**Tutorial Interactivo: Estructuras de Control en Java**

**¡Bienvenido! En este tutorial, aprenderás sobre las estructuras de control en Java. Estas estructuras te permiten controlar el flujo de ejecución de tu programa, determinando qué instrucciones se ejecutan y cuándo.**

Vamos a trabajar paso a paso para que entiendas cada concepto clave, realices ejercicios prácticos y resuelvas preguntas que te permitirán aplicar lo que aprendas de forma interactiva. ¡Empecemos!

**1. Estructuras de Control Condicionales**

Las estructuras de control condicionales permiten que un programa ejecute un bloque de código si se cumple una condición específica.

**Conceptos Clave:**

* **If**: Ejecuta un bloque de código solo si la condición es verdadera.
* **Else**: Ejecuta un bloque de código si la condición del if es falsa.
* **Else if**: Evalúa una nueva condición si las anteriores son falsas.

**Ejemplo de código:**

java

Copiar código

int number = 10;

if (number > 0) {

System.out.println("El número es positivo.");

} else if (number == 0) {

System.out.println("El número es cero.");

} else {

System.out.println("El número es negativo.");

}

**Tarea 1:**

Escribe un código en Java que verifique si un número ingresado por el usuario es **par** o **impar**. Usa if-else para resolverlo.

**2. Estructura de Control: Switch**

El **switch** permite seleccionar entre múltiples alternativas basadas en el valor de una expresión. Es útil cuando tienes muchas condiciones que dependen del mismo valor.

**Conceptos Clave:**

* **Switch**: Evalúa una expresión y ejecuta el bloque de código correspondiente al caso que coincida con el valor de la expresión.
* **Case**: Cada posible valor de la expresión que puede coincidir.
* **Default**: Se ejecuta si ninguno de los casos coincide.

**Ejemplo de código:**

java

Copiar código

int day = 3;

switch (day) {

case 1:

System.out.println("Lunes");

break;

case 2:

System.out.println("Martes");

break;

case 3:

System.out.println("Miércoles");

break;

default:

System.out.println("Día no válido");

}

**Tarea 2:**

Escribe un programa que reciba un número del 1 al 7 e imprima el día de la semana correspondiente usando un switch.

**3. Estructuras de Control Repetitivas**

Las estructuras repetitivas permiten ejecutar un bloque de código múltiples veces. Java tiene tres principales: **for**, **while** y **do-while**.

**3.1. Bucle for**

Un bucle que se ejecuta un número específico de veces. Es útil cuando sabes cuántas veces quieres que se repita.

**Conceptos Clave:**

* **Inicialización**: El valor inicial de la variable de control.
* **Condición**: El bucle sigue ejecutándose mientras la condición sea verdadera.
* **Incremento/Decremento**: Cómo se modifica la variable de control después de cada iteración.

**Ejemplo de código:**

java

Copiar código

for (int i = 1; i <= 5; i++) {

System.out.println("Iteración: " + i);

}

**Tarea 3:**

Crea un programa que imprima los números del 1 al 10 usando un bucle for.

**3.2. Bucle while**

Ejecuta un bloque de código mientras una condición sea verdadera. Se utiliza cuando no se sabe cuántas iteraciones se necesitarán.

**Conceptos Clave:**

* **Condición**: El bucle se ejecuta mientras la condición sea verdadera.

**Ejemplo de código:**

java

Copiar código

int i = 1;

while (i <= 5) {

System.out.println("Iteración: " + i);

i++;

}

**Tarea 4:**

Escribe un programa en Java que use un bucle while para sumar los números del 1 al 100.

**3.3. Bucle do-while**

El bloque de código se ejecuta al menos una vez, y luego sigue ejecutándose mientras la condición sea verdadera.

**Ejemplo de código:**

java

Copiar código

int i = 1;

do {

System.out.println("Iteración: " + i);

i++;

} while (i <= 5);

**Tarea 5:**

Escribe un programa que pida al usuario ingresar un número. El programa seguirá pidiendo números **hasta que** el usuario ingrese un número **positivo**. Usa un bucle do-while.

**Ejercicios Interactivos:**

Ahora que has aprendido los conceptos básicos de las **estructuras de control** en Java, es hora de poner en práctica lo aprendido. Responde las siguientes preguntas y realiza los ejercicios prácticos.

**Preguntas:**

1. **¿Qué estructura de control usarías si solo necesitas que un bloque de código se ejecute bajo una condición específica?**
2. **¿Cómo harías para que un programa evalúe varias condiciones basadas en el valor de una variable?**
3. **¿Cuál es la diferencia entre un bucle for y un bucle while?**
4. **¿En qué caso usarías un do-while en lugar de un while?**

**Respuestas:**

1. Usarías la estructura de control **if**.
2. Para evaluar varias condiciones basadas en el valor de una variable, usarías la estructura **switch**.
3. La diferencia principal es que en el bucle **for** conoces de antemano cuántas veces se va a ejecutar el ciclo, mientras que en el bucle **while** el ciclo continúa mientras la condición siga siendo verdadera.
4. Usarías un do-while cuando quieres que el bloque de código se ejecute al menos una vez, independientemente de si la condición es verdadera desde el principio.