Rajalakshmi Engineering College

Name: HARI PRASANNA S
Email: 241501064@rajalakshmi.edu.in
Roll no: 241501064
Phone: 9042038178
Branch: REC
Department: AI & ML - Section 2
Batch: 2028
Degree: B.E - AI & ML

Scan to verify results

2024_28_III_OOPS Using Java Lab

REC_2028_OOPS using Java_Week 6_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

Arun wants to calculate the age gap between the grandfather and the son and determine the father's age after 5 years.

Your task is to assist him in developing a program using three classes: GrandFather, Father, and Son, where the GrandFather stores the grandfather's age, the Father extends GrandFather to include the father's age and calculates his age after 5 years, and Son extends Father to include the son's age and calculate the age difference between the grandfather and the son.

Input Format

The input consists of three integers representing the ages of the grandfather, father, and son, one per line.

Output Format

The first line of output prints "Grandfather and son's age gap:" followed by an integer representing the age gap between the grandfather and the son, ending with "years".

The second line prints "Father's Age:" followed by an integer representing the father's age after 5 years, ending with "years".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 50
30
3
Output: Grandfather and son's age gap: 47 years
Father's Age: 35 years

Answer

```
import java.util.Scanner;

class GrandFather{
   protected int grandfatherAge;
   public void setGrandfatherAge(int grandfatherAge){
      this.grandfatherAge=grandfatherAge;
   }
}
class Father extends GrandFather{
   protected int FatherAge;
   public void setFatherAge(int FatherAge){
      this.FatherAge=FatherAge;
   }
   public int calculateFatherAgeAfter5Years(){
      return FatherAge+=5;
   }
}
class Son extends Father{
   protected int SonAge;
   public void setSonAge(int SonAge){
      this.SonAge=SonAge;
```

```
    }
    public int calculateGrandfatherSonAgeDifference(){
        return grandfatherAge-SonAge;
    }
}

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

        int grandfatherAge = scanner.nextInt();
        son.setGrandfatherAge(grandfatherAge);

        int fatherAge = scanner.nextInt();
        son.setFatherAge(fatherAge);

        int sonAge = scanner.nextInt();
        son.setSonAge(sonAge);

        System.out.println("Grandfather and son's age gap: "+
son.calculateGrandfatherSonAgeDifference() + " years");

        int fatherAgeAfter5Years = son.calculateFatherAgeAfter5Years();
        System.out.println("Father's Age: " + fatherAgeAfter5Years + " years");
    }
}
```

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

2. Problem Statement

Adams has a reputation company with a great number of employees. He must calculate the salary weekly according to the hourly rate and working hours. Create a program to define a class Employee with attributes name and hourly rate. Create a subclass HourlyEmployee that calculates the weekly salary based on the number of hours worked.

(The first 40 hours are based on the regular hour rate. If the work hours are greater than 40 then the work wage is 1.5 times the hourly rate)

Note: Use Math(Math.max, Math.min) functions .

Example

Input:

Chris

10

45

Output:

Weekly Salary: Rs.475.00

Explanation:

Calculation:

The first 40 hours are paid normally: 40 × 10 = 400.00The extra 5 hours are paid at 1.5 times the hourly rate: 5 × (10×1.5) = 5 × 15 = 75.00Total salary: 400.00 + 75.00 = 475.00

Input Format

The first line of input consists of a string that represents the name of the employee.

The second line consists of a double value that represents the rate for an hour.

The last line consists of an integer that represents the total hours worked.

Output Format

The output displays the total salary of the employee, where salary is rounded to two decimal places in the format: "Weekly Salary: Rs.<double value>".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: Dave
10.0

