

## COURSE OUTCOME 4

**DATE: 3-12-2024**

1. Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two Rectangle objects by their area.

**PROGRAM:**

```
class Rectangle:
    def __init__(self,length,breadth):
        self.length=length
        self.breadth=breadth
    def area(self):
        return self.length * self.breadth

    def perimeter(self):
        return 2 * (self.length + self.breadth)

length1=int(input("Enter length of the Rectangle 1:"))
breadth1=int(input("Enter breadth of the Rectangle 1:"))
length2=int(input("Enter length of the Rectangle 2:"))
breadth2=int(input("Enter breadth of the Rectangle 2:"))
rect1=Rectangle(length1,breadth1)
rect2=Rectangle(length2,breadth2)
print("Area of the Rectangle 1:", rect1.area())
print("Perimeter of the Rectangle 1:", rect1.perimeter())
print("Area of the Rectangle 2:", rect2.area())
print("Perimeter of the Rectangle 2:", rect2.perimeter())
a=rect1.area()
b=rect2.area()
if a > b:
    print("Area of Recatangle 1 is greater")
elif b > a:
    print("Area of Rectangle 2 is greater")
else:
    print("Equal Area")
```

## OUTPUT:

Enter length of the Rectangle 1:12  
Enter breadth of the Rectangle 1:2  
Enter length of the Rectangle 2:21  
Enter breadth of the Rectangle 2:2  
Area of the Rectangle 1: 24  
Perimeter of the Rectangle 1: 28  
Area of the Rectangle 2: 42  
Perimeter of the Rectangle 2: 46  
Area of Rectangle 2 is greater

Enter length of the Rectangle 1:22  
Enter breadth of the Rectangle 1:3  
Enter length of the Rectangle 2:10  
Enter breadth of the Rectangle 2:1  
Area of the Rectangle 1: 66  
Perimeter of the Rectangle 1: 50  
Area of the Rectangle 2: 10  
Perimeter of the Rectangle 2: 22  
Area of Recatangle 1 is greater

Enter length of the Rectangle 1:10  
Enter breadth of the Rectangle 1:10  
Enter length of the Rectangle 2:10  
Enter breadth of the Rectangle 2:10  
Area of the Rectangle 1: 100  
Perimeter of the Rectangle 1: 40  
Area of the Rectangle 2: 100  
Perimeter of the Rectangle 2: 40  
Equal Area

**DATE: 5-12-2024**

2. Create a Bank account with members account number, name, type of account and balance. Write constructor and methods to deposit at the bank and withdraw an amount from the bank.

**PROGRAM:**

```
class Account:
    def __init__(self,ano,aname,atype,abalance):
        self.ano=ano
        self.aname=aname
        self.atype=atype
        self.abalance=abalance

    def deposit(self,amt):
        if amt>0:
            self.abalance+=amt
            print("Successfully deposited Rs.",amt)
        else:
            print("Invalid Amount!!")

    def withdraw(self,amt):
        if amt>self.abalance:
            print("Insufficient Balance!!")
        else:
            print("Successfully withdrawn Rs.", amt)
            self.abalance-=amt

    def viewdetails(self):
        print("Account Number:",self.ano)
        print("Account Name:",self.aname)
        print("Account Type:",self.atype)
        print("Account Balance: Rs.",self.abalance)

ano=int(input("Enter the Account Number:"))
aname=input("Enter the Account Name:")
```

```

atype=input("Enter the Account Type:")
abalance=int(input("Enter the Balance: Rs. "))

c1=Account(ano,aname,atype,abalance)
while True:
    print("*****MENU*****\n1.Deposit\n2.Withdraw\n3.Current
Balance\n4.View Details\n5.Exit")
    ch=int(input("Enter your choice:"))
    if ch==1:
        amt=int(input("Enter the amount to be deposited: Rs. "))
        c1.deposit(amt)
    elif ch==2:
        amt=int(input("Enter the amount to be withdrawn: Rs. "))
        c1.withdraw(amt)
    elif ch==3:
        print("Current Balance= Rs.",c1.abalance)
    elif ch==4:
        c1.viewdetails()
    elif ch==5:
        break
    else:
        print("Invalid choice!!")

```

### OUTPUT:

```

Enter the Account Number:2345
Enter the Account Name:Christina
Enter the Account Type:Savings
Enter the Balance: Rs.56748

```

```

*****MENU*****
1.Deposit
2.Withdraw
3.Current Balance
4.View Details
5.Exit

```

Enter your choice:1

Enter the amount to be deposited: Rs.2500

Successfully deposited Rs. 2500

\*\*\*\*\*MENU\*\*\*\*\*

1.Deposit

2.Withdraw

3.Current Balance

4.View Details

5.Exit

Enter your choice:2

Enter the amount to be withdrawn: Rs.2000

Successfully withdrawn Rs. 2000

\*\*\*\*\*MENU\*\*\*\*\*

1.Deposit

2.Withdraw

3.Current Balance

4.View Details

5.Exit

Enter your choice:3

Current Balance= Rs. 57248

\*\*\*\*\*MENU\*\*\*\*\*

1.Deposit

2.Withdraw

3.Current Balance

4.View Details

5.Exit

Enter your choice:4

Account Number: 2345

Account Name: Christina

Account Type: Savings

Account Balance: Rs. 57248

\*\*\*\*\*MENU\*\*\*\*\*

1.Deposit

2.Withdraw

3.Current Balance

4.View Details

5.Exit

Enter your choice:5

**DATE:7-12-2024**

3. Create a class Rectangle with private attributes length and width. Overload '<' operator to compare the area of 2 rectangles.

