

ACTIVIDAD

Arrays Funciones y Objetos

Desarrollo de Aplicaciones Web

Desarrollo web en entorno cliente



Linkia FP



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.

```
class Extra{ //CLASE EXTRA, CONSTRUCTOR CON ATRIBUTOS PRECIO Y URL
  constructor(precio, url){
    this.precio = precio;
    this.url = url;
  }
  getHTML() { //FUNCION QUE RETORNA ESA LÍNEA DE CÓDIGO
    return "<span><img src='" + this.url + "'/>" + this.precio + "</span>";
  }
};
```

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: “ 10€”

```
let extra = new Extra(); //CREAMOS VARIABLE EXTRA Y EN ELLA SE GUARDA OBJETO EXTRA
extra.precio = "10€"; //SE ASIGNA VALOR AL ATRIBUTO PRECIO
extra.url = "imgs/concha_azul.jpg"; //SE ASIGNA VALOR AL ATRIBUTO URL
console.log("LO QUE IMPRIME extra.getHTML = " + extra.getHTML()); //SE IMPRIME POR CONSOLA LA FUNCION getHTML CON ATRIBUTOS DE extra

LO QUE IMPRIME extra.getHTML = <span><img src='imgs/concha_azul.jpg'/>10€</span>
```

4. 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

```
class Coche{ //CLASE COCHE, CONSTRUCTOR CON ATRIBUTOS NOMBRE Y VELOCIDAD
  constructor(nombre, velocidad){
    this.nombre = nombre;
    this.velocidad = velocidad;
  }

  extras = new Array(); //ATRIBUTO EXTRAS, GUARDADOS EN UN ARRAY LLAMADO EXTRAS

  addExtra(extra){ //FUNCION QUE RECIBE UN EXTRA (DE LA OTRA CLASE) Y LO AÑADE AL ARRAY
    this.extras.push(extra.getHTML()); //MÉTODO PUSH, PARA AÑADIR VALORES A UN ARRAY, LE AÑADIMOS EL OBJETO extra CON getHTML
  }

  getHTML(){
    let cocheMasExtras = ""; //VARIABLE PARA GUARDAR LO QUE NOS DEVUELVA EL FOR
    for (let index = 0; index < this.extras.length; index++) { //FOR PARA RECORRER EL ARRAY extras AL QUE ANTES HEMOS AÑADIDO UN VALOR
      cocheMasExtras += this.extras[index];
    }
    return "<span>" + this.nombre+ " " + this.velocidad + "[" + cocheMasExtras + "]"</span><br/>"
  }
}
```

5. 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: Carricoche 10km/h [10€
"

```
let coche = new Coche(); // SE CREA OBJETO coche DE LA CLASE Coche
coche.nombre = "Carricoche"; //SE ASIGNA VALOR AL ATRIBUTO NOMBRE
coche.velocidad = "10km/h"; // SE ASIGNA VALOR AL ATRIBUTO VELOCIDAD
coche.addExtra(extra); // SE ASIGNA VALOR extra POR PARAMETRO A TRAVÉS DE LA FUNCION addExtra
console.log("LO QUE IMPRIME coche.getHTML = "+coche.getHTML()); // IMPRIMIR POR CONSOLA LO QUE NOS RETORNA LA FUNCION getHTML DE LA CLASE
```

LO QUE IMPRIME coche.getHTML = Carricoche 10km/h10€

6. 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.

```
let extrasDisponibles = new Array(); //DECLARO ARRAY extrasDisponibles
extrasDisponibles.push(extra); //AÑADO VALOR "extra" AL ARRAY

let extra2 = new Extra(); //CREO NUEVO OBJ DE LA CLASE Extra
extra2.precio = "50€"; //VALOR DE PRECIO
extra2.url = "imgs/estrella_amarilla.jpg" //VALOR DE URL

extrasDisponibles.push(extra2); //AÑADO NUEVO VALOR AL ARRAY extrasDisponibles, EL OBJETO extra2

console.log(extrasDisponibles); //CON ESTO COMPRUEBO QUE EL ARRAY CONTIENE LOS 2 EXTRAS

for(let i = 0; i < extrasDisponibles.length; i++) {
  console.log(extrasDisponibles[i].getHTML());
}
```

