



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 :

- ✓ Teoría de Conjuntos
- ✓ Lógica Proposicional
- ✓ Operadores lógicos
- ✓ Tablas de la Verdad

# LEARNING PATHWAY

4.2

Start! 🏁

Fundamentos de la  
programación en Java

Estructuras Condicionales  
Simples y Dobles

Estructuras Condicionales  
Simples

Contraseña Incorrecta

Estructuras Condicionales  
Dobles

¿Es Mayor o Menor?

# OBJETIVOS DE APRENDIZAJE

¿Qué aprenderemos?



**Comprender el objetivo de las estructuras de control condicionales**



**Implementar estructuras condicionales simples**



**Implementar estructuras condicionales dobles**





# Rompehielo

## "¿Qué harías si...?":

Vamos a exponer algunas situaciones hipotéticas, del estilo "¿Qué harías si...?":

- ¿Qué harías si llueve mucho mañana?
- ¿Qué harías si te ganas la lotería?
- ¿Qué harías si se va la luz mientras estudias?
- ¿Qué harías si te regalan entradas para un concierto de tu banda favorita?

Ahora, levantando la mano, respondan estas preguntas como afirmaciones condicionales.  
Por ejemplo: Si llueve mañana, entonces llevaré paraguas.

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

# › Estructuras de Control Condicionales





# Estructuras de Control Condicionales



Las estructuras de control condicionales nos permiten **tomar decisiones basadas en condiciones específicas**.

Imagina que estás en un cruce de caminos y debes elegir una dirección a seguir. Las estructuras de control condicionales son como señales de tráfico que te indican qué camino tomar según ciertas condiciones.



En un algoritmo, si una condición se cumple (o no), se ejecutará un bloque de código determinado.



# › Estructura Condicional Simple



# Estructuras de Control Condicional Simple



Los condicionales simples permiten controlar el flujo del programa y la lógica de decisión basada en **evaluar si una condición booleana es verdadera o falsa**. Son una pieza fundamental en cualquier lenguaje de programación.



De esta forma, un condicional simple permite tomar decisiones en un programa. El flujo se divide en dos caminos: ejecutar un bloque de código o no ejecutarlo, dependiendo del valor de verdad de la condición.

Esto permite controlar el comportamiento del programa y especificar reglas lógicas para orientar su funcionamiento.





# Condicional Simple

## Si - Entonces:

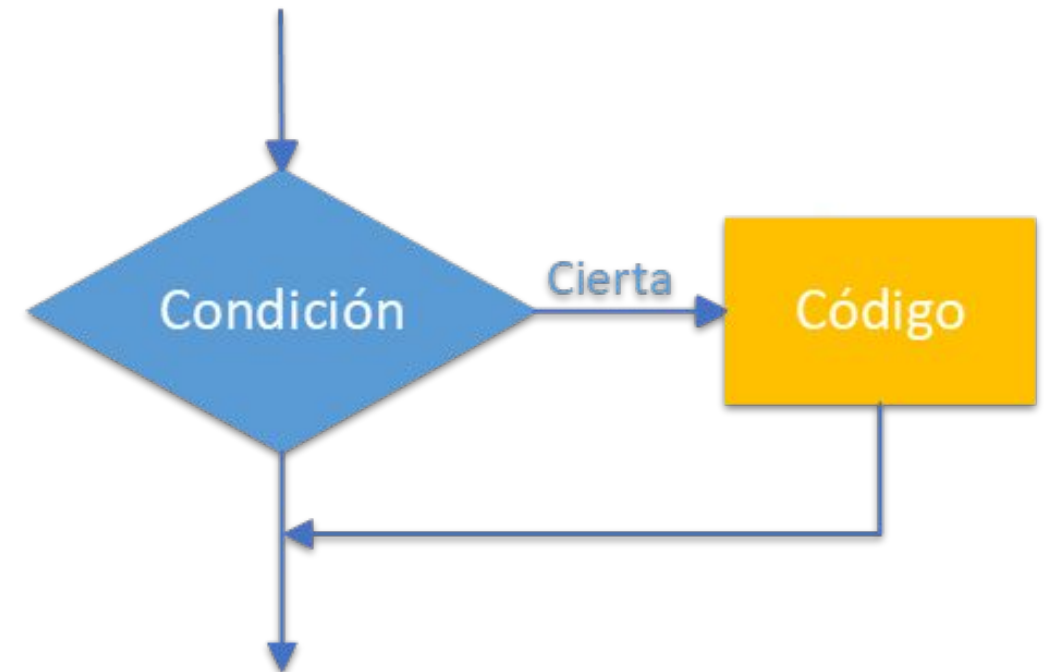
La estructura “si - entonces”, es donde **evaluamos una condición y si es verdadera, ejecutamos un bloque de código**. Es como decir: "Si llueve, entonces lleva un paraguas".

