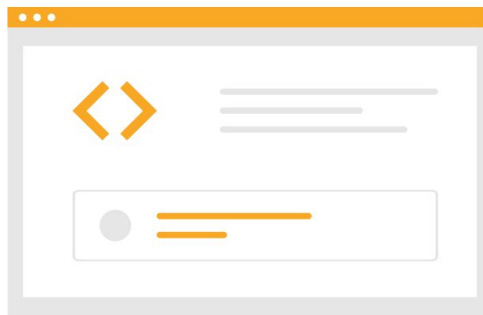


Callback & Promises JS

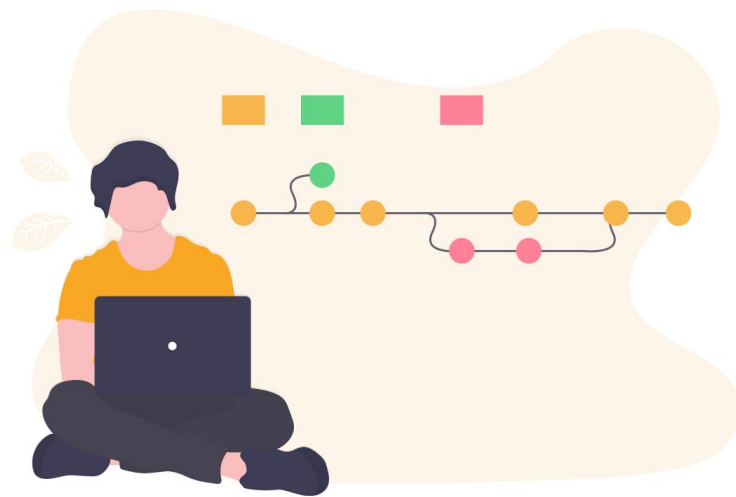
César Arellano
Diana Aviles
Jair Escamilla
Mauricio de Garay

En

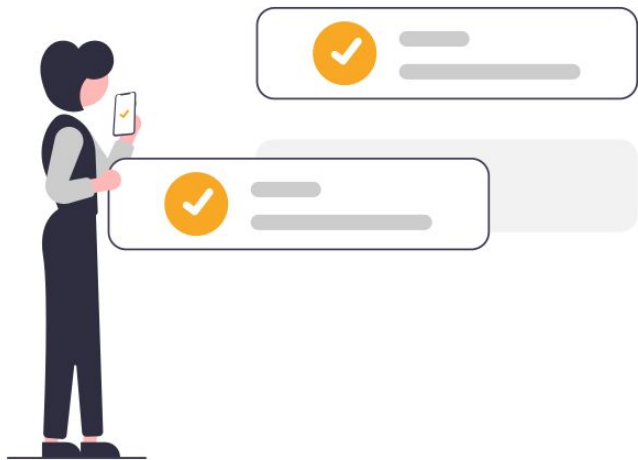


Callbacks

01



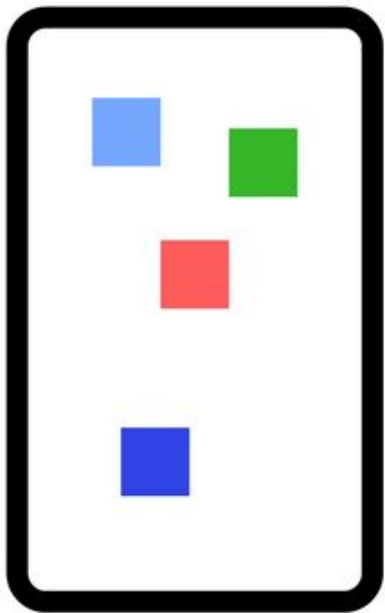
Función que se utiliza como
parametro en otra función



Eventloop

Es un patrón diseñado que espera y distribuye eventos o mensajes del programa.

02



Memory Heap

Espacio de memoria donde se asignan objetos, variables, scope (no hay un orden en particular).



Call Stack

Apila de forma organizada las instrucciones de nuestro programa

Task Queue



Donde se agregan las tareas que están listas para pasar al Call Stack

Microtask Queue



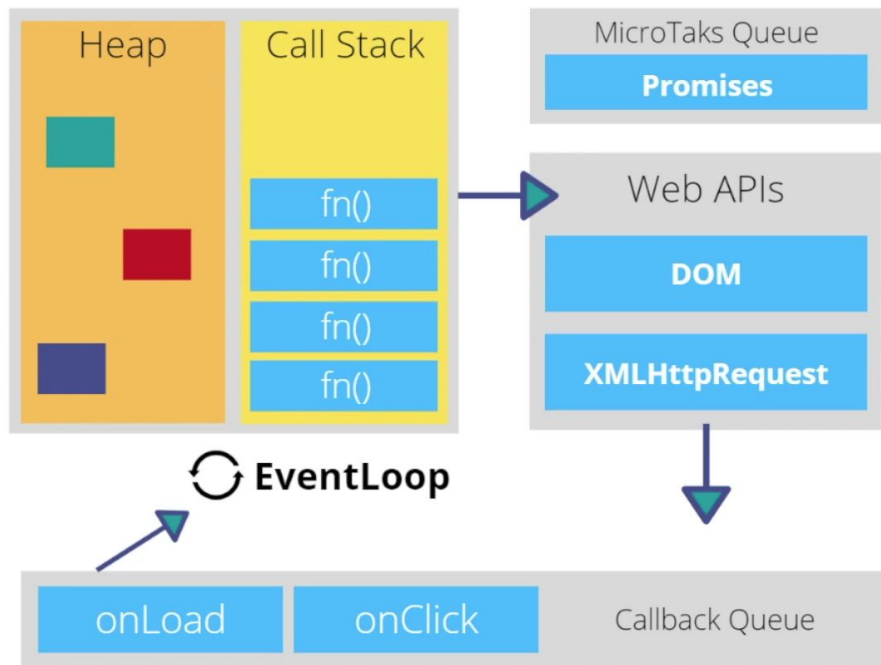
Tiene una prioridad mayor y trabaja las promesas.

Web APIs

- Event listeners
- AJAX calls
- Timeout methods

Ejemplos de: setTimeout, Elementos del DOM, fetch, XMLHttpRequest

Event Loop a detalle



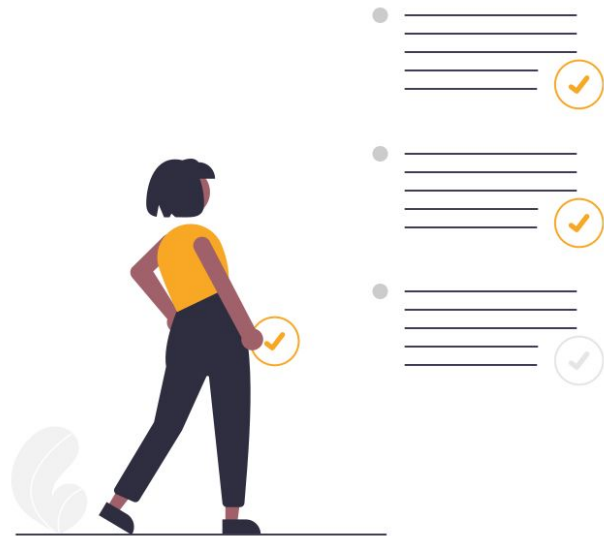
Revisa las tareas en la cola y pasa al Call Stack en dado caso que esté vacía.

Demos un vistazo

Ejemplo de Event Loop:
latentflip.com/loupe/

Promesas

03



En una función normal...

Sabemos lo que
va a pasar y
cuándo.

```
main()
{
    ...
    ...
    x=miFuncion(a,b)
    print(x)
}
```

```
miFuncion(a,b)
{
    x=a+b;
    return x;
}
```

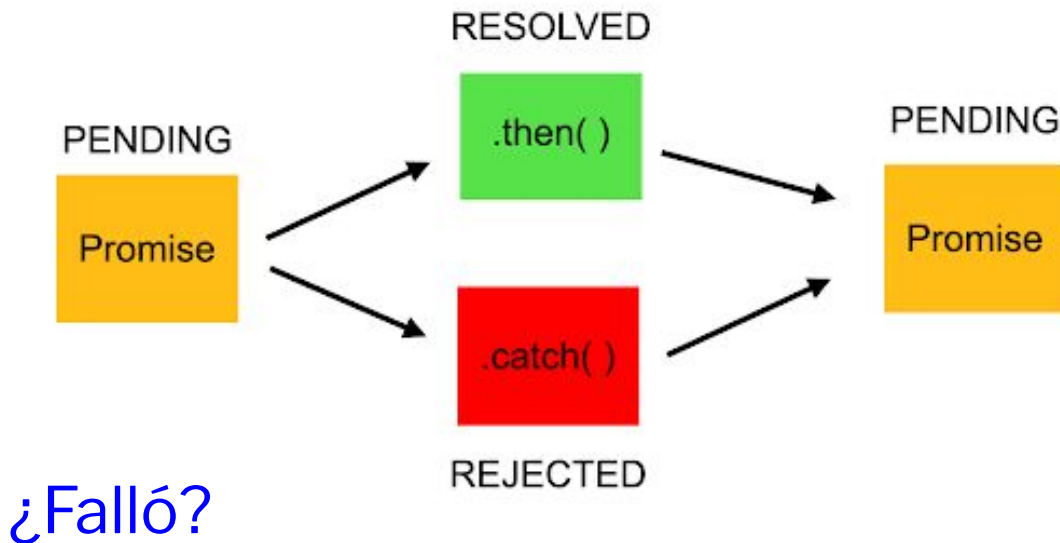
Pero para una función llamada asincrónicamente

...

Javascript necesita
garantizar
su funcionamiento

Promises: la promesa de que, en algún momento, algo tiene que regresar.

¿Funcionó?



```
function myDisplayer(some) {  
    document.getElementById("demo").innerHTML = some;  
}  
  
let myPromise = new Promise(function(myResolve, myReject) {  
    let x = 0;  
  
    // The producing code (this may take some time)  
  
    if (x == 0) {  
        myResolve("OK");  
    } else {  
        myReject("Error");  
    }  
});  
  
myPromise.then(  
    function(value) {myDisplayer(value);},  
    function(error) {myDisplayer(error);}  
);
```

```
$.ajax({  
    url: 'URL',  
    type: 'POST',  
    data: yourData,  
    datatype: 'json',  
    success: function (data) { successFunction(data); },  
    error: function (jqXHR, textStatus, errorThrown) { errorFunction(); }  
});
```

Ejemplo

04