

Hemos visto cómo trabajan las funciones, hemos visto también lo relativo al ámbito de las variables, también vimos cómo utilizar condicionales operadores lógicos y relacionales. ¿Cómo podríamos poner todo esto en práctica? Para utilizar todas estas pequeñas, piezas móviles, en un mecanismo de relojería, en un mecanismo que funcione de manera amalgamada y nos produzca un resultado, les voy a leer un problema, el enunciado de un problema y luego lo pensaremos y lo resolveremos juntos.

No será largo, lo prometo.

“Mariana nos dice que para ella un número es “de la suerte”, si cumple con las siguientes tres condiciones: escribe una función para determinar esto, ¿cuáles son las condiciones de las que nos habla Mariana? Su número de la suerte será tal, si es un número positivo, si es múltiplo de dos o de tres y si no es el número 15”.

Bueno, con esta función que tenemos que escribir, vamos a ponerle de número, ¿es número de la suerte?, por ejemplo, a la función que va a recibir un número y le va a decir a Mariana si es un número de la suerte. El número debe cumplir con las tres condiciones antes mencionadas, es decir, debe ser positivo, tiene que ser múltiplo de dos o de tres y además, no tiene que ser el número 15. ¿Okey?

Vamos al código para descubrir cómo podríamos resolver esto, que puede parecer complejo como un problema, si estamos iniciando con el mundo de la programación, pero que, cuando vean la resolución, entenderán que es más sencillo de lo que parece.

Bien hemos dejado aquí el el enunciado, por si quieren repasarlo, si quieren tenerlo ahí visualmente, vamos a ver cómo podemos construir este algoritmo.

A ver.

Bien, escribamos nuestra función, “function”, la función debe llamarse “el número de la suerte”, “de la suerte”, es bastante largo el nombre, pero bueno, eso es lo que nos pide el enunciado, así que vamos a cumplirlo. Va a resumir, va a recibir aquí un parámetro que será el número que le pasemos y ahora vamos a hacer una serie de preguntas. Vamos a preguntar si número es mayor que cero, porque esta es la manera de ver que sea un número positivo, que es una de las condiciones, y la primera, perfecto. También tenemos que cumplir una condición y es que el número sea distinto de 15, que no sea 15, por lo tanto, vamos a decir que sea distinto de 15 y ya está, bueno, en este caso, es la tercera condición cumplida. Okey. Y ahora también tienen que darse uno de dos resultados, por lo tanto aquí, como va a ser uno de dos, vamos a separar esto, apropiadamente entre paréntesis, para que no se falsee el resultado. Tenemos que preguntar si es múltiplo el número es múltiplo de dos o si es múltiplo de tres, se tiene que dar una de las dos, con que se dé una está bien, no necesitamos que se cumplan ambas, por eso las ponemos entre paréntesis y acá preguntamos “or”, ¿okey? No es múltiplo, de tres con que se cumpla una de estas condiciones, vamos a estar bien. Entonces si esto se cumple estamos ante un número de la suerte, podemos decir “console log” “suerte”, okey, y si no podremos decir lo contrario.

Bien, ahora al llamar esta función, podemos llamarla con distintos parámetros, podríamos hacerlo aquí directamente, “document.write” para mostrarlo sin tener que abrir la consola, lo veríamos en el documento, ¿verdad?, bien y aquí entonces ahora podríamos llamar a nuestras funciones, “número de la suerte”, con distintos parámetros, si los llamamos con un número negativo, ¿qué va a ocurrir?, nos va a decir que no es un número de la suerte. Si los

llamamos con el cero, debería decirnos que no es un número de la suerte, si lo llamamos con el 15 debería decirnos que no es un número de la suerte, pero un tres, debería pasar la condición, porque es positivo, es múltiplo de tres, uno de dos, pero sí de tres y tiene que darse una, fíjense que es "or" y además es distinto del 15, por lo tanto pasemosle un tres y veremos que aquí, tenemos suerte, lo mismo si pasamos un dos, si pasamos un cinco no va a pasar, pero si un seis. Bien, ahora, podríamos mejorar un poco nuestro, nuestro algoritmo, nuestra función, en realidad para que sea un poco más profesional, podríamos indicar que directamente retorne "true", aquí o, por el contrario, retorne "false". Cuando ponemos un "return" no necesitamos escribir un "else" porque el "return", si se cumple, nos va a sacar de la función, por lo tanto, nunca se va a leer el "else".

Entonces nos podemos ahorrar esa línea y aquí directamente retornar "false", ¿por qué?, porque si la condición, alguna de las condiciones se cumplió, se va a ejecutar el "return true" y esta línea nunca va a correr. Ya vamos a estar fuera de la función, por lo tanto o, mejor dicho, por el contrario, si no se cumplió ninguna de estas condiciones, va a ejecutar esta línea, sin necesidad de escribir el "else". Okey, bien y, ¿cómo podemos mostrar esto aquí afuera? Bueno, aquí afuera podríamos directamente correr la función, ¿okey?. Vamos a hacerlo así, vamos a correr la función y aquí vamos a ver otra manera de escribir un "if", con un operador ternario, luego de correr la función, que recordamos nos devuelve "true" o "false", vamos a escribir un signo de pregunta. Esto es como hacer un "if" y aquí diremos lo siguiente: "si es true, o sea, la primera condición, vamos a hacer un "document write", "document", punto, "write", "suerte", luego, con un, con dos puntos, eso sería el "else", diríamos "document", punto, "write", "sin suerte". Okey. Y esto, fíjense que está funcionando de la misma manera, si le pasamos cero, "sin suerte", si le pasamos 15, "sin suerte", si le pasamos un 12, "suerte".

Esto que acabamos de escribir aquí, el operador condicional ternario, se lo voy, se los voy a explicar aquí con más código.

Supongamos que lo hacemos en un "if", entonces podríamos decir "if es número de la suerte" y aquí pasar un número, por ejemplo, un cero, diríamos esto, perdón, lo borré, quería copiarlo en realidad, sin dos puntos, "else", diríamos esto. Okey. Y ahí bueno vamos a ver qué está corriendo dos veces la función, si comento esto aquí abajo, va a correr una sola vez y esto funciona exactamente igual, pero noten que con él ternario se escribe quizá mucho más limpio, hasta podríamos ponerlo en una sola línea. Vamos a comentar esto y aquí vamos a descomentar nuestro ternario, fíjense que podemos escribirlo así, en una misma línea. Bien y el resultado es el mismo.

Recuerden, este "if" y este ternario es exactamente igual, ¿okey?. Ejecutamos el "if" y luego tenemos la opción, por si la condición se cumple, o después de dos puntos el "else", si de cualquiera forma, de cualquier forma que ustedes lo corran, de todas maneras va a funcionar. Aquí hemos aplicado entonces muchos de los conceptos que hemos visto hasta ahora, condicionales operadores y mucho más.