**1 write a program to print the numbers from 10 to 50 using for loop/while loop. output =negative**

// Program to print negative numbers from -10 to -50 using a for loop and a while loop.

public class NegativeNumberPrinter {

    /\*\*

     \* Prints numbers from -10 to -50 using a for loop.

     \* It iterates by decrementing the counter.

     \*/

    public static void printNegativeNumbersWithForLoop() {

        System.out.println("--- Numbers from -10 to -50 using a FOR loop ---");

        // Initialize 'i' to -10.

        // Continue as long as 'i' is greater than or equal to -50.

        // Decrement 'i' by 1 in each iteration.

        for (int i = -10; i >= -50; i--) {

            System.out.println(i);

        }

        System.out.println("----------------------------------------------\n");

    }

    /\*\*

     \* Prints numbers from -10 to -50 using a while loop.

     \* It uses a counter that decrements in each step.

     \*/

    public static void printNegativeNumbersWithWhileLoop() {

        System.out.println("--- Numbers from -10 to -50 using a WHILE loop ---");

        // Initialize 'num' outside the loop.

        int num = -10;

        // Continue the loop as long as 'num' is greater than or equal to -50.

        while (num >= -50) {

            System.out.println(num);

            // Decrement 'num' inside the loop to move towards -50.

            num--;

        }

        System.out.println("----------------------------------------------\n");

    }

    /\*\*

     \* Main method to execute both loop demonstrations.

     \*/

    public static void main(String[] args) {

        // Call the method to print numbers using a for loop.

        printNegativeNumbersWithForLoop();

        // Call the method to print numbers using a while loop.

        printNegativeNumbersWithWhileLoop();

    }

}

**2 write a program that find a given number is negative or positive .   
 input =123   
 output = positive   
 input =100   
 output=678**

import java.util.Scanner; // Import the Scanner class to read user input

public static void main(String[] args) {

        // Create a Scanner object to read input from the console

        Scanner scanner = new Scanner(System.in);

        System.out.println("--- Positive/Negative Number Checker ---");

        // Loop to allow multiple checks until the user decides to exit

        while (true) {

            System.out.print("Enter an integer number (or type 'exit' to quit): ");

            // Check if the next input is an integer

            if (scanner.hasNextInt()) {

                int number = scanner.nextInt(); // Read the integer input

                // Determine if the number is positive, negative, or zero

                if (number > 0) {

                    System.out.println("Output: Positive");

                } else if (number < 0) {

                    System.out.println("Output: Negative");

                } else {

                    // If the number is neither greater than 0 nor less than 0, it must be 0

                    System.out.println("Output: Zero");

                }

            } else {

                // Read the non-integer input to check if it's the exit command

                String input = scanner.next();

                if (input.equalsIgnoreCase("exit")) {

                    System.out.println("Exiting the program. Goodbye!");

                    break; // Exit the while loop

                } else {

                    System.out.println("Invalid input! Please enter an integer number or 'exit'.");

                }

            }

            System.out.println(); // Add a blank line for better readability between checks

        }

        // Close the scanner to release system resources

        scanner.close();

    }

}

**3 write down the program to reverse the given number using loops.**

**input == 876**   
 import java.util.Scanner; // Import the Scanner class to read user input

// This program reverses a given integer number using a while loop.

public class NumberReverser {

    public static void main(String[] args) {

        // Create a Scanner object to read input from the console

        Scanner scanner = new Scanner(System.in);

        System.out.println("--- Number Reverser ---");

        System.out.print("Enter an integer number to reverse: ");

        int originalNumber = 0;

        try {

            // Read the integer input from the user

            originalNumber = scanner.nextInt();

        } catch (java.util.InputMismatchException e) {

            System.err.println("Invalid input! Please enter a valid integer number.");

            scanner.close(); // Close scanner before exiting on error

            return; // Exit the program

        }

        int number = originalNumber; // Use a working copy of the number

        int reversedNumber = 0;       // Initialize a variable to store the reversed number

        // Loop as long as the number is not 0

        // In each iteration, we extract the last digit, build the reversed number,

        // and then remove the last digit from the original number.

        while (number != 0) {

            // Get the last digit of the number

            int digit = number % 10; // The modulo operator (%) gives the remainder when divided by 10

            // Build the reversed number:

            // Multiply the current reversedNumber by 10 to shift existing digits to the left,

            // then add the extracted digit.

            reversedNumber = reversedNumber \* 10 + digit;

            // Remove the last digit from the number

            number = number / 10; // Integer division (/) discards the decimal part

        }

        System.out.println("Original Number: " + originalNumber);

        System.out.println("Reversed Number: " + reversedNumber);

        // Close the scanner to release system resources

        scanner.close();

    }

}

**4 write a java program to find the smallest number among three numbers.** import java.util.InputMismatchException; // For handling non-integer input

import java.util.Scanner;             // For reading user input

// This program finds the smallest number among three given integers.

public class SmallestNumberFinder {

    public static void main(String[] args) {

        // Create a Scanner object to read input from the console

        Scanner scanner = new Scanner(System.in);

        System.out.println("--- Find the Smallest Number Among Three ---");

        int num1, num2, num3;

        // --- Input Section ---

        try {

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

            num1 = scanner.nextInt();

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

            num2 = scanner.nextInt();

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

            num3 = scanner.nextInt();

        } catch (InputMismatchException e) {

            System.err.println("Invalid input! Please enter only integer numbers.");

            scanner.close(); // Close scanner before exiting

            return; // Exit the program if input is invalid

        }

        // --- Method 1: Using if-else if-else statements ---

        System.out.println("\n--- Using if-else if-else ---");

        int smallestUsingIfElse;

        if (num1 <= num2 && num1 <= num3) {

            // If num1 is less than or equal to both num2 and num3, then num1 is the smallest

            smallestUsingIfElse = num1;

        } else if (num2 <= num1 && num2 <= num3) {

            // If num2 is less than or equal to both num1 and num3, then num2 is the smallest

            smallestUsingIfElse = num2;

        } else {

            // If neither num1 nor num2 is the smallest, then num3 must be the smallest

            smallestUsingIfElse = num3;

        }

        System.out.println("The smallest number (using if-else) is: " + smallestUsingIfElse);

        // --- Method 2: Using Math.min() function ---

        System.out.println("\n--- Using Math.min() ---");

        // Math.min(a, b) returns the smaller of the two arguments.

