



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 :

- ✓ Concepto e implementación de las Estructuras condicionales
- ✓ Estructuras condicionales simples
- ✓ Estructuras condicionales dobles

LEARNING PATHWAY

4.2

Start! 🏁

Fundamentos de la programación en Java

Estructuras condicionales anidadas y estructuras condicionales múltiples.

Estructuras anidadas

Donantes

Estructuras múltiples

Calculadora

OBJETIVOS DE APRENDIZAJE

¿Qué aprenderemos?



Comprender el uso e implementación de las estructuras condicionales anidadas.



Reconocer e implementar las estructuras condicionales múltiples.





Rompehielo

"¿Qué harías si...?" Versión múltiple:



Vamos a exponer una situación hipotética:

- "Imagina que vas de viaje a la playa y llueve todo el día. ¿Qué harías en ese caso?"



Ahora, levantando la mano, respondan estas preguntas:

- "¿Y si sigue lloviendo al día siguiente?"
- "¿Y si continúa lloviendo más días?"

Además, te pedimos que respondas una alternativa, es decir... ¿Y qué harías si no?



› Estructura Condicional Anidada



Condicionales Anidados

¿Para qué se utilizan?:

Los condicionales anidados permiten **evaluar múltiples condiciones** de forma encadenada y ejecutar diferentes códigos según combinaciones de verdadero y falso. Proporcionan complejidad y versatilidad a la lógica de decisión en los programas.

Se pueden anidar tantos condicionales como se necesite para modelar una situación determinada. El orden de evaluación es **secuencial** y de adentro hacia afuera.

Las condiciones internas sólo se evalúan si todas las condiciones externas fueron verdaderas.





Condicional Anidado

“Si - Anidados”:

