

1 Programación Asíncrona en JavaScript

Cuando una aplicación JavaScript es ejecutada en un equipo, esta corre en *single-threaded* (un solo hilo de ejecución).

1.1 Sumario

- 1 Callbacks
- 2 Promise
- 3 Generator
- 4 Herramientas de tiempo en JavaScript
 - ◆ 4.1 setTimeout()
 - ◆ 4.2 setInterval()

1.2 Callbacks

..

1.3 Promise

..

1.4 Generator

..

1.5 Herramientas de tiempo en JavaScript

1.5.1 setTimeout()

El método setTimeout() permiten programar la ejecución de alguna parte del código de JavaScript después del paso de una cantidad definida de milisegundos.

Veamos un ejemplo muy sencillo:

```
function susto() { alert('Boooooo!') };
setTimeout(susto, 3000);
```

Un ejemplo que puede servirnos como base para hacer el **Juego de Buscar parejas**:

• HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="estilos.css">
  <title>Document</title>
</head>
<body>

  <div id="contenido">

  </div>

  <script src="scripts.js"></script>
</body>
```

```
</html>
```

• CSS

```
body {
    font-weight: bold;
    color: white;
}
#contenido {
    /* Ancho de 700px y centrado en el navegador */
    width: 400px; height: 400px;
    margin: auto;
    /* Borde sólido y fondo gris clarito */
    border: solid 1px black;
    background-color: #CCCCCC;
}
/* Creamos una ficha centrada y de color verdoso */
.ficha {
    width: 100px; height: 100px;
    background-color: olivedrab;
    border: solid 1px black;
    margin-left: 20px; margin-right: auto;
    margin-top: 20px; margin-bottom: 10px;
    text-align: center;
    float: left;
}

.fichaVolt {
    background-color: orangered;
}
```

• JavaScript

```
function girarCarta(e) {
    //Función que se ejecuta al hacer click

    //Eliminamos temporalmente el escuchador para no acumular más de un click
    const losDivs = document.getElementsByClassName('ficha');
    for (const div of losDivs) {
        div.removeEventListener("click", girarCarta, false);
    }

    var ficha = this;

    //Guardamos el contenido inicial de la ficha
    var contInicial = ficha.innerHTML;
    //Lo cambiamos por el texto que nos interese
    let contFicha = ficha.getAttribute('id');
    ficha.innerHTML = contFicha;
    //Y le añadimos una clase más para cambiar aspecto
    ficha.classList.add('fichaVolt');

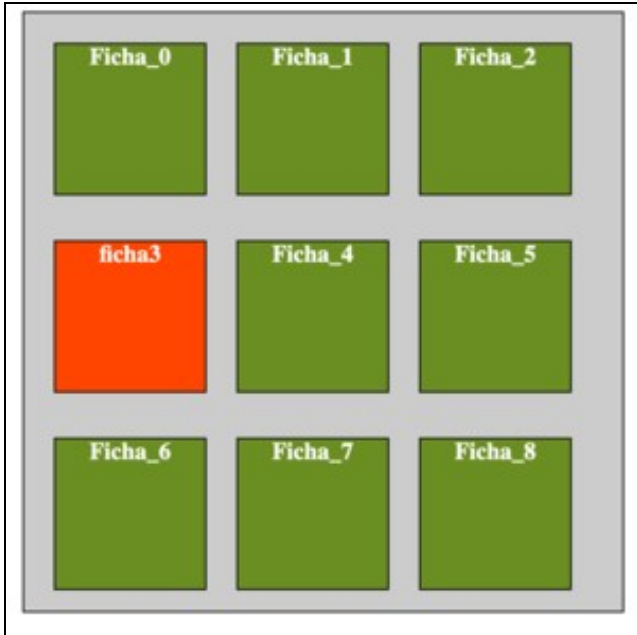
    //Realizamos una pausa de 3 segundos con setTimeout
    setTimeout(function () {
        //Creamos una función de Callback que
        //Reescribe el contenido anterior
        ficha.innerHTML = contInicial;
        ficha.classList.remove('fichaVolt');
    }, 3000); //No olvidar configurar el tiempo de pausa
}

//Número de fichas a crear
const numFichas = 9;
//Creamos las fichas como objetos DOM
for (let i = 0; i < numFichas; i++) {
    //Div donde vamos a crear las fichas
    const contenido = document.getElementById("contenido");
    //Creamos el div
```

```

const div = document.createElement("div");
//Contenido del div, un simple texto
const innerDiv = document.createTextNode(`Ficha_${i}`);
div.appendChild(innerDiv);
//Configuramos el atributo id distinto a cada ficha
div.setAttribute("id", `ficha${i}`);
//Le añadimos la clase 'ficha'
div.classList.add("ficha");
//Configuramos el escuchador click -> girarCarta
div.addEventListener("click", girarCarta, false);
//Lo añadimos al div 'contenido'
contenido.appendChild(div);
}

```



setTimeout

1.5.2 setInterval()

setInterval() ejecuta una función repetidamente con un intervalo de tiempo configurado en milisegundos.

Esta función devuelve un entero que representa el ID de la ejecución. Este ID es utilizado para cancelar la ejecución, empleando para ello el método **clearInterval()**.

Veamos el siguiente ejemplo donde se crea una web con un título que cambia de color cada 2 segundos. Paramos ese cambio de color con un botón que anula la función **setInterval()**:

• HTML

```

<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8" />
<title>JavaScript</title>
</head>
<body>
<h1 id="miH1">Este título cambia de color</h1>
<br />
<button id="para">Para clearInterval</button>

<script src="scripts.js"></script>
</body>
</html>

```

• JavaScript

```

//Dado un número devuelve uno aleatorio de 1-val

```

```
function randomVal(val) {
return Math.floor(Math.random() * val)
}

//Devuelve un string de la forma : rgb(255,255,255)
function randomColor() {
return ( `rgb(${randomVal(255)},${randomVal(255)},${randomVal(255)})` )
}

//Cambia de color el elemento 'miH1'
function cambiaColor() {
const miH1 = document.getElementById('miH1')
const nColor = randomColor()
miH1.style.color = nColor
}

//Cada 2 segundos se ejecuta la función cambiaColor
let miSetIntervalID = setInterval(cambiaColor, 2000)

//Paramos el setInterval() con el botón para
const botonPara = document.getElementById('para')
botonPara.addEventListener('click', function(){
clearInterval(miSetIntervalID)
console.log('Parao!!')
}, false)
```

[Volver](#)