



2.A. Estructura del lenguaje JavaScript

5.- Operadores.

5.1.- Operadores de comparación.

Operadores de comparación en JavaScript



Operadores de comparación en JavaScript

Sintaxis	Nombre	Tipos de operandos	Resultados
==	Igualdad.	Todos.	Boolean.
!=	Distinto.	Todos.	Boolean.
===	Igualdad estricta.	Todos.	Boolean.
!==	Desigualdad estricta.	Todos.	Boolean.
>	Mayor que .	Todos.	Boolean.
>=	Mayor o igual que.	Todos.	Boolean.
<	Menor que.	Todos.	Boolean.
<=	Menor o igual que.	Todos.	Boolean.

En valores numéricos, los resultados serían los mismos que obtendríamos con cálculos algebraicos. Por ejemplo:

```
30 == 30          // true
30 == 30.0        // true
5  != 8           // true
9  > 13           // false
7.29 <= 7.28      // false
```

También podríamos comparar cadenas a este nivel:

```
"Marta" == "Marta"    // true
"Marta" == "marta"    // false
"Marta" > "marta"     // false
"Mark" < "Marta"      // true
```

Para poder comparar cadenas, JavaScript convierte cada carácter de la cadena de un string, en su correspondiente valor ASCII. Cada letra, comenzando con la primera del operando de la izquierda, se compara con su correspondiente letra en el operando de la derecha. Los valores ASCII de las letras mayúsculas, son más pequeños que sus correspondientes en minúscula, por esa razón "Marta" no es mayor que "marta". En JavaScript hay que tener muy en cuenta la sensibilidad a mayúsculas y minúsculas.

Si por ejemplo comparamos un número con su cadena correspondiente:

```
"123" == 123          // true
```

JavaScript cuando realiza esta comparación, convierte la cadena en su número correspondiente y luego realiza la comparación. También dispones de otra opción, que consiste en convertir mediante las funciones `parseInt()` o `parseFloat()` el operando correspondiente:

```
parseInt("123") == 123 // true
```

Los operadores `===` y `!==` comparan tanto el dato como el tipo de dato. El operador `===` sólo devolverá true, cuando los dos operandos son del mismo tipo de datos (por ejemplo ambos son números) y tienen el mismo valor.

