

Ejercicio 9: “Leer y archivar datos”

Existen diferentes tipos de datos, por ejemplo números, caracteres o letras, dentro de los números podemos definir diferentes tipos, como pueden ser enteros, naturales, racionales, reales, también imágenes, etc.

Hemos usado una función “**parseInt**” para convertir un tipo de dato que no sabemos cual es, por ejemplo letras en números. Operar con diferentes tipos de datos es complejo.

Hoy vais a ver tipos de datos asociados.

Un objeto es una estructura de datos (en archivo tipo **JavaScript**) por ejemplo es:

```
let car={
  Marca:"Opel",
  Modelo:"Astra",
  Plazas:5
}
```

Nosotros no vamos a manejar este tipo de dato, debido a su complejidad de procesado.

Os voy a explicar un ejemplo, y luego vosotros haréis un ejercicio diferente. Copiarlo os puede servir para modificarlo luego

En vez de usar el tipo de estructura anterior usamos lo siguiente:

```
MarcaCoche=["Opel","Renault","BMW"];
Modelo=["Corsa","Megane","i4"];
NumerodeAsientos=[5,5,5];
```

La posición **0,1 y 2** de cada variable son 3 datos completos:

Dato 1

MarcaCoche[0]="Opel"

Modelo[0]="Corsa"

NumerodeAsientos[0]=5

Dato 2

MarcaCoche[1]="Renault"

Modelo[1]="Megane"

NumerodeAsientos[1]=5

Dato 3

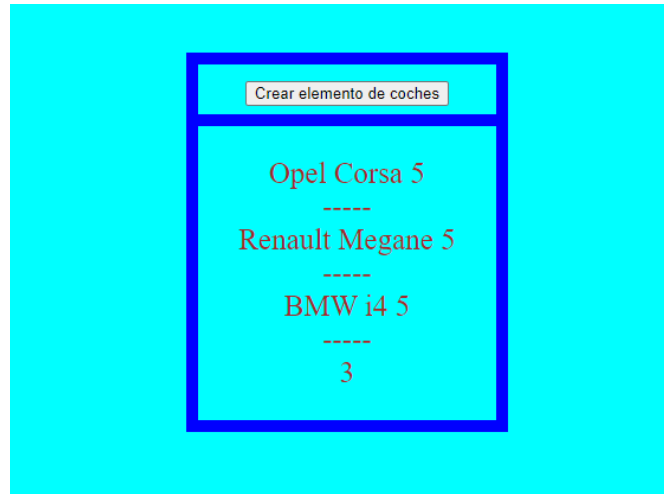
MarcaCoche[2]="Opel"

Modelo[2]="Corsa"

NumerodeAsientos[2]=5

Esto ya lo hemos visto en las gráficas.

Vosotros vais a realizar una tabla, donde se vea nuestra base de datos y podamos añadir un nuevo elemento al pulsar un botón:



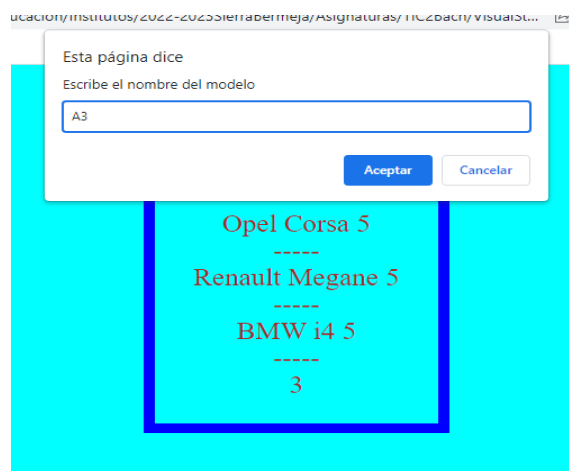
A screenshot of a web form with a blue border. At the top, there is a button labeled "Crear elemento de coches". Below the button, the text "Opel Corsa 5" is displayed. Underneath, there are three dashed lines, followed by the text "Renault Megane 5". Below that, there are three dashed lines, followed by the text "BMW i4 5". At the bottom, there are three dashed lines, followed by the text "3".

Si pulsamos el botón aparecerá lo siguiente, recogiendo sucesivamente los 3 datos para la base de datos:



A screenshot of a web browser showing a dialog box. The dialog box has a title bar that says "Esta página dice". Below the title bar, there is a label "Escribe el nombre de la marca" and a text input field. The input field is empty. Below the input field, there are two buttons: "Aceptar" and "Cancelar". In the background, the same web form as in the previous screenshot is visible.

