­­­­JAVA METHODS  
WEEK-1- LEVEL-1

import java.util.\*;

// Program 1: Simple Interest Calculator

class SimpleInterestCalculator {

public double calculateSimpleInterest(double principal, double rate, double time) {

return (principal \* rate \* time) / 100;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter Principal: ");

double principal = sc.nextDouble();

System.out.print("Enter Rate: ");

double rate = sc.nextDouble();

System.out.print("Enter Time: ");

double time = sc.nextDouble();

SimpleInterestCalculator obj = new SimpleInterestCalculator();

double si = obj.calculateSimpleInterest(principal, rate, time);

System.out.println("The Simple Interest is " + si + " for Principal " + principal + ", Rate of Interest " + rate + " and Time " + time);

}

}

// Program 2: Combination Formula for Handshakes

class HandshakeFormula {

public int calculateUsingCombination(int n) {

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

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter number of students: ");

int n = sc.nextInt();

HandshakeFormula obj = new HandshakeFormula();

int result = obj.calculateUsingCombination(n);

System.out.println("Handshakes using combination formula: " + result);

}

}

// Program 3: Display Handshake Count

class DisplayHandshakes {

public int calculateHandshakes(int n) {

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

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter number of students: ");

int n = sc.nextInt();

DisplayHandshakes obj = new DisplayHandshakes();

int handshakes = obj.calculateHandshakes(n);

System.out.println("Maximum number of handshakes: " + handshakes);

}

}

// Program 4: Rounds in Triangular Park

class TriangularPark {

public double calculateRounds(double a, double b, double c) {

double perimeter = a + b + c;

return 5000 / perimeter;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter side A: ");

double a = sc.nextDouble();

System.out.print("Enter side B: ");

double b = sc.nextDouble();

System.out.print("Enter side C: ");

double c = sc.nextDouble();

TriangularPark obj = new TriangularPark();

double rounds = obj.calculateRounds(a, b, c);

System.out.println("Rounds needed to complete 5km: " + rounds);

}

}

// Program 5: Positive, Negative or Zero

class NumberCheck {

public int checkNumber(int num) {

if (num > 0) return 1;

else if (num < 0) return -1;

else return 0;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

NumberCheck obj = new NumberCheck();

int result = obj.checkNumber(num);

if (result == 1) System.out.println("Positive number");

else if (result == -1) System.out.println("Negative number");

else System.out.println("Zero");

}

}

// Program 6: Spring Season Check

class SpringSeason {

public boolean isSpringSeason(int month, int day) {

return (month == 3 && day >= 20) || (month > 3 && month < 6) || (month == 6 && day <= 20);

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter month: ");

int month = sc.nextInt();

System.out.print("Enter day: ");

int day = sc.nextInt();

SpringSeason obj = new SpringSeason();

if (obj.isSpringSeason(month, day)) System.out.println("It's a Spring Season");

else System.out.println("Not a Spring Season");

}

}

// Program 7: Sum of N Natural Numbers

class NaturalNumberSum {

public int sumNaturalNumbers(int n) {

int sum = 0;

for (int i = 1; i <= n; i++) sum += i;

return sum;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int n = sc.nextInt();

NaturalNumberSum obj = new NaturalNumberSum();

int sum = obj.sumNaturalNumbers(n);

System.out.println("Sum of first " + n + " natural numbers is: " + sum);

}

}

// Program 8: Smallest and Largest of 3 Numbers

class MinMaxOfThree {

public int[] findSmallestAndLargest(int a, int b, int c) {

int smallest = Math.min(a, Math.min(b, c));

int largest = Math.max(a, Math.max(b, c));

return new int[]{smallest, largest};

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter three numbers: ");

int a = sc.nextInt();

int b = sc.nextInt();

int c = sc.nextInt();

MinMaxOfThree obj = new MinMaxOfThree();

int[] result = obj.findSmallestAndLargest(a, b, c);

System.out.println("Smallest: " + result[0] + ", Largest: " + result[1]);

}

}

// Program 9: Quotient and Remainder

class QuotientRemainder {

public int[] findRemainderAndQuotient(int number, int divisor) {

return new int[]{number % divisor, number / divisor};

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter number: ");

int number = sc.nextInt();

System.out.print("Enter divisor: ");

int divisor = sc.nextInt();

QuotientRemainder obj = new QuotientRemainder();

int[] result = obj.findRemainderAndQuotient(number, divisor);

System.out.println("Quotient: " + result[1] + ", Remainder: " + result[0]);

}

}

// Program 10: Chocolate Division

class ChocolateDivision {

public int[] divideChocolates(int chocolates, int children) {

return new int[]{chocolates / children, chocolates % children};

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter number of chocolates: ");

int chocolates = sc.nextInt();

System.out.print("Enter number of children: ");

int children = sc.nextInt();

ChocolateDivision obj = new ChocolateDivision();

int[] result = obj.divideChocolates(chocolates, children);

System.out.println("Each child gets: " + result[0] + ", Remaining chocolates: " + result[1]);

}

}

// Program 11: Wind Chill Temperature

class WindChillCalculator {

public double calculateWindChill(double temperature, double windSpeed) {

return 35.74 + 0.6215 \* temperature - 35.75 \* Math.pow(windSpeed, 0.16)

+ 0.4275 \* temperature \* Math.pow(windSpeed, 0.16);

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter temperature in Fahrenheit: ");

double temp = sc.nextDouble();

System.out.print("Enter wind speed in mph: ");

double speed = sc.nextDouble();

WindChillCalculator obj = new WindChillCalculator();

double wct = obj.calculateWindChill(temp, speed);

System.out.println("Wind Chill Temperature: " + wct);

}

}

// Program 12: Trigonometric Functions

class TrigonometricFunctions {

public double[] calculateTrigonometricFunctions(double angle) {

double rad = Math.toRadians(angle);

return new double[]{Math.sin(rad), Math.cos(rad), Math.tan(rad)};

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

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

double angle = sc.nextDouble();

TrigonometricFunctions obj = new TrigonometricFunctions();

double[] results = obj.calculateTrigonometricFunctions(angle);

System.out.println("Sine: " + results[0] + ", Cosine: " + results[1] + ", Tangent: " + results[2]);

}  
}