# Level 1 Practice Programs

1. Write a program to input the Principal, Rate, and Time values and calculate Simple Interest.

**Hint =>**

1. Simple Interest = Principal \* Rate \* Time / 100
2. Take user input for principal, rate, time
3. Write a method to calculate the simple interest given principle, rate and time as parameters
4. Output “The Simple Interest is \_\_\_ for Principal \_\_\_, Rate of Interest \_\_\_ and Time \_\_\_”

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

double Principle=input.nextDouble(),Rate=input.nextDouble(),Time=input.nextDouble();

System.out.printf("The Simple Interest is %.3f for Principal %.3f, Rate of Interest %.3f and Time %.3f ", SimpleInterest(Principle,Rate,Time), Principle, Rate, Time);

}

public static double SimpleInterest(double P, double R, double T){

return (P\*T\*R)/100;

}

}

1. Create a program to find the maximum number of handshakes among N number of students.

**Hint =>**

1. Get integer input for number of students
2. Use the combination = (n \* (n - 1)) / 2 formula to calculate the maximum number of possible handshakes.
3. Write a method to use the combination formulae to calculate the number of handshakes

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int num=input.nextInt();

System.out.println(MaxCombinations(num));

}

public static int MaxCombinations(int n){

return n\*(n-1)/2;

}

}

1. Create a program to find the maximum number of handshakes among N number of students.

**Hint =>**

1. Get integer input for numberOfStudents variable.
2. Use the combination = (n \* (n - 1)) / 2 formula to calculate the maximum number of possible handshakes.
3. Display the number of possible handshakes.

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int num=input.nextInt();

System.out.println(MaxCombinations(num));

}

public static int MaxCombinations(int n){

return n\*(n-1)/2;

}

}

1. An athlete runs in a triangular park with sides provided as input by the user in meters. If the athlete wants to complete a 5 km run, then how many rounds must the athlete complete

**Hint =>**

1. Take user input for 3 sides of a triangle
2. The perimeter of a triangle is the addition of all sides and rounds is distance/perimeter
3. Write a Method to compute the number of rounds user needs to do to complete 5km run

import java.util.Scanner;

public class AthleteRun {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.print("Enter the length of side 1 (in meters): ");

double side1 = input.nextDouble();

System.out.print("Enter the length of side 2 (in meters): ");

double side2 = input.nextDouble();

System.out.print("Enter the length of side 3 (in meters): ");

double side3 = input.nextDouble();

int rounds = calculateRounds(side1, side2, side3);

if (rounds > 0) {

System.out.println("The athlete needs to complete " + rounds + " rounds to run 5 km.");

} else {

System.out.println("Invalid triangle sides! Please enter valid lengths.");

}

input.close();

}

public static int calculateRounds(double side1, double side2, double side3) {

double perimeter = side1 + side2 + side3;

if (side1 <= 0 || side2 <= 0 || side3 <= 0 || (side1 + side2 <= side3) || (side1 + side3 <= side2) || (side2 + side3 <= side1)) {

return -1; // Invalid triangle

}

double totalDistance = 5000; // 5 km = 5000 meters

return (int) Math.ceil(totalDistance / perimeter);

}

}

1. Write a program to check whether a number is positive, negative, or zero.

**Hint =>** Get integer input from the user. Write a Method to return -1 for negative number, 1 for positive number and 0 if number is zero

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int num=input.nextInt();

System.out.println(CheckNo(num));

}

public static String CheckNo(int n){

if (n<0){

return "Negative";

}

else if(n>0){

return "Positive";

}

else {

return "0";

}

}

}

1. Write a program SpringSeason that takes two int values month and day from the command line and prints “Its a Spring Season” otherwise prints “Not a Spring Season”.

