

Vamos a comenzar ahora a ver efectivamente qué podemos hacer con un lenguaje de programación como Javascript.

Hasta el momento hemos maquetado un sitio web, pero queremos convertirlo en algo que tenga no solamente una estética y una estructura, sino también vida, que reaccione a la interacción del usuario. Para eso, entre utilizarlo además para otras muchas cosas, vamos a utilizar Javascript.

Comenzamos con los conceptos generales.

Perfecto, tenemos nuestro editor de código favorito, por lo menos el que usamos en este curso, Visual Studio Code, abierto en una nueva carpeta, la que hemos nombrado aquí.

Bueno, de esta manera están viéndolo ustedes ahí arriba, a la izquierda, no importa el nombre ¿correcto?.

Para empezar a trabajar con Javascript, vamos a tener en cuenta que como es un lenguaje de programación, que es un lenguaje interpretado, no necesitaremos descargarlo ni compilarlo, ni hacer un paso previo, ni posterior. Con escribir directamente en nuestro navegador o en nuestro editor de código, “código Javascript”, ya podremos correrlo directamente, de manera automática, digámoslo así: Javascript ya está incluido en vuestras computadoras, en vuestros navegadores, tabletas y teléfonos. Así que, podríamos comenzar escribiendo un archivo HTML para vincularlo, “index.HTML”, aquí vamos a escribir simplemente un archivo muy sencillo, “Introducción a Javascript”. Y aquí podríamos poner, por ejemplo, un título “Conceptos básicos”. Perfecto, bien.

Nosotros vamos a ver este documento aquí, lo vamos a abrir y verán ustedes que, por supuesto, no contiene nada, salvo lo que hemos escrito recién. Ahora, podemos escribir Javascript directamente en nuestro documento, abriendo aquí unas etiquetas “script” y escribiendo directamente nuestro código, aquí, entre estas etiquetas. Aunque la manera más prolija es, por supuesto, utilizar un archivo vinculado. Para eso, lo que haremos es cargar un un archivo externo, ¿okey?, vamos a cargar un archivo externo de Javascript, vamos a indicar que nuestro archivo se llama, por ejemplo, “index js”, bien, ya tenemos entonces nuestro archivo “index.html” vinculado a un archivo “index.javascript” que, de momento, no existe, pero que podemos crear aquí mismo junto al archivo index. Vamos a indicar “Index punto Javascript”, debe llevar el mismo nombre, no puede llevar cualquier nombre, simplemente elegí ese. Bueno, como podría haber elegido otro, ¿verdad? Sí, obviamente tiene que tener una extensión “punto js”, para decirle a nuestro sistema que se trata de un archivo Javascript,

Correcto. Vamos entonces a ir a la sintaxis básica de Javascript y algunos conceptos muy generales. Para comenzar el primero, no se tienen en cuenta los espacios en blanco, ni los las nuevas líneas, por ejemplo, si escribimos así, espacios, esto no va a ser tenido en cuenta, es igual que HTML. Se distingue entre mayúsculas y minúsculas, por lo tanto, no es lo mismo, por ejemplo, decir “variable” que decir “Variable” o decir “VARIABLE”, ¿okey?, son cosas distintas y eso tenemos que tenerlo en cuenta.

Podemos escribir comentarios con dos barras inclinadas, esto es un comentario de una línea y también podemos escribir comentarios de múltiples líneas, de este modo todo lo que esté aquí, será tenido en cuenta como un comentario y no, digamos, no nos dará un error. Si escribiéramos esto así, sin indicar que es un comentario, tendríamos un error, porque Javascript no reconocería esto. Okey, muy bien, vamos a algo muy importante que son las variables. Vamos a hablar de variables.

Variables, ¿qué son las variables?, son cajas en las que, o contenedores, donde podemos almacenar información. Podríamos decir que es un insumo básico para el trabajo con nuestro código. Entonces, primero deberemos definir una variable. ¿Cómo podemos definir una variable? Bueno, de tres maneras, aunque usaremos básicamente dos, luego, un poquitito más adelante, les explicaré por qué. Por ahora sepan que, para definir una variable, podemos utilizar las siguientes palabras clave “var”, “let”, oops, y “const”. Vamos a usar las tres y veremos si hay alguna diferencia.

La manera de crear una variable es declarándola, ¿cómo se declara una variable? Con una de las palabras reservadas “var”, “let” o “const” y el nombre que deseemos asignarle, ¿okey? No podemos utilizar palabras reservadas del lenguaje, todos los lenguajes tienen una lista de palabras reservadas, por ejemplo, “var” es una palabra reservada, porque se utiliza para definir, para designar una nueva variable, por lo tanto, no podría hacer esto, no podría ponerle “var” a una variable de nombre, porque el intérprete aquí no sabría exactamente qué quiero hacer, pero sí podría ponerle “variable”, por ejemplo.

Seamos un poco más descriptivos. Digamos que esta variable va a ser igual a “number” o, mejor dicho, el nombre va a ser “number”, ¿okey? Aquí acabo de definir una variable que se llama “number”, vamos a definir, o a declarar, una segunda variable, no podría hacer esto, porque ya existe una variable con ese nombre, podríamos poner “number two” y podríamos también escribir “const number three”, okey, “const” y aquí sí o sí, en este caso, estoy obligado, no solamente a declarar la variable, sino también a inicializarla. El caso de “const” es un caso particular y ya veremos por qué.

Esta primera parte se llama “declarar la variable”, le damos existencia y decimos que este es su nombre, pero no tiene contenido aún, luego con un signo igual, haremos una asignación. Cuando asignemos un valor, podremos decir que esa variable habrá sido inicializada.

Aquí podríamos poner, por ejemplo, 120, da igual, okey, bien. Y aquí podemos ver que “number” en realidad, pongámosle “tres” para no tener problemas, más, más adelante. Podríamos decir que “number tres” tiene el valor 120, okey.

Okey, nota para el editor, tuve que levantarme a cerrar una ventana porque comenzaron en una obra en construcción con unas máquinas infernales, perdón por eso.

Retomo.

Por otra parte, aquí “number one” y “number two” no tienen ningún valor. Vamos a asignarles un valor inicial aquí podríamos decir que este es 118 y aquí 119, no importa el valor ni la correlación, ¿verdad? Aquí puede ser cualquier número. Perfecto, entonces tenemos tres variables, tres espacios en memoria con un nombre propio y un valor asociado, ¿cómo podemos saber, conocer, qué valores tienen estas variables? Bueno, tenemos una herramienta muy práctica para los desarrolladores, que es la instrucción “console log”, ahora veremos de qué se trata.

Podemos escribir, por ejemplo aquí “console log”, “number” y esto va a darnos por pantalla el resultado 118. Si escribimos “number dos” y “number tres”, obtendremos 119 y 120, okey, esta instrucción, “console log” es una función incluida dentro de Javascript que se utiliza para el trabajo interno nunca le mostraremos a un usuario información de esta manera, pero sí es muy útil para los desarrolladores. Por lo tanto si queremos inspeccionar el valor de algunas de estas variables, sólo deberemos abrir nuestro “index.js”, vamos a abrirlo aquí en un costado, perdón, nuestro “index.html”, y aquí podemos dar click derecho, inspeccionar y esto nos abrirá la consola.

La consola de desarrollo, una herramienta importante, reitero, para los desarrolladores.

Podemos venir aquí “console log” y fíjense que aquí nos aparece 118, 119 y 120, con una

indicación al lado. Esto quiere decir que en esta línea, en la línea seis del archivo "index.js", se ejecutó una instrucción cuya salida es esta que está aquí. Si inspeccionamos la línea seis, es el "console log", para mostrar el valor de "number", de la variable en "number". Bien, cuando una variable se crea, decimos que está declarada cuando además de crearse se inicializa, con el signo igual y un valor, decimos que es una variable inicializada. Tengan en cuenta lo siguiente, cuando declaramos una variable, pero no le ponemos un valor, Javascript le va a asignar por defecto el valor "undefined", indefinido.

Okey, si ahora guardamos este archivo, verán que en la línea seis ya no tiene un valor, tampoco le estamos poniendo este valor. Esto es un comentario, esta línea es un comentario, simplemente tendrá ese valor porque Javascript se lo asignará por defecto. Bien, entonces este es un concepto importante, tengan en cuenta que cuando declaramos una variable, más no la inicializamos, su valor será "undefined".

Ahora, ¿cómo podemos mutar las variables?, simplemente asignándoles un valor con el operador. Igual, aquí podríamos decir que el "number", que ha sido creada, ha sido declarada aquí arriba, pero nunca fue inicializada bueno ahora va a ser inicializada con un valor, le vamos a asignar el valor 10.

Okey y ahora nuestro "console log", este de aquí, ya nos dirá que el valor no es ni 118, ni "undefined", es 10, porque este es el valor que le hemos agregado. Ahora, hoy cuando intentamos declarar el número "tres", con la palabra reservada "const", sin inicializarla, no lo permitió, nos marca un error. De hecho sí guardo aquí, veremos un problema y nos dice "error de sintaxis", está, falta, "missing", falta, el inicializador en una declaración de constante. Porque "var" no necesita un valor inicial y tampoco "let", pero si "const". Okey, "const" indica que esta variable es una constante, es decir, el valor que va a almacenar no podrá cambiar, no va a ser mutable, por lo tanto, aquí estoy cometiendo un error de interpretación básico, ¿cómo puedo declarar una constante, que no tenga un valor inicial, si luego me será prohibido asignarle un valor nuevo? Es decir, estoy creando una variable que nunca tendrá un valor, lo puedo explicar mejor así, supongan que aquí tenemos "number three", "número tres", con el valor de once, bien y luego queremos mutarlo, entonces "number three" ahora, o "número tres" es igual a 19. Okey y aquí nos vuelve a dar un error, noten que me marca la línea once, línea once, estoy queriendo asignar un nuevo valor a esta constante "number three" y ¿qué nos dice? estoy tratando de asignar algo un valor a una variable constante y eso está prohibido en la especificación del lenguaje "número tres", "number three", nunca podrá cambiar porque es una constante. Por ahora, son conceptos sueltos que quiero que aprendan o que, por lo menos, vean. Luego veremos casos prácticos para todo esto.

Muy bien, vamos a continuar entonces, la mutación de las variables puede ser directa, puede ser por medio de una operación, supongan que aquí vamos a comentar "number" y tendremos solamente el "console log" de la línea nueve y 10, pero aquí, luego de haber definido el "number" como, con un valor de 10, vamos a mutarla, vamos a decir que "number" es igual a "number" más 20. ¿Qué ocurrirá con esto? Hagamos un "console log" de "number", por ahora vamos a borrar estos dos de aquí, y el resultado será 30, porque esta línea está asignando a esta variable el siguiente valor: vamos a seguir la historia de "number", la creamos, pero nunca la inicializamos. En la siguiente línea, le asignamos el valor 10. Aquí, "number" que vale 10, va a recibir un nuevo valor, que será el propio valor de "number", el valor actual más 20, si el valor actual es "10 más 20", aquí tenemos el resultado, 30. Bueno, aquí podemos ver entonces cómo podemos asignar, cómo podemos mutar el contenido de una variable, ¿bien? Podríamos aquí haber hecho, por ejemplo, lo siguiente: "number",

vamos a comentar, vamos a hacerlo aquí, "number" será igual a "number" más "number dos".  
¿Okey? Y esto nos va a dar el resultado de 129 porque "number dos" vale, aquí la hemos  
inicializado en la declaración, 119, y "number" valía 10, por lo tanto, el nuevo valor de  
"number" será el que resulte de sumar el valor actual, que es 10, más el valor de otra variable  
"number dos", ¿okey? Ahí podemos ver la interrelación entonces entre las distintas variables.