Desarrollo de Aplicaciones Web

Arrays Funciones y Objetos

Desarrollo web en entorno cliente

Actividad

Arrays Funciones y Objetos

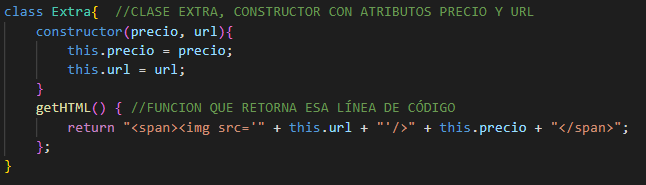
**Objetivos**

* Crear y utilizar funciones propias y del propio JavaScript.
* Crear y utilizar estructuras Array.
* Crear y utilizar objetos.

|  |
| --- |
| **¿Cómo lo hago?** |
| 1. Rellena los datos que se piden en la tabla “Antes de empezar”. 2. Haz uso de fuentes comunes como Arial, Calibri, Times New Roman etc. 3. Utiliza el color negro para desarrollar tus respuestas y usa otros colores para destacar contenidos o palabras que creas necesario resaltar. 4. Entrega un zip que contenga todos los archivos .html, .css y .js que has creado. Para poder aprobar un ejercicio, éste debe poder ejecutarse sin errores. 5. Recuerda nombrar el archivo zip siguiendo estas indicaciones:  * Ciclo\_Módulo o crédito\_Tema\_ACT\_número actividad\_Nombre y apellido   + Ejemplo: AF\_M01\_T01\_ACT\_01\_Maria Garcia |

|  |  |
| --- | --- |
| **Antes de empezar…** | |
| Nombre |  |
| Apellidos |  |
| Módulo/Crédito |  |
| UF (solo ciclos LOE) |  |
| Título de la actividad |  |

**Debes comentar como mínimo los bloques de código correspondiente a cada ejercicio.**

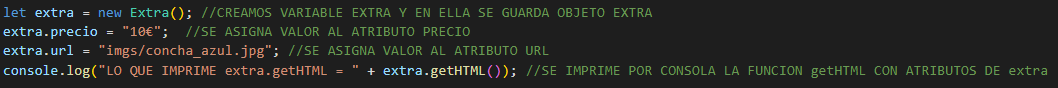
1. Crea un nuevo documento HTML con el nombre “**index.html** y un nuevo JS con el nombre “**actividad03.js**”. Vincula el HTML con el CSS y el JavaScript.
2. Crea una nueva clase “Extra” para almacenar: el precio y la url de la imagen del extra y una función “getHTML()” que al ejecutarse retorne un código HTML con la imagen y el precio del extra.
3. Una vez realizada la clase “Extra” deberá ser capaz de validar el siguiente código:

let extra = new Extra();

extra.precio = "10€";

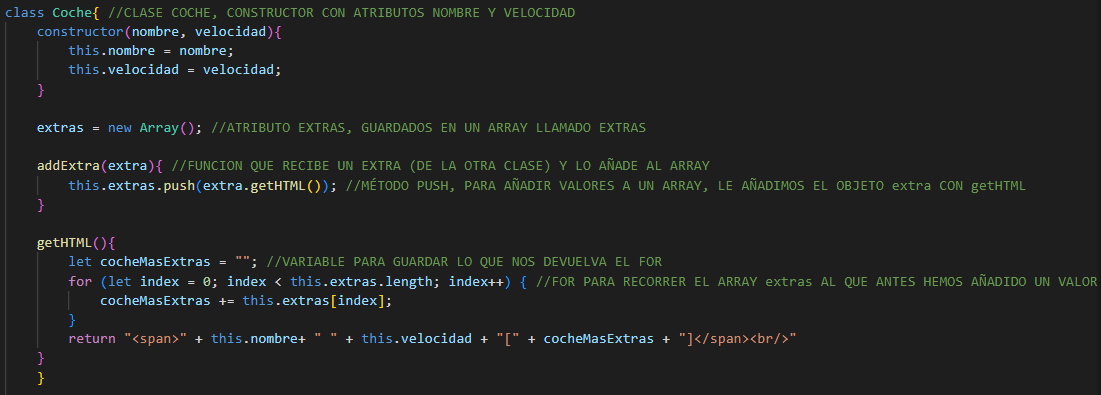
extra.url = "imgs/concha\_azul.jpeg";

console.log(extra.getHTML());

