

A thick, solid red diagonal stripe runs from the top right corner towards the bottom left, separating the white background on the left from the red background on the right.

LOS **CONDICIONALES** **IF TERNARIO** **SWITCH**

“

Nos permiten **evaluar condiciones** y realizar diferentes acciones **según el resultado** de esas evaluaciones.



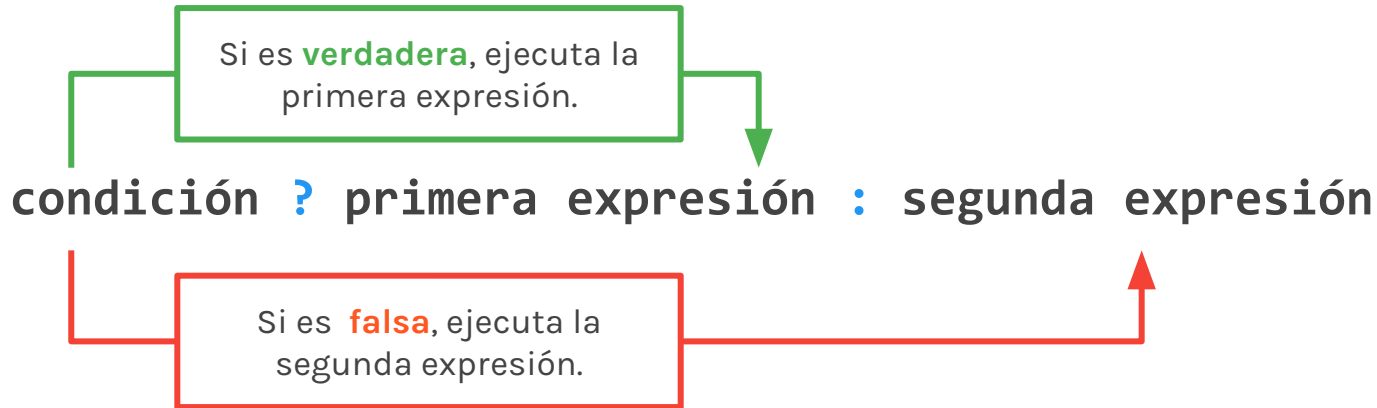
A thick, solid red diagonal stripe runs from the top right corner towards the bottom left, separating the white background on the left from a solid red background on the right.

1.

IF TERNARIO

DEFINIENDO UN IF TERNARIO

A diferencia de un **if** tradicional, el **if ternario** se escribe de forma **horizontal**. Ambas estructuras tienen el mismo flujo interno (*si esta condición es verdadera hacé esto, si no, hacé ésto otro*) pero en este caso no hace falta escribir la palabra **if** ni la palabra **else**.



ESTRUCTURA DE UN IF TERNARIO

Condición

Declaramos una expresión que se evalúa como true o false.



```
4 > 10 ? 'El 4 es más grande' : 'El 10 es más grande'
```

ESTRUCTURA DE UN IF TERNARIO

Primera expresión

Si la condición es **verdadera**, se ejecuta el código que está después del signo de interrogación.




A diagram showing a purple dot at the top left. A vertical line descends from the dot, then a horizontal line extends to the right, and finally a vertical line descends again to point at the opening curly brace of the first string in the code below.

```
4 > 10 ? 'El 4 es más grande' : 'El 10 es más grande'
```

ESTRUCTURA DE UN IF TERNARIO

Segunda expresión

Si la condición es **falsa**, se ejecuta el código que está después de los dos puntos.



```
4 > 10 ? 'El 4 es más grande' : 'El 10 es más grande'
```



Para el if ternario **es obligatorio poner código en la segunda expresión**, si no queremos que pase nada, podemos usar un string vacío "".

A thick, solid red diagonal stripe runs from the top right corner towards the bottom left, separating the white background on the left from the solid red background on the right.

2.

SWITCH

DEFINIENDO UN SWITCH

El **switch**, al igual que los otros condicionales, evalúa una condición y, según el resultado, ejecuta sólo las líneas de código que correspondan.

Nos propone una **sintaxis más legible** para esos casos en los que queremos evaluar que un valor sea igual a alguno de los casos que proponemos.

También **nos permite agrupar casos** y ejecutar un mismo bloque de código para cualquier caso de ese grupo.

