



Extracto de las clases

Introducción a JavaScript (Clases 2 – 5)
Mentor: Joshua Eduardo González Ruíz

ÍNDICE

- Operadores y estructuras condicionales pág. 3
- Funciones y métodos String pág. 14
- Estructuras cíclicas pág. 21
- Referencias pág. 23

Operadores y estructuras condicionales



- Asignar un valor a una variable:
 - $x = 4; c = 2;$
- Asignar el resultado de sumar a una variable el valor de otra:
 - $x += c;$
- Asignar el resultado de restarle a una variable el valor de otra:
 - $x -= c;$
- Asignar el resultado de multiplicarle a una variable el valor de otra:
 - $x *= c;$

- Asignar el resultado de dividirlo a una variable el valor de otra:
 - `x /= c;`
- Asignar el residuo de dividirlo a una variable el valor de otra:
 - `x %= c;`
- Comparar dos variables y verificar si contienen el mismo valor sin importar su tipo de dato:
 - `x == c;`
- Comparar dos variables y verificar si contienen el mismo valor y tipo de dato:
 - `x === c;`

- Comparar dos variables y verificar si no contienen el mismo valor sin importar su tipo de dato:
 - `x != c;`
- Comparar dos variables y verificar si no contienen el mismo valor y tipo de dato:
 - `x !== c;`
- Comparar dos variables y verificar si uno es mayor que el otro:
 - `x > c;`
- Comparar dos variables y verificar si uno es menor que el otro:
 - `x < c;`

- Comparar dos variables y verificar si uno es mayor o igual que el otro:
 - `x >= c;`
- Comparar dos variables y verificar si uno es menor o igual que el otro:
 - `x <= c;`
- Incrementar el valor de una variable en una unidad antes o después:
 - `++x;` //establece el valor de x en 5 y eso devuelve x.
 - `x++;` //devuelve el valor de x en ese momento [4] y luego establece en x el valor 5.

- Decrementar el valor de una variable en una unidad antes o después:
 - `--x;` //establece el valor de x en 3 y eso devuelve x.
 - `x--;` //regresa el valor de x en ese momento [4] y luego establece en x el valor 3.
- Convertir el valor numérico de alguna variable en negativo:
 - `-x;`
- Tratar de convertir a un número el valor de alguna variable:
 - `+x;`
- Convertir el valor booleano positivo de alguna variable a negativo y viceversa:
 - `!x;`

- Evaluar alguna condición y realizar algún proceso síncrono u asíncrono si se cumple o no:
 - `if(x > c) { console.log("La variable x es mayor"); } else { console.log("La variable c es mayor"); }`
 - `if(x === 4 && c == 2) { console.log("Ambas condiciones se cumplen"); } else { console.log("Alguna de las condiciones no se cumple"); }`
 - `if(x === 4 || c == 2) { console.log("Alguna de las condiciones se cumple"); } else { console.log("Ninguna de las condiciones se cumple"); }`

- Evaluar alguna expresión y realizar algún proceso síncrono u asíncrono si existe algún caso que coincida con la misma:

- **switch(c) {**

- case 1:**

- console.log**("El valor ingresado es el 1");

- break;**

- case 2:**

- console.log**("El valor ingresado es el 2");

- break;**

- default:**

- console.log**("El valor ingresado no está entre las opciones

- disponibles");**

- }**

- Anidar evaluación de condiciones:

- `if(x > c) {`

- `console.log("La variable x es mayor");`

- `if(x === 4 && c == 2) {`

- `console.log("Ambas condiciones se cumplen");`

- `} else {`

- `console.log("Alguna de las condiciones no se cumple");`

- `if(x === 4 || c == 2) {`

- `console.log("Alguna de las condiciones se cumple");`

- `} else {`

- `console.log("Ninguna de las condiciones se cumple");`

- `}`

- `}`

- `} else {`

- `console.log("La variable c es mayor");`

- `}`

- Concatenar evaluación de condiciones:

- `if(x > c) {`

- `console.log("La variable x es mayor");`

- `} else if(x === 4 && c == 2) {`

- `console.log("Ambas condiciones se cumplen");`

- `} else if(x === 4 || c == 2) {`

- `console.log("Alguna de las condiciones se cumple");`

- `} else {`

- `console.log("Ninguna de las condiciones se cumple");`

- `}`

- Realizar la evaluación de alguna condición en una línea:
 - `x > c ? console.log("La variable x es mayor") :`
`console.log("Ninguna de las condiciones se cumple")`

