

OPERADORES LÓGICOS:

ASIGNACIÓN DE VALORES POR DEFECTO CON || (OR)

En funciones se puede utilizar para asignar valores por defecto

Ejemplo

```
Const port = process.env.PORT || 5000;
```

Si no está definida process.env.PORT toma el valor de 5000

OPERADORES TERNARIOS

OPERADORES INCREMENTALES

a++ incremento posterior

```
let x=10
```

```
let y=x++      se incrementa x después de asignarla a y
```

```
console.log(x,y)
```

++a incremento anterior

```
let x=10
```

```
let y=++x      se incrementa x antes de asignarla a y
```

```
console.log (x,y)
```

OPERADORES DE TIPO

Posibles valores de typeof

- Undefined
- Object
- Boolean
- Number
- BigInt
- String
- Symbol
- Function

OPERADORES PRECEDENCIA DE OPERADORES. Introducción

La precedencia de operadores determina el orden en el cual los operadores son evaluados uno respecto del otro. Los operadores con mayor precedencia se convierten en los operandos de los operadores con menor precedencia

```
let year= 2050;
```

```
let newYear = year = 2051; // se ejecuta de derecha a izquierda
```

significa lo mismo que:

```
let year = 2050;
```

```
year= 2051;
```

```
let newYear= year;
```

TIPOS DE DATOS – PRINCIPALES TIPOS

JavaScript tiene “tipificación débil y dinámica”, es decir, que se puede cambiar el tipo de un dato y realizar conversiones automáticamente al hacer operaciones como comparaciones, operaciones aritméticas, de cadena..

TIPOS DE DATOS – CONVERSIONES ENTRE TIPOS (CASTING)

JS es interpretado y débilmente tipado por lo que se asignará por defecto a una variable el tipo de dato más acertado

Existen funciones de conversión (métodos globales). `String()` `toString()` (hace lo mismo) y `Number()`

Además existen:

`parseInt()` → convierte cadena a entero

`parseFloat()` → convierte cadena a decimal

`toDateStr()` → convierte variable de tipo fecha a String

```
let fecha = new Date();
```

```
let cadena= fecha.toDateStr (); [También fecha.toString()]
```

TIPOS DE DATOS – CASTING

Sirven para cambiar una variable de tipo:

De String a Number

- `Number(variable)`
- `Variable parseInt()`
- `Variable parseFloat()`

De Number a String

- `String(variable)`
- `Variable.toString()`

CONVERSIONES IMPLICITA

Tipos de dato COERCIÓN

TIPOS DE DATOS - TEMPLATE LITERALS

Uso de ` (acento grave) en cadenas

Sirve para poner dentro de una cadena variables y expresiones que serán evaluadas al ponerlas dentro de `\${}`

Se respetan los saltos de línea, tabuladores, etc. Que haya dentro

```
let edad=25;
```

```
console.log(` El visitante tiene: ${edad} años` )
```

Mostrará en la consola:

El visitante tiene:

25 años

TIPOS DE DATOS – OPERACIONES ESPECIALES

- Valor no definido (undefined). Valor predeterminado de todas las variables después de una declaración sin asignación de valor.
- Cadena vacía (“”)
- Valor nulo (null). Es “nada” (algo que no existe en cualquier lenguaje de programación). Variable sin un valor significativo.
- Generalmente se le asigna a lo que va a ser después de un objeto.
- NaN (Not a Number): indica que el resultado de la operación no puede ser convertido a un número.

TIPOS DE DATOS – BOOLEAN()

Existen funciones de conversión: Boolean()

El valor false siempre se devuelve para:

- 0
- Nan (Not a number)
- cadena vacía
- undefined
- null

TIPOS DE DATOS – PRECISIÓN DE NÚMEROS Y CONVERSIONES

Método toString()

Convierte número en cadena de caracteres

```
NombreVariable.toString;  
67).toString();
```

Método toFixed() que devuelve este numero toFixed 2 ejemplo de ejercicio en EXAMEN

Ajuste de decimales

```
var numero= 9.99;  
numero.toFixed(0);  
    //devuelve 10.  
(redondea)  
numero.toFixed(1);  
    //devuelve 10.0  
(redondea)  
numero.toFixed(2);  
    //devuelve 9.99  
Numero.toFixed(3);  
    //devuelve 9.990
```

Método toPrecision{}

Ajusta la precisión de un número

```
var numero=9.99  
numero.toPrecision(0);  
    //devuelve 9.99  
numero.toPrecision(1);  
    //devuelve 1e+1  
numero.toPrecision(2);  
    //devuelve 10;  
numero.toPrecision(3);  
    //devuelve 9.99;
```

TIPOS DE DATOS – ENTRADA DE DATOS

- `window.prompt`

Si no se introduce valor, guarda “null” en la variable

```
let libroLeido = prompt (“¿Cuál es el título del último libro que has leído?”, default);
```

- `<p> Introduzca edad y pulse botón: </p>`
- `<input id =”edad” value=”18”/>`

Window.confirm dato para confirmar

TIPOS DE DATOS – EJEMPLO

Pedir nombre

Mostrarlo por consola dentro de un saludo

Repetir utilizando “template literals”

Mostrar el tipo de variable que es

```
<script>
```

```
var nombre = prompt("Dime tu nombre");
```

```
console.log(` Hola, ${nombre} `);
```

```
console.log(` El tipo de variable es: ${typeof nombre} `);
```

```
</script>
```