



Recibe una cálida:

¡Bienvenida!

Te estábamos esperando 😊 

» Algoritmos

Plan formativo: Desarrollo de Aplicaciones Full Stack Java Trainee V2.0

HOJA DE RUTA

¿Cuáles **skill** conforman el programa?



REPASO CLASE ANTERIOR

En la clase anterior trabajamos :

- ✓ Estructura repetitiva Mientras
- ✓ Estructura repetitiva Repetir

LEARNING PATHWAY

4.2

Start! 🏁

Fundamentos de la
programación en Java

Estructuras de control
repetitiva: Para

Estructura Para

Sumando del 1 al 10

Imprimiendo Pares

OBJETIVOS DE APRENDIZAJE

¿Qué aprenderemos?



Estructura y aplicación del ciclo Para



› Estructura “*Para*”



Bucle Para

¿Para qué sirve?:

El bucle **Para** (for) es una estructura de control en programación que permite repetir un bloque de código **un número determinado de veces**. Sus características principales son:

- La **inicialización** define y asigna valor a la **variable de control** del bucle.
Por ejemplo: `int i = 0;`
- La **condición evaluada** en cada iteración determina si el bucle continúa o se detiene. Por ejemplo: `i < 10;`
- El **incremento** modifica el valor de la variable de control en cada iteración.
Por ejemplo: `i++`
- Es muy útil cuando se conoce previamente la cantidad de iteraciones que se requieren.

Es más apropiado que un Mientras, cuando se requiere un número fijo y conocido de repeticiones.



Bucle Para

Para:

Es como contar las vueltas que das en una pista de carreras. Especificamos un **valor inicial**, una **condición de finalización** y un **incremento**, y el bucle se ejecutará hasta que se cumpla la condición.

Para (**valor inicial**; **condición de finalización**; **incremento**)

acción 1

acción 2

acción 3

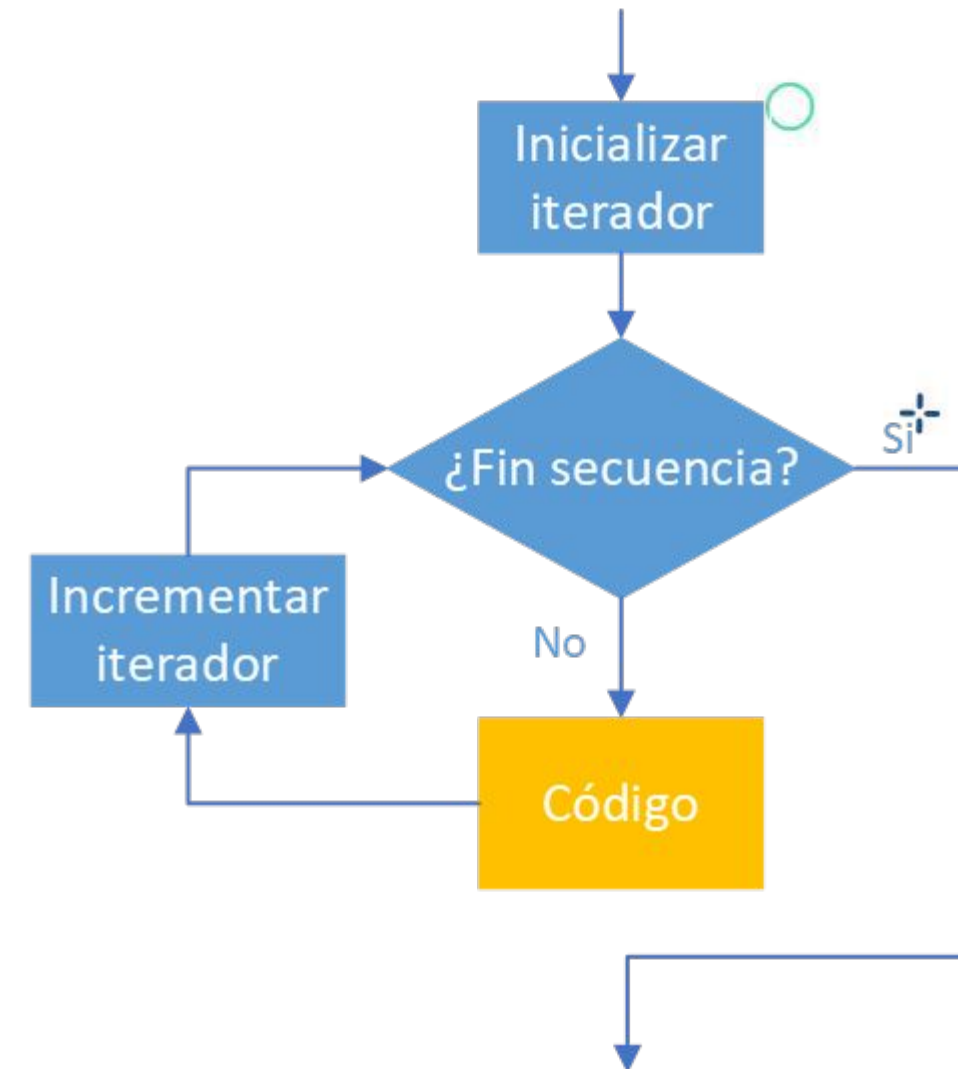
....

