



Instituto de Enseñanza Secundaria
Santiago Hernández

DESARROLLO WEB EN ENTORNO CLIENTE

(Curso 2021 – 2022)

UD5. jQUERY.

¿QUÉ ES JQUERY?

- Framework Javascript con librerías de código que contienen procesos o rutinas ya listos para usar
- Ayuda en la creación de:
 - interfaces de usuario
 - efectos dinámicos
 - aplicaciones que hacen uso de Ajax
 - ...
- Funciona de forma exacta en todas las plataformas más habituales: Evita programar en función del explorador.
- Facilidad para la creación de aplicaciones complejas del lado del cliente
- Licencia para uso en cualquier tipo de plataforma, personal o comercial

CÓMO USAR JQUERY

- Conocer las librerías del framework y utilizar sus
 - clases
 - propiedades
 - métodos.
- Incluir en nuestras páginas un script JavaScript que contiene el código de jQuery.
 - Se puede usar desde un CDN en la web (Content Delivery Network):
 - Google: `<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>`
 - Microsoft: `<script src="https://ajax.aspnetcdn.com/ajax/jquery/jquery-3.4.1.min.js"></script>`
 - jQuery: `<script src="jquery-3.4.1.min.js"></script>`
 - Se puede descargar y usar en local. Ocupa unos 56 KB (19 KB comp).
 - Lo enviará al cliente la primera vez y para las siguientes lo cogerá de la caché.

VENTAJAS DE JQUERY

Ventajas

- Gran aceptación por los programadores.
- Es estable.
- Bien documentado
- Actualizaciones gracias al equipo de desarrolladores.
- Plugins, componentes para interfaces de usuario, galerías, efectos...

Funcionalidades que aporta

- Mejoras de la web 2.0
- Efectos dinámicos
- Ajax
- interacción
- Interfaces de usuario avanzadas

SINTAXIS

Función \$()

- Alias del objeto jQuery: <https://code.jquery.com/jquery-3.4.1.js> (puede cambiarse)
- Recibe un nombre de elemento XHTML o CSS y devuelve un elemento o conjunto de elementos del DOM que encajan con lo recibido. Ejemplo:
 - `$("#prueba");` //devuelve el elemento con id="prueba.
 - `$("div");` // devuelve el conjunto de elementos del DOM con tag div
 - Selectores: https://www.w3schools.com/jquery/jquery_selectors.asp
- Realiza una acción sobre el/los elemento/s. Ejemplo:
 - `$(".alumno").removeClass("alumno").addClass("exAlumno");`
 - //los elementos de la clase "alumno" dejan de ser parte de esa clase. La funcion `removeClass` devuelve los elementos afectados como conjunto y a este se le aplica `addClass` y pasan a ser todos esos elementos de la clase "exAlumno."
 - <https://api.jquery.com/>

FUNCIONES

Ejemplos

- Función CSS: acceso a cualquier propiedad CSS:
 - `css(propiedad)`: devuelve el valor de dicha propiedad:
 - `var color = $("#miElemento").css("background-color"); // Obtiene el color`
 - `css(propiedad,valor)`: modifica el valor de dicha propiedad.
 - `$("#miElemento").css("background-color","red"); // Pone el fondo rojo`
 - <https://api.jquery.com/category/css/>
- Filtros: aplicar filtros a los elementos obtenidos en la consulta con ":"
 - `:first, :last, :odd, :hide, :visible`
 - `$("li").filter(":even").css("background-color","red"); //Fondo rojo rojo a los "li" impares`
 - <https://api.jquery.com/filter/>

MODIFICAR EL DOM

Funciones típicas:

- `html`: contenido html de un elemento
 - `miHTML=$("#miDiv").html();` // Obtiene el HTML del elemento
 - `$("#miDiv").html("Hola");` // Modifica el HTML del elemento
- `append/prepend`: añade elemento hijo al final o al principio:
 - `$("#miDiv").append("<p>Probando</p>");` // Añade el elemento al final.
- `empty()`: vacía el contenido html de un elemento(s) (hijos incluidos)
 - `$("#p").empty();` // Vacía todos los elementos P
- `remove()`: elimina el elemento(s)
 - `$("#div").remove();` // Elimina todos los divs
- <http://api.jquery.com/category/manipulation/dom-insertion-inside/>
<https://api.jquery.com/category/manipulation/dom-removal/>

MANEJO DE EVENTOS

- Igual que en js, podremos modificar los eventos asociados como propiedades del elemento identificado a través de jQuery:
 - `$("#target").click(function() {`
 - `alert("Función asociada al evento click del elemento con id target.");`
 - `});`
- `one()` la función asignada al evento se ejecuta una única vez:
 - `$('#boton1').one('click',function(){`
 - `//código que se ejecuta cuando se hace click sobre el botón la primera vez.`
 - `});`
- `on/off`: permiten asignar y desasignar eventos:
 - `$('#boton1').on('click',function(){`
 - `//código que se ejecuta cuando se hace click sobre el botón la primera vez.`
 - `});`
 - `$('#boton1').off();`

DELEGACIÓN DE EVENTOS

- Asignar automáticamente eventos a un elemento por el hecho de pertenecer a un grupo
- El evento se asocia al padre y se llama con un filtro que identifica el hijo que lo llama → El evento se lanza con target el hijo correspondiente.
- .on(). Sintaxis:
 - `.on(events [, selector] [, data], handler)`
 - events: eventos a asociar con la selección, 1 o varios separados por espacios.
 - selector: opcional, permitir filtrar los elementos hijos de la selección desde los que queremos que se lance el handler. Permite usar la delegación funcional.
 - data: opcional, cualquier tipo de dato que necesitemos pasar a la función.
 - handler: Función asociada al evento. Puede ser anónima.

ANIMACIONES

Mostrar y ocultar

- Show y hide:
 - `$('#elemento').show(3000);` // muestra el elemento con id elemento.
 - `$('.clase').hide(4000);` //muestra todos los elementos de clase clase.
 - Los números indican el tiempo en ms que dura la animación.
 - El tiempo se puede dar como `slow(600ms)`, `normal (400ms)` y `fast(200ms)`
- fadeIn y fadeOut: varían la opacidad.
- fadeTo: ajusta el nivel de opacidad:
 - `$('#p').fadeTo("normal", 0.6, function() {`
 - `// función callback`
 - `});`
- Si se añade una función callback al final de la animación, ésta se ejecuta al finalizar la animación para cada elemento.

ANIMACIONES

Función animate()

- Permite modificar las propiedades css de los elementos: velocidad, posición, función callback, etc.
- <http://api.jquery.com/animate/>

AJAX EN JQUERY

El método ajax()

➤ Sustituye al objeto XMLHttpRequest

➤ Sintaxis: →

➤ Métodos para trabajar con JSON:

- .getJSON: carga datos codificados en JSON desde un servidor usando una solicitud de tipo GET.
- .parseJSON **NO SE USA**
- https://www.w3schools.com/jquery/jquery_ref_ajax.asp

- \$.ajax({
 method:"POST",
 url:"prueba.php",
 data:{ nombre: "Casimiro", apellido: "Buenavista" }
})
- .done(function(msg) {
 alert("Recibo: " + msg);
})
- .fail(function() {
 alert("Error");
})
- .always(function() {
 alert("Trabajo realizado");
});