

## Introducción

JavaScript es un lenguaje de programación interpretado (El navegador lee directamente el código, sin necesidad de terceros) de alto nivel. Es el lenguaje de programación encargado de dotar de mayor interactividad y dinamismo a las páginas web.



¿Cuáles son los principales objetivos de estos ejercicios?

- Entender cómo utilizar funciones flecha
- Aprender a utilizar foreach, reduce, map & filter

## 1. Funciones flecha

- Convierte la siguiente función en una función flecha:

```
function greetings() {  
    return "Hola";  
}
```

- Convierte la siguiente función en una función flecha en línea:

```
function division(a,b) {  
    return a / b;  
}
```

- Convierte la siguiente función en una función flecha:

```
function myName(name) {  
    return `Mi nombre es ${name}`;  
}
```

- Convierte las siguientes funciones en funciones flecha:

```
function test2() {  
    console.log("Función test 2 ejecutada.");  
}  
  
function test1(callback) {  
    callback();  
}
```

## 2. Foreach

- Utiliza la siguiente array para resolver los próximos ejercicios:

```
let people = [  
  { name: 'Jamiro', age: 45 },  
  { name: 'Juan', age: 35 },  
  { name: 'Paco', age: 34 },  
  { name: 'Pepe', age: 14 },  
  { name: 'Pilar', age: 24 },  
  { name: 'Laura', age: 24 },  
  { name: 'Jenny', edad: 10 },  
]
```

- Crea un array con la gente mayor de 25 años y muéstralo por consola.
- Crea un array con la gente que empieza por J.

## 3. Map

- Utilizando el array de antes crea un array con la gente mayor de 25 años y muéstralo por consola.
- Crea un array con la gente que empieza por J.
- Dado el siguiente array, crear otro que sea el resultado de elevar cada número a si mismo:

```
const numbers = [ 4, 5, 6, 7, 8, 9, 10]  
  
// Resultado esperado  
  
// [256, 3125, 46656, 823543, 16777216, 387420489, 10000000000]
```

## 4. Filter

- Crea un segundo array que devuelva los impares

```
const numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];
```

- Dado el siguiente array, genera un segundo array que filtre los platos veganos y saque una sentencia como la del ejemplo:

```
const foodList = [  
  { name: 'Tempeh', isVeggie: true },  
  { name: 'Cheesbacon burger', isVeggie: false },  
  { name: 'Tofu burger', isVeggie: true },  
  { name: 'Entrecot', isVeggie: false }  
]  
  
/* [  
  'Que rico Tempeh me voy a comer!',  
  'Que rica Tofu burger me voy a comer!'  
]  
*/
```

## 5. Reduce

- Dado el siguiente array, obtén la multiplicación de todos los elementos del array:

```
const numbers = [39, 2, 4, 25, 62]  
  
// Salida--> 483600
```

## Extras

- Map
  - Dado el siguiente array, crear un segundo array que forme frases como en el ejemplo accediendo a las propiedades del objeto proporcionado:

```
const staff = [  
  
  {  
  
    name: "Pepe",  
  
    role: "The Boss",  
  
    hobbies: ["leer", "ver pelis"],  
  
  },  
  
  {  
  
    name: "Ana",  
  
    role: "becaria",  
  
    hobbies: ["nadar", "bailar"],  
  
  },  
  
  {  
  
    name: "Luis",  
  
    role: "programador",  
  
    hobbies: ["dormir", "comprar"],  
  
  },  
  
  {  
  
    name: "Carlos",  
  
    role: "secretario",  
  
    hobbies: ["futbol", "queso"],  
  
  },  
  
]
```

```

]

// Resultado esperado
/*
    [
        'Pepe es TheBoss y le gusta leer y ver pelis',
        'Ana es becaria y le gusta nadar y bailar',
        'Luis es programador y le gusta dormir y comprar',
        'Ana es becaria y le gusta nadar y bailar',
        'Carlos es secretario y le gusta fútbol y queso'
    ]
*/
*/

```

- Dado el siguiente array, generar un segundo array que consiga generar de salida el resultado esperado:

```

const foodList = ["Pizza", "Ramen", "Paella", "Entrecot"];

//Resultado esperado
/* [
    'Como soy de Italia, amo comer Pizza',
    'Como soy de Japón, amo comer Ramen',
    'Como soy de Valencia, amo comer Paella',
    'Aunque no como carne, el Entrecot es sabroso'
]
*/
*/

```

- Filter
  - Dado el siguiente array, devolver un array con los nombres de los elementos que valgan más de 300 euros

```
const inventory = [  
  
  {  
  
    name: 'Mobile phone',  
  
    price: 199  
  
  },  
  
  {  
  
    name: 'TV Samsung',  
  
    price: 459  
  
  },  
  
  {  
  
    name: 'Viaje a Cancún',  
  
    price: 600  
  
  },  
  
  {  
  
    name: 'Mascarilla',  
  
    price: 1  
  
  }  
  
];  
  
/*  
  
[  
  
  'TV Samsung',  
  
  ]  
*/
```

```
        'Viaje a Cancún'  
    ]  
    */
```

- Reduce
  - Concatena todos los elementos del array con reduce para que devuelva una sola frase

```
const sentenceElements = [  
    'Me',  
    'llamo',  
    /* Tu nombre aquí! */,  
    'y',  
    'quiero',  
    'sentir',  
    'la',  
    'fuerza',  
    'con',  
    'javascript'  
];  
  
// Resultado--> 'Me llamo XX y quiero sentir la fuerza  
con javascript'
```



## Entregables

- Subir el ejercicio a un repositorio en github
- Dejar el enlace del repositorio adjuntado en Classroom