acción n

Fin_para



Bucle iterativo





Bucles

Estas estructuras de control repetitivas nos permiten ahorrar tiempo y evitar repetir líneas de código innecesarias. Son herramientas valiosas para procesar listas de datos, realizar cálculos repetitivos y realizar tareas iterativas de manera eficiente.

Aquí algunas recomendaciones generales sobre **cuándo utilizar cada tipo de bucle** en programación:



Mientras: Cuando no se conoce previamente cuántas iteraciones se necesitan y depende de una condición. Por ejemplo, mientras no se ingrese un valor válido.

Repetir: Cuando se requiere que el bloque de código se ejecute al menos una vez antes de evaluar la condición. Por ejemplo, en un menú de opciones.

Para: Cuando se conoce de antemano la cantidad de iteraciones. Por ejemplo, para iterar sobre todos los elementos de un arreglo. También cuando se necesita llevar un control específico de la variable iteradora. Por ejemplo para imprimir números del 1 al 100.

LIVE CODING

Ejemplo en vivo

Imprimiendo tablas:

Vamos a **imprimir la tabla del 2** utilizando un bucle de iteración **Para**.

1. Declaramos la variable contador i que va desde 1 hasta 10. Esta será la iteradora del bucle.
2. En cada iteración se calcula el resultado como 2 multiplicado por el valor actual de i .
3. Se imprime en pantalla la operación completa con el número 2, el valor de i y el resultado.



LIVE CODING

Ejemplo en vivo

Imprimiendo tablas:

*Vamos a **imprimir la tabla del 2** utilizando un bucle de iteración **Para**.*

- 4. Al ir incrementando i en cada vuelta, se va calculando y mostrando la tabla de multiplicar del 2.*
- 5. El bucle se repetirá 10 veces, una por cada valor de 1 a 10 en i .*

 **Tiempo:** 15 minutos





Ejercicio N° 1

Sumando del 1 al 10



Sumando del 1 al 10



Contexto: 🙌

Los bucles de repetición Para nos ayudan a realizar iteraciones cuando conocemos cuántas veces queremos que se ejecuten. Además, nos brinda la variable iteradora en su declaración.



Consigna: 📝

Crear un algoritmo que, utilizando un bucle Para, calcule la suma de los números del 1 al 10.

Paso a paso: ⚙️

- Utiliza el **índice para recorrer** cada número de **uno en uno**.
- **Acumula** el valor en una variable **suma**.

Tiempo 🕒: 20 minutos



Ejercicio N° 2

Imprimiendo Pares



Imprimiendo Pares



Contexto: 🙌

Los bucles de repetición también puede anidar otras estructuras, como por ejemplo, estructuras condicionales. Practiquemos con este ejercicio.



Consigna: ✍️

Crear un algoritmo que, utilizando un bucle Para, imprima los números pares del 1 al 100:

Tiempo 🕒: 20 minutos



Paso a paso: ⚙️

- Utiliza el **índice para recorrer** cada número de **uno en uno**.
- Utiliza un **condicional simple dentro del bucle Para**, y evaluar el valor de i. (recuerda la función MOD)

○

¿Alguna consulta?

+



RESUMEN

¿Qué logramos en esta clase?

- ✓ **Comprender la sintaxis e implementación de la estructura repetitiva *Para*.**
- ✓ **Reconocer cuándo utilizar cada tipo de bucle.**



#WorkingTime

Continuemos ejercitando

¡Antes de cerrar la clase! Te invitamos a: 📌 📌 📌

1. Repasar nuevamente la grabación de esta clase
2. Revisar el material compartido en la plataforma de Moodle (lo que se vio en clase y algún ejercicio adicional)
 - a. Material 1 (Foro)
 - b. *Lectura Módulo 4, Lección 2: página 17*
3. Traer al próximo encuentro, todas tus dudas y consultas para verlas antes de iniciar nuevo tema.

¡Muchas Gracias!

Nos vemos en la próxima clase 🙌



Momento: ✚

Time-out!

🕒 5 min.

