**¿Para qué sirven los operadores lógicos?**

* **Controlar el flujo de ejecución:** Decidir qué parte del código se ejecuta en función de ciertas condiciones.
* **Crear expresiones condicionales:** Combinar múltiples condiciones para tomar decisiones más complejas.
* **Validar datos:** Verificar si los datos ingresados ​​por el usuario cumplen con ciertos criterios.

**Tipos de operadores lógicos:**

* **AND (&&):** Devuelve verdadero solo si ambas expresiones son verdaderas.
* **OR (||):** Devuelve verdadero si al menos una de las expresiones es verdadera.
* **NOT (!):** Invierte el valor de una expresión. Si es verdadera, la vuelve falsa, y viceversa.

**Ejemplos:**

JavaScript

// AND (&&)let edad = 25; let tieneLicencia = true;if (edad >= 18 && tieneLicencia) { console.log("Puedes conducir"); } else { console.log("No puedes conducir"); }// OR (||)let esDia = true; let estaLloviendo = false;if (esDia || !estaLloviendo) { console.log("Podemos salir a caminar"); }// NOT (!)let estaConectado = false;if (!estaConectado) { console.log("Debes conectarte a internet"); }

Utilice el código [con precaución.](file:///C:/faq#coding)

**Explicación de los ejemplos:**

* **Ejemplo 1:** Solo se imprimirá "Puedes conducir" si la edad es mayor o igual a 18 Y tiene licencia. Ambas condiciones deben cumplirse.
* **Ejemplo 2:** Se imprimirá "Podemos salir a caminar" si es de día O si no está lloviendo. Solo una de las condiciones necesita cumplirse.
* **Ejemplo 3:** Se imprimirá "Debes conectarte a internet" si la variable estaConectado es falsa. El operador NO invierte el valor.

**Aplicaciones comunes:**

* **Validación de formularios:** Verificar que los campos estén llenos, que los formatos sean correctos, etc.
* **Condicionales en bucles:** Decidir cuándo detener un bucle o realizar una acción específica.
* **Creación de funciones condicionales:** Definir el comportamiento de una función en función de diferentes parámetros.

**Resumen:**

Los operadores lógicos son herramientas poderosas en JavaScript que te permiten crear expresiones condicionales más complejas y controlar el flujo de ejecución de tu código. Dominar su uso es esencial para escribir programas más robustos y eficientes.

**¿Quieres ver más ejemplos o tienes alguna pregunta específica?**

**Ejercicios para practicar:**

1. Escribe una condición que verifique si un número es par y mayor que 10.
2. Crea una función que tome dos números como parámetros y devuelva "Ambos son positivos" si ambos son mayores que 0, "Ambos son negativos" si ambos son menores que 0, o "Son de distinto signo" en caso contrario.
3. Escribe un programa que pida al usuario su edad y si tiene licencia de conducir. Luego, muestre un mensaje indicando si puede alquilar un coche.