

**Asignatura:** Tecnologías para el Desarrollo de Aplicaciones Web.

**Tema:** Introducción a JQuery.

## INTRODUCCIÓN.

JQuery es una biblioteca que posee funcionalidades adicionales que permiten agregarle a las páginas diversas interacciones y efectos. Se pueden utilizar eventos, como `click`, `dblclick`, `hover`, `keypress`, `keyup`, `focus` y otros.

En la figura la figura 1 se muestra la sintaxis básica del uso de JQuery:

```
<script>
$( "#c1" ).click(function() {
    $( "#cont1" ).hide(1500) ;
});
</script>
```

**Figura 1.** Sintaxis de una instrucción con JQuery.

Donde:

<code>&lt;/script&gt;</code> <code>&lt;/script&gt;</code>	Representan las etiquetas que engloban la instrucción JQuery.
<code>"#c1"</code>	Es un selector que representa la identidad del elemento que dispara la acción.
<code>"#cont1"</code>	Es un selector que representa la identidad del elemento al cual se le asigna una acción.
<code>hide()</code>	Es un método, el cual en este caso es la acción de ocultar.
<code>click()</code>	Es el evento del clic.

Métodos de JQuery para efectos y animaciones:

Método	Función
<code>animate()</code>	Ejecuta una animación personalizada en los elementos seleccionados.
<code>clearQueue()</code>	Elimina todas las funciones restantes en cola de los elementos seleccionados.
<code>delay()</code>	Establece un retardo para todas las funciones en cola en los elementos seleccionados.
<code>dequeue()</code>	Elimina la siguiente función de la cola y luego ejecuta la función.
<code>fadeIn()</code>	Se desvanece en los elementos seleccionados.
<code>fadeOut()</code>	Desvanece los elementos seleccionados.
<code>fadeTo()</code>	Desvanece los elementos seleccionados a una opacidad dada.
<code>fadeToggle()</code>	Alterna entre los métodos <code>fadeIn()</code> y <code>fadeOut()</code> .
<code>finish()</code>	Detiene, elimina y completa todas las animaciones en cola para los elementos seleccionados.
<code>hide()</code>	Oculto los elementos seleccionados.
<code>queue()</code>	Muestra las funciones en cola en los elementos seleccionados.
<code>show()</code>	Muestra los elementos seleccionados.
<code>slideDown()</code>	Desliza (muestra) los elementos seleccionados.
<code>slideToggle()</code>	Alterna entre los métodos <code>slideUp()</code> y <code>slideDown()</code> .
<code>slideUp()</code>	Desliza (oculta) los elementos seleccionados.
<code>stop()</code>	Detiene la animación en ejecución para los elementos seleccionados.
<code>toggle()</code>	Alterna entre los métodos <code>hide()</code> y <code>show()</code> .

Métodos para agregar, eliminar o modificar estilos CSS o modificar directamente el contenido HTML.

Método	Función
<code>addClass()</code>	Agrega uno o más nombres de clase a los elementos seleccionados.
<code>after()</code>	Inserta contenido después de elementos seleccionados.
<code>append()</code>	Inserta contenido al final de los elementos seleccionados.
<code>appendTo()</code>	Inserta elementos HTML al final de los elementos seleccionados.
<code>attr()</code>	Establece o devuelve atributos / valores de elementos seleccionados.
<code>before()</code>	Inserta contenido antes de elementos seleccionados.