Consisten en tener **una instrucción condicional dentro de otra.**

SI <condición 1> ENTONCES

Acción 1

SI NO

SI <condición 2> ENTONCES

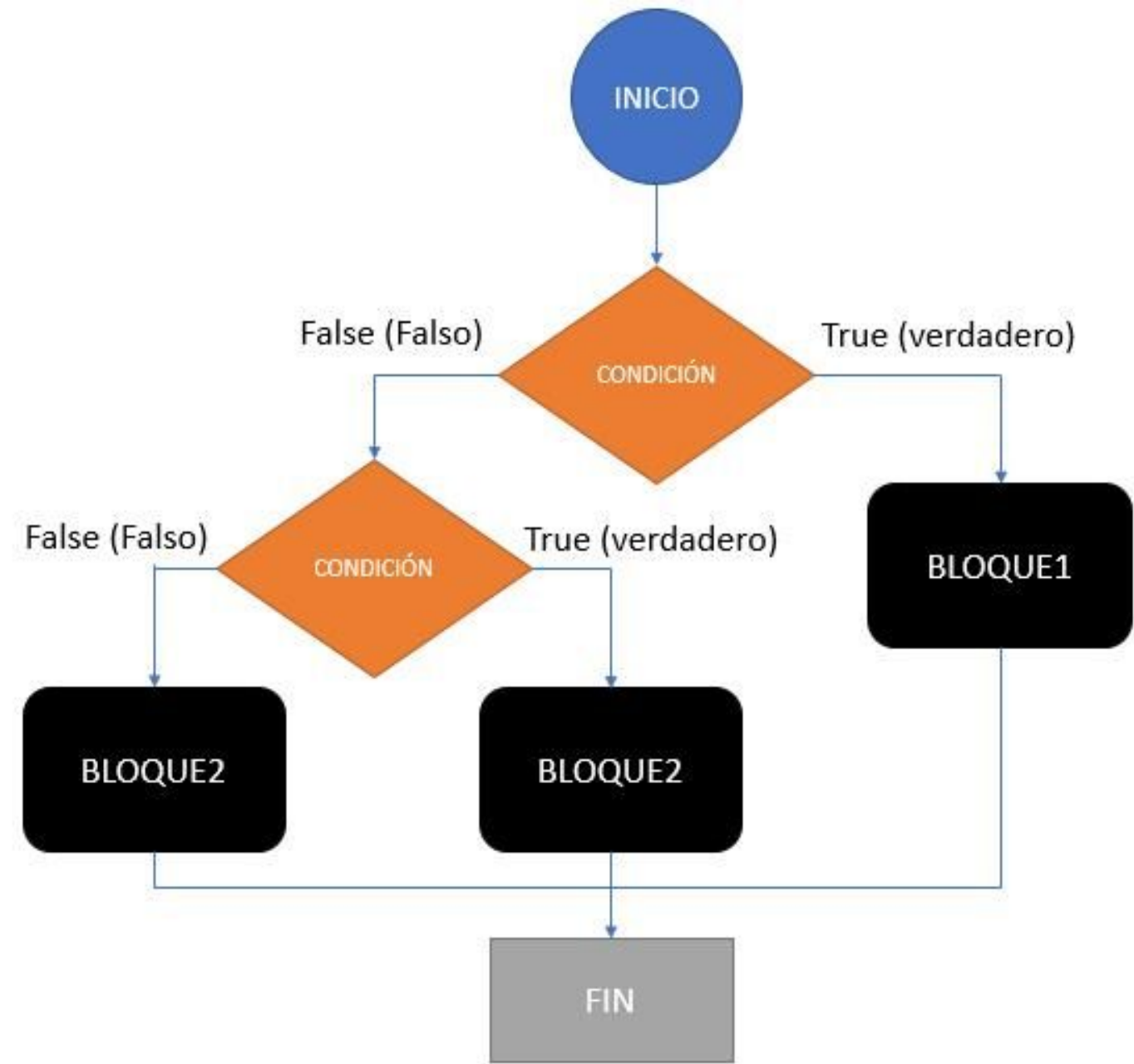
Acción 2

SI NO

Acción 3

FIN SI

FIN SI



LIVE CODING

Ejemplo en vivo

Calculando entradas:

*Vamos a crear un algoritmo para una empresa que tiene salas de juegos para todas las edades y quiere calcular de forma automática el precio que debe cobrar a sus clientes por entrar. **El programa debe preguntar al usuario la edad del cliente y mostrar el precio de la entrada.***

- 1. Si el cliente es menor de 4 años puede entrar gratis*
- 2. Si el cliente tiene entre 4 y 18 años debe pagar 5€*
- 3. Si el cliente es mayor de 18 años, 10€.*

 **Tiempo:** 20 minutos






Ejercicio N° 1

Donantes



Donantes

Los condicionales anidados nos ayudan a tomar diferentes decisiones y expanden las posibilidades del flujo de control de nuestros programas. ¡Vamos a ponerlo en práctica!

 En este ejercicio se requiere crear un algoritmo que diga si una persona puede donar sangre o no, **basándose en las siguientes condiciones:**

- Si tiene entre 18 y 65 años y pesa 50 kg o más, puede donar.
- Si tiene menos de 18 o más de 65, no puede donar.
- Si está en el rango de edad pero pesa menos de 50 kg, no puede donar.

Tiempo : 20 minutos

Paso a paso:

1. Pedir al usuario su edad y peso
2. Con condicionales anidados:
 - Evaluar si tiene entre 18 y 65 años
 - Dentro de este rango, evaluar si pesa 50 kg o más.
 - Si cumple ambas, indicar que puede donar sangre.
 - Si no cumple alguna, indicar que no puede donar.
3. Imprimir el resultado si puede o no puede donar sangre.

› Estructura Condicional Múltiple



Condicional Múltiple

Permite **evaluar una expresión** y ejecutar diferentes bloques de código según el valor que coincida.



Las estructuras de selección múltiple son útiles cuando queremos tomar diferentes acciones **dependiendo del valor de una variable**. Es como un menú de opciones donde elegimos una **según lo que necesitamos**. Por ejemplo: "Si el día de la semana es 1, entonces es lunes; si es 2, entonces es martes, y así sucesivamente".



Los condicionales múltiples permiten evaluar múltiples casos de forma directa y organizada, en lugar de usar muchos si - entonces anidados.



