Assignments

Q1. Create class furniture. using constructor initialize height, width and color use display method display ().

Create another class chair which is inherited from furniture with the attribute legs and use display method to display all attributes.

```
->
Class furniture:
    def __init__(self):
        self. height = "8ft"
        self. width = 784.5
        self.colour = "Brown".
    def display (self):
        print ("Weight is:", self.height)
        print ("Width is:", self.width)
        print ("colour is:", self.colour)
class Chair (Furniture):
    legs = "fourlegs".
obj = Chair().
Obj.display
print(obj, legs)
```

Q2. Create class rectangle, using constructor initialize length and breadth find of rectangle using method.

```
-> from Area import area
Class Rectangle:
    def input(self):
    Self. length = int (input ("Enter length: ")
    Self. breadth = int (input ("Enter breadth: ")
    def display(self):
        print ("Area of Rectangle is: ", area(self.length, self.breadth))
obj = Rectangle()
obj. input ()
obj. display ()
```

Q3. Create class circle, using constructor initialize radius and find area and perimeter of circle.

```
->
From Area import circle.
    def input (self):
        self. radius = int (input ("Enter Radius": "))
    def display (self):
    print("Area of Circle is : ",Circle(self.radius))
obj = Circle()
obj. input()
obj. display ()
```

Q4. Create class the bank, using constructor take the values of account, name and balance. Create method deposit (self, amt) with one parameter amount to increase the balance, create another method withdraw (self, amt) with one parameter to decrease the balance. Create method Statement () to display all the attribute of class.

->

```
Class Bank:
    def __init__(self):
        Self. bal = 10000
        Self. account = int (input ("Account Number:")
        Self. name = input ("Account Holder Name:")
        Self. balance = print ("Available Balance:", self. bal)

    det deposit (self, amt = 0):
        self. bal = amt + self. bal
        print ("Currently Available Balance is ", self. bal)
    def withdraw (self, amt = 0):
        totalBalance = self.bal - amt
        print ("Currently Available balance: ",totalBalance
    def statement(self):
        b = int (input ("Enter Amount:"))
        a = input ("You want to depositor or withdraw money? / D(W)\n")
        if a == "D" or a == "d":
            obj.deposit(b)
        elseif a == "W" or a == "w":
            obj.withdraw(b)
        print (self.account, "\n", self.name, "\n", self.bal, "\n")
obj = Bank ()
obj.statement()
```