<code>clone()</code>	Hace una copia de los elementos seleccionados.
<code>css()</code>	Establece o devuelve una o más propiedades de estilo para los elementos seleccionados.
<code>detach()</code>	Elimina elementos seleccionados (mantiene datos y eventos).
<code>empty()</code>	Elimina todos los nodos secundarios y el contenido de elementos seleccionados.
<code>hasClass()</code>	Comprueba si alguno de los elementos seleccionados tiene un nombre de clase especificado.
<code>height()</code>	Establece o devuelve la altura de los elementos seleccionados.
<code>html()</code>	Establece o devuelve el contenido de los elementos seleccionados.
<code>innerHeight()</code>	Devuelve la altura de un elemento (incluye el relleno, pero no el borde).
<code>innerWidth()</code>	Devuelve el ancho de un elemento (incluye el relleno, pero no el borde).
<code>insertAfter()</code>	Inserta elementos HTML después de los elementos seleccionados.
<code>insertBefore()</code>	Inserta elementos HTML antes de elementos seleccionados.
<code>offset()</code>	Establece o devuelve las coordenadas de desplazamiento de los elementos seleccionados (en relación con el documento).
<code>offsetParent()</code>	Devuelve el primer elemento padre.
<code>outerHeight()</code>	Devuelve la altura de un elemento (incluye relleno y borde).
<code>outerWidth()</code>	Devuelve el ancho de un elemento (incluye relleno y borde).
<code>position()</code>	Devuelve la posición (relativa al elemento padre) de un elemento.
<code>prepend()</code>	Inserta contenido al principio de los elementos seleccionados.
<code>prependTo()</code>	Inserta elementos HTML al principio de los elementos seleccionados.
<code>prop()</code>	Establece o devuelve propiedades / valores de elementos seleccionados.
<code>remove()</code>	Elimina los elementos seleccionados (incluidos datos y eventos).
<code>removeAttr()</code>	Elimina uno o más atributos de elementos seleccionados.
<code>removeClass()</code>	Elimina una o más clases de elementos seleccionados.
<code>removeProp()</code>	Elimina una propiedad establecida por el método <code>prop()</code> .
<code>replaceAll()</code>	Reemplaza elementos seleccionados con nuevos elementos HTML.
<code>replaceWith()</code>	Sustituye elementos seleccionados por nuevos contenidos.
<code>scrollLeft()</code>	Establece o devuelve la posición de la barra de desplazamiento horizontal de los elementos seleccionados.
<code>scrollTop()</code>	Establece o devuelve la posición de la barra de desplazamiento vertical de los elementos seleccionados.
<code>text()</code>	Establece o devuelve el contenido de texto de los elementos seleccionados.
<code>toggleClass()</code>	Alterna entre agregar / eliminar una o más clases de elementos seleccionados.
<code>unwrap()</code>	Elimina el elemento padre de los elementos seleccionados.
<code>val()</code>	Establece o devuelve el atributo de valor de los elementos seleccionados (para elementos de formulario).
<code>width()</code>	Establece o devuelve el ancho de los elementos seleccionados.
<code>wrap()</code>	Envuelve elementos HTML alrededor de cada elemento seleccionado.
<code>wrapAll()</code>	Envuelve elementos HTML alrededor de todos los elementos seleccionados.
<code>wrapInner()</code>	Envuelve elementos HTML alrededor del contenido de cada elemento seleccionado.

### Métodos de JQuery para AJAX

AJAX permite intercambiar de datos y actualizar contenido de una página sin recargarla, directamente con el servidor.

Método	Función
<code>\$.ajax()</code>	Realiza una solicitud asíncrona AJAX.
<code>\$.ajaxPrefilter()</code>	Maneje las opciones personalizadas de AJAX o modifique las opciones existentes antes de enviar cada solicitud y antes de que sean procesadas por <code>\$.ajax()</code> .
<code>\$.ajaxSetup()</code>	Establece los valores predeterminados para futuras solicitudes AJAX.
<code>\$.ajaxTransport()</code>	Crea un objeto que maneja la transmisión real de datos AJAX.
<code>\$.get()</code>	Carga datos desde un servidor utilizando una solicitud AJAX HTTP GET.
<code>\$.getJSON()</code>	Carga los datos codificados en JSON de un servidor que utiliza una solicitud HTTP GET.
<code>\$.getScript()</code>	Carga (y ejecuta) un JavaScript desde un servidor utilizando una solicitud AJAX HTTP GET.
<code>\$.param()</code>	Crea una representación serializada de una matriz o un objeto (se puede utilizar como cadena de consulta URL para solicitudes AJAX).

<code>\$.post()</code>	Carga datos desde un servidor mediante una solicitud AJAX HTTP POST.
<code>ajaxComplete()</code>	Especifica una función que se ejecutará cuando se complete la solicitud AJAX.
<code>ajaxError()</code>	Especifica una función que se ejecutará cuando la solicitud AJAX finaliza con un error.
<code>ajaxSend()</code>	Especifica una función que se ejecutará antes de enviar la solicitud AJAX.
<code>ajaxStart()</code>	Especifica una función que se ejecutará cuando comience la primera solicitud AJAX.
<code>ajaxStop()</code>	Especifica una función que se ejecutará cuando todas las solicitudes AJAX hayan finalizado.
<code>ajaxSuccess()</code>	Especifica una función que se ejecutará cuando se complete una solicitud AJAX correctamente.
<code>load()</code>	Carga los datos de un servidor y coloca los datos devueltos en el elemento seleccionado.
<code>serialize()</code>	Codifica un conjunto de elementos de formulario como una cadena para su envío.
<code>serializeArray()</code>	Codifica un conjunto de elementos de formulario como una matriz de nombres y valores.

