Nuestro código lo ejecutamos en Google Chrome, con el compilador de Google Chrome, pero sí usamos otro navegador, otro browser, como por ejemplo Internet Explorer, podemos ver que los caracteres especiales, como las acentuaciones, están viniendo con errores, no los está reconociendo.

**¿Por qué pasa esto?**Los navegadores tenemos que definir el tipo de conjunto de datos que estamos usando. Para el caso de Google Chrome, por defecto, ya está considerando los caracteres especiales, que en este caso se usan en lengua española que son las acentuaciones. En el caso de Internet Explorer no lo considera por defecto. Entonces es aconsejable siempre en el código definir al inicio una instrucción que es la siguiente:

* **<meta charset="UTF-8">:** Instrucción para que los navegadore puedan reconocer las acentuaciones y caracteres.

Existen convenciones de código que se las llama, específicamente de códigos HTML donde nos indican o nos sugieren que siempre usemos minúscula para definir las etiquetas.

Siempre que vamos a crear un proyecto en HTML deben ir las siguientes instrucciones:

* **<meta charset="UTF-8">**
* **<script></script>**

En nuestra página web podemos hacer instrucciones en JavaScript, pero también podemos hacer instrucciones HTML en JavaScript con la siguiente estructura:

* **<script> document.write("la edad de Christian es ... <br> ?") </script>:** Con la función **“document”** estamos diciendo que pertenece al lenguaje HTML.

**¿cuál es el objetivo de usar código HTML dentro del script de JavaScript?**

Podemos trabajar en forma dinámica, podemos usar variables, porque imagínense que yo creo una función donde a cada 15 minutos vaya contando la edad, por ejemplo, empieza con 10 y cada 15 minutos quiero que aparezca 11, 12, 13, 14. Esa variación no la puedo hacer afuera del código de JavaScript porque recordemos que el código de HTML es un código estático.

**¿Qué pasa en JavaScript si yo quito las comillas, si mando un texto sin comillas?**

Por ejemplo, tenemos el siguiente código:

<script>

document.write("la edad de Christian es ... <br> ?");

document.write("<br>");

document.write(25);

</script>

Cuando ponemos un numero dentro de JavaScript sin comillas lo considera un dato numérico, es decir, un “Number”, sin embargo, los que llevan comillas van a ser considerados Textos o “String”.

El beneficio de usar JavaScript es que podemos hacer cálculos matemáticos con los números de la siguiente manera:

<script>

document.write("la edad de Christian es ... <br> ?");

document.write("<br>");

document.write(25 + 15);

</script>

Pero también si ponemos en comillas los números podemos concatenar o unir los dos números:

<script>

document.write("la edad de Christian es ... <br> ?");

document.write("<br>");

document.write(“25” + “15”);

</script>

Cuando nosotros queremos desarrollar operaciones matemáticas dentro del código JavaScript debemos tener en cuenta que también se rige por las siglas PAPOMUDAS, por ejemplo:

No es lo mismo el resultado de esta operación:

<script>

document.write(25 + 20 + 15 / 3);

</script>

Que de este:

<script>

document.write( (25+20+15) / 3);

</script>

Ya que el primer resultado es 50 mientras que el segundo es 20.

Lo que aprendimos

* Convenios y buenas prácticas de programación.
* A concatenar caracteres (strings).
* Operaciones conjuntas entre textos y números.