Luego



A screenshot of a web browser showing a dialog box. The dialog box has a title bar that says "Esta página dice". Below the title bar, there is a label "Escribe el nombre del modelo" and a text input field. The input field contains the text "A3". Below the input field, there are two buttons: "Aceptar" and "Cancelar". In the background, the same web form as in the previous screenshot is visible.

Lo último



Si los datos se introducido correctamente se actualiza la base de datos:



El **HTML** y el **CSS** no hace falta que os lo explique. Os incluyo el **JS**. Os hago una breve explicación.

El código que tendréis que modificar será el siguiente:

```

1  let MarcaCoche=[],Modelo=[],NumerodeAsientos=[];
2  MarcaCoche=["Opel","Renault", "BMW"];
3  Modelo=["Corsa","Megane","i4"];
4  NumerodeAsientos=[5,5,5];
5  function add (){
6      let a,b,c,numero;
7      a=prompt("Escribe el nombre de la marca");
8      b=prompt("Escribe el nombre del modelo");
9      c=prompt('Escribe el número de plazas');
10     if((a!=0)&&(b!=0)&&(c!=0))
11     {
12         numero=MarcaCoche.length;
13         MarcaCoche[numero]=a;
14         Modelo[numero]=b;
15         NumerodeAsientos[numero]=c;
16     }
17 }
18 function tiempo(){
19     dato="";
20     for(i=0;i<MarcaCoche.length;i++)
21     {
22         dato=""+dato+" "+MarcaCoche[i]+" "+Modelo[i]+" "+NumerodeAsientos[i]+"<br>"+"-----"+"<br>";
23     }
24     dato=dato+" "+MarcaCoche.length;
25     document.getElementById("coche").innerHTML=dato;
26 }
27 setInterval(tiempo,1000);

```

Lo primero que hacemos es declarar la variables globales y los 3 primeros datos de la estructura:

```

let MarcaCoche=[],Modelo=[],NumerodeAsientos=[];
MarcaCoche=["Opel","Renault", "BMW"];
Modelo=["Corsa","Megane","i4"];
NumerodeAsientos=[5,5,5];

```

La función **add()**, que permite añadir los datos a pinchar el botón del HTML.

```

function add (){
    let a,b,c,numero;
    a=prompt("Escribe el nombre de la marca");
    b=prompt("Escribe el nombre del modelo");
    c=prompt('Escribe el número de plazas');
    if((a!=0)&&(b!=0)&&(c!=0))
    {
        numero=MarcaCoche.length;
        MarcaCoche[numero]=a;
        Modelo[numero]=b;
        NumerodeAsientos[numero]=c;
    }
}

```

Si te acuerdas la función “**prompt**” permite recoger datos mediante una ventana.

La expresión siguiente indica que si has introducido los 3 datos correctamente se guardarán, si has pinchado el “**Enter**” sin incluir cualquiera de los datos las variables no se guardan.

```
if((a!=0)&&(b!=0)&&(c!=0))
```

“**a!=0**” significa que si a es diferente a 0. Esto es que has incluido un dato.

“**&&**” esto significa y.

La expresión general quiere decir que si a, b y c son diferentes a 0 se guardan los datos.

`numero=MarcaCoche.length;` esto recoge la longitud del “**array**” de datos, o lo que es lo mismo el número de elementos que hay por cada término.

Aquí se guardan los datos si se dan las condiciones anteriores.

```
MarcaCoche[numero]=a;
Modelo[numero]=b;
NumerodeAsientos[numero]=c;
```

La función **tiempo()** hace que se vea en pantalla todos los elementos de la pequeña base de datos:

```
function tiempo(){
    dato="";
    for(i=0;i<MarcaCoche.length;i++)
    {
        dato=""+dato+" "+MarcaCoche[i]+" "+Modelo[i]+" "+NumerodeAsientos[i]+"<br>"+"-----"+"<br>";
    }
    dato=dato+" "+MarcaCoche.length;
    document.getElementById("coche").innerHTML=dato;
}
```

Por último uso la función para actualizar la base de datos cada segundo o 1000 ms.

```
setInterval(tiempo,1000);
```

TAREA A REALIZAR

Vais a realizar una base de datos de 4 elementos de estructura que no tenga nada que ver con los coches.

Pensadlo. Por ejemplo con datos de una persona.

TAREA EXTRA.

Haced lo mismo que en el ejercicio anterior, pero con 2 columnas.

Hay que actualizar las 2 bases de datos a la vez.

Esto es una base de dato actualizable al pulsar un botón. De 3 elementos de estructura.

Otra base de datos de otros 3 elementos de estructura actualizables cuando se pulse un botón.

Es hacer el doble que en el ejercicio anterior en un HTML.