COURSE OUTCOME 4

DATE:03/12/2024

1. Create a 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)

print("Rectangle1")

length=int(input("enter the length"))

breadth=int(input("enter the breadth"))

rectangle1=Rectangle(length,breadth)

print("Rectangle2")

length=int(input("enter the length"))

breadth=int(input("enter the breadth"))
```

```
    rectangle2=Rectangle(length,breadth)
    print("Area",rectangle1.area())
    print("Perimeter:",rectangle1.perimeter())
    print("Area",rectangle2.area())
    print("Perimeter:",rectangle2.perimeter())
    a1=rectangle1.area()
    a2=rectangle2.area()
    print("compare the area of two rectangle")
    print("Rectangle1 Area:",a1)
    print("Rectangle2 Area:",a2)
    if a1> a2:
        print("Rectangle1 has a larger area.")
    elif a2 > a1:
       print("Rectangle2 has a larger area.")
    else:
       print("Both rectangles have the same
area.")
```

OUTPUT

```
Rectangle1
enter the length5
enter the breadth8
```

```
Rectangle2
enter the length9
enter the breadth3
Area 40
Perimeter: 26
Area 27
Perimeter: 24
compare the area of two rectangle
Rectangle1 Area: 40
Rectangle2 Area: 27
Rectangle1 has a larger area.

Rectangle1
enter the length4
enter the breadth3
Rectangle2
enter the length6
enter the breadth5
Area 12
Perimeter: 14
Area 30
Perimeter: 22
compare the area of two rectangle
Rectangle1 Area: 12
Rectangle2 Area: 30
Rectangle2 has a larger area.
```

DATE:05/12/2024

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,a_no,a_name,a_type,a_balance):
  self.a_no=a_no
  self.a_name=a_name
  self.a_type=a_type
  self.a_balance=a_balance

 def deposite(self,amt):
  if amt>0:
   self.a_balance +=amt
   print("succesfully deposited amount")
   print("New balance:₹",self.a_balance)
  else:
   print("Invalide amount")

 def withdraw(self,amt):
   if amt>self.a_balance:
    print("Insuffetient balance")
   else:
    print("Succesfully withdrawn amount")
    self.a_balance -=amt

 def viewdetails(self,amt):
  print("Account number:",self.a_no)
  print("Name:",self.a_name)
  print("Account type:",self.a_type)
  print("Account balance:₹",self.a_balance)
a_no=int(input("enter the account number:"))
a_name=input("enter the name:")
a_type=input("enter the type of account:")
a_balance=int(input("enter the balance:"))
c1=account(a_no,a_name,a_type,a_balance)
while True:
 print("Menu\n1.deposite\n2.Withdraw\n3.Current balance\n4.View
details\n5.Exit\n")
 ch=int(input("enter your choice"))
```

```
if ch==1:
 amt=int(input("enter the amount to be deposited"))
 c1.deposite(amt)
elif ch==2:
 amt=int(input("enter the amount to be withdraw"))
 c1.withdraw(amt)
elif ch==3:
 print("current balance=₹",c1.a_balance)
elif ch==4:
 c1.viewdetails(amt)
elif ch==5:
 print("Exiting...")
 break
```

OUTPUT

```
enter the account number:1010123
enter the name:Jake
enter the type of account:savings
enter the balance:1000
Menu
 1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice1
enter the amount to be deposited1
succesfully deposited amount
New balance:₹ 1001
Menu
 1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit
```

enter your choice3
current balance=₹ 1001
Menu
1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice2
enter the amount to be withdraw1000
Succesfully withdrawn amount Menu
1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice3
current balance=₹ 0
Menu
1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice4
Account number: 1010123
Name: Jake
Account type: savings
Account balance:₹ 0
Menu
1.deposite
2.Withdraw
3.Current balance
4.View details

5.Exit

enter your choice5
Exiting...

enter the account
number:10123 enter the
name:Tessa
enter the type of
account:savings enter the
balance:1000
Menu
1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice1
enter the amount to be
deposited10 succesfully deposited
amount New balance:₹ 1010
Menu
1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice3
current balance= ₹1010
Menu
1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice2
enter the amount to be withdraw1050
Insuffetient balance
Menu
1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice3
current balance=₹ 1010
Menu
1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice4
Account number: 10123
Name: Tessa
Account type: savings
Account balance:₹ 1010
Menu
1.deposite
2.Withdraw
3.Current balance
4.View details
5.Exit

enter your choice5
Exiting...