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

EXTRAS DISPONIBLES



```
window.onload = function(){//PARA QUE AL CARGAR LA PAGINA CORRAN ESAS FUNCIONES
  mostrarExtras();
  mostrarCoche();
}


function mostrarExtras(){//MOSTRAR TODOS LOS EXTRAS QUE ESTABAN EN UN ARRAY GUARDADOS
  let codigoExtras = ""; //SE CREA VARIABLE
  for (let index = 0; index < extrasDisponibles.length; index++) { //SE RECORRE extrasDisponibles y su funcion getHTML
    codigoExtras += extrasDisponibles[index].getHTML(); //A LA VARIABLE codigoExtras SE LE VA AÑADIENDO CADA RRECORRIDO QUE HACE EL FOR
  }
  document.getElementById("divMostrarExtras").innerHTML = codigoExtras; //SE CAMBIA EL CONTENIDO DEL DIV POR LA VARIABLE codigoExtras
}
```

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

```
let cochesDisponibles = new Array(); //SE CREA ARRAY cochesDisponibles
cochesDisponibles["Carricoche"] = coche.getHTML(); //SE AGREGA EL VALOR ASOCIADO A Carricoche, SE ASOCIA coche.getHTML CREADO ANTES P.5
console.log(cochesDisponibles["Carricoche"]); //PROBAMOS POR CONSOLA SI SE IMPRIME
```

9. 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**

COCHES DISPONIBLES

Carricoche 10km/h [ 10€]

```
function mostrarCoches(){//SE CREA FUNCION
  let codigoCoche = ""; //VARIABLE PARA GUARDAR LO QUE NOS RECORRA EL FOR
  for(let key in cochesDisponibles){//RECORREMOS MEDIANTE UN FOR EL ARRAY cochesDisponibles
    console.log("for del array cochesDisponibles = " + key + ":" + cochesDisponibles[key]); //COMPROBACION POR CONSOLA
    console.log("for del array cochesDisponibles solo nombres " + key + ":" + cochesDisponibles)
    codigoCoche += cochesDisponibles[key] //A LA VARIABLE codigoCoche SE LE AÑADE LO QUE RECORRE EL FOR
  }
  document.getElementById("divMostrarCoches").innerHTML = codigoCoche; //SE CAMBIA EL CONTENIDO DEL DIV POR LA VARIABLE
}
```

10. 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()**)

GESTIONA EXTRAS

```
<h4>GESTIONA EXTRAS</h4>
<input id="nuevoPrecio" type="number" placeholder="precio">
<select id="nuevaUrl">
  <option value="imgs/concha_azul.jpg">concha_azul</option>
  <option value="imgs/estrella_amarilla.jpg">estrella_amarilla</option>
</select>
<button id="addExtra" onclick="nuevoExtra()">ADD EXTRA</button>
```

```
function nuevoExtra(){//FUNCION PARA AÑADIR NUEVO EXTRA, A TRAVES DEL BOTON ADD EXTRA, CON UN ONCLIC, EN EL HTML
  let nuevoPrecio = document.getElementById("nuevoPrecio").value;//CAPTAMOS EL VALOR DEL PRECIO EN EL HTML
  let nuevaUrl = document.getElementById("nuevaUrl").value;// CAPTAMOS EL VALOR DE LA URL DE LA IMG EN EL SELECT DEL HTML
  let newExtra = new Extra (nuevoPrecio, nuevaUrl);//SE CREA NUEVO OBJETO DE LA CLASE EXTRA CON ESOS ATRIBUTOS
  extrasDisponibles.push(newExtra); //SE AGREGA AL ARRAY DE EXTRAS, EL EXTRA CREADO ANTES
  mostrarExtras();//LLAMAMOS A LA FUNCION PARA MOSTRAR TODOS LOS EXTRAS
}
```