Mostrando por consola un mensaje parecido a: “<span> <img src='imgs/concha\_azul.jpeg'/> 10€</span>”



1. Crea una clase “Coche” que almacene: el nombre del coche, su velocidad, una array vacía de extras, una función (addExtra) que reciba un extra y lo añada a su array y una función (getHTML) que retorne un código HTML con los datos del Coche y de los extras



1. Una vez realizada la clase “Coche”, deberá ser capaz de validar el siguiente código:

let coche = new Coche();

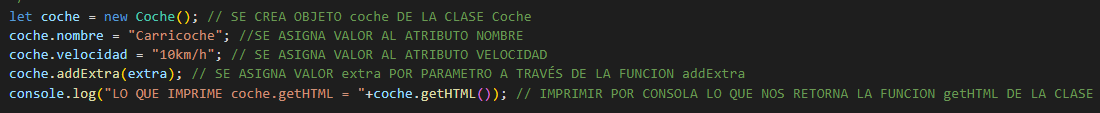
coche.nombre = "Carricoche";

coche.velocidad = "10km/h";

coche.addExtra(extra);

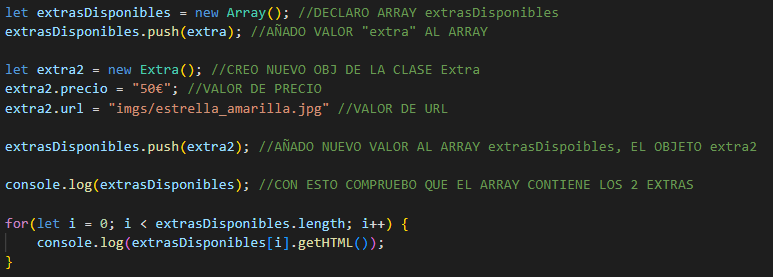
console.log(coche.getHTML());

Mostrando por consola un mensaje parecido a: <span> Carricoche 10km/h [<span> <img src='imgs/concha\_azul.jpeg'/> 10€</span>]</span><br />”

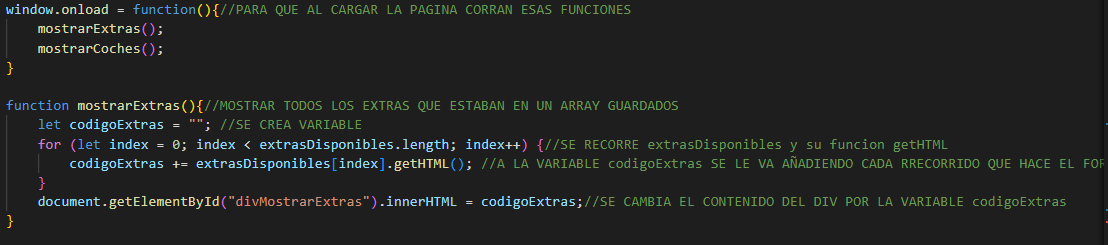




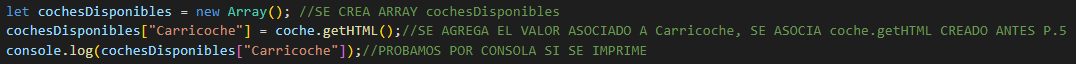
1. Crea un array global **extrasDisponibles** para almacenar todos los extras disponibles y añádele el extra creado anteriormente y otro extra. Éste array contendrá los extras que se pueden añadir a cualquier coche.



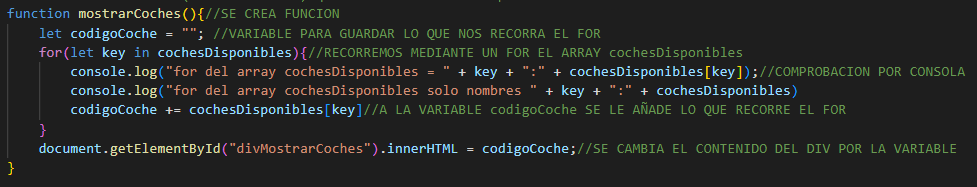
1. Programa que una vez cargada la web llame a una función **mostrarExtras()** que muestre en el HTML todos los extras que contiene **extrasDisponibles.**

****

1. Crea un array asociativa global de nombre **cochesDisponibles** para almacenar todos los coches creados según su nombre y añádele el coche creado anteriormente.

