Operador Spread en objetos

En esta sección vamos a ver la diferencia al comparar objetos con los operadores de igualdad e identidad, también aplicaremos el operador spread para la clonación de objetos.

Operador de igualdad (==)

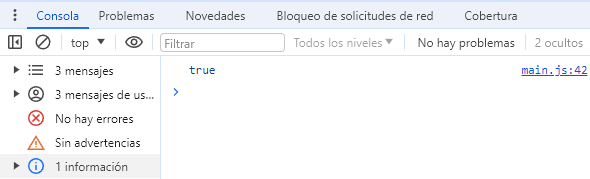
El operador **==** se emplea para comparar dos valores y devuelve **true** si son iguales, o **false** en caso contrario. Sin embargo, es crucial comprender su comportamiento, especialmente con objetos y referencias.

Cuando se utiliza **==** para comparar objetos, se están comparando las referencias a los objetos, no sus valores internos. Por lo tanto, dos variables que apuntan al mismo objeto en la memoria darán como resultado **true** al usar **==**.

Ten en cuenta que el objeto **invoice** es el mismo de la sección anterior, además no se van a realizar cambios dentro del objeto **invoice**.



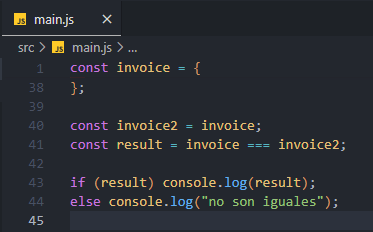


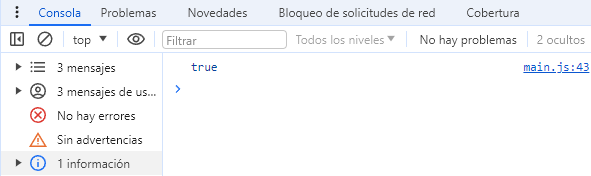


En el ejemplo mostrado, **invoice** y **invoice2** hacen referencia al mismo objeto en la memoria, por lo tanto, la comparación devuelve **true**.

Operador de identidad (===)

El operador **===** compara dos valores teniendo en cuenta tanto su contenido como su tipo de dato. A diferencia de **==**, que solo evalúa el contenido, **===** verifica que ambos valores sean idénticos en tipo y contenido.



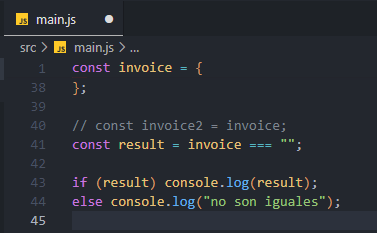


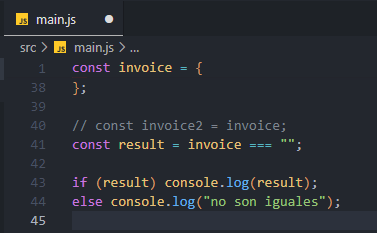
En el ejemplo mostrado, **invoice** y **invoice2** son idénticos en tipo y contenido, por lo que la comparación devuelve **true**.

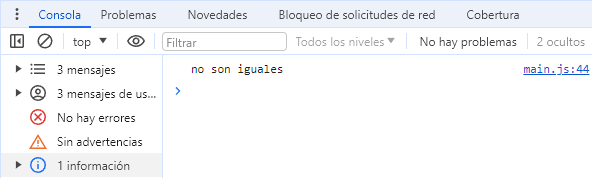
**Nota:** Cuando la operación en un condicional **if** es de una sola línea, se pueden omitir las llaves **{ }**.

Objetos que no son idénticos

Es esencial tener en cuenta que, al utilizar **===**, incluso una pequeña diferencia en el tipo de dato o contenido hará que la comparación devuelva **false**.







En el ejemplo mostrado, **invoice** es un objeto y **" "** es una cadena de texto, por lo que la comparación devuelve **false**.