**Hint =>** Spring Season is from March 20 to June 20. Write a Method to check for Spring season and return a boolean true or false

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int day=input.nextInt(),month=input.nextInt();

System.out.println(CheckSpring(day,month));

}

public static String CheckSpring(int day,int month){

if ((month == 3 && day >= 20) || (month == 4) || (month == 5) || (month == 6 && day <= 20)) {

return "Spring Season";

}

return "Not a spring season";

}

}

1. Write a program to find the sum of n natural numbers using loop

**Hint =>** Get integer input from the user. Write a Method to find the sum of n natural numbers using loop

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int num=input.nextInt();

System.out.println(SumOfN(num));

}

public static int SumOfN(int n){

return n\*(n+1)/2;

}

}

1. Write a program to find the smallest and the largest of the 3 numbers.

**Hint =>**

1. Take user input for 3 numbers
2. Write a single method to find the smallest and largest of the three numbers

***public static int[] findSmallestAndLargest(int number1, int number2, int number3)***

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int num1=input.nextInt(),num2=input.nextInt(),num3=input.nextInt();

int[] result= SmallLarge(num1,num2,num3);

System.out.printf("The smallest value is %d\n",result[0]);

System.out.printf("The smallest value is %d",result[1]);

}

public static int[] SmallLarge(int num1,int num2,int num3){

int smallest=Math.min(num1,Math.min(num2,num3));

int largest=Math.max(num1,Math.max(num2,num3));

return new int[]{smallest,largest};

}

}

1. Write a program to take 2 numbers and print their quotient and reminder

**Hint =>**

1. Take user input as integer
2. Use division operator (/) for quotient and moduli operator (%) for reminder
3. Write Method to find the reminder and the quotient of a number

***public static int[] findRemainderAndQuotient(int number, int divisor)***

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int num1=input.nextInt(),num2=input.nextInt();

int[] result= SmallLarge(num1,num2);

System.out.printf("The Quotient is %d\n",result[0]);

System.out.printf("The remainder is %d",result[1]);

}

public static int[] SmallLarge(int num1,int num2){

int quotient=num1/num2;

int remainder=num1%num2;

return new int[]{quotient,remainder};

}

}

1. Create a program to divide N number of chocolates among M children. Print the number of chocolates each child will get and also the remaining chocolates

**Hint =>**

1. Get an integer value from user for the numberOfchocolates and numberOfChildren.
2. Write the method to find the number of chocolates each child gets and number of remaining chocolates

***public static int[] findRemainderAndQuotient(int number, int divisor)***

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int num1=input.nextInt(),num2=input.nextInt();

int[] result= SmallLarge(num1,num2);

System.out.printf("The Quotient is %d\n",result[0]);

System.out.printf("The remainder is %d",result[1]);

}

public static int[] SmallLarge(int num1,int num2){

int quotient=num1/num2;

int remainder=num1%num2;

return new int[]{quotient,remainder};

}

}

1. Write a program calculate the wind chill temperature given the temperature and wind speed

**Hint =>**

1. Write a method to calculate the wind chill temperature using the formula

***public double calculateWindChill(double temperature, double windSpeed)***

import java.util.Scanner;

public class EmployeeBonus {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int num1=input.nextInt(),num2=input.nextInt();

System.out.println(WindChill(num1,num2));

}

public static double WindChill(double temp,double speed){

double windchill= 35.74+(0.6215\*temp)+ (0.4275\*temp-35.75)\* Math.pow(speed,0.16);

return windchill;

}

}

1. Write a program to calculate various trigonometric functions using Math class given an angle in degrees

**Hint =>**

1. Method to calculate various trigonometric functions, Firstly convert to radians and then use Math function to find sine, cosine and tangent.

***public double[] calculateTrigonometricFunctions(double angle)***

import java.util.Scanner;

public class TrigonometryCalculator {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.print("Enter an angle in degrees: ");

double angle = input.nextDouble();

double[] trigValues = calculateTrigonometricFunctions(angle);

System.out.println("Sine: " + trigValues[0]);

System.out.println("Cosine: " + trigValues[1]);

System.out.println("Tangent: " + trigValues[2]);

input.close();

}

public static double[] calculateTrigonometricFunctions(double angle) {

double radians = Math.toRadians(angle);

double sine = Math.sin(radians);

double cosine = Math.cos(radians);

double tangent = Math.tan(radians);

return new double[]{sine, cosine, tangent};

}

}