

Programación

UT3 – TAREA12

Programa: “Gestor de temperaturas diarias”

Diseña un programa que permita registrar temperaturas diarias con **opciones de menú** para:

1. (2 puntos) Añadir una nueva temperatura (hasta tres, como máximo).
2. (0,5 puntos) Mostrar la temperatura más alta registrada.
3. (0,5 puntos) Mostrar la temperatura más baja registrada.
4. (1 punto) Mostrar la temperatura media.

Observaciones:

- El valor inicial de las tres temperaturas será de 0.
- Una vez que el usuario ha introducido las tres primeras temperaturas, las siguientes temperaturas que introduzca irán machacando el valor de las anteriores de forma cíclica. Por ejemplo, la cuarta temperatura machacará el valor de la primera, la quinta temperatura machacará el valor de la segunda y así sucesivamente.

Criterios de calificación adicionales:

- a) (1 punto) Se han usado comentarios Javadoc para documentar TODO el código.
- b) (0,5 puntos) Se ha usado el tipo de datos adecuado para almacenar las temperaturas.
- c) (1,5 puntos) Se han usado manejadores de excepciones para controlar y prevenir errores.
- d) (1 punto) Se han definido métodos (al menos tres) para la ejecución de cálculos, devolviendo como respuesta el resultado de dicho cálculo.
- e) (1 punto) Se ha implementado una solución eficiente (sin usar código repetitivo o redundante) y totalmente funcional (el programa hace lo que debe).
- f) (1 punto) Se ha desarrollado un programa que compila y se ejecuta sin “petar” bajo ninguna circunstancia.