        // We can nest it to compare three numbers:

        // First, find the smaller of num1 and num2.

        // Then, compare that result with num3.

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

        System.out.println("The smallest number (using Math.min) is: " + smallestUsingMathMin);

        // Close the scanner to release system resources

        scanner.close();

    }

}  
**5 write a java program that takes the purchase amount as input and calculate the final payable amount after applying the discount .   
 1 . if the purchase is less than 500 , no discount is applied.   
 2 . if the purchase amount is between 500 and 1000 , a 10% discount is applied.   
 3 . if the purchase amount is greater than 1000 a 20%discount is applied.** import java.util.InputMismatchException; // For handling non-numeric input

import java.util.Scanner;             // For reading user input

import java.text.DecimalFormat;       // For formatting the output to two decimal places

// This program calculates the final payable amount after applying discounts

// based on the purchase amount.

public class DiscountCalculator {

    public static void main(String[] args) {

        // Create a Scanner object to read input from the console

        Scanner scanner = new Scanner(System.in);

        // Create a DecimalFormat object to format currency to two decimal places

        DecimalFormat df = new DecimalFormat("#.00");

        System.out.println("--- Purchase Discount Calculator ---");

        double purchaseAmount = 0.0;

        double discountPercentage = 0.0;

        double discountAmount = 0.0;

        double finalPayableAmount = 0.0;

        // Loop to ensure valid numeric input for purchase amount

        while (true) {

            System.out.print("Enter the purchase amount: $");

            try {

                purchaseAmount = scanner.nextDouble(); // Read the purchase amount

                // Validate if the purchase amount is positive

                if (purchaseAmount < 0) {

                    System.out.println("Purchase amount cannot be negative. Please enter a valid amount.");

                } else {

                    break; // Exit loop if valid positive input is received

                }

            } catch (InputMismatchException e) {

                System.out.println("Invalid input! Please enter a numeric value for the purchase amount.");

                scanner.next(); // Consume the invalid input to prevent an infinite loop

            }

        }

        // --- Apply Discount Logic ---

        if (purchaseAmount < 500) {

            // Condition 1: Less than 500, no discount

            discountPercentage = 0.0;

            discountAmount = 0.0;

            finalPayableAmount = purchaseAmount;

            System.out.println("\nNo discount applied.");

        } else if (purchaseAmount >= 500 && purchaseAmount < 1000) {

            // Condition 2: Between 500 and 1000 (exclusive of 1000), 10% discount

            discountPercentage = 10.0;

            discountAmount = purchaseAmount \* (discountPercentage / 100);

            finalPayableAmount = purchaseAmount - discountAmount;

            System.out.println("\n10% discount applied!");

        } else { // purchaseAmount >= 1000

            // Condition 3: Greater than or equal to 1000, 20% discount

            discountPercentage = 20.0;

            discountAmount = purchaseAmount \* (discountPercentage / 100);

            finalPayableAmount = purchaseAmount - discountAmount;

            System.out.println("\n20% discount applied!");

        }

        // --- Display Results ---

        System.out.println("\n--- Purchase Summary ---");

        System.out.println("Original Purchase Amount: $" + df.format(purchaseAmount));

        System.out.println("Discount Percentage: " + df.format(discountPercentage) + "%");

        System.out.println("Discount Amount: $" + df.format(discountAmount));

        System.out.println("----------------------------------");

        System.out.println("Final Payable Amount:   $" + df.format(finalPayableAmount));

        System.out.println("----------------------------------");

        // Close the scanner to release system resources

        scanner.close();

    }

}

**6 write a java program to bellowed pattern -> i and j and k =>5   
 5 5 5 5 5   
 5 4 4 4 4   
 5 4 3 3 3   
 5 4 3 2 2   
 5 4 3 2 1** // This program generates a specific number pattern based on given rules.

public class NumberPattern {

    public static void main(String[] args) {

        // Define the size of the pattern (e.g., 5 for a 5x5 pattern).

        // The problem states i, j, k => 5, implying a size of 5.

        int size = 5;

        System.out.println("--- Generated Number Pattern (Size " + size + ") ---");

        // Outer loop: Controls the rows.

        // It iterates from 'size' down to 1.

        // 'i' represents the current row's base number.

        for (int i = size; i >= 1; i--) {

            // Inner loop: Controls the columns for the current row.

            // It also iterates from 'size' down to 1.

            // 'j' represents the current column's value from the rightmost (smallest) to leftmost (largest).

            for (int j = size; j >= 1; j--) {

                // Determine the number to print at the current (i, j) position.

                // The logic is to print the maximum of the current row number (i)

                // and the current column number (j).

                // Example:

                // If i=5, j=5 -> max(5,5)=5

                // If i=5, j=4 -> max(5,4)=5

                // If i=4, j=5 -> max(4,5)=5

                // If i=4, j=4 -> max(4,4)=4

                // If i=3, j=2 -> max(3,2)=3

                // If i=2, j=1 -> max(2,1)=2

                System.out.print(Math.max(i, j));

            }

            // After all columns for the current row are printed, move to the next line.

            System.out.println();

        }

        System.out.println("----------------------------------------------");

    }

}