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):
        area= self.length*self.breadth
        print("\nArea of Rectangle  is", area)
        return area

    def perimeter(self):
        perimeter=2*(self.length+self.breadth)
        print("Perimeter of Rectangle is",perimeter)



l = int(input("\nEnter the length of the first rectangle: "))
b = int(input("Enter the breadth of the first rectangle: "))

rect1 = Rectangle(l,b)
```

```
a=rect1.area()
rect1.perimeter()

l = int(input("Enter the length of the second rectangle: "))
b = int(input("Enter the breadth of the second rectangle: "))

rect2 = Rectangle(l,b)
b=rect2.area()
rect2.perimeter()

if a < b:
   print("\n Rectangle 1 has a smaller area than Rectangle 2.")
elif a == b:
   print("\n Both rectangles have the same area.")
else:
   print("\n Rectangle 1 has a larger area than Rectangle 2.")
```

OUTPUT

```
Enter the length of the first rectangle: 5
Enter the breadth of the first rectangle: 4

Area of Rectangle  is 20
Perimeter of Rectangle is 18

Enter the length of the second rectangle: 4
Enter the breadth of the second rectangle: 6
```

```
Area of Rectangle  is 24
Perimeter of Rectangle is 20

Rectangle 1 has a smaller area than Rectangle 2.
```

OUTPUT

```
Enter the length of the first rectangle: 5
Enter the breadth of the first rectangle: 4

Area of Rectangle  is 20
Perimeter of Rectangle is 18

Enter the length of the second rectangle: 3
Enter the breadth of the second rectangle: 5

Area of Rectangle  is 15
Perimeter of Rectangle is 16

Rectangle 1 has a larger area than Rectangle 2.
```

DATE:22-10-2024

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 BankAccount:
   def __init__(self, number, name, atype, balance=0):
      self.number = number
      self.name = name
      self.atype = atype
      self.balance = balance

   def deposit(self, amt):
      if amt > 0:
         self.balance += amt
         print("Successfully deposited amount")
      else:
         print("Invalid amount")

   def withdraw(self, amt):
      if amt > self.balance:
         print("Insufficient balance")
      else:
         print("Successfully withdrawn amount")
         self.balance -= amt
```

```
    def view_details(self):
        print("Name:", self.name)
        print("Number:", self.number)
        print("Type:", self.atype)
        print("Balance:", self.balance)

name =  input("Enter the Name:")
number =  int(input("Enter the Number:"))
atype =  input("Enter the Type:")
balance =  int(input("Enter the Balance:"))

customer = BankAccount(number, name, atype, balance)

while True:
    print("\n....menu.......\n")
    print("1) Deposit")
    print("2) Withdraw")
    print("3) Current balance")
    print("4) View details")
    print("5) Exit")

    ch = int(input("Enter your choice: "))
    if ch == 1:
        amt = int(input("Enter the amount to deposit: "))
        customer.deposit(amt)
    elif ch == 2:
        amt = int(input("Enter the amount to withdraw: "))
        customer.withdraw(amt)
    elif ch == 3:
        print("Current Balance:", customer.balance)
    elif ch == 4:
        customer.view_details()
    elif ch == 5:
        break
    else:
        print("Invalid choice. Please try again.")
```

OUTPUT

```
Enter the Name: Nirmal
Enter the Number:123
Enter the Type: Savings
Enter the Balance:2000

....menu.......

1) Deposit
2) Withdraw
3) Current balance
4) View details
5) Exit
Enter your choice: 1
Enter the amount to deposit: 2000
Successfully deposited amount
```

```
 ....menu.......

1) Deposit
2) Withdraw
3) Current balance
4) View details
5) Exit
Enter your choice: 3
Current Balance: 4000

....menu.......

1) Deposit
2) Withdraw
3) Current balance
4) View details
5) Exit
Enter your choice: 2
Enter the amount to withdraw: 1500
Successfully withdrawn amount

....menu.......

1) Deposit
2) Withdraw
3) Current balance
4) View details
```

```
5) Exit
Enter your choice: 4
Name: Nirmal
Number: 123
Type: Savings
Balance: 2500

....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()

leng=int(input("Enter the length :"))
widt=int(input("Enter the width :"))
rectangle1=Rectangle(leng,widt);
leng=int(input("Enter the length :"))
widt=int(input("Enter the width :"))
rectangle2=Rectangle(leng,widt);
if rectangle1 < rectangle2:
   print("Area of recatangle 1 is smallerthan area of rectangle 2")
elif rectangle1 > rectangle2:
   print("Area of recatangle 1 is largerthan area of rectangle 2")
```

```
else:
   print("Both rectangle has same area")
```

OUTPUT

```
Enter the length :4
Enter the width :5
Enter the length :6
Enter the width :5
Area of recatangle 1 is smallerthan area of rectangle 2
```

OUTPUT

```
Enter the length :5
Enter the width :8
Enter the length :2
Enter the width :3
Area of recatangle 1 is largerthan area of rectangle 2
```

DATE:5-12-2024

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 __add__(self, other):
        second = self.second + other.second
        minute = self.minute + other.minute + second // 60
        hour = self.hour + other.hour + minute // 60
        return Time(hour % 24, minute % 60, second % 60)

    def display(self):
        print("Time:",self.hour,self.minute,self.second)

s=int(input("Enter second:"));
m=int(input("Enter minute:"));
h=int(input("Enter hour:"));
time1 = Time(h, m, s)
s=int(input("Enter second:"));
m=int(input("Enter minute:"));
```