Nota: No debes usar arrays para resolver la tarea.

```
import java.util.Scanner;

import java.util.InputMismatchException;

/***
 * @author Carlos Guaraca Nagua
 * @since 19/11/25 - 13:37
 *
 *      Programa para registrar hasta tres temperaturas de forma cíclica
 *      y ofrecer cálculos básicos (máximo, mínimo, media).
 *
 *      Requisitos cumplidos:
 *
 *      - No usa arrays (solo tres variables double).
 *
 *      - Manejo de excepciones para entradas no válidas.
 *
 *      - Contiene al menos tres métodos que devuelven resultados.
 *
 *      - Documentación Javadoc en clase, campos y métodos.
 */
public class Temperatura1 {

    /** Creación de las temperaturas al principio del código */

    private static double temp1 = 0.0;
    private static double temp2 = 0.0;
    private static double temp3 = 0.0;

    /** Índice cíclico (indica qué temperatura se sobrescribe) */

    private static int indice = 1;

    /**
     * Método principal con el menú interactivo.
     */
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int opcion = 0;
```

```
do {  
    System.out.println("\n--- MENÚ DE TEMPERATURAS ---");  
  
    System.out.println("1. Añadir nueva temperatura");  
  
    System.out.println("2. Mostrar temperatura más alta");  
  
    System.out.println("3. Mostrar temperatura más baja");  
  
    System.out.println("4. Mostrar temperatura media");  
  
    System.out.println("5. Salir");  
  
    System.out.print("Elige una opción: ");  
  
  
    try {  
  
        opcion = sc.nextInt();  
  
        sc.nextLine();// limpiar el final de línea  
  
        switch (opcion) {  
  
            case 1:  
  
                agregarTemperatura(sc);  
  
                break;  
  
            case 2:  
  
                System.out.printf("La temperatura más alta es: %.2f%n", calcularMax());  
  
                break;  
  
            case 3:  
  
                System.out.printf("La temperatura más baja es: %.2f%n", calcularMin());  
  
                break;  
  
            case 4:  
  
                System.out.printf("La temperatura media es: %.2f%n", calcularMedia());  
  
                break;  
  
            case 5:  
  
                System.out.println("Saliendo del programa...");  
  
                break;  
        }  
    } catch (Exception e) {  
        System.out.println("Opción no válida");  
    }  
}
```

```
        default:  
  
            System.out.println("Valor no válido. Intenta de nuevo");  
  
            break;  
  
        }  
  
    } catch (InputMismatchException e) {  
  
        System.out.println("Error: debes introducir un valor entero");  
  
        sc.nextLine();// descartar entrada inválida  
  
    }  
  
    } while (opcion != 5);  
  
    sc.close();  
  
}  
  
/**  
 * Lee una nueva temperatura desde el Scanner y la almacena de forma cíclica  
 * en t1, t2 o t3 (1->2->3->1...).  
 *  
 * @param sc Scanner único para leer la entrada del usuario.  
 */  
  
private static void agregarTemperatura(Scanner sc) {  
  
    System.out.print("Introduce la temperatura(decimal con ','): ");  
  
    try {  
  
        double temp = sc.nextDouble();  
  
        sc.nextLine();  
  
        switch (indice) {  
  
            case 1:  
  
                temp1 = temp;
```

```
        break;

    case 2:

        temp2 = temp;

        break;

    case 3:

        temp3 = temp;

        break;

    }

indice++;

if (indice > 3) {

    indice = 1;// Reiniciar ciclo

}

System.out.println("Temperatura guardada automáticamente");

}

} catch (InputMismatchException e) {

    System.out.println("Error: introduce un valor válido para la temperatura");

    sc.nextLine();// descartar entrada inválida

}

}

/**

 * Calcula y devuelve la temperatura máxima entre las tres almacenadas.

 *

 * @return valor máximo (double).

 */

private static double calcularMax() {

    return Math.max(temp1, Math.max(temp2, temp3));

}
```

```
}

/** 
 * Calcula y devuelve la temperatura mínima entre las tres almacenadas.
 *
 * @return valor mínimo (double).
 */
private static double calcularMin() {
    return Math.min(temp1, Math.min(temp2, temp3));
}

/** 
 * Calcula y devuelve la temperatura media de las tres almacenadas.
 *
 * @return media (double).
 */
private static double calcularMedia() {
    return (temp1 + temp2 + temp3) / 3.0;
}
```

```

1 import java.util.Scanner;
2 import java.util.InputMismatchException;
3
4 /**
5 * @author Carlos Guaraca Nagua
6 * @since 19/11/25 - 13:37
7 *
8 * Programa para registrar hasta tres temperaturas de forma cíclica
9 * y ofrecer cálculos básicos (máximo, mínimo, media).
10 * Requisitos cumplidos:
11 * - No usa arrays (solo tres variables double).
12 * - Manejo de excepciones para entradas no válidas.
13 * - Contiene al menos tres métodos que devuelven resultados.
14 * - Documentación Javadoc en clase, campos y métodos.
15 */
16 public class Temperatura {
17     /** Creación de las temperaturas al principio del código */
18     private static double temp1 = 0.0;
19     private static double temp2 = 0.0;
20     private static double temp3 = 0.0;
21     /** Índice cíclico (indica qué temperatura se sobreescribe) */
22     private static int indice = 1;
23
24     /**
25      * Método principal con el menú interactivo.
26      */
27     public static void main(String[] args) {
28         Scanner sc = new Scanner(System.in);
29         int opcion = 0;
30
31         do {
32             System.out.println("\n--- MENÚ DE TEMPERATURAS ---");
33             System.out.println("1. Añadir nueva temperatura");
34             System.out.println("2. Mostrar temperatura más alta");
35             System.out.println("3. Mostrar temperatura más baja");
36             System.out.println("4. Mostrar temperatura media");
37             System.out.println("5. Salir");
38             System.out.print("Elige una opción: ");
39
40             try {
41                 opcion = sc.nextInt();
42                 sc.nextLine(); // Limpiar el final de línea
43                 switch (opcion) {
44                     case 1:
45                         agregarTemperatura(sc);
46                         break;
47                     case 2:
48                         System.out.printf("La temperatura más alta es: %.2f\n", calcularMax());
49                         break;
50                     case 3:
51                         System.out.printf("La temperatura más baja es: %.2f\n", calcularMin());
52                         break;
53                     case 4:
54                         System.out.printf("La temperatura media es: %.2f\n", calcularMedia());
55                         break;
56                     case 5:
57                         System.out.println("Saliendo del programa...");
58                         break;
59                     default:
60                         System.out.println("Valor no válido. Intenta de nuevo");
61                         break;
62                 }
63             } catch (InputMismatchException e) {
64                 System.out.println("Error: debes introducir un valor entero");
65                 sc.nextLine(); // Descartar entrada inválida
66             }
67         } while (opcion != 5);
68         sc.close();
69     }
70
71     /**
72      * Lee una nueva temperatura desde el Scanner y la almacena de forma cíclica
73      * en t1, t2 o t3 (1->2->3->1...).
74      *
75      * @param sc Scanner único para leer la entrada del usuario.
76      */
77     private static void agregarTemperatura(Scanner sc) {
78         System.out.print("Introduce la temperatura(decimal con ','): ");
79         try {
80             double temp = sc.nextDouble();
81             sc.nextLine();
82             switch (indice) {
83                 case 1:
84                     temp1 = temp;
85                     break;
86                 case 2:
87                     temp2 = temp;
88                     break;
89                 case 3:
90                     temp3 = temp;
91                     break;
92                 default:
93                     break;
94             }
95             indice++;
96             if (indice > 3) {
97                 indice = 1; // Reiniciar ciclo
98             }
99             System.out.println("Temperatura guardada automáticamente");
100        } catch (InputMismatchException e) {
101            System.out.println("Error: introduce un valor válido para la temperatura");
102            sc.nextLine(); // Descartar entrada inválida
103        }
104    }
105
106    /**
107     * Calcula y devuelve la temperatura máxima entre las tres almacenadas.
108     *
109     * @return valor máximo (double).
110     */
111     private static double calcularMax() {
112         return Math.max(temp1, Math.max(temp2, temp3));
113     }
114
115     /**
116     * Calcula y devuelve la temperatura mínima entre las tres almacenadas.
117     *
118     * @return valor mínimo (double).
119     */
120     private static double calcularMin() {
121         return Math.min(temp1, Math.min(temp2, temp3));
122     }
123
124     /**
125     * Calcula y devuelve la temperatura media de las tres almacenadas.
126     *
127     * @return media (double).
128     */
129     private static double calcularMedia() {
130         return (temp1 + temp2 + temp3) / 3.0;
131     }
132 }
133
134
135
136
137 }
138

```