

MIND HUB.

An aerial night photograph of a city, with a prominent multi-lane highway running vertically through the center. The highway is filled with light trails from cars, creating a bright, streaky path. The surrounding city is densely packed with lights, and the overall image has a strong blue color cast. Various geometric shapes and lines are overlaid on the image, including a large 'U' on the left, a checkered pattern at the bottom left, and several brackets and dashes in pink and white.

OPERADORES

Índice

1 Operadores de Asignación

2 Operadores Aritméticos

3 Operadores de Comparación

4 Operadores Lógicos

Operadores de Asignación o acción

son usados para *asignar valores* a las variables de JavaScript, se asigna o reasigna con el signo =

```
//asignación: defino variables
```

```
let nombre = "ignacio"
```

```
let edad = 32
```

```
const nacimiento = 1990
```

```
//reasignación: modifiko variables
```

```
nombre = "igna"
```

```
edad = 37
```

```
edad = edad + 5
```

```
edad += 5 //es lo mismo
```

Operadores Aritméticos

son usados para **realizar operaciones aritméticas** entre variables y/o valores (suma, resta, división, multiplicación)

```
//aritmética

let añoActual = nacimiento + edad //suma con +

let nombreCompleto = nombre + "mindhub" //suma con +

let haceCincoAños = añoActual - 5 //resta con -

let cantidadDiasVividos = edad * 365 //producto con *

let mitadDeEdad = edad / 2 //división con /
```

El operador **+** también puede ser usado para agregar (**concatenar**) strings.

Operadores de Comparación

son usados para **determinar diferencias o igualdades** entre variables/valores y siempre retornan booleanos (**true/false**)

```
//comparar con signo mayor >

let comparacion1 = 3 > 5 //false

let comparacion2 = 100 > 5 //true

//comparar con signo menor <

let comparacion3 = 17 < 52 //true

let comparacion4 = 17 < 5 //false
```

```
//comparar variables

let num1 = 15

let num2 = 25

let comparacion5 = num1 < 100 //true

let comparacion6 = num2 > 50 //false

let comparacion7 = num1 < num2 //true
```

Para poder incluir el valor de comparación, existen: **>=** (mayor o igual) **<=** (menor o igual)

Operadores de Comparación

son usados para **determinar diferencia o igualdad** entre variables/valores y siempre retornan booleanos (**true/false**)

```
//comparar con ==  
  
let num1 = "70"  
  
let comp1 = 70 == 19 //false  
  
let comp2 = 70 == 70 //true  
  
let comp3 = 70 == "70" //true  
  
let comp4 = 70 == num1 //true
```

```
//comparar con ===  
  
let num2 = 70  
  
let comp5 = 70 === num2 //true  
  
let comp6 = num1 === "70" //true  
  
let comp7 = 70 === "70" //false  
  
let comp8 = num1 === num2 //false
```

RECORDATORIO: = asigna == compara (solo valor) === compara estrictamente (tipo de dato)

Operadores de Comparación

son usados para **determinar diferencia o igualdad** entre variables/valores y siempre retornan booleanos (**true/false**)

```
//comparar con !=  
  
let comp1 = 70 != 19 //true  
  
let comp2 = 70 != "opeador" //true  
  
let comp3 = 70 != "70" //false  
  
let comp4 = 70 != 70 //false
```

```
//comparar con !==  
  
let comp5 = 70 !== 70 //false  
  
let comp6 = 70 !== "70" //true  
  
let comp7 = 70 !== "operador" //true  
  
let comp8 = 70 !== 100 //true
```

RECORDATORIO:

!= distinto de

!== distinto estricto de

Operadores Lógicos

son usados para **combinar múltiples expresiones booleanas** y siempre retornan booleanos (**true/false**)

```
//combinar con &&
let num1 = 15
let num2 = 25

let comp1 = (num1>=15) && (num2<50)
           true    &&    true
           true

let comp2 = num1>10 && num2==="25"
           true &&   false
           false
```

```
//combinar con ||
let comp3 = num1>10 || num2==="25"
           true  ||   false
           true

let comp4 = num1==="15" || num2<50
           true

let comp5 = num1==="15" || num1<0
           false
```

RECORDATORIO: && AND (y) || OR (o) ! NOT (no)

Operadores Lógicos

son usados para **combinar múltiples expresiones booleanas** y siempre retornan booleanos (**true/false**)

```
let comp6 = (num1 != "10") && (num1 > 100) && (num2 <= 25) //false
              true      &&      false      &&      true

let comp10 = ( (num1 >= 15) || (num1 == 10) ) && (num2 != 0) //true
              ( true      ||      false )      &&      true
              true                                &&      true

let comp10 = ((num1 >= 15) || (num1 == 10)) && ((num2 != 0) && (num2 >= 20))
              ( true      ||      false )      &&      ( true      &&      true )
              true                                &&                                true
              true
```

RECORDATORIO: && AND (y) || OR (o) ! NOT (no)

Uso de Operadores

+	suma dos números o añade dos cadenas: si se agrega más de un tipo de variable, incluida una string anexa a un número o viceversa, el resultado será una string
-	resta el segundo número del primero
/	divide el primer número por el segundo
*	multiplica dos números
%	divide el primer número por el segundo y retorna el resto
=	asigna el valor de la derecha al objeto en la izquierda
+=	el objeto en la izquierda = el objeto en la izquierda + el valor en la derecha - esto también funciona al añadir strings
-=	el objeto en la izquierda = el objeto en la izquierda - el valor en la derecha
>	el número a la izquierda debe ser más grande que el número a la derecha -esto también funciona para strings y valores
<	el número a la izquierda debe ser menor que el número a la derecha -esto también funciona para strings y valores
>=	el número a la izquierda debe ser más grande o igual que el número a la derecha -esto también funciona para strings y valores
<=	el número a la izquierda debe ser menor o igual que el número a la derecha -esto también funciona para strings y valores

++	incrementa el número
--	decrece el número
==	los números u objetos o valores deben ser iguales
!=	los números u objetos o valores no deben ser iguales
!	logical NOT (la declaración no debe ser true)
&&	logical AND (ambas declaraciones deben ser true)
	logical OR (alguna de las declaraciones debe ser true)
===	los números u objetos o valores deben ser iguales, y deben ser del mismo tipo de variable
!==	los números u objetos o valores no deben ser iguales, o no deben ser del mismo tipo de variable

¡Muchas gracias!

MIND
HUB.