Rajalakshmi Engineering College

Name: KRITHESHWARAN R
Email: 241901049@rajalakshmi.edu.in
Roll no: 241901049
Phone: 9843565002
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

2024_28_III_OOPS Using Java Lab

REC_2028_OOPS using Java_Week 7_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

Maria, an online store owner, is looking to implement a pricing system that calculates the final price of products after applying discounts. She needs a program that takes the original price of a product and the discount percentage as input and computes the final discounted price. The discount is applied as a percentage of the original price. Maria wants to ensure that the final price is formatted to display exactly two decimal places.

Implement this functionality using the PriceCalculator interface and the DiscountCalculator class.

Input Format

The first line of input consists of the original price (a double value).

The second line of input consists of a discount percentage (a double value).

Output Format

The output displays the final price after the discount, adhering to the following format: "Final Price after discount: $[final_price]".

Here, [final_price] should be replaced with the calculated final price, formatted as a currency value with two decimal places.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 100.0
10.0

Output: Final Price after discount: $90.00

Answer

```
import java.util.Scanner;

// You are using Java
interface PriceCalculator{
   double calculatePrice(double originalPrice,double discount);
}
class DiscountCalculator implements PriceCalculator{
   public double calculatePrice(double originalPrice,double discount){
      return originalPrice - (originalPrice * (discount/100));
   }
}

class Main {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      double originalPrice = scanner.nextDouble();
      double discount = scanner.nextDouble();
      PriceCalculator calculator = new DiscountCalculator();
      double finalPrice = calculator.calculatePrice(originalPrice, discount);
      System.out.printf("Final Price after discount: $%.2f%n", finalPrice); //
Formats output to 2 decimal places
   }
}
```

Status : Correct ***Marks : 10/10***

2. Problem Statement:

Rathish is planning a road trip and needs a program to convert speeds between miles per hour (MPH) and kilometers per hour (KPH).

Create an interface, SpeedConverter, with a method convertSpeed(double mph). Implement the interface with MPHtoKPHConverter class, allowing Rathish to input MPH and receive the converted speed in KPH, rounded to two decimal points.

Formula: Speed in KPH = 1.60934 * Speed in MPH.

Input Format

The input consists of a single double-point number representing the speed in miles per hour (MPH).

Output Format

The output displays the converted speed (double-point number) in kilometers per hour (KPH) rounded off to two decimal points in the following format:

"Speed in KPH: <<converted speed>>".

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 1.0
Output: Speed in KPH: 1.61

Answer

```
import java.util.Scanner;

// You are using Java
interface SpeedConverter{

    double convertSpeed(double speedInMPH);
```

```
}
class MPHtoKPHConverter implements SpeedConverter{

    public double convertSpeed(double speedInMPH){
        return speedInMPH * 1.60934;
    }
}
class SpeedConversionApp {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        double speedInMPH = scanner.nextDouble();

        SpeedConverter converter = new MPHtoKPHConverter();

        double speedInKPH = converter.convertSpeed(speedInMPH);

        System.out.printf("Speed in KPH: %.2f\n", speedInKPH);

        scanner.close();
    }
}
```

Status : Correct **Marks : 10/10**

3. Problem Statement

Alex and Bob are designing a control system for household appliances, and one of the appliances is a washing machine. You want to create a program to help them that models the washing machine as a motor and calculates its electricity consumption based on its capacity.

Define an interface named Motor with the following methods:

void run() double consume(double capacity)

Create a class called WashingMachine that implements the Motor interface.

In the WashingMachine class:

Implement the run() method to print "Washing machine is running."Implement a consume() method to print "Washing machine is consuming electricity."Implement the consume(double capacity) method to calculate the electricity consumption (in kWh) of the washing machine based on its capacity. The formula for electricity consumption is (capacity * 0.05).

Input Format

The input consists of a double value representing the capacity of the washing machine in kW.

Output Format

The first line of output prints "Washing machine is running."

The second line prints "Washing machine is consuming electricity."

The third line prints "Electricity consumption: X kWh" where X is a double value, rounded off to two decimal places, representing the electricity consumption.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 2.5
Output: Washing machine is running.
Washing machine is consuming electricity.
Electricity consumption: 0.13 kWh

Answer

```
import java.util.Scanner;

// You are using Java
interface motor {
    void run();
    void consume();
    double consume(double capacity);
}
class WashingMachine implements motor{
```

```
    public void run(){
        System.out.println("Washing machine is running.");
    }
    public void consume(){
        System.out.println("Washing machine is consuming electricity.");
    }
    public double consume(double capacity){
        return capacity * 0.05;
    }
}

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        WashingMachine washingMachine = new WashingMachine();

        double capacity = scanner.nextDouble();

        washingMachine.run();
        washingMachine.consume();


        double consumption = washingMachine.consume(capacity);
        System.out.printf("Electricity consumption: %.2f kWh" ,consumption);

        scanner.close();
    }
}
```

Status : Correct **Marks : 10/10**

4. Problem Statement

John is developing a car loan calculator and has structured his program using two interfaces, Principal and InterestRate, defining methods for principal and interest rate retrieval.

The Loan class implements these interfaces, taking principal and annual interest rates as parameters. User input is solicited for these values, and the program ensures their validity before performing calculations. If input

values are invalid (less than or equal to zero), an error message is displayed.

Note: Total interest = principal * interest rate * years

Input Format

The first line of input consists of a double value P, representing the principal.

The second line consists of a double value R, representing the annual interest rate.

The third line consists of an integer value N, representing the loan duration in years.

Output Format

If the input values are valid, print "Total interest paid: Rs. " followed by a double value, representing the total interest paid, rounded off to two decimal places.

If the input values are invalid (negative or zero values for principal, annual interest rate, or loan duration), print "Invalid input values!".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 20000.00
0.05
5
Output: Total interest paid: Rs.5000.00

Answer

```
import java.util.Scanner;

// You are using Java
//Type your code here
interface Principal{

}
interface InterestRate{
```

```
}
class Loan implements Principal,InterestRate{
    public double carPrice;
    public double AnnualInterestRate;
    public int loanDuration;

    public Loan(double carPrice,double AnnualInterestRate){
        this.carPrice = carPrice;
        this.AnnualInterestRate = AnnualInterestRate;
    }
    public double calculateTotalInterest(int loanDuration){
        return carPrice * AnnualInterestRate * loanDuration;
    }
}
public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        double carPrice = scanner.nextDouble();

        double annualInterestRate = scanner.nextDouble();

        int loanDuration = scanner.nextInt();

        if (carPrice <= 0 || annualInterestRate <= 0 || loanDuration <= 0) {
            System.out.println("Invalid input values!");
            return;
        }

        Loan carLoan = new Loan(carPrice, annualInterestRate);
        double totalInterest = carLoan.calculateTotalInterest(loanDuration);

        System.out.printf("Total interest paid: Rs.%.2f%n", totalInterest);
    }
}
```

Status : Correct ***Marks : 10/10***