

ACTIVIDADES DE CLASE:

Título Actividades funciones de orden superior

Objetivos

Aprender a utilizar funciones de orden superior en JavaScript.

Temporalización

Dedicación estimada: 4 horas.

Actividades:

Para este bloque de actividades deberás implementar las funciones de orden superior predefinidas en JavaScript `setInterval` / `setTimeout` para establecer ejecuciones de código en intervalos temporales y `map` para trabajar sobre arrays. Recuerda que cuando utilices los selectores `getElementsByTagName`, `getElementsByClassName...`, los elementos seleccionados no se almacenan en un array, sino en HTML Collection. HTML collection no es iterable con `map`. Para poder hacerlo, es necesario convertir HTML Collection a array. Para ello deberás utilizar la siguiente instrucción:

```
let photos = document.getElementsByClassName('photo');  
  
Array.from(photos).map((element, index) => {  
    // Lógica a implementar  
});
```

Nota: aunque en el enunciado sólo se indica que utilices `map`, es recomendable que también utilices el resto de funciones de orden superior sobre arrays `forEach`, `filter`, `reduce...`.

Actividad 1 – Utilizando map, ubica una foto distinta en cada uno de los elementos HTML con class photo.

Actividad 2 – Implementa un script para que cuando se cargue la página se realice un cambio progresivo en el valor de la opacidad de las imágenes al 60%. Dicho cambio se producirá de la siguiente manera: cada 100 milisegundos se cambiará el valor de una imagen. De este modo, pasados los primeros 100 milisegundos sólo la primera imagen tendrá la opacidad al 60%. Pasados 200 milisegundos, las imágenes cambiadas serán la primera y la segunda. Al transcurrir 600 milisegundos, todas las imágenes tendrán su opacidad cambiada. Tras este intervalo, se producirá la operación inversa: pasados 100 milisegundos la última imagen pasará a recuperar su valor inicial -100%-. Pasados otros 100 milisegundos, la quinta y sexta imágenes habrán recuperado su valor inicial. Y así sucesivamente hasta que todas las imágenes hayan recuperado su valor inicial. El script debe permitir realizar dos barridos como el descrito. Uno al cargarse la página y otro 1,8 segundos después.

Nota: si tienes dudas puedes consultar el video adjunto.

Actividad 3 – Implementa un script para que 3,7 segundos después de cargarse la página, las seis imágenes cambien el valor de su opacidad al '50%' e incrementen sus valores de ancho y alto a un '105%'. Transcurridos 4 segundos de cargarse la página, la opacidad y el ancho y alto de las seis imágenes volverán a sus valores iniciales. Transcurridos 4,4 segundos de cargarse la página, la opacidad y el ancho y alto de las seis imágenes volverán a cambiar sus valores de opacidad al 50% y las dimensiones ambas al '105%'. A los 4,8 segundos de cargar la página, las imágenes mostraran los valores iniciales.

Actividad 4 – Utilizando map(), implementa la lógica necesaria para que cuando el usuario haga click sobre cualquiera de las seis imágenes, se eliminen todas ellas y se muestre únicamente en pantalla la imagen pulsada. Esta imagen pasará a ocupar el espacio que ocupaban las seis anteriores.

Actividad 5 – Implementa un script para que cinco segundos después de cargarse la página se realicen los siguientes cambios:

- Las tres imágenes pasarán a tener un alto de 0px.
- El elemento con la class photo-container pasará a 600px.
- El color de los textos ubicados sobre las imágenes pasará a ser blanco.
- El color del título de primer nivel (h1) pasará a ser negro.

Cada 5000 milisegundos, los cambios anteriores oscilarán con los siguientes, creando un bucle de repetición:

- Las tres imágenes pasarán a tener los siguientes altos: la primera 200px, la segunda 400px y la tercera 600px.
- Las tres imágenes pasarán a tener los siguientes line-height: la primera 150px, la segunda 300px y la tercera 400px.
- El elemento con la class photo-container pasará a 200px.
- El color de los textos ubicados sobre las imágenes pasará a ser negro.
- El color del título de primer nivel (h1) pasará a ser blanco.

El color del fondo de pantalla oscilará entre blanco y negro y se aplicará un valor a la propiedad CSS transition de 2s para todos los elementos animados.