**PROGRAM:**

```
class Rectangle:
    def __init__(self,length,width):
        self.length=length
        self.width=width
    def area(self):
        return self.length*self.width
    def __lt__(self,other):
        return self.area() < other.area()
l1=int(input("Enter the length of Rectangle 1"))
w1=int(input("Enter the width of Reactangle 1"))
l2=int(input("Enter the length of Rectangle 2"))
w2=int(input("Enter the width of Reactangle 2"))
r1=Rectangle(l1,w1) #creating object r1
r2=Rectangle(l2,w2) #creating object r2
if r1<r2:
    print("Area of Rectangle 1 is Smaller that Area of rectangle 2")
elif r1>r2:
    print("Area of Rectangle 1 is larger than Area of Rectangle 2")
else:
    print("Both Rectangles have same Area")
```

**OUTPUT:**

```
Enter the length of Rectangle 1:12
Enter the width of Reactangle 1:23
Enter the length of Rectangle 2:45
Enter the width of Reactangle 2:67
Area of Rectangle 1 is Smaller that Area of rectangle 2
```

Enter the length of Rectangle 1:34

Enter the width of Rectangle 1:4

Enter the length of Rectangle 2:7

Enter the width of Rectangle 2:9

Area of Rectangle 1 is larger than Area of Rectangle 2

Enter the length of Rectangle 1:2

Enter the width of Rectangle 1:2

Enter the length of Rectangle 2:2

Enter the width of Rectangle 2:2

Both Rectangles have same Area



**DATE:4-12-2024**

4. Create a class Time with private attributes hour, minute and second. Overload '+' operator to find sum of 2 time.

### **PROGRAM**

```
class Time:
    def __init__(self,h=0,m=0,s=0):
        self.hours=h
        self.minutes=m
        self.seconds=s
    def __add__(self,other):
        t=Time()
        t.seconds=self.seconds+other.seconds
        t.minutes=self.minutes+other.minutes
        t.hours=self.hours+other.hours
        if t.seconds>=60:
            t.minutes+=t.seconds//60
            t.seconds%=60
        if t.minutes>=60:
            t.hours+=t.minutes//60
            t.minutes%=60
        return t
    def __str__(self):
        return "%d:%d:%d"%(self.hours,self.minutes,self.seconds)

h1=int(input("Enter Hour 1: "))
m1=int(input("Enter Minute 1: "))
s1=int(input("Enter Second 1: "))
t1=Time(h1,m1,s1)
h2=int(input("Enter Hour 2: "))
m2=int(input("Enter Minute 2: "))
s2=int(input("Enter Second 2: "))
t2=Time(h2,m2,s2)
tsum=t1 + t2
print(tsum)
```

## **OUTPUT**

Enter Hour 11

Enter Minute 11

Enter Second 11

Enter Hour 21

Enter Minute 21

Enter Second 21

2:2:2

Enter Hour 123

Enter Minute 123

Enter Second 145

Enter Hour 232

Enter Minute 245

Enter Second 277

56:10:2

**DATE:6-12-2024**

5. Create a class Publisher (name). Derive class Book from Publisher with attributes title and author. Derive class Python from Book with attributes price and no\_of\_pages. Write a program that displays information about a Python book. Use base class constructor invocation and method overriding.

### **PROGRAM**

```
class Publisher:
    def __init__(self,name):
        self.name=name
    def display():
        pass

class Book(Publisher):
    def __init__(self,name,title,author):
        super().__init__(name) #invoking the base class publisher
        self.title=title
        self.author=author
    def display():
        pass

class Python(Book):
    def __init__(self,name,title,author,price,pages):
        super().__init__(name,title,author) #invoking the base classes
        self.price=price
        self.pages=pages
    def display(self):
        print("\n**Book Details**")
        print("Publisher:",self.name)
```

```
        print("Title:",self.title)
        print("Author:",self.author)
        print("Price:",self.price)
        print("No. of Pages:",self.pages)
pub=input("Enter the Publisher Name:")
t=input("Enter the Title:")
a=input("Enter the Author Name:")
p=int(input("Enter the Price of the Book:"))
pa=int(input("Enter the no. of Pages:"))
b=Python(pub,t,a,p,pa)
b.display()
```

## **OUTPUT**

```
Enter the Publisher Name:Macmillan
Enter the Title:Jungle Book
Enter the Author Name:Rudyard Kipling
Enter the Price of the Book:550
Enter the no. of Pages:357
```

**\*\*Book Details\*\***

```
Publisher: Macmillan
Title: Jungle Book
Author: Rudyard Kipling
Price: 550
No. of Pages: 357
```