Funciones y métodos String



- Realizar una función usando la forma general y llamarla para que se ejecute:

- ```
function showSum(x, c) {
 let result = x + c;
 console.log(result);
 return result;
}
```

```
let sum = showSum(x, c);
```

- Anidar funciones para realizar diferentes procesos síncronos o asíncronos:

- ```
function showSum(x, c) {  
    let result = x + c;  
    function printSum(result){  
        console.log(result);  
    }  
    return printSum(result);  
}
```

```
let sum = showSum(x, c);
```


- Muestras de Strings primitivos:
 - `let a = "Hola Koders";`
 - `let b = 'Feliz 2k22';`
 - `let d = `Espero todos sus deseos y propósitos se cumplan`;`
- Muestra de un objeto String:
 - `let e = new String("Vamos por más logros juntos");`
- Obtener un caracter de alguna cadena de texto:
 - `a.charAt(2);`
 - `b[4]`
 - `c.at(6);` //Uso ya no recomendado

- Crear una cadena de texto extensa:
 - `let a = "Hola Koders," + "les deseo un feliz 2k22. " + "Espero que todos sus deseos y propósitos se cumplan," + "vamos por más logros juntos";`
 - `let a = "Hola Koders, \les deseo un feliz 2k22. \Espero que todos sus deseos y propósitos se cumplan, \vamos por más logros juntos";`
- Obtener los caracteres de una cadena dado un índice desde el cual van a empezar y un índice que es el que no aparecerá:
 - `a.slice(2, 6);` //índices positivos y negativos
 - `a.substring(2, 6);` //solo índices positivos
- Concatenar cadenas de texto:
 - `a.concat(" ", b);`
- Verificar si existe alguna palabra en una cadena de texto:
 - `a.includes("Hola");`

- Reemplazar alguna palabra por otra (1 vez) en una cadena de texto:
 - `a.replace("Hola", "Hola de nuevo");`
- Crear un arreglo a partir de una cadena de texto con base en algún parámetro incluido en la misma:
 - `a.split(" ");`
- Eliminar espacios al principio y al final de una cadena de texto:
 - `a.trim();`
- Convertir todos los caracteres de una cadena de texto en mayúsculas o minúsculas:
 - `a.toUpperCase();`
 - `a.toLowerCase();`
- Tratar de convertir cualquier tipo de dato a cadena de texto:
 - `x.toString();`

- Reemplazar alguna palabra por otra en una cadena de texto:
 - `a.replaceAll(" ", "");`
- Encontrar el índice donde se encuentra la primera coincidencia de un caracter o el inicio de una palabra en una cadena de texto:
 - `a.indexOf("Koders");`

Estructuras cíclicas



- Iterar un proceso mientras la condición principal se cumpla:
 - `while(x > c) { console.log("La x es mayor que la c"); }`
- Iterar un proceso por lo menos una vez y después verificar se cumpla una condición para volverlo a ejecutar:
 - `do { console.log("La x es mayor que la c"); } while(x > c)`
- Iterar un proceso siempre y cuando la condición lo permita con base en un índice previamente especificado en el ciclo:
 - `for(let i = 0; i < x; i++){ console.log(a); }`

Referencias

- **Expresiones y operadores – JavaScript | MDN. (2021, 15 noviembre). MDN Web Docs.**
https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Expressions_and_Operators
- **Tomando decisiones en tu código – condicionales – Aprende sobre desarrollo web | MDN. (2021, 17 noviembre). MDN Web Docs.**
https://developer.mozilla.org/es/docs/Learn/JavaScript/Building_blocks/conditionals
- **String – Cadena de caracteres – JavaScript | MDN. (2021, 20 noviembre). MDN Web Docs.**
https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Global_Objects/String
- **while – JavaScript | MDN. (2021, August 18). MDN Web Docs.**
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/while>

- **do...while** – JavaScript | MDN. (2021, August 18). MDN Web Docs.
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/do...while>
- **for** – JavaScript | MDN. (2021, August 18). MDN Web Docs.
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for>
- **for...in** – JavaScript | MDN. (2021, July 20). MDN Web Docs.
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...in>
- **for...of** – JavaScript | MDN. (2021, October 30). MDN Web Docs.
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...of>
- **for await...of** – JavaScript | MDN. (2021, October 22). MDN Web Docs.
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for-await...of>