

And / Or

&& y **||** son operadores lógicos utilizados para combinar condiciones en expresiones condicionales.

js

 Copiar código

```
if( 5 == 4 || 6 == 8 ){
  console.log("Si cumple la condición ");
}else{
  console.log("No cumple la condición ");
}
```

Switch

La sentencia **switch** en JavaScript es una estructura de control que te permite ejecutar diferentes bloques de código dependiendo del valor de una expresión. Es útil cuando tienes múltiples casos posibles y quieres ejecutar acciones específicas para cada uno.

Recuerda que es importante usar la palabra clave **break** después de cada caso para evitar que se ejecuten accidentalmente los bloques de código siguientes. Si no incluyes **break**, JavaScript continuará ejecutando los casos siguientes hasta encontrar un **break** o hasta llegar al final del switch.

También puedes usar el caso **default** como una opción por defecto si ninguno de los casos coinciden con el valor de la expresión. Esto te permite manejar casos no previstos de manera adecuada.

js

 Copiar código

```
variable1= 4;

switch (variable1){
  case 1:
    console.log("Variable es igual a uno");
    break;
  case 2:
    console.log("Variable es igual a dos");
    break;
  case 3:
    console.log("Variable es igual a tres");
    break;
  default:
    console.log("El valor no coincide con ninguno de los casos definidos");
}
```

js - Ejercicio 1

 Copiar código

```
// Ejercicio 1.
let creditos;
let ingreso = "Salas: ";
creditos = 2;

switch (creditos){
  case 1:
    console.log("Tiene acceso a la sala niños de 5 a 8 años");
    break;

  case 2:
    console.log("Tiene acceso a la sala niños de 7 a 9 años");
    break;

  case 3:
    console.log("Tiene acceso a la sala niños de 9 a 12 años");
    break;

  case 4:
    console.log("Tiene acceso a la sala niños de 12 a 15 años");
    break;

  default:
}
```

js - Ejercicio 2 / Usando Prompt

 Copiar código

```
let creditos;
let ingreso = "Salas: ";
creditos = parseInt(prompt("Cuantos creditos vas a comprar ?"));

switch (creditos){
  case 1:
    ingreso += " de niños a 5 a 8 años";
    break;

  case 2:
    ingreso += " de niños a 8 a 10 años";
    break;

  case 3:
    ingreso += " de niños a 10 a 13 años";
    break;


  case 4:
    ingreso += " de niños a 14 a 16 años";
    break;

  default:
}
alert(ingreso);
```

For

En este caso, se utiliza el **bucle for** para **calcular el factorial** de un número ingresado por el usuario. El bucle for se encarga de iterar sobre los números desde 1 hasta el número ingresado, multiplicándolos entre sí para calcular el factorial.

js

 Copiar código

```
// Factorial
let n;
n = parseInt(prompt("Ingresa tu número para hallar su factoría"));
var fact = 1;

for(let i = 1; i<= n; i++){
    fact *= i;
}
alert(" El factorial del número " + n + " es " + fact);
```

js - Factoria otra forma

 Copiar código

```
let num;
let result = ""; // Inicializamos la variable result
num = parseInt(prompt("Ingresa tu número para hallar su factoría"));

for (let i = 1; i <= num; i++){

    if (i % 2 === 0){
        result += i + "\n";
        alert(result);
    }
}
```

For of/in

El bucle **for...of** se utiliza para recorrer los **valores** de un iterable, como un array. Mientras tanto, el bucle **for...in** se usa para recorrer las **propiedades** de un **objeto**. El primero sigue el orden de los elementos, mientras que el segundo no garantiza un orden específico.

js

 Copiar código

```
let lista_personas = [
  {
    "nombre" : "carlos",
    "tipo" : "persona",
    "id" : "435",
  },
  {
```


```
"nombre": "hugo",
"tipo" : "persona",
"id" : "876",
},
{
  "nombre": "daniel",
  "tipo" : "persona",
  "id" : "673",
}
]
for(let personas of lista_personas){
  console.log(personas.nombre);
  console.log(personas.id);

  for (let propiedad in personas){
    console.log(propiedad);
  }
}
```

While

Es como una puerta que se abre y cierra según una condición. Imagina que tienes una tarea que quieres repetir, pero no sabes cuántas veces necesitarás hacerlo. Con `while`, puedes hacer que esa tarea se repita mientras se cumpla una condición específica.

js

 Copiar código

```
n = 0;
x = 0;

while (n < 3){
  n++;
  x += n;
}
console.log(x);

let contadorr = 0;
let iterarr = true;

while(iterarr == true){
  console.log(contadorr);
  if(contadorr == 5){
    iterarr = false;
  }
  contadorr++;
}
console.log(contador);
```

Enlace while: [click](#), Enlace Ejercicios: [click](#), Estudiar **ciclos**