal3:
    def __init__(self,p,r,t):
        self.p=p
        self.r=r
        self.t=t

    def calInterest(self):
        self.Interest=self.p*self.r*self.t

    def display(self):
        print(self.Interest)

a = cal3(1000,3,1)
a.calInterest()
a.display()
```

3000

```
In [28]: #4. Create a class cal4 that will calculate square of a number. Create
#setdata() method which has one parameters that contain number.
#Create display() method that will calculate sum.(Function should
#return value)

class cal4:
    def setdata(self,a):
        self.a=a

    def display(self):
        self.area=self.a*self.a
        return self.area

a=cal4()
a.setdata(4)
a.display()
```

Out[28]: 16

```
In [41]: #5. Consider an employee class, which contains fields such as name and
#designation. And a subclass, which contains a field salary. Write a
#program for inheriting this relation.

class employee:
    def __init__(self,name,designation):
        self.name=name
        self.designation=designation

    def print(self):
        print(self.name,self.designation)

class salary(employee):
    def __init__(self,name,designation,salary):
        super().__init__(name,designation)
        self.salary=salary

a=salary("Romil","HeadProgrammer",200000)

a.print()
a.salary
```

Romil HeadProgrammer

Out[41]: 200000

```
In [44]: #6. Create a class cal5 that will calculate area of a rectangle. Create
#constructor method which has two parameters .Create calArea()
#method that will calculate area of a rectangle. Create display() method
#that will display area of a rectangle.

class cal5:

    def __init__(self,length,width):
        self.length=length
        self.width=width

    def calArea(self):
        self.area=self.length*self.width

    def display(self):
        print("Area=",self.area)

a=cal5(5,4)
a.calArea()
a.display()
```

Area= 20

```
In [1]: #7. Create a class cal6 that will calculate area of a square. Create set
data()
#method that should take length from the user. Create area() method
#that will calculate area . Create display() method that will display ar
ea .

class cal6:
    def setdata(self):
        self.l = float(input("Enter the lenght"))

    def area(self):
        self.area=self.l*self.l

    def display(self):
        print(self.area)

a=cal6()
a.setdata()
a.area()
a.display()
```

Enter the lenght4

16.0

```
In [3]: #8. Write a program with use of inheritance: Define a class publisher th
at
#stores the name of the title. Derive two classes book and tape, which
#inherit publisher. Book class contains member data called page no and
#tape class contain time for playing. Define functions in the appropriat
e
#classes to get and print the details.

class publisher:
    def title(self):
        self.title = input("Enter the name of the book")

    def print_title(self):
        print(self.title)

class book(publisher):
    def pages(self):
        self.pages = int(input("Enter the no of pages"))

    def print_pages(self):
        print(self.pages)

class time(book):
    def time(self):
        self.time = float(input("Enter the time required for playing"))

    def print_time(self):
        print(self.time)
```

a = time()
a.title()
a.print_title()
a.pages()
a.print_pages()
a.time()
a.print_time()

Enter the name of the bookAtomic Habits
Atomic Habits
Enter the no of pages233
233
Enter the time required for playing122
122.0

```
In [6]: #9. Create a class called scheme with scheme_id,
#scheme_name,outgoing_rate, and message_charge. Derive customer
#class form scheme and include cust_id, name and mobile_no
#data.Define necessary functions to read and display data.

class scheme:
    def scheme_info(self):
        self.scheme_id=int(input("Enter scheme id"))
        self.scheme_name=input("Enter scheme name")
        self.outgoing_rate=float(input("Enter outgoing rate"))
        self.message_charge=int(input("Enter message charges"))

class customer(scheme):
    def cust_info(self):
        self.cust_id=int(input("Enter customer id"))
        self.name=input("Enter customer name")
        self.mobile_no=int(input("Enter mobile number"))

    def display(self):
        print("scheme id:",self.scheme_id)
        print("scheme name:",self.scheme_name)
        print("outgoing rate:",self.outgoing_rate)
        print("Message charges:",self.message_charge)
        print("Customer id:",self.cust_id)
        print("Customer name:",self.name)
        print("Mobile number:",self.mobile_no)
```

a=customer()
a.scheme_info()
a.cust_info()
a.display()

Enter scheme id2
Enter scheme namexyz
Enter outgoing rate1233
Enter message charges12
Enter customer id3
Enter customer nameromil
Enter mobile number8767656564
scheme id: 2
scheme name: xyz
outgoing rate: 1233.0
Message charges: 12
Customer id: 3
Customer name: romil
Mobile number: 8767656564

```
In [18]: #10.Create a arith class. The class should have a parameterized construc
tor
#and methods to add, subtract and multiply two numbers and to return
#the answers

class arith:
    def __init__(self,num1,num2):
        self.num1=int(num1)
        self.num2=int(num2)

    def add(self):
        self.sum=self.num1+self.num2
        print(self.sum)

    def subtract(self):
        self.sub=self.num1-self.num2
        print(self.sub)

    def multiply(self):
        self.mul=self.num1*self.num2
        print(self.mul)
```

a=arith(5,4)
a.add()
a.subtract()
a.multiply()

9
1
20

In []: