//  Program Name: Total, Average, Max, Min, and Interest

//  Author: Greg Fritz

//  University: CSU Global

//  Professor: Dr. Gonzalez

//  Class: Programming 1 CSC320-2

//  Date: 6 July 2025

// -------------------------------------------

//  Pseudocode:

// -------------------------------------------

// START

// INITIALIZE count to 0

// INITIALIZE total to 0

// INITIALIZE max to smallest possible number

// INITIALIZE min to largest possible number

// DISPLAY "Enter 5 floating-point values:"

// WHILE count is less than 5

//     DISPLAY "Enter value #(count + 1):"

//     IF input is a valid floating-point number THEN

//         READ value

//         ADD value to total

//         IF value > max THEN

//             SET max = value

//         ENDIF

//         IF value < min THEN

//             SET min = value

//         ENDIF

//         INCREMENT count

//     ELSE

//         DISPLAY "Invalid input. Please enter a valid floating-point number."

//         DISCARD invalid input

//     ENDIF

// ENDWHILE

// COMPUTE average as total / 5

// COMPUTE interest as total \* 0.20

// DISPLAY "Results:"

// DISPLAY total

// DISPLAY average

// DISPLAY max

// DISPLAY min

// DISPLAY interest

// END

import java.util.Scanner;

public class CriticalThinkingMod4Opt1Fritz {

    public static void main(String[] args) {

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

        Scanner scanner = new Scanner(System.in);

        // Initialize a counter to track the number of valid inputs

        int count = 0;

        // Initialize variables to store total, max, and min values

        double total = 0;

        double max = Double.NEGATIVE\_INFINITY; // Start with the smallest possible value

        double min = Double.POSITIVE\_INFINITY; // Start with the largest possible value

        // Prompt the user to begin input

        System.out.println("Enter 5 floating-point values:");

        // Use a while-loop to continue prompting until 5 valid numbers are entered

        while (count < 5) {

            System.out.print("Enter value #" + (count + 1) + ": ");

            // Check if the user input is a valid floating-point number

            if (scanner.hasNextDouble()) {

                double value = scanner.nextDouble(); // Read the input

                total += value; // Add to total

                // Update maximum if this value is larger

                if (value > max) {

                    max = value;

                }

                // Update minimum if this value is smaller

                if (value < min) {

                    min = value;

                }

                count++; // Increase the count of valid inputs

            } else {

                // Handle invalid input (non-number)

                System.out.println("Please enter a valid floating-point number.");

                scanner.next(); // Clear the invalid input to avoid an infinite loop

            }

        }

        // Calculate the average of the 5 values

        double average = total / 5;

        // Calculate 20% interest on the total

        double interest = total \* 0.20;

        // Display all results to the user with proper formatting

        System.out.println("\nResults:");

        System.out.printf("Total: %.2f%n", total);

        System.out.printf("Average: %.2f%n", average);

        System.out.printf("Maximum: %.2f%n", max);

        System.out.printf("Minimum: %.2f%n", min);

        System.out.printf("Interest on the total of %.2f at 20%%: %.2f%n", total, interest);

        // Close the scanner to free up system resources

        scanner.close();

    }

}

Screenshots  
  
Executing the program testing while loop  
A screenshot of a computer

AI-generated content may be incorrect.

This step sets up the necessary variables and establishes what the user will validly input  
A screen shot of a computer program

AI-generated content may be incorrect.

This step is a while loop that only breaks when 5 floating point numbers are entered. If an invalid number is input, then the loop will not iterate the count and will retry that step until the condition is met.  
A computer screen shot of code

AI-generated content may be incorrect.

This last step does all the required calculations and prints them for the user   
A computer screen shot of a program code

AI-generated content may be incorrect.

Github Links

<https://github.com/gfritzcsu/Programming1/tree/master/Mod4CTOpt1>

<https://github.com/gfritzcsu/Programming1/commits/master/Mod4CTOpt1>