1. ¿Por qué es conveniente tener instalados al menos dos navegadores diferentes en tu ordenador? Indica el nombre del motor HTML y JavaScript de 2 navegadores que elijas. (4 ptos)

Debido a que cada navegador tiene unas integraciones diferentes, de manera que puede llegar a visualizar las etiquetas HTML de forma distinta por la versión de la etiqueta HTML que soportan o directamente no permitir la etiqueta, haciendo que no se visualice.

También, cada navegador tiene estilos personalizados para los elementos básicos, como podrían ser el tamaño de los saltos de línea.

**Opera**

Motor HTML: Blink.

Motor JavaScript: V8.

**Firefox**

Motor HTML: Gecko.

Motor JavaScript: SpiderMonkey.

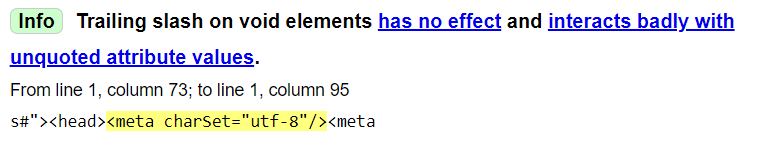
1. ¿Por qué se habla de intérprete de JavaScript y no de compilador? Busca las diferencias entre ambas formas de traducir el código (3 ptos)

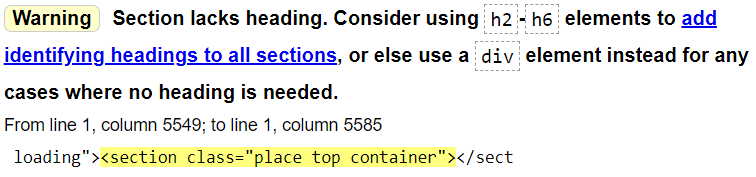
Por la forma de traducir el código que tiene JavaScript y los lenguajes interpretados en general.

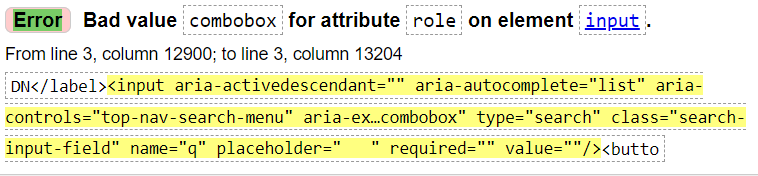
Los lenguajes interpretados ejecutan línea a línea los códigos, por lo que son más lentos debido a la cantidad de ejecuciones que tienen que llevarse a cabo, como ventaja si hay un error de ejecución, te marca la línea donde ocurrió.

En cambio, los lenguajes compilados traducen el código a un lenguaje intermedio (bytecode), y ejecutan todo el código de una vez, tienen la ventaja que una vez compilado, puede volver a ejecutarse las veces que quieras sin tener que volver a realizar el proceso.

1. Usando la dirección de validación de W3C, realiza la validación de la página https://developer.mozilla.org/es/docs/Web/JavaScript. Indica la solución de un error de tipo Info, otro Warning y otro Error (3 ptos)



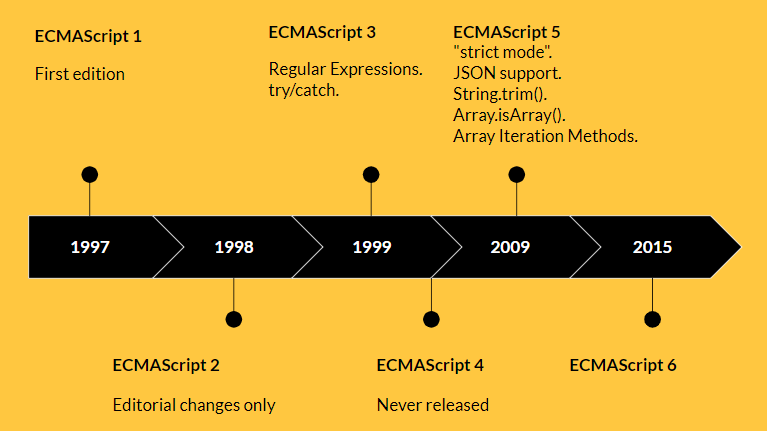




Para realizar los scripts, solamente se utilizará lo visto por la unidad. Para la entrada de datos se utilizará el método prompt () y console.log () para mostrar los resultados. Es importante utilizar comentarios para ir aclarando las funcionalidades del script.