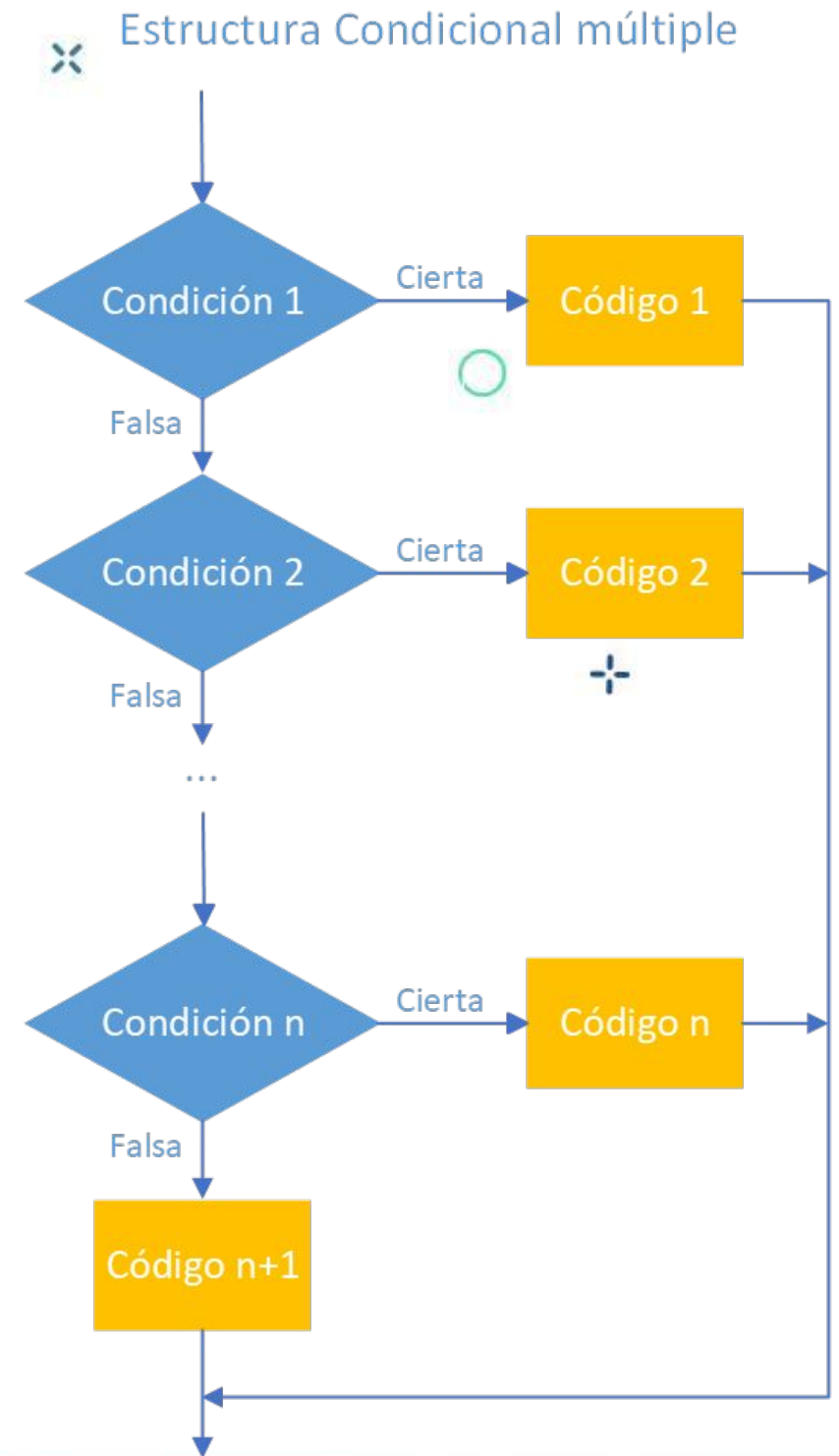


Condicional Múltiple

Según:

```
SEGÚN(variable/expresión){  
  caso 1 : println("lunes")  
  caso 2 : println("martes")  
  caso 3 : println("miércoles")  
  caso 4 : println("jueves")  
  caso 5 : println("viernes")  
  caso 6 : println("sábado")  
  caso 7 : println("domingo")  
}
```

La palabra **Según** está antes de las acciones que se realizan si se cumple la condición de la variable expresión. Y, dependiendo del valor de la expresión, define cuál es el “**caso**” que se ejecuta. **Cada caso tiene un bloque de código determinado.**



LIVE CODING

Ejemplo en vivo

Frutería:

Vamos a escribir un algoritmo que nos devuelva el valor de cada kg de fruta según la fruta seleccionada. Las opciones disponibles son: “plátano”, “naranja”, “papaya”, “frutilla”, “manzana”.

El usuario debe ingresar el dato como cadena (el nombre de la fruta), y el programa debe imprimir el valor del kg de fruta seleccionada.

 **Tiempo: 15 minutos**





Ejercicio N° 2

Calculadora



Calculadora

Creando un menú de operaciones: 🙌

Las condiciones múltiples son ideales para construir menús de opciones, clasificaciones, opciones múltiples, entre otras. En este ejercicio pondremos en práctica una **Calculadora**.

Consigna: 🖋 Realizar un algoritmo que pida dos números enteros positivos por teclado y muestre por pantalla el siguiente menú de opciones de operaciones que se pueden realizar con los números ingresados.:

1. Sumar
2. Restar
3. Multiplicar
4. Dividir

El usuario deberá elegir una opción y el programa deberá mostrar el resultado por pantalla.

Tiempo 🕒: 20 minutos



Calculadora



Paso a paso:

1. Declarar las variables num1, num2 y opción.
2. Pedir al usuario dos números enteros (num1 y num2).
3. Pedir al usuario la opción que desea del menú (cada opción representa una operación aritmética).
4. Usar una estructura Según para evaluar la opción:
5. Mostrar el resultado de la opción elegida.



○

¿Alguna consulta?

+



RESUMEN

¿Qué logramos en esta clase?

- ✓ Reconocer el uso e implementación de las estructuras de condición anidadas.
- ✓ Comprender el uso e implementación de las estructura de condición múltiple.
- ✓ Identificar cuándo es recomendable utilizar cada estructura



#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 15
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 🙌