```
h=int(input("Enter hour:"));
time2 = Time(h, m, s)

result = time1 + time2
result.display()
```

OUTPUT

```
Enter second:45
Enter minute:60
Enter hour:5
Enter second:56
Enter minute:34
Enter hour:9

Time: 15 35 41
```

OUTPUT

```
Enter second:34
Enter minute:23
Enter hour:12
Enter second:34
Enter minute:21
Enter hour:23

Time: 11 45 8
```

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 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, nopage):
        super().__init__(name, title, author)
        self.price = price
        self.nopage = nopage

    def display(self):
        print("Name:", self.name)
        print("Title:", self.title)
        print("Author:", self.author)
        print("Price:", self.price)
        print("NO of Pages:", self.nopage)

name=input("Enter the Name :")
title=input("Enter the Title :")
author=input("Enter the Author :")
price=int(input("Enter the Price :"))
nopage=int(input("Enter the No of pages :"))
book=Python(name, title, author, price, nopage)
book.display()
```

OUTPUT

```
Enter the Name :O'Reilly Media
Enter the Title : Learning Python
Enter the Author :Mark Lutz
Enter the Price :50
Enter the No of pages :1000
Name: O'Reilly Media
Title:  Learning Python
Author: Mark Lutz
```

```
Price: 50
NO of Pages: 1000
```

OUTPUT

```
Enter the Name : Pearson
Enter the Title : Python for Data Science
Enter the Author : John Smith
Enter the Price :40
Enter the No of pages :350
Name: Pearson
Title:  Python for Data Science
Author: John Smith
Price: 40
NO of Pages: 350
```

COURSE OUTCOME 5

DATE:7-11-2024

1. Write a Python program to read a file line by line and store it into a list.

PROGRAM

```
f=open("file.txt","r")
l=[i.split() for i in open("file.txt")]
print(l)
f.close()
```

file.txt

```
Hello! Welcome to demofile.txt
This file is for testing purposes.
Good Luck!
```

OUTPUT

```
[['Hello!', 'Welcome', 'to', 'demofile.txt'], ['This', 'file', 'is', 'for', 'testing',
'purposes.'], ['Good', 'Luck!']]
```

DATE:8-11-2024

2. Python program to copy odd lines of one file to other

PROGRAM

```
with open("file.txt", "r") as x:
    with open("file4.txt", "w") as y:
        line_number = 1
        for line in x:
            if line_number % 2 != 0:
                y.write(line)
            line_number += 1
x.close()
y.close()
s=open("file4.txt","r")
print(s.read())
```

file.txt

```
Hello! Welcome to demofile.txt
This file is for testing purposes.
Good Luck!
```

OUTPUT

```
Hello! Welcome to demofile.txt
Good Luck!
```

DATE:22-10-2024

3. Write a Python program to read each row from a given csv file and print a list of strings.

PROGRAM

```
import csv
with open("student.csv","r") as f:
    csvr=csv.reader(f)
    for row in csvr:
            print(row)
```

student.csv

```
roll,name,age,course
101, Rasim,21,mca
102,Farook,21,,mca
103,Aswin,24,mca
104,nirmal ,21,mca
105,arun,21,mca
106,midhun,21,mca
107,amal,21,mca
```

OUTPUT

```
['roll', 'name', 'age', 'course']
['101', 'rasim', '21', 'mca']
['102', 'Farook', '21', '', 'mca']
['103', 'Aswin', '24', 'mca']
['104', 'nirmal', '21', 'mca']
```

```
['105', 'arun', '21', 'mca']
['106', 'midhun', '21', 'mca']
['107', 'amal', '21', 'mca']
```

DATE:15-11-2024

4. Write a Python program to read specific columns of a given CSV file and print the content of the columns

PROGRAM

```
import csv


data = {
   'Name': ['Farook', 'Aswin', 'Vishnu'],
   'Age': [25, 30, 22],
   'depart': ['Mca', 'Bca', 'Mba']
}


with open('output.csv', 'w') as file:
   writer = csv.DictWriter(file, fieldnames=data.keys())
   writer.writeheader()
   writer.writerow(data)


print("Dictionary written to CSV file 'output.csv'.")


with open('output.csv','r') as file:
   reader = csv.DictReader(file)
   for row in reader:
      print(row)
```

OUTPUT

```
Dictionary written to CSV file 'output.csv'.

{'Name': "['Farook', 'Aswin', 'Vishnu']", 'Age': '[25, 30, 22]', 'depart': "['Mca',
'Bca', 'Mba']"}
```

DATE:16-11-2024

5. Write a Python program to write a Python dictionary to a csv file. After writing the CSV file read the CSV file and display the content.

PROGRAM

```
import csv
columns_to_read = ['Name', 'City']


with open("dictionary.csv","r") as file:
   csv_reader = csv.DictReader(file)
   for row in csv_reader:
      selected_data = {column: row[column] for column in
columns_to_read}
      print(selected_data)
```

dictionary.csv

```
Name,Age,City,Occupation
rasim,30,New York,Engineer
nirmal,25,london,Designer
naji,35,america,Teacher
```

OUTPUT

```
{'Name': 'rasim', 'City': 'New York'}
{'Name': 'nirmal', 'City': 'london'}
{'Name': 'naji', 'City': 'america'}
```