**COURSE OUTCOME 4**

**Date: 27/11/2023**

**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)**

**def compare\_area(self, other\_rectangle):**

**if self.area() > other\_rectangle.area():**

**return "The first rectangle has a larger area."**

**elif self.area() < other\_rectangle.area():**

**return "The second rectangle has a larger area."**

**else:**

**return "Both rectangles have the same area."**

**print("RECTANGLE 1 \n")**

**length = int(input("Enter length: "))**

**breadth = int(input("Enter breadth: "))**

**rectangle1 = Rectangle(length, breadth)**

**print("\nArea = ", rectangle1.area())**

**print("Perimeter = ", rectangle1.perimeter())**

**print("\n\n RECTANGLE 2 \n")**

**length = int(input("Enter length: "))**

**breadth = int(input("Enter breadth: "))**

**rectangle2 = Rectangle(length, breadth)**

**print("\nArea = ", rectangle2.area())**

**print("Perimeter = ", rectangle2.perimeter())**

**comparison\_result = rectangle1.compare\_area(rectangle2)**

**print("\nComparison of areas: ", comparison\_result)**

**Output:**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$ python3 co4\_1.py**

**RECTANGLE 1**

**Enter length: 5**

**Enter breadth: 3**

**Area = 15**

**Perimeter = 16**

**RECTANGLE 2**

**Enter length: 4**

**Enter breadth: 2**

**Area = 8**

**Perimeter = 12**

**Comparison of areas: The first rectangle has a larger area.**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$**

**COURSE OUTCOME 4**

**Date: 27/11/2023**

**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, account\_number, account\_holder\_name, account\_type, balance=0):**

**self.account\_number = account\_number**

**self.account\_holder\_name = account\_holder\_name**

**self.account\_type = account\_type**

**self.balance = balance**

**def deposit(self, amount):**

**if amount > 0:**

**self.balance = self.balance + amount**

**print("\nDeposition Successful!")**

**else:**

**print("\nInvalid amount!")**

**def withdraw(self, amount):**

**if 0 < amount < self.balance:**

**self.balance = self.balance - amount**

**print("Withdrawal successful")**

**print("New Balance : ",self.balance)**

**elif amount > self.balance:**

**print("Not possible to withdraw. Insufficient funds.")**

**else:**

**print("Invalid amount!")**

**def get\_balance(self):**

**return self.balance**

**account1 = BankAccount("250783", "Manuel", "Savings", 20000)**

**print("current balance:",account1.get\_balance())**

**deposit\_amount = float(input("Enter the deposit amount: "))**

**account1.deposit(deposit\_amount)**

**withdrawal\_amount = float(input("Enter the withdrawal amount: "))**

**account1.withdraw(withdrawal\_amount)**

**account1.get\_balance()**

**Output:**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$ python3 co4\_2.py**

**current balance: 20000**

**Enter the deposit amount: 1000**

**Deposition Successful!**

**Enter the withdrawal amount: 2000**

**Withdrawal successful**

**New Balance : 19000.0**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$**

**COURSE OUTCOME 4**

**Date: 29/11/2023**

**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()**

**a=int(input("Enter the length of rectangle1:"))**

**b=int(input("Enter the width of rectangle1:"))**

**x=int(input("Enter the length of rectangle2:"))**

**y=int(input("Enter the length of rectangle2:"))**

**rectangle1 = Rectangle(a,b)**

**rectangle2 = Rectangle(x,y)**

**if rectangle1 < rectangle2:**

**print("Area of Rectangle 1 is smaller than the area of Rectangle 2.")**

**elif rectangle1 > rectangle2:**

**print("Area of Rectangle 1 is larger than the area of Rectangle 2.")**

**else:**

**print("Both rectangles have the same area.")**

**Output:**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$ python3 co4\_3.py**

**Enter the length of rectangle1:3**

**Enter the width of rectangle1:2**

**Enter the length of rectangle2:4**

**Enter the length of rectangle2:2**

**Area of Rectangle 1 is smaller than the area of Rectangle 2.**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$**

**COURSE OUTCOME 4**

**Date: 29/11/2023**

**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=0, minute=0, second=0):**

**self.\_hour = hour # Private attribute**

**self.\_minute = minute # Private attribute**

**self.\_second = second # Private attribute**

**def \_\_add\_\_(self, other):**

**total\_seconds = self.\_hour \* 3600 + self.\_minute \* 60 + self.\_second + \**

**other.\_hour \* 3600 + other.\_minute \* 60 + other.\_second**

**new\_hour, remainder = divmod(total\_seconds, 3600)**

**new\_minute, new\_second = divmod(remainder, 60)**

**return Time(new\_hour, new\_minute, new\_second)**

**def \_\_str\_\_(self):**

**return f"{self.\_hour:02d}:{self.\_minute:02d}:{self.\_second:02d}"**

**a=int(input("Enter the hour of time1:"))**

**b=int(input("Enter the minute of time1:"))**

**c=int(input("Enter the second of time1:"))**

**x=int(input("Enter the hour of time2:"))**

**y=int(input("Enter the minute of time2:"))**

**z=int(input("Enter the second of time2:"))**

**time1 = Time(a,b,c)**

**time2 = Time(x,y,z)**

**sum\_time = time1 + time2**

**print("Time 1:", time1)**

**print("Time 2:", time2)**

**print("Sum of Time 1 and Time 2:", sum\_time)**

**Output:**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$ python3 co4\_4.py**

**Enter the hour of time1:3**

**Enter the minute of time1:23**

**Enter the second of time1:40**

**Enter the hour of time2:4**

**Enter the minute of time2:21**

**Enter the second of time2:12**

**Time 1: 03:23:40**

**Time 2: 04:21:12**

**Sum of Time 1 and Time 2: 07:44:52**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$**

**COURSE OUTCOME 4**

**Date: 29/11/2023**

**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**

**class Book(Publisher):**

**def \_\_init\_\_(self, name, title, author):**

**super().\_\_init\_\_(name)**

**self.title = title**

**self.author = author**

**def display\_info(self):**

**print("Publisher:", self.name)**

**print("Title:", self.title)**

**print("Author:", self.author)**

**class Python(Book):**

**def \_\_init\_\_(self, name, title, author, price, no\_of\_pages):**

**super().\_\_init\_\_(name, title, author)**

**self.price = price**

**self.no\_of\_pages = no\_of\_pages**

**def display\_info(self):**

**super().display\_info()**

**print("Price:", self.price)**

**print("Number of Pages:", self.no\_of\_pages)**

**publisher=input("Enter the name of publisher:")**

**book\_name=input("Enter the name of book:")**

**author=input("Enter the name of author:")**

**book\_price=input("Enter the price of book:")**

**book\_pages=input("Enter the number of pages of the book:")**

**python\_book = Python("publisher", "book\_name", "author", book\_price, book\_pages)**

**python\_book.display\_info()**

**Output:**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$ python3 co4\_5.py**

**Enter the name of publisher:h&c**

**Enter the name of book:hellen keller**

**Enter the name of author:shakespeare**

**Enter the price of book:455**

**Enter the number of pages of the book:980**

**Publisher: publisher**

**Title: book\_name**

**Author: author**

**Price: 455**

**Number of Pages: 980**

**mits@mits-HP-280-Pro-G6-Microtower-PC:~/Desktop/sahala$**