

# JavaScript desde cero

Módulo 6

# jQuery

# jQuery

“jQuery es una librería de JavaScript rápida, pequeña y llena de características. Hace cosas como manipulación de HTML, manejo de eventos animaciones y AJAX mucho más simple mediante una interfaz fácil de usar compatible con una multitud de navegadores.”

<https://jquery.com/>



# Instalación

Para instalar jQuery simplemente debemos añadirla a nuestro documento HTML antes de utilizarla. En [la sección de Descargas de la documentación oficial](#) disponemos de varias versiones. Podemos utilizar, por ejemplo, [la última versión de producción](#).

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta http-equiv="X-UA-Compatible" content="IE=edge">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <title>Document</title>
8  </head>
9  <body>
10
11    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
12    <script>
13      // Disponemos de jQuery
14    </script>
15  </body>
16  </html>
```

# Primeros pasos

Para verificar si se ha instalado todo correctamente, podemos utilizar **console.log** para mostrar la variable **\$**

*¿Por qué esa?*

jQuery expone todas sus funciones a través de ese símbolo. Una de las ventajas de jQuery es que requiere poco código para utilizarlo.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>

  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
  <script>
    // Disponemos de jQuery
    console.log($) // Objeto de jQuery
  </script>
</body>
</html>
```

# Seleccionar elementos HTML

Una de las operaciones más comunes con jQuery es seleccionar elementos HTML. Esta será la base para eventos, animaciones, etc.

El primer paso es siempre seleccionar un elemento.

*¿Cómo se puede hacer?*

El signo **\$** de jQuery es, en principio, una función que recibe un texto y retorna un elemento HTML. Ese texto es un selector CSS que usará jQuery para elegir el elemento en cuestión.

```
<div id="panel"></div>
<div class="paneles"></div>
<button>Boton</button>
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
  // Disponemos de jQuery
  const panel = $('#panel');
  const paneles = $('.paneles');
  const botones = $('button');
</script>
```

# Operaciones con HTML

Una vez elegido un elemento HTML, podemos realizar operaciones comunes, como son agregar una clase CSS, quitarla, animar ciertas propiedades de estilos.

También es posible aplicar un mismo código para muchos elementos HTML con la función **each()**.

```
<div id="panel"></div>
<div class="paneles"></div>
<button>Boton</button>
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
  // Disponemos de jQuery
  $('card').addClass('nueva_clase'); // Añadimos una clase CSS
  $('card').removeClass('clase_a_remove'); // Quitamos clases CSS
  $('card').animate({           // Animamos estas propiedades CSS en este tiempo
    backgroundColor: 'red'
  }, 2000)
  $('paneles').each(function(i, elemento){
    // Esta función sirve para aplicar código a muchos elementos HTML
  });
</script>
```

# Eventos con HTML

Uno de los principales aportes que hace JavaScript al diseño es la posibilidad que brinda para responder a eventos como el click del botón en un elemento, la introducción de datos en un campo de texto, etc.

Esos eventos se pueden “escuchar”, es decir, asignar funciones para que sean ejecutadas cuando ocurran. Algunos eventos, como el click, tienen un “acceso directo”.

```
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
  // Disponemos de jQuery
  $('boton').on('click', function(){
    alert('Hiciste click en el boton')
  });

  $('boton').click(function(){
    alert('Hiciste click en el boton')
  });
</script>
```



# Obtener datos de una fuente externa

Muchos diseños web tienen que obtener datos de terceros (como APIs u otros recursos). Para obtener datos de un sitio externo, podemos utilizar el **\$.get()** de jQuery (nótese que es **\$.get()** y no **\$get()**; debido a que **get()**

es una propiedad especial dentro del selector **\$**). Esta función **get** recibe dos parámetros: la url que estas pidiendo y la función que se ejecutará cuando obtenga los datos.

```
<script>
  // Disponemos de jQuery
  const boton = $('boton')
  const divEstado = $('#estado')

  boton.click(function(){
    $.get('https://jsonplaceholder.typicode.com/posts/',function(data){
      divEstado.html('Items cargados') // con .html() cambiamos el contenido de un elemento
      console.log(data)
    });
  });
</script>
```

# Consultar con la documentación

Siempre que queramos estudiar una nueva herramienta conviene aprender a buscar en su documentación.

La documentación oficial estará siempre actualizada y, si bien muchas veces puede no ser didáctica o fácil de entender, es la fuente primaria a la que debemos consultar.

En el caso de jQuery, pueden ver la documentación oficial [aquí](#).



# ¡Muchas gracias!

¡Sigamos trabajando!