

```
In [3]: #Square Numbers and Return Their Sum
class point:
    x = 0
    y = 0
    z = 0

    def __init__(self, x, y, z):
        self.x = x
        self.y = y
        self.z = z

    def display(self):
        print("First number = " + str(self.x))
        print("Second number = " + str(self.y))
        print("third number = " + str(self.z))

    def sqSum(self):
        answer = 0
        self.x = self.x * self.x
        self.y = self.y * self.y
        self.z = self.z * self.z
        answer = self.x + self.y + self.z
        return answer

print("hi")
p = point(1,3,5)
p.display()
print(p.sqSum())

hi
First number = 1
Second number = 3
third number = 5
35
```

```
In [4]: #Implement the claculator class:
class cal():
    def __init__(self,num1,num2):
        self.num1=num1
        self.num2=num2
    def add(self):
        return self.num1+self.num2
    def sub(self):
        return self.num1-self.num2
    def mul(self):
        return self.num1*self.num2
    def div(self):
        return self.num1/self.num2
num1=int(input("Enter the first number:"))
num2=int(input("Enter the second number:"))
obj=cal(num1,num2)
choice=1
while choice!=0:
    print("0.Exit")
    print("1.Add")
    print("2.Subtraction")
    print("3.Multiplication")
    print("4.division")
    choice=int(input("Enter choice:"))
    if choice==1:
        print("Result:",obj.add())
    elif choice==2:
        print("Result:",obj.sub())
    elif choice==3:
        print("Result:",obj.mul())
    elif choice==4:
        print("Result:",obj.div())
    elif choice==0:
        print("Exiting!")
    else:
        print("Invalid choice!")

Enter the first number:15
Enter the second number:20
0.Exit
1.Add
2.Subtraction
3.Multiplication
4.division
Enter choice:1
Result: 35
0.Exit
1.Add
2.Subtraction
3.Multiplication
4.division
Enter choice:2
Result: -5
0.Exit
1.Add
2.Subtraction
3.Multiplication
4.division
Enter choice:4
Result: 0.75
0.Exit
1.Add
2.Subtraction
3.Multiplication
4.division
Enter choice:3
Result: 300
0.Exit
1.Add
2.Subtraction
3.Multiplication
4.division
Enter choice:0
Exiting!
```

```
In [5]: # Implement the Complete Student Class
class student:
    name="bharat"
    rollNumber=1

    def getName(self):
        return self.name
    def getRollNumber(self):
        return self.rollNumber
    def setName(self,name):
        self.name=name
    def setRollNumber(self,rollNumber):
        self.rollNumber=rollNumber

obj1=student()
obj1.setName("Mayur")
obj1.setRollNumber(123)
print("My name is " + obj1.getName())
print("My Rollnumber is " + str(obj1.getRollNumber()))

My name is Mayur
My Rollnumber is 123
```

```
In [6]: #Implement a Banking Account
class Account:

    def __init__(self,title,balance):
        self.title=title
        self.balance=balance

    def disp_Acc_details(self):
        print(self.title,self.balance,self.interestRate)

class SavingsAccount(Account):

    def __init__(self,title,balance,interestRate):
        Account.__init__(self,title,balance)
        self.interestRate=interestRate
obj=SavingsAccount('Ashish',5000,5)
obj.disp_Acc_details()

Ashish 5000 5
```

```
In [20]: #Handling a Bank Account
class Account:
    def __init__(self, title, balance):
        self.title = title
        self.balance = balance

    def withdrawal(self, amount):
        self.balance-=amount

    def deposit(self,amount):
        self.balance+=amount

    def getBalance(self):
        return self.balance

class SavingsAccount(Account):
    def __init__(self,title,balance,interestRate):
        Account.__init__(self,title,balance)
        self.interestRate = interestRate

    def interestAmount(self):
        return (self.interestRate*self.balance)/100
```

```
In [21]: obj=SavingsAccount("Ashish", 2000,5)
```

```
In [22]: obj.getBalance()
```

Out[22]: 2000

```
In [24]: obj.deposit(500)
```

```
In [25]: obj.getBalance()
```

Out[25]: 2500

```
In [26]: obj.withdrawal(700)
```

```
In [27]: obj.getBalance()
```

Out[27]: 1800

```
In [28]: obj.interestAmount()
```

Out[28]: 90.0

```
In [ ]:
```