SI <condición> ENTONCES  
    Acciones  
FIN SI

La palabra **SI** indica el comando de evaluación lógica, **<condición>** indica la condición a evaluar y **Acciones** son las instrucciones que se realizarán **sólo si** la evaluación resulta en **VERDADERO**.



## Estructura Condicional simple



# LIVE CODING

Ejemplo en vivo

## ¿Par o Impar?:

*Vamos a crear un programa que nos devuelva si un número es **par**, y de ser así mostrar un mensaje por pantalla. Para esto, utilizaremos una regla matemática:*

- *Un número es par siempre que al dividirlo en 2, el resto sea 0.*

*Recuerda la función del operador **MOD**.*

 **Tiempo: 15 minutos**





# **Ejercicio N° 1**

# **Contraseña Incorrecta**



# Contraseña Incorrecta

**Vamos a probar la seguridad de nuestro programa:** 🙌

Las estructuras condicionales se utilizan para evaluar el flujo de nuestro programa. Se implementan en casi todos los algoritmos, y aquí veremos un ejemplo que nos resultará familiar.

**Consigna:** 📝

Escribir un algoritmo que almacene la cadena de caracteres **“contraseña”** en una variable, luego pregunte al usuario por la contraseña e imprima por pantalla si la contraseña introducida por el usuario coincide con la guardada en la variable (sin tener en cuenta mayúsculas y minúsculas).

**Tiempo** 🕒: 20 minutos

**Recomendaciones:** ⚙️

- Sólo habrá una variable de cadena “contraseña” previamente inicializada con el valor que elijas. La otra variable (para comparar) puede tener el nombre que quieras, y no tendrá valor inicial, el valor se lo dará el usuario.
- Debes comparar ambas cadenas, y definir si son iguales para mostrar el mensaje por pantalla.

# › Estructura Condicional Doble





# Estructuras de Control Condicional Doble



El condicional doble es una extensión del condicional simple que **añade un comportamiento alternativo en caso de que la condición sea falsa.**



De esta forma el flujo se divide en **dos caminos** excluyentes: si la condición es **verdadera se ejecuta un bloque** y si es **falsa se ejecuta otro bloque diferente.**

Esto permite definir comportamientos alternativos dependiendo de una condición. Los condicionales dobles proporcionan un control de flujo más completo en la programación.