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

Posicion a borrar

```
<h4>BORRAR EXTRAS</h4>
Posición a borrar <input type="number" placeholder="número" id="borrarExtra">
<button onclick="borrarExtra()">BORRAR</button>
```

```
function borrarExtra(){//FUNCION PARA BORRAR UN EXTRA DEL ARRAY extrasDisponibles
  let numBorrar = document.getElementById("borrarExtra").value;//CAPTAMOS EL VALOR DEL INPUT, NÚMERO DE EXTRA QUE QUEREMOS BORRAR
  numBorrar = numBorrar-1//RESTAMOS UNO PORQUE LOS ARRAYS SE CUENTAN DESDE 0. SI PONEMOS EN EL INPUT EL 1, NOS BORRARÍA EL 2 EN WEB
  extrasDisponibles.splice(numBorrar, 1);
  mostrarExtras();
}
```

12. 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()**)

CREA UN COCHE DISPONIBLE

Km/h

```
<h4>CREA UN COCHE DISPONIBLE</h4>
<input type="text" id="nuevoNombreCoche" placeholder="nombre">
<input type="number" id="nuevaVelocidadCoche" placeholder="velocidad">
<button onclick="nuevoCoche()">ADD EXTRA COCHE</button>
```



```
function nuevoCoche(){//FUNCION PARA AÑADIR NUEVO COCHE
  let nuevoNombreCoche = document.getElementById("nuevoNombreCoche").value;//CAPTAMOS VALOR DEL INPUT
  let nuevaVelocidadCoche = document.getElementById("nuevaVelocidadCoche").value;//CAPTAMOS VALOR INPUT
  let newCoche = new Coche (nuevoNombreCoche, nuevaVelocidadCoche);//NUEVO OBJETO DE CLASE COCHE CON NUEVOS ATRIBUTOS
  cochesDisponibles.push(newCoche.nombre + " " +newCoche.velocidad + " ");//SE AGRAGA EL OBJETO AL ARRAY cochesDisponibles
  mostrarCoches();
  console.log(cochesDisponibles["nombre"]);
}
```

13. 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.

ASIGNA UN EXTRA A UN COCHE

Extra: Nombre Coche:

```
<h4>ASIGNA UN EXTRA A UN COCHE</h4>
Extra: <select id="addExtraCoche">
  <option value="1">1</option>
  <option value="2">2</option>
  <option value="3">3</option>
  <option value="4">4</option>
  <option value="5">5</option>
</select>
Nombre Coche: <select id="nombreCocheExtraAsignar">
</select>
<button id="addExtraCoche">ADD EXTRA COCHE</button>
```

```
function selectNombreCoches(){
  let selectName = "";
  for (let j = 0; j < cochesDisponibles.length; j++) {
    selectName += "<option value='" + cochesDisponibles[nombre] + "'></option>";
  }
  console.log("IMPRIMIR NOMBRE COCHES " + selectName)
  document.getElementById("nombreCocheExtraAsignar").innerHTML = selectName;
}
```

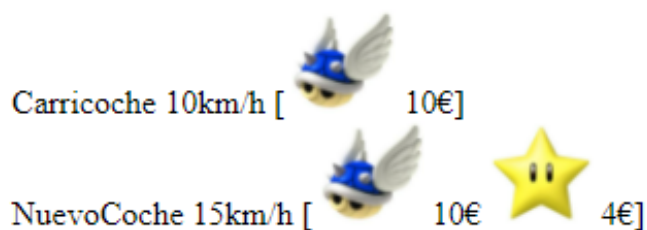
14. Programa el botón ADD EXTRA COCHE para que añada al coche seleccionado el número de extra seleccionado y actualice visualmente todos los coches disponibles.

A continuación se muestra una captura de ejemplo:

EXTRAS DISPONIBLES



COCHES DISPONIBLES



GESTIONA EXTRAS

Posicion a borrar

CREA UN COCHE DISPONIBLE

Km/h

ASIGNA UN EXTRA A UN COCHE

Extra: Nombre Coche: