

Bien, pero, ¿cómo podemos hacer para recoger información de un usuario? ¿cómo podemos interactuar con él? ¿tendremos alguna cajita? ¿Alguna especie de sitio donde el usuario pueda enviarnos información? y nosotros luego, ¿cómo podríamos responderle? Estas son las cuestiones que vamos a comenzar a ver en esta clase.

Hasta aquí hemos utilizado la consola a través de `console.log` para mostrar resultados de las operaciones con nuestras variables. Ahora, cerremos la consola, y vamos a utilizar directamente el documento para recibir aquí los mensajes que sean pertinentes. Vamos a pensar en un algoritmo. Un algoritmo es básicamente, bueno, como una suerte de receta, un proceso en el cual están todos los pasos indicados uno por uno para arribar a un resultado. Supongamos que queremos pedirle al usuario su nombre, y luego de eso saludarle. Podríamos utilizar el método, por ejemplo, `prompt`, método `prompt`. Nos permite pedir información al usuario. Podríamos indicar: “`prompt` ingrese su nombre”, ok, y se nos mostraría en el documento lo siguiente: una ventana que nos dice “ingrese su nombre”, correcto. Ahora, esto no tiene demasiado sentido porque no estamos almacenando el valor que ingresa el usuario. Por lo tanto, aquí podríamos hacer lo siguiente, podríamos indicar “`const` =” o podríamos hacerlo con `let` también, ¿verdad? Simplemente uso `const` porque este valor no va a cambiar, al menos no en para esto que tengo pensado, y siempre que podamos utilizar `const` vamos a preferir su uso. Luego, si queremos crear una variable que siempre va a estar cambiando lo que sabemos que al menos una vez va a cambiar su valor, bueno, utilizaremos `let`. Bien, entonces aquí vamos a definir una constante que se llama “nombre”, perdón, aquí “`const nombre`”, aquí declaro la variable y para inicializarla utilizo “`=`”. Entonces dije: “Esta variable que se llama ‘nombre’ que acaba de crear aquí va a tener este valor”. Este texto no. Va a recibir el valor que surja de lo que ingrese aquí el usuario. Ok, si ahora escribimos “Marcelo” esta variable ‘nombre’ contiene precisamente ese valor.

Abramos la consola, y por ejemplo, aquí busquemos la variable ‘nombre’. Fíjense que su valor es “Marcelo”, es lo que acaba de ingresar el usuario. Ok. Y ahora utilicemos, por ejemplo, un “`alert`”. Un `alert` nos va a desplegar también, una ventana similar a la del `prompt` con un mensaje. Aquí podríamos indicar “hola”, y aquí vamos a concatenar con un “`+`” la variable.

Nuestro resultado sería el siguiente. En el primer, la primera línea le pedimos al usuario que ingrese un dato y lo almacenamos en esta variable, y en la segunda línea mostramos un cartel, aquí, que dice “hola” más el valor almacenado en la variable. Perfecto, ahora, supongamos que no queremos mostrarlo con un `alert` queremos escribirlo aquí en el documento, bueno, eso podríamos hacerlo con un método del documento. Este objeto, todo este objeto, digamos nuestra pantalla por decirlo de alguna manera o para ser más exactos la porción de la pantalla donde se pinta, donde se renderiza la información es nuestro objeto “`document`”. Ese objeto tiene un método. Tiene muchos en realidad pero vamos a usar uno ahora. Si nosotros escribimos “`document.write`”, es decir, escribir en el documento, ¿escribir qué cosa? Esto mismo. Nos va a pedir un nombre

y el resultado lo va a imprimir directamente en la pantalla, ¿ok? bien, estas son algunas de las posibilidades. Ahora, ¿qué ocurre si, por ejemplo, le pedimos a

el usuario que ingrese los números para luego sumarlos? Bueno, entonces, vayamos a ello. Definamos. “`number1`” va a ser igual a `prompt`

“Primer número”. Luego definiremos “`number2`”

“Segundo número” y luego mostraremos el resultado. Bien. Y el resultado podríamos hacerlo algo así: “`document.write`”

“El resultado es”. Y aquí escribimos...

podríamos escribir, por ejemplo, resultado. Aquí haremos resultado. “Const resultado = number1 + number2”

Perfecto, y entonces aquí ya tenemos el resultado. Vamos a mostrarlo. Nos va a pedir un, bueno, perdón, seguimos con esto no lo borré. Bue, no importa, “Emma”. Ahora nos va a pedir un número, “10”, el segundo número es 1 y el resultado es 101. Bueno, primero resolvamos esto. Miren, con un pequeño ajuste, en vez de usar el método “write”, voy a usar el método “writeln”, write ele ene. La diferencia es que después de cada impresión

hace un salto de línea, entonces, va a imprimir lo que sigue debajo. Vamos a mostrarlo otra vez. “Ingrese su nombre”. Vamos a ingresar un nombre. Luego un número, el número 10 y luego, el número 3. Por lo tanto, debería decirnos “hola juana el resultado es 13” pero nos dice “hola juana el resultado es 103” Perdón, aquí les dije que íbamos a tener un salto de línea, en realidad, lo que tenemos es un espacio. Antes no nos... no nos mostraba un espacio y ahora, nos muestra un espacio. Bueno, aquí tenemos el siguiente problema: el método prompt siempre devuelve un string. Esto siempre devuelve un string. Por lo tanto, borremos la primera parte para resumir. Cuando almacenamos un valor en “number1” estamos almacenando un string. No importa que el usuario escriba un número. Lo que retorna javascript es un string. Es decir, si el usuario ingresa “3” javascript está retornando un 3 como un string. ¿Recuerda lo que ocurría cuando queríamos sumar strings? Bueno, javascript en vez de dar un error, concatenaba como si fuera texto. Por lo tanto, aquí podríamos parsear la información, es decir, convertir el número a un entero. Esto lo podemos hacer de varias maneras. Una es con el método “parseInt”. Este, esto de parsear o de hacer un “parseInt” sobre los datos es una práctica muy habitual en cualquier lenguaje de programación. Entonces, hagamos un “parseInt” de todo esto. Al envolver con el parseInt en todo este método, lo que este método retorne como string se convertirá a un número entero, ¿ok? Por eso pongo “parseInt”. Hay otra variante que es “parseFloat” para convertir en números con decimales. Y ahora hacemos “parseInt” aquí también, y ahora, cuando el usuario aquí escriba un 4, en esta variable se va a almacenar un 4, pero un 4 no como string sino como número. Y en el segundo, bueno, vamos a poner un 5. El resultado ahora, no debe... no debería ser 45, sino 9, ¿ok? Bien, solamente nos falta mostrar ese resultado. Nos hemos saltado un pequeño paso. Entonces, aquí tenemos “number1 + number2” y vamos a poner el resultado, “10”, “11”, y veamos a ver dónde está nuestro error. Aquí tenemos un error. Perdón, estoy tratando de mostrar algo que no existe, no sé por qué me apareció ahí en ese nombre “writelnJuana”. En alguna operación de borrado, copié y pegué algo sin querer. Ahora sí, vamos, el número es 20 y el segundo es 3 y no debe decirnos 203, debe decir 23. Y efectivamente muestra 23. Recuerden, si no hacemos el parse

tomará lo que retornen estos métodos como un string. Y entonces “20 + 3” será 203 ¿estamos claros? Perfecto, ésta es una de las de las cuestiones que hay que tener en cuenta cuando el valor que retorna de una entrada de usuario, es de un tipo string y nosotros queremos operar como si fuera un número y no lo es.