

# CALLBACKS



# TIPOS DE CALLBACK

## ANÓNIMO

En este caso la función que pasamos como **callback** no tiene nombre, es decir que es una **función anónima**.

Como las funciones anónimas no pueden ser llamadas luego por su nombre, necesitamos escribirla dentro de la función que se encargará de al callback.

```
setTimeout( function(){  
    console.log('Hola Mundo!');  
} , 1000)
```

{}

# TIPOS DE CALLBACK

## DEFINIDO

La función que pasamos como **callback** puede ser una **función definida** previamente. Al momento de pasarla como parámetro de otra función, nos referiremos a la misma por su nombre.

```
let miCallback = () => console.log('Hola Mundo!');  
setTimeout( miCallback, 1000);
```



Cuando enviamos una función como parámetro la escribimos **sin los parentesis**, ya que no queremos que se ejecute en ese momento. Será la función que la recibe quien se encargue de ejecutarla.

```
{ código }
```

```
function nombreCompleto(nombre, apellido) {  
    return nombre + ' ' + apellido;  
};
```

```
function saludar(nombre, apellido, callback) {  
    return '¡Hola ' + callback(nombre, apellido) + '!';  
};
```

```
saludar('Juanito', 'Sánchez', nombreCompleto);
```

```
{ código }
```

```
function nombreCompleto(nombre, apellido) {  
    return nombre + ' ' + apellido;  
};
```

```
function saludar(nombre, apellido, callback) {  
    return '¡Hola ' + callback(nombre, apellido) + '!';  
};
```

```
saludar('Juanito', 'Sánchez', nombreCompleto);
```

Definimos la función **nombreCompleto()**.

La misma se encarga de unir con un espacio un nombre y un apellido.

Nos **devuelve** un **string**.

```
{ código }
```

```
function nombreCompleto(nombre, apellido) {  
    return nombre + ' ' + apellido;  
};
```

```
function saludar(nombre, apellido, callback) {  
    return '¡Hola ' + callback(nombre, apellido) + '!';  
};
```

```
saludar('Juanito', 'Sánchez', nombreCompleto);
```

Definimos la función **saludar()**.

La misma recibe un nombre, un apellido y un **callback** como parámetros.

Este último será la función que vamos a querer ejecutar internamente.

```
{ código }
```

```
function nombreCompleto(nombre, apellido) {  
    return nombre + ' ' + apellido;  
};
```

```
function saludar(nombre, apellido, callback) {  
    return '¡Hola ' + callback(nombre, apellido) + '!';  
};
```

```
saludar('Juanito', 'Sánchez', nombreCompleto);
```

Lo que queremos devolver es un string completo.

La primera parte la tenemos en el return: '¡Hola (...)' !.

El resto (...) vendrá de lo que nos retorne el **callback** en el momento en el que se ejecute.



```
{ código }
```

```
function nombreCompleto(nombre, apellido) {  
    return nombre + ' ' + apellido;  
};
```

```
function saludar(nombre, apellido, callback) {  
    return '¡Hola ' + callback(nombre, apellido) + '!';  
};
```

```
saludar('Juanito', 'Sánchez', nombreCompleto);
```

```
// Devolverá '¡Hola Juanito Sánchez!'
```

Ejecutamos la función **saludar**, le pasamos como parámetros un nombre, un apellido y la función **nombreCompleto()**.

**Primero** se ejecutará el **callback**, que va a devolver el nombre completo y **luego** se ejecutará la función **saludar()**, que va a devolver el saludo completo.



```
{ código }
```

```
function iniciales(nombre, apellido) {  
    return nombre[0] + apellido[0];  
};
```

```
function saludar(nombre, apellido, callback) {  
    return '¡Hola ' + callback(nombre, apellido) + '!';  
};
```

```
saludar('Juanito', 'Sánchez', iniciales);
```

Podríamos definir otra función que se encargue de devolvernos las iniciales del nombre y el apellido que nos pasen.

```
{ código }
```

```
function iniciales(nombre, apellido) {  
    return nombre[0] + apellido[0];  
};
```

```
function saludar(nombre, apellido, callback) {  
    return '¡Hola ' + callback(nombre, apellido) + '!';  
};
```

```
saludar('Juanito', 'Sánchez', iniciales);
```

```
// Devolverá '¡Hola JS!'
```

Esta vez cuando ejecutamos la función **saludar()**, le pasamos la función **iniciales()** como callback.

Nuevamente se ejecutará el callback, esta vez va a devolver las iniciales y luego se ejecutará la función **saludar()**, que va a devolver el saludo completo.