ESTRUCTURA DE UN SWITCH

El **switch** está compuesto por una **expresión** a evaluar, seguida de diferentes casos, tantos como queramos, cada uno contemplando un escenario diferente.

```
switch (expresión) {  
    case valorA:  
        //código que se ejecuta si valorA es verdadero  
        break;  
    case valorB:  
        //código que se ejecuta si valorB es verdadero  
        break;  
}
```

```
{ código }
```

```
let edad = 5
```

```
switch (edad) {  
  case 10:  
    console.log('Tiene 10 años')  
    break;  
  case 5:  
    console.log('Tiene 5 años')  
    break;  
}
```

```
{ código }
```

```
let edad = 5
```

```
switch (edad) {  
  case 10:  
    console.log('Tiene 10 años')  
    break;  
  case 5:  
    console.log('Tiene 5 años')  
    break;  
}
```

Definimos la variable **edad** y le asignamos el número 5.

```
{ código }
```

```
let edad = 5
```

```
switch (edad) {
```

```
  case 10:
```

```
    console.log('Tiene 10 años')
```

```
    break;
```

```
  case 5:
```

```
    console.log('Tiene 5 años')
```

```
    break;
```

```
}
```

Iniciamos el condicional con la palabra reservada **switch** y, entre paréntesis, la expresión / condición que queremos evaluar.

En este caso vamos a evaluar **qué valor tiene la variable edad**.

```
{ código }
```

```
let edad = 5
```

```
switch (edad) {
```

```
  case 10:
```

```
    console.log('Tiene 10 años')
```

```
    break;
```

```
  case 5:
```

```
    console.log('Tiene 5 años')
```

```
    break;
```

```
}
```

Por cada caso escribimos la palabra reservada **case** y a continuación el valor que queremos evaluar.

En este caso, **preguntamos si** el valor de la variable **edad** es 10.

Como este caso **NO es verdadero**, Javascript ignora el código de este caso y pasa a evaluar el siguiente caso.

```
{ código }
```

```
let edad = 5
```

```
switch (edad) {
```

```
  case 10:
```

```
    console.log('Tiene 10 años')
```

```
    break;
```

```
  case 5:
```

```
    console.log('Tiene 5 años')
```

```
    break;
```

```
}
```

Este caso **ES verdadero**, por lo tanto **Javascript ejecuta el código** que está asociado: en este caso un `console.log()`.

La palabra reservada **break** corta las siguientes evaluaciones. Si quisiéramos que a pesar de ser verdadero este caso Javascript continúe evaluando los siguientes, simplemente no la escribimos.

¿Y SI NINGÚN CASO ES VERDADERO?

En ese caso introducimos la palabra reservada **default**, que nos permite tener un comportamiento por defecto cuando ningún caso evalúa como verdadero.

```
switch (expresión) {  
    case valorA:  
        // código que se ejecuta si valorA es verdadero  
        break;  
    default:  
        // código que se ejecuta si ningún caso es  
        verdadero  
}
```



```
{ código }
```

```
let fruta = 'wefwef';
```

```
switch (fruta) {  
  case 'manzana':  
    console.log('Qué rica es la manzana');  
    break;  
  case 'naranja':  
    console.log('¡Me encanta!');  
    break;  
  default:  
    console.log('No conozco esa fruta');  
    break;  
}
```

```
{ código }
```

```
let fruta = 'wefwef';
```

```
switch (fruta) {  
  case 'manzana':  
    console.log('Qué rica es la manzana')  
    break;  
  case 'naranja':  
    console.log('¡Me encanta!')  
    break;  
  default:  
    console.log('No conozco esa fruta');  
    break;  
}
```

Definimos la expresión que vamos a evaluar en el **switch**.

En este caso queremos preguntar por el valor de la variable **fruta**.

```
{ código }
```

```
let fruta = 'wefwef';
```

```
switch (fruta) {
```

```
  case 'manzana':
```

```
    console.log('Qué rica es la manzana');
```

```
    break;
```

```
  case 'naranja':
```

```
    console.log('¡Me encanta!')
```

```
    break;
```

```
  default:
```

```
    console.log('No conozco esa fruta');
```

```
    break;
```

```
}
```

Este caso **es falso**, por lo tanto no se ejecuta su código.

```
{ código }
```

```
let fruta = 'wefwef';
```

```
switch (fruta) {  
  case 'manzana':  
    console.log('Qué rica es la manzana');  
    break;  
  case 'naranja':  
    console.log('¡Me encanta!');  
    break;  
  default:  
    console.log('No conozco esa fruta');  
    break;  
}
```

Este caso también **es falso**, por lo tanto no se ejecuta su código.

```
{ código }
```

```
let fruta = 'wefwef';
```

```
switch (fruta) {  
  case 'manzana':  
    console.log('Qué rica es la manzana');  
    break;  
  case 'naranja':  
    console.log('¡Me encanta!');  
    break;  
  default:  
    console.log('No conozco esa fruta');  
    break;  
}
```

Como ningún caso evaluó a verdadero, se ejecuta el código dentro del bloque **default**.