1. ECMAScript es la especificación de referencia que controla la evolución del lenguaje JavaScript. Investiga sobre la especificación y revisa el historial de cambios más importantes que se han producido en el lenguaje en los últimos años. Indica la última versión y los cambios aportados. (1.5 ptos)

Es una especificación de lenguaje de programación basado en JavaScript, actualmente la mayoría de los navegadores incluyen una implementación de este estándar.



La primera edición se publicó en 1997.

El año siguiente cambio de editorial y publicó la segunda versión para mantener la especificación alineada con el estándar ISO/IEC 16262. En 1999 con la versión 3 se agregaron expresiones regulares, mejor manejo de strings y manejo de excepciones con try/catch.

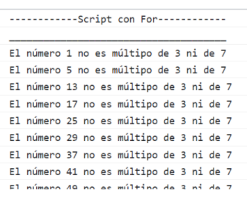
La versión 5 (2009) agrega el modo estricto, subconjunto destinado a proporcionar una mejor comprobación de errores, agrega características como getters y setters, librería para el soporte de JSON y nuevas propiedades para los objetos.

La versión 6 (2015) agregó cambios significativos en la sintaxis para escribir aplicaciones complejas, incluyendo clases y módulos, incluyeron iteradores for/of loops, generadores, datos binarios y colecciones.

La versión 8 (2017) introdujo async/await, que revolucionó la forma de manejar el código asíncrono haciendo más fácil escribirlo y leerlo sin anidar promesas (Promise), simplificando la programación asíncrona y mejorando la legibilidad y mantenimiento del código.

La última versión (versión 14 2023) trae mejoras para el manejo de tuplas, registros y expresiones regulares, mejoras de rendimiento y nuevos métodos para el manejo de arrays.

2. Números: Crea un script que muestre los números impares que no sean múltiplo de 3 ni de 7 que se encuentren entre el 100 al 1. Realizar versiones del programa usando bucles while, do while y for (2.5 ptos)



1. Realiza un script con if y otro con switch que permita introducir la edad de una persona, mediante la función prompt y muestre un mensaje de pendiendo la edad. (2.5 ptos)

- Si la edad es menor de 18 años, se mostrar el mensaje, "Eres menor de edad".

- Si la edad es mayor de 18 y menor o igual de 30 de años, se mostrar el mensaje, "Eres muy joven".

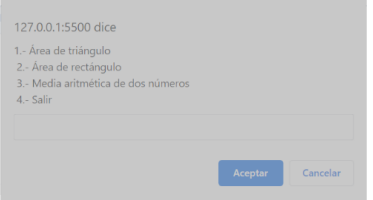
- Si la edad es mayor de 30 y menor o igual de 60 de años, se mostrar el mensaje, "Eres una persona adulta".

- Si la edad es mayor de 60, se mostrar el mensaje, "Eres una persona adulta mayor".

- Si la edad es un dato incorrecto, mostrar un mensaje "Error, debe ser un número"

4. Escribe un programa que ponga en práctica la siguiente lógica (3.5 ptos):

a) Mostrar el siguiente menú y el usuario deberá elegir una de las opciones



b) En función de la opción que elija el usuario se le pedirán los datos necesarios en cada caso.

c) El programa calculará el resultado y lo mostrará en la consola.

d) La lógica anterior se repetirá indefinidamente hasta que el usuario pulse la opción 4 para salir que mostrará un mensaje por consola, Fin del ejercicio.

e) Si la opción no está entre el rango 1-4 se mostrará un error mediante la función alert () y se volverá a pedir otra opción.



Notas:

• Cada ejercicio tendrá un archivo HTML y un fichero externo donde se encontrará el código javascript.

• Los comentario y calidad del código también son un CE que se evaluar.

• Si hay una parte del script que se utilizan métodos/funciones que no se hayan visto en esta unidad, no se evaluará el item