DATE:05/12/2024

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()
print("rectangle 1")
length=int(input("enter the length"))
width=int(input("enter the width"))
rectangle1=rectangle(length,width)
print("Area of rectangle1",rectangle1.area())
print("rectangle 2")
length=int(input("enter the length"))
width=int(input("enter the width"))
rectangle2=rectangle(length,width)
print("Area of rectangle2",rectangle2.area())
if rectangle1<rectangle2:
 print("area of rectangle1 is smaller than area of rectangle2")
elif rectangle1 > rectangle2:
 print("area of rectangle1 is larger than area of rectangle2")
else:
 print("Both rectangles have same area")
```

OUTPUT

```
rectangle 1
enter the length5
enter the width6
Area of rectangle1 30
rectangle 2
enter the length7
enter the width8
Area of rectangle2 56
area of rectangle1 is smaller than area of rectangle2
```

rectangle 1
enter the length9
enter the width5
Area of rectangle1 45
rectangle 2
enter the length3
enter the width7
Area of rectangle2 21
area of rectangle2 is smaller than area of rectangle1 58

DATE:09/12/2024

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,hour,minute,second):
  self.hour=hour
  self.minute=minute
  self.second=second

 def sum(self,other):
  tot_sec = self.second + other.second
  tot_min = self.minute + other.minute + tot_sec // 60
  tot_hr = self.hour + other.hour + tot_min // 60
  tot_sec %= 60
  tot_min %= 60
  return time(tot_hr,tot_min,tot_sec)

 def __add__(self,other):
  return self.sum(other)
print("Time1")
hour=int(input("enter the hour"))
minute=int(input("enter the minute"))
 second=int(input("enter the second"))
time1=time(hour,minute,second)
print("Time2")
hour=int(input("enter the hour"))
minute=int(input("enter the minute"))
 second=int(input("enter the second"))
time2=time(hour,minute,second)

t3=time1+time2
print("sum of time:"+str(t3.hour)+":"+str(t3.minute)+":"+str(t3.second))
```

OUTPUT

```
Time1
enter the hour3
```

enter the minute25
enter the second30
Time2
enter the hour4
enter the minute25
enter the second30
sum of time:7:51:0

Time1
enter the hour7
enter the minute30
enter the second50
Time2
enter the hour6
enter the minute40
enter the second20
sum of time:14:11:10

DATE:05/12/2024

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)
  self.title=title
  self.author=author
 def display():
  pass
class python(book):
 def__init__(self,name,title,author,price,no_pages:
  super().__init__(name,title,author)
  self.price=price
  self.no_pages=no_pages
 def display(self):
  print("title\n",self.title)
  print("publisher name\n",self.name)
  print("author\n",self.author)
  print("Price of book\n",self.price)
  print("Number of pages\n",self.no_pages)
name=input("Enter the name")
```

```
title=input("Enter the title")
author=input("Enter the author")
price=int(input("Enter the price"))
no_pages=int(input("Enter the number of pages"))
b=python(name,title,author,price,no_pages)
b.display()
```

OUTPUT

```
Enter the nametessa
Enter the titlesky
Enter the authoranu
Enter the price150
Enter the number of pages100
title sky
publisher name tessa
 author anu
Price of book 150
Number of pages 100
Enter the name Jake
Enter the title water
Enter the author Johns O
Enter the price 200
Enter the number of pages150
title water
publisher name Jake
 author Johns O
Price of book 200
Number of pages 150
```