Problema de referencia en objetos

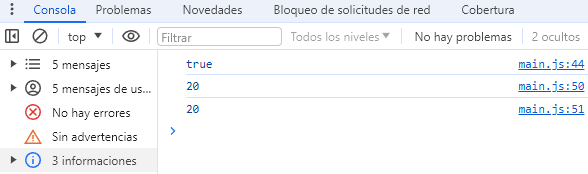
Cuando se asigna un objeto a otra variable usando el operador de asignación (**=**), ambas variables apuntan a la misma instancia del objeto en la memoria. Esto significa que cualquier cambio realizado a través de una de las variables se reflejará en la otra.

const invoice = {
  id: 10,
  name: "Compras de oficina",
  date: new Date(),
  client: {
    id: 2,
    name: "Jhon",
    lastname: "Doe",
    age: 20,
  },
  items: [
    {
      product: "keyboard",
      price: 399,
      quantity: 2,
    },
    {
      product: "mouse",
      price: 200,
      quantity: 1,
    },
    {
      product: "papel",
      price: 100,
      quantity: 10,
    },
  ],
  total: function () {
    let total = 0;
    this.items.forEach((item) => {
      total = total + item.price * item.quantity;
    });
    return total;
  },
  greeting: () => {
    return `Hola ${invoice.client.name}`;
  },
};

const invoice2 = invoice;
const result = invoice === invoice2;

if (result) {
  console.log(result);
} else {
  console.log("no son iguales");
}

invoice2.id = 20;
console.log(invoice.id);
console.log(invoice2.id);

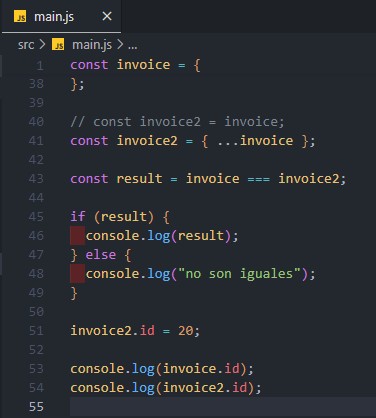


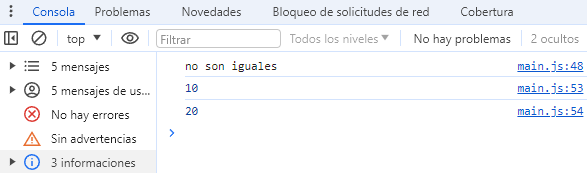
En el ejemplo mostrado, ambos objetos (**invoice** e **invoice2**) son idénticos en contenido y referencia. Ambos objetos muestran el valor **20** al imprimir el valor de su atributo **id**, confirmando que ambas variables apuntan al mismo objeto.

Clonación de objetos con el operador Spread

El operador spread (**...**) se emplea para crear una copia superficial o una instancia de un objeto con sus respectivos atributos. Esto nos permite clonar el objeto original, evitando el problema de referencia.

Al clonar un objeto mediante el operador spread, los dos objetos resultantes son distintos, aunque tengan el mismo contenido.





En el ejemplo mostrado, **invoice** y **invoice2** son dos objetos distintos y cualquier modificación en **invoice2** (como el valor de su atributo **id**) no afectará a **invoice**.

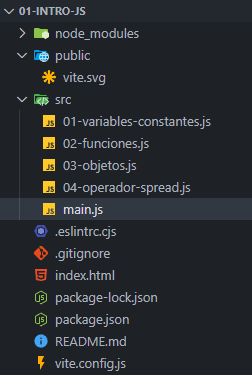
Antes de pasar al siguiente tema

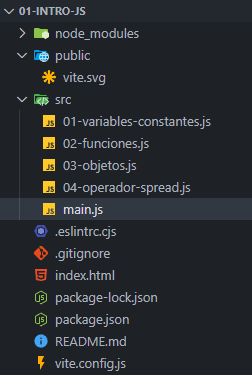
Realiza estas acciones en Visual Studio Code:

1. Abre el panel izquierdo y busca el archivo **main.js**.
2. Haz clic derecho en el archivo y selecciona **Cambiar nombre**.
3. Cambia el nombre del archivo a **04-operador-spread.js** y presiona **Enter**.
4. Ahora, crea un nuevo archivo, dentro la carpeta raíz **src**, haciendo clic en el botón **Nuevo archivo** en la parte superior del panel izquierdo.



1. Nombre el nuevo archivo como **main.js** y presiona **Enter**.





A partir de este momento, se realizarán cambios en el archivo **main.js** que has creado.