

DEV.F.:

Clase 2

**Variables, Operadores y
Estructuras de Control**



| Objetivos de la clase (180 min)

Variables (20-25 min)

¿qué son?

var, let y const

Operadores (35-40 min)

¿Qué son?

De comparación (>, <, >=, <=)

Lógicos (&&, ||, !)

== y ===

Estructuras de control (35-40 min)

if

if y else

else if

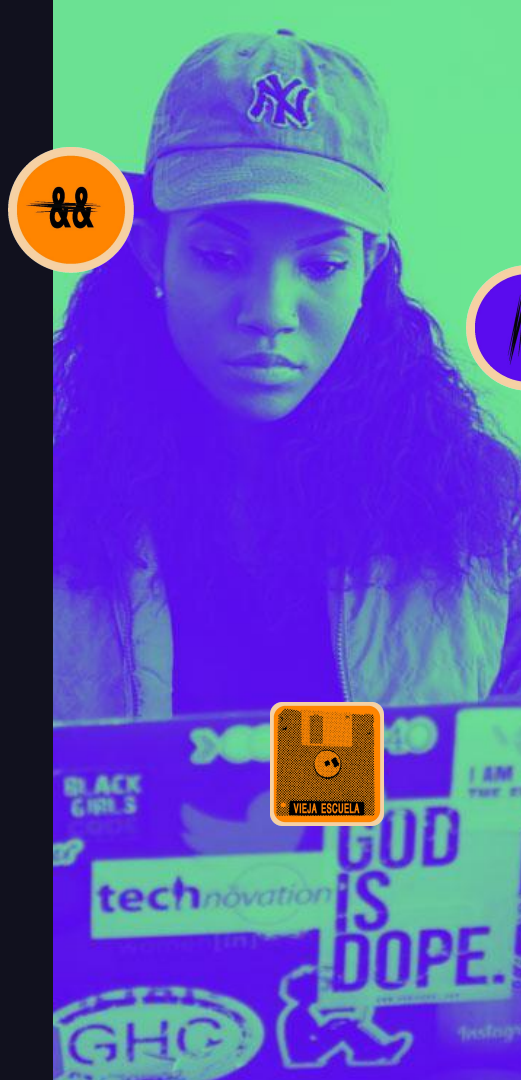


| Objetivos de la clase (180 min)

Al finalizar esta clase deberías ser capaz de escribir condiciones **if** para tomar decisiones en tu código.

El ejercicio de esta clase consiste en crear un programa que genere un mensaje de acuerdo a tu calificación numérica. (chechar Campus)

(¿De cuántas maneras se puede resolver el ejercicio?)



| Variables

¿Dónde vamos a guardar información?

¿Qué cosas podemos guardar?

¿La información guardada siempre es la misma?



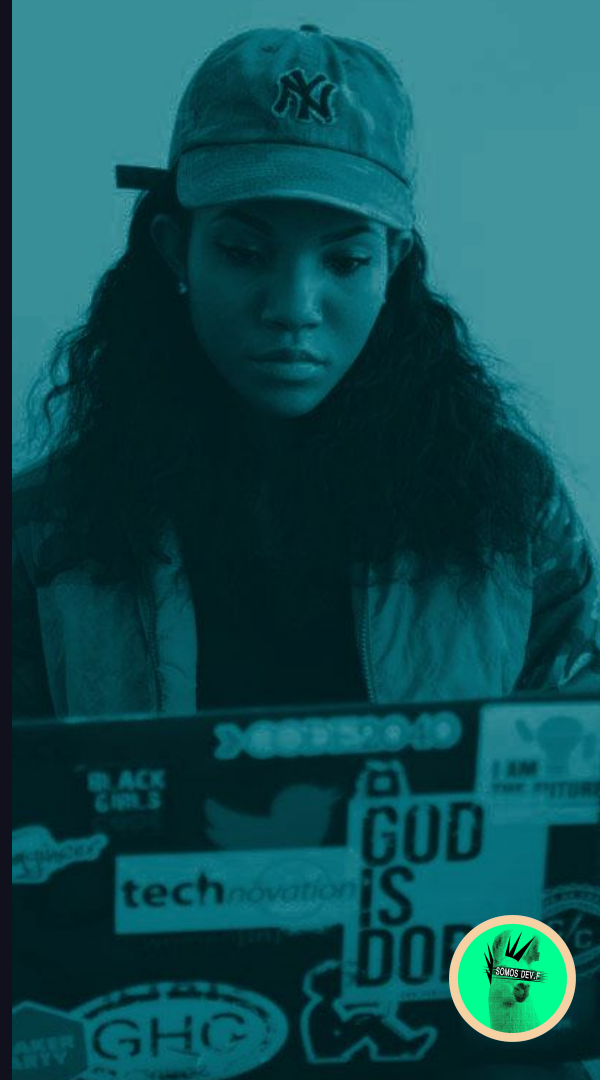
DEV.E::



| **var, let y const**

Todas las variables sirven para guardar valores pero no son lo mismo

- El uso de **var** es antiguo, no está prohibido pero usarla puede ocasionar problemas de lógica
- **let** suele usarse cuando el valor guardado se podría modificar
- Cuando un valor o la información será siempre la misma la guardaremos en una caja **const**



| Operadores Básicos

- **Asignación**
 - =
- **Matemáticos**
 - +, -, *, /, %
- **Combinados**
 - +=
 - -=

x = 5

persona = José

**perro =
Snoopy**



| Ejemplos de operadores básicos

- No dejar solo a la imaginación cómo se ven los operadores matemáticos, sobre todo los combinados