# Condicional Doble

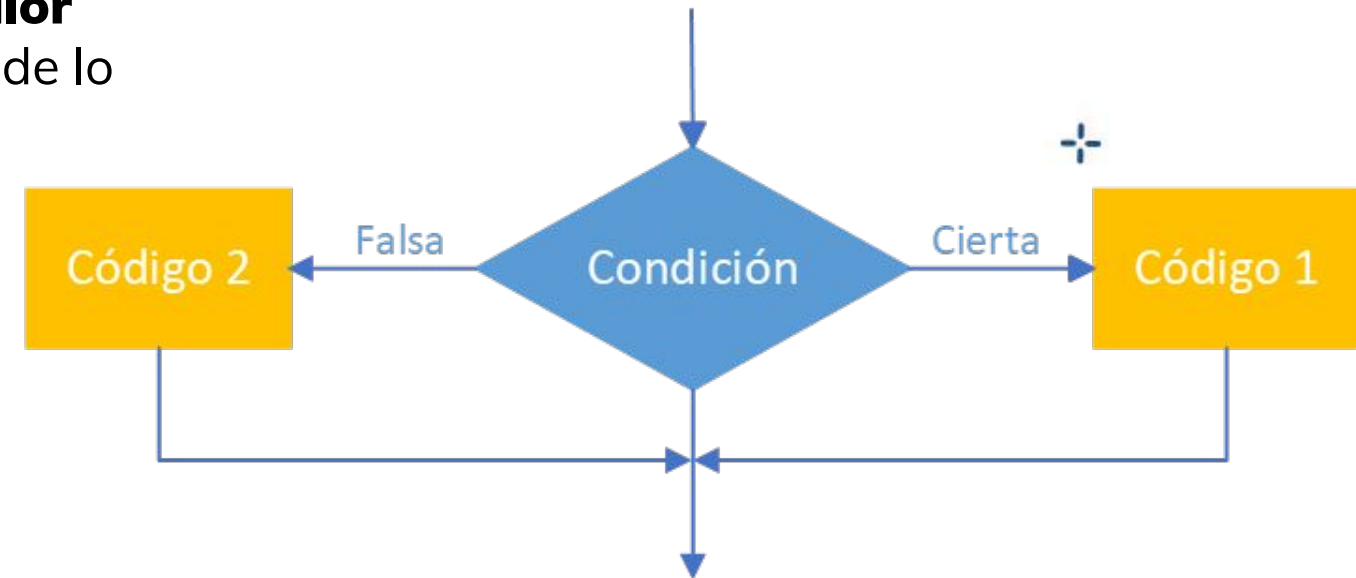
## Si - Entonces - Sino:

Este tipo de estructura añade una acción a ejecutarse **en el caso de que la condición evaluada devuelve el valor FALSO**. Es como decir: "Si llueve, lleva un paraguas, de lo contrario, no lo lleves".

SI <condición> ENTONCES  
    Acciones  
SI NO  
    Acciones  
FIN SI

La palabra **ENTONCES** está antes de las acciones que se realizan si se cumple la condición (**verdadero**) y la expresión **SI NO** está antes de las expresiones que se realizan si no se cumple la condición (**falso**).

Estructura Condicional doble



# LIVE CODING

Ejemplo en vivo

## ¿Par o Impar?

*Vamos a crear un programa que nos devuelva si un número es **impar**. De no ser así, debemos informar por pantalla que “el número es par”.*

- *Un número es impar siempre que al dividirlo en 2, el resto sea distinto de 0.*

 **Tiempo: 10 minutos**





## **Ejercicio N° 2**

# **¿Es Mayor o Menor?**



# ¿Es Mayor o Menor?

**Vamos a probar la seguridad de nuestro programa:** 🙌

Las estructuras condicionales se utilizan para evaluar el flujo de nuestro programa. Se implementan en casi todos los algoritmos, y aquí veremos un ejemplo que nos resultará familiar.

**Consigna:** 📝

Escribir un algoritmo que pregunte al usuario su edad y muestre por pantalla un mensaje si es mayor de edad: “Es mayor” o, caso de no cumplir la condición, muestre “Es menor”.

Para este ejemplo, consideramos mayor cuando la edad sea igual o superior a 18.

**Recomendaciones:** ⚙️

- Debes crear una variable entera **edad**, que será inicializada según el valor ingresado por el usuario.

**Tiempo** 🕒: 10 minutos

○

# ¿Alguna consulta?

+



# RESUMEN

¿Qué logramos en esta clase?

- ✓ Reconocer la importancia de las estructuras de control en la programación.
- ✓ Comprender e implementar la estructura condicional simple.
- ✓ Comprender e implementar la estructura condicional doble.



# #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 14*
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.

