

# Diseño de Interfaces Web

Integración de  
contenido interactivo

# Introducción

Es una librería de JavaScript que permite simplificar las instrucciones nativas de JavaScript de forma que se pueden hacer cambios en el DOM de forma sencilla y rápida.

Una vez se aprenda la sintaxis de jQuery, será muy fácil aplicar las nociones que se tienen en JavaScript sobre nuestros proyectos Web.

# Introducción

Se puede resumir en que jQuery hace que sea muy fácil y rápido escribir JavaScript, y muy sencillo leerlo.

# Eventos y efectos

Existen multitud de métodos que provocan efectos y eventos que nos permiten añadir dinamismo a nuestro HTML, como son:

- ☐ Click.
- ☐ Hide.
- ☐ Show.
- ☐ FadeIn.
- ☐ FadeOut.
- ☐ Change.
- ☐ Mouseenter.
- ☐ Etc.

# Eventos y efectos

Lista de eventos

[https://www.w3schools.com/jquery/jquery\\_ref\\_events.asp](https://www.w3schools.com/jquery/jquery_ref_events.asp)

Lista de efectos

[https://www.w3schools.com/jquery/jquery\\_ref\\_effects.asp](https://www.w3schools.com/jquery/jquery_ref_effects.asp)

# Cómo añadir la librería

Con objeto de que nuestra inclusión de la librería jQuery no afecte a nuestro HTML y CSS, se añadirán las líneas de código para añadir esta librería al final del body:

...

```
<script type="text/javascript"  
src="https://ajax.googleapis.com/ajax/libs/jquery  
/3.5.1/jquery.min.js"></script>  
</body>
```

# Funcionamiento

El funcionamiento es simple, se seleccionan elementos HTML de forma análoga a cómo se vio con los selectores en CSS, y después se aplican acciones sobre estos.

Para definir un comando de JavaScript que debe interpretarse como jQuery, se emplea el carácter \$.

# Document ready

Para prevenir que se ejecute nuestro código jQuery antes de que esté cargado completamente el DOM, se incluye todo nuestro código jQuery dentro del evento “document ready” de la siguiente forma:

```
$(document).ready(function(){  
...  
});
```



# Document ready

Existe otro método alternativo a este pero más corto:

```
$(function(){  
  
});
```

# Selectores

La sintaxis es análoga a los selectores vistos en CSS. Hay que indicar previamente el carácter \$ y poner entre comillas el selector.

Ej.: \$('span')

Existe un selector particular que es \$(this) y se emplea para seleccionar el elemento HTML actual. Muy útil tras seleccionar un elemento del DOM en un evento y posteriormente querer aplicar alguna acción sobre este.

# Actividad 1

Hacer clic en un botón que desplegará una capa con contenido. Si se vuelve a hacer clic en ese mismo botón, se cerrará dicha capa de contenido.

## Actividad 2

Crea un HTML con varios encabezamientos h2:

- ☐ Ocúltalos todos con jQuery.
- ☐ Ahora en lugar de ocultarlos, se quiere que se oculten solo cuando se pase por encima de estos.
- ☐ Para terminar, se prefiere que los h2 desaparezcan solo cuando se haga clic sobre ellos.

## Actividad 3

Se tienen dos botones, uno de ellos oculto al principio, y una capa de contenidos debajo oculta. Al hacer clic sobre el único botón visible, se muestra el contenido, el botón oculto, y se quita la visibilidad de este primer botón.

En caso de que se pulse ese nuevo botón, se ocultarán dicho botón y el contenido, y se mostrará únicamente el botón inicial.

# Juego persigue el bloque

Crea un recuadro 400x400px a modo de tablero donde, tras pulsar un botón de inicio en el centro, aparecerá un bloque de 25x25px en una posición aleatoria dentro de ese recuadro.

El usuario tiene que mover el ratón para pasar por encima. En ese momento, desaparecerá ese bloque y aparecerá otro en otro punto aleatorio. Una vez el usuario haya "capturado" 10 bloques acaba el juego.

Se debe registrar el tiempo que pasa desde que el usuario pulsa empezar hasta que captura el décimo bloque y mostrarlo por pantalla.

# Juego persigue el bloque

Eventos y efectos interesantes:

- ☐ Click.
- ☐ Hide.
- ☐ TimeStamp.
- ☐ Math.random().
- ☐ Math.floor.
- ☐ ParseInt.
- ☐ Css.
- ☐ Show.
- ☐ AppendTo.
- ☐ Etc.



# CEU

*Centro de Estudios  
Profesionales*

---

*Fundación San Pablo Andalucía*