Aterrizar los ejemplos a lo que se tratará el proyecto del día

Crearán un programa que evalúe una nota y genere un mensaje personalizado según su rango (Excelente, Bien, Suficiente o No aprueba).

| Operadores de comparación

Siempre nos van a devolver un valor booleano: true o false.

- == igualdad débil
- === igualdad estricta
- != desigualdad débil
- !== desigualdad estricta
- > mayor
- < menor
- >= mayor o igual
- <= menor o igual

DEV.E.:

¿==? o ¿===?

Al usar == JavaScript trata de hacer una conversión automática de tipos que no siempre sigue una lógica sencilla, lo que puede producir resultados que no esperamos.

El uso de **igualdad estricta** (===) es **más recomendable** porque te asegura que la comparación tenga en cuenta tanto el valor como el tipo de dato. Esto te ayuda a evitar conversiones automáticas que podrían causar confusión o errores en tu programa.

*¿aplicará lo mismo para desigualdades?



| Operadores lógicos

Los operadores lógicos permitirán combinar más de una condición y así cumplir diferentes reglas.

- Para obtener un crédito en un banco debes de ser cliente y tener buen historial
- Para tener un pasaporte mexicano puedes ser mexicano o naturalizado
- ¿Qué condiciones se deben de cumplir para votar en elecciones?
- ¿Qué condiciones se deben de cumplir para ver una película en Netflix?
- ¿Siempre se deben de combinar las condiciones?





Operadores de comparación y lógicos

“Para comenzar a darle sentido a las soluciones a los problemas o programas que vamos a desarrollar es útil saber combinar operadores.”

| Estructura de control: **if**

if

"Si una **condición** es verdadera, haz esto."

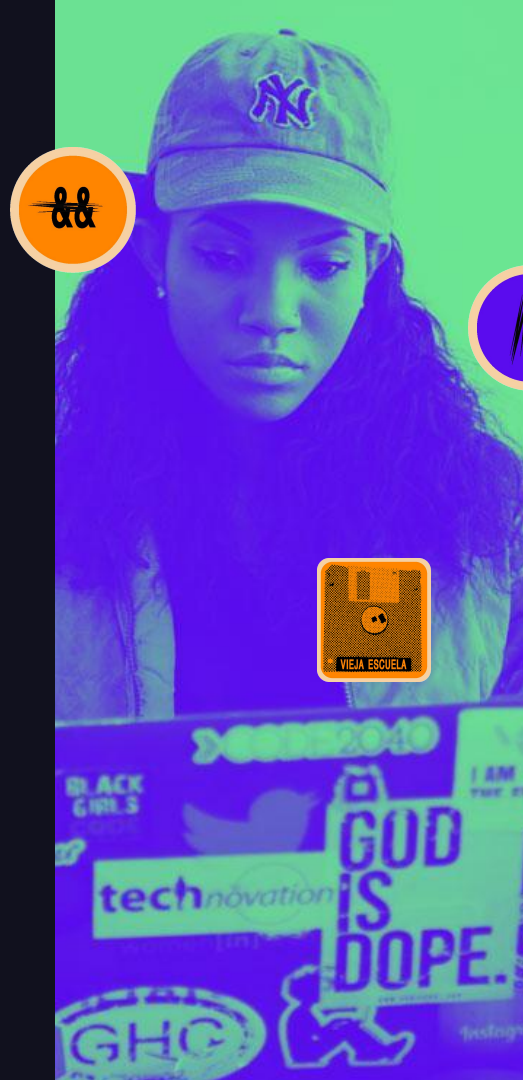
**if con
else**

"Si pasa esto, hazlo. Si no pasa, entonces haz algo diferente."

else if

"Si pasa esto, hazlo. Si no, revisa si pasa esto otro, y entonces hazlo. Si no se cumple ninguna de las dos, haz algo diferente."

DEV.F.



Aterrizar los ejemplos a lo que se tratará el proyecto del día

- No dejar solo a la imaginación cómo se usa el if y el if else.
- Si los alumnos no tienen una imagen mental de cómo funciona no lo podrán aplicar



¿Cómo
comenzamos a
resolver el
ejercicio de
campus?

Mensajes a alumnos por
sus notas