40
Output: Weekly Salary: Rs.400.00

Answer

```
import java.util.Scanner;
import java.text.DecimalFormat;

class Employee{
   protected String name;
   protected double hourlyRate;
   public Employee(String name,double hourlyRate){
      this.name=name;
      this.hourlyRate=hourlyRate;
   }
}
class HourlyEmployee extends Employee{
   protected int hoursWorked;
   public HourlyEmployee(String name,double hourlyRate,int hoursWorked){
      super(name,hourlyRate);
      this.hoursWorked=hoursWorked;
   }
   public double calculateWeeklySalary(){
      if (hoursWorked<=40){
         return hoursWorked*hourlyRate;
      }
      else{
         return (40*hourlyRate + (hoursWorked-40)*hourlyRate*1.5);
   }
}}
public class Main {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);

      String name = scanner.nextLine();
      double hourlyRate = scanner.nextDouble();
      int hoursWorked = scanner.nextInt();

      HourlyEmployee employee = new HourlyEmployee(name, hourlyRate,
hoursWorked);

      double weeklySalary = employee.calculateWeeklySalary();
      DecimalFormat df = new DecimalFormat("#.00");
```

```
        String formattedSalary = df.format(weeklySalary);
        System.out.println("Weekly Salary: Rs." + formattedSalary);
        scanner.close();
    }
}
```

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

3. Problem Statement

Teena's retail store has implemented a Loyalty Points System to reward customers based on their spending. The program calculates and displays the loyalty points based on whether the customer is a regular or a premium customer.

For regular customers (class Customer), the loyalty points are calculated as:

Loyalty points = amount spent / 10

For premium customers (class PremiumCustomer, which inherits from Customer), the loyalty points are calculated as:

Loyalty points = 2 * (amount spent / 10)

The program should use method overriding for premium customers to calculate their loyalty points. The method that needs to be overridden is calculateLoyaltyPoints in the Customer class.

Input Format

The first line of input consists of an integer representing the amount spent by the customer.

The second line consists of a string representing the premium customer status:

- "yes" if the customer is a premium customer.
- "no" if the customer is not a premium customer.

Output Format

The output should display the loyalty points earned based on the amount spent and the customer type.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 50
yes
Output: 10

Answer

```
import java.util.Scanner;

// You are using Java
class Customer {
    protected int amountSpent;
    public int calculateLoyaltyPoints(int amountSpent){
        return amountSpent/10;
    }
}

class PremiumCustomer extends Customer {
    public int calculateLoyaltyPoints(int amountSpent){
        return 2*(amountSpent/10);
    }
}
public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        int amountSpent = scanner.nextInt();

        String isPremium = scanner.next().toLowerCase();

        Customer customer;

        if (isPremium.equals("yes")) {
            customer = new PremiumCustomer();
        } else {
            customer = new Customer();
        }
```

```
        int loyaltyPoints = customer.calculateLoyaltyPoints(amountSpent);

        System.out.println(loyaltyPoints);
    }
}
```

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

4. Problem Statement

Bob has been tasked with creating a program using CircleUtils class to calculate and display the circumference and area of the circle.

The program should allow Bob to input the radius of a circle as both an integer and a double and compute both the circumference and area of the circle using separate overloaded methods:

calculateCircumference- To calculate the circumference using the formula 2 * 3.14 * radiuscalculateArea- To calculate the area 3.14 * radius * radius

Write a program to help Bob.

Input Format

The first line of input consists of an integer m, representing the radius of the circle as a whole number.

The second line consists of a double value n, representing the radius of the circle as a decimal number.

Output Format

The first line of output displays two space-separated double values, rounded to two decimal places, representing the circumference of the circle with the integer radius and the double radius, respectively.

The second line displays two space-separated double values, rounded to two decimal places, representing the area of the circle with the integer radius and the double radius, respectively.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
3.50

Output: 31.40 21.98
78.50 38.47

Answer

```
import java.util.Scanner;

class CircleUtils{
    public double calculateCircumference(int radius){
        return (2*3.14*radius);
    }
    public double calculateCircumference(double radius){
        return 2*3.14*radius;
    }
    public double calculateArea(int radius){
        return 3.14*radius*radius;
    }
    public double calculateArea(double radius){
        return 3.14*radius*radius;
    }
}
class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        int radiusInt = scanner.nextInt();
        double radiusDouble = scanner.nextDouble();

        CircleUtils circleUtils = new CircleUtils();

        double circumferenceInt = circleUtils.calculateCircumference(radiusInt);
        double circumferenceDouble =
circleUtils.calculateCircumference(radiusDouble);
        double areaInt = circleUtils.calculateArea(radiusInt);
        double areaDouble = circleUtils.calculateArea(radiusDouble);
```

```
        System.out.format("%.2f %.2f\n", circumferenceInt, circumferenceDouble);
        System.out.format("%.2f %.2f", areaInt, areaDouble);

        scanner.close();
    }
}
```

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