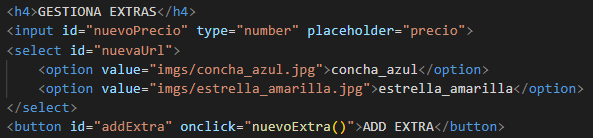


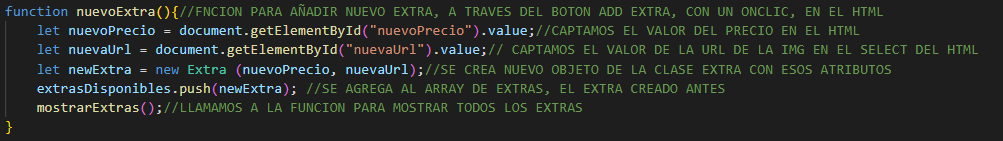
1. Programa que una vez cargada la web llame a una función **mostrarCoches()** que muestre en el HTML todos los coches (con sus extras) que contiene **coches disponibles**

****

1. Crea la estructura HTML que se muestra a continuación y programa el botón ADD EXTRA para que añada en **extrasDisponibles** un nuevo extra con el precio indicado y la imagen seleccionada del *select* y actualice visualmente todos los extras disponibles (llama a mostrarExtras() )

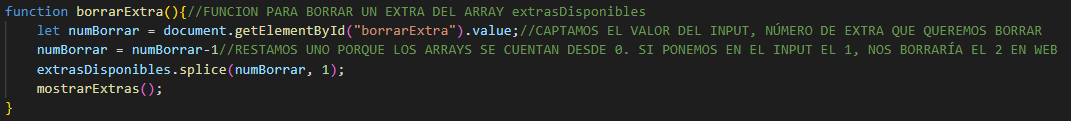






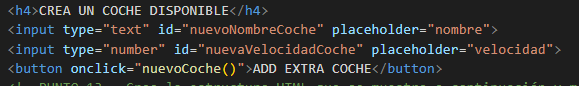
1. Crea la estructura HTML que se muestra a continuación y programa el botón BORRARpara que borre de **extrasDisponibles** el número de extra indicado (el número es la posición que ocupa el extra dentro del array de **extrasDisponibles**)

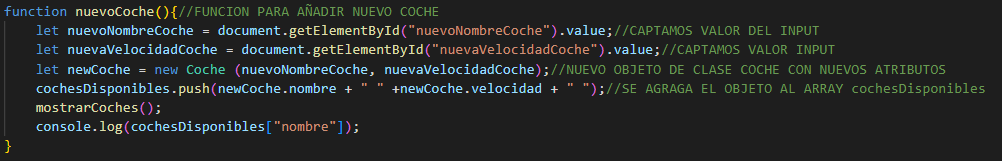
C:\Users\Cristian\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Untitled-5.jpg



1. Crea la estructura HTML que se muestra a continuación y programa el botón ADD COCHE para que añada a **cochesDisponibles** un nuevo coche con el nombre y la velocidad máxima indicada y actualice visualmente todos los coches disponibles (llama a mostrarCoches() )

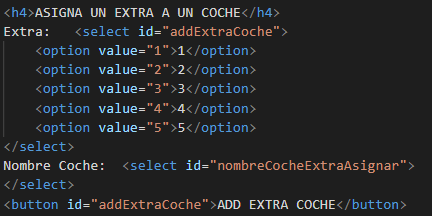


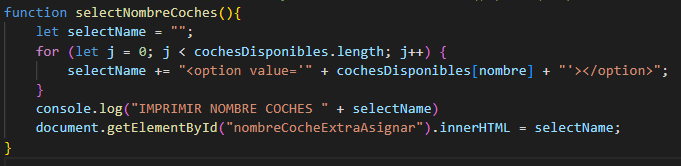




1. Crea la estructura HTML que se muestra a continuación y modifica mostrarExtras() para que por cada extra se añada un option al select con el número de extra , y modifica mostrarCoches() para que por cada coche se añada un option al select con el nombre del coche.







1. Programa el botón ADD EXTRA COCHE para que añada al coche seleccionado el número de extra seleccionado y actualice visualmente todos los coches disponibles.

A continuación se muestra una captura de ejemplo:

