

## Ejercicio 1: Declaración de variables con `let` y `const`

1. Declara una variable con `let` que almacene tu edad y reasígnala con un nuevo valor.
  2. Declara una constante `const` para representar el valor de la velocidad de la luz (299792458 m/s). Intenta reasignar su valor y observa qué ocurre.
- 

## Ejercicio 2: Buenas prácticas en nombres de variables

1. Declara variables usando la notación **camelCase** para los siguientes conceptos:
    1. Tu nombre completo
    2. Tu color favorito
    3. Tu ciudad actual
  2. Intenta declarar una variable con un número al inicio de su nombre. ¿Qué ocurre?
- 

## Ejercicio 3: Tipos de datos básicos

1. Declara variables de los siguientes tipos y escribe un ejemplo para cada uno:
    - Número entero
    - Número decimal
    - Cadena (string)
    - Booleano
    - Null
  2. Usa `console.log()` para imprimir el tipo de cada variable utilizando `typeof`.
- 

## Ejercicio 4: Diferencias entre `undefined` y `null`

1. Declara una variable sin asignarle ningún valor y verifica su tipo.

```
javascript
let sinValor;
console.log(typeof sinValor); // ¿Qué imprime?
```

2. Declara otra variable asignándole el valor `null` y verifica su tipo.

```
let vacio = null;
console.log(typeof vacio); // ¿Qué imprime?
```

3. Explica la diferencia entre `undefined` y `null`.
- 

## Ejercicio 5: Operadores aritméticos

1. Declara dos variables `x` y `y` con valores 10 y 3 respectivamente.
  2. Usa los operadores aritméticos para realizar:
    - Suma
    - Resta
    - Multiplicación
    - División
    - Módulo (resto de la división)
  3. Escribe los resultados esperados antes de ejecutarlos. Después ejecútalos y comprueba el resultado.
-

## Ejercicio 6: Operadores de asignación

1. Declara una variable `z` con valor inicial 20.
2. Usa los operadores de asignación (`+=`, `-=`, `*=`, `/=`, `%=`) para modificar su valor y muestra los resultados:

```
let z = 20;
z += 5;
z -= 10;
z *= 2;
z /= 3;
z %= 4;
console.log(z);
```

3. ¿Qué valor tiene `z` al final?

## Ejercicio 7: Operadores de comparación

1. Declara dos variables `a` y `b` con valores `5` y `"5"` respectivamente.
2. Usa los siguientes operadores para comparar las variables y explica los resultados:

- `a == b`
- `a === b`
- `a != b`
- `a !== b`

## Ejercicio 8: Operadores lógicos

1. Declara tres variables:

```
let esMayorDeEdad = true;
let tieneLicencia = false;
let puedeConducir = esMayorDeEdad && tieneLicencia;
```

2. Responde:
  - ¿Qué valor tiene `puedeConducir` y por qué?
  - Cambia `tieneLicencia` a `true`. ¿Qué ocurre?
3. Crea una nueva condición usando el operador OR (`||`):

```
let tieneCarnetProvisional = true;
let acceso = tieneLicencia || tieneCarnetProvisional;
console.log(acceso);
```

4. ¿Qué imprime `acceso`?

## \*\*Ejercicio 9: Plantillas de cadenas

1. Declara dos variables `nombre` y `edad` con tu nombre y tu edad.
2. Usa *template literals* para construir una frase:

```
let saludo = `Hola, me llamo ${nombre} y tengo ${edad} años.`;
console.log(saludo);
```

## Ejercicio 10: Conversión entre tipos básicos

1. Declara una variable que almacene un número como texto (`"123"`).
2. Realiza una operación matemática directamente con esa variable y observa qué ocurre.

```
let numeroTexto = "123";
let suma = numeroTexto + 10; // ¿Qué resultado da?
let resta = numeroTexto - 10; // ¿Qué resultado da?
```

3. Reflexiona: ¿Por qué el operador ``+`` concatena y el operador ``-`` convierte?

---

### \*\*Ejercicio 11: Objeto básico\*\*

1. Crea un objeto llamado ``persona`` con las propiedades ``nombre``, ``edad`` y ``ciudad``.

---

### \*\*Ejercicio 12: Conversión de tipos\*\*

1. Declara una variable que almacene el número ``"42"`` como cadena.

```
````javascript
let numeroComoCadena = "42";
````
```

2. Convierte esta variable a un número usando ``Number()`` y verifica el tipo:

```
````javascript
let numero = Number(numeroComoCadena);
console.log(typeof numero); // "number"
````
```

3. Declara otra variable booleana y conviértela en cadena usando ``String()``:

---