Otros métodos de JQuery.

Método	Función
<code>data()</code>	Adjunta datos a, o obtiene datos de elementos seleccionados.
<code>each()</code>	Ejecutar una función para cada elemento coincidente.
<code>get()</code>	Obtener los elementos DOM acompañados por el selector.
<code>index()</code>	Buscar un elemento dado entre los elementos emparejados.
<code>\$.noConflict()</code>	Libere el control de JQuery de la variable \$.
<code>\$.param()</code>	Crear una representación serializada de una matriz o un objeto (se puede utilizar como cadena de consulta de URL para solicitudes AJAX).
<code>removeData()</code>	Elimina una pieza de datos previamente almacenada.
<code>size()</code>	Eliminado en la versión 3.0. Utilice la propiedad <code>length</code> en su lugar.
<code>toArray()</code>	Recuperar todos los elementos DOM contenidos en el conjunto JQuery, como una matriz.

Eventos de JQuery que disparan las acciones necesarias para ejecutar cualquier función.

Evento	Función
<code>blur()</code>	Conecta/desencadena el evento de desenfoque.
<code>change()</code>	Conecta/activa el evento de cambio.
<code>click()</code>	Conecta/activa el evento de clic.
<code>dblclick()</code>	Conecta/activa el evento de doble clic.
<code>event.currentTarget</code>	El elemento DOM actual dentro de la fase de burbujeo del evento.
<code>event.data</code>	Contiene los datos opcionales transferidos a un método de evento cuando el controlador de ejecución actual esta enlazado.
<code>event.delegateTarget</code>	Devuelve el elemento donde se ha conectado el controlador de eventos JQuery actualmente llamado.
<code>event.preventDefault()</code>	Devuelve si <code>event.preventDefault()</code> fue llamado para el objeto de evento.
<code>event.isImmediatePropagationStopped()</code>	Devuelve si <code>event.stopImmediatePropagation()</code> se llama para el objeto de evento.
<code>event.isPropagationStopped()</code>	Devuelve si <code>event.stopPropagation()</code> fue llamado para el objeto de evento.
<code>event.namespace</code>	Devuelve el espacio de nombres especificado cuando se desencadena el evento.
<code>event.pageX</code>	Devuelve la posición del ratón en relación con el borde izquierdo del documento.
<code>event.pageY</code>	Devuelve la posición del mouse en relación con el borde superior del documento.
<code>event.preventDefault()</code>	Evita la acción predeterminada del evento.
<code>event.relatedTarget</code>	Devuelve el elemento entrado o salido en el movimiento del ratón.

<code>event.result</code>	Contiene el ultimo / anterior valor devuelto por un manejador de eventos disparado por el evento especificado.
<code>event.stopImmediatePropagation()</code>	Evita que se llamen a otros controladores de eventos.
<code>event.stopPropagation()</code>	Evita que el evento burbujee el árbol DOM, evitando que los manejadores de origen sean notificados del evento.
<code>event.target</code>	Devuelve el elemento DOM que ha desencadenado el evento.
<code>event.timeStamp</code>	Devuelve el número de milisegundos desde el 1 de enero de 1970, cuando se desencadena el evento.
<code>event.type</code>	Devuelve que tipo de evento se ha activado.
<code>event.which</code>	Devuelve la tecla del teclado o el botón del ratón para el evento.
<code>focus()</code>	Conecta/activa el evento de enfoque.
<code>focusin()</code>	Conecta un controlador de eventos al evento <code>focus</code> .
<code>focusout()</code>	Conecta un controlador de eventos al evento de enfoque.
<code>hover()</code>	Conecta dos manejadores de eventos al evento <code>hover</code> .
<code>keydown()</code>	Conecta/activa el evento <code>keydown</code> .
<code>keypress()</code>	Conecta/activa el evento de pulsaciones de teclas.
<code>keyup()</code>	Conecta/activa el evento <code>keyup</code> .
<code>mousedown()</code>	Conecta/activa el evento <code>mousedown</code> .
<code>mouseenter()</code>	Conecta/activa el evento <code>mouseenter</code> .
<code>mouseleave()</code>	Adjunta/activa el evento <code>mouseleave</code> .
<code>mousemove()</code>	Adjunta/activa el evento <code>mousemove</code> .
<code>mouseout()</code>	Conecta/activa el evento <code>mouseout</code> .
<code>mouseover()</code>	Conecta/activa el evento <code>mouseover</code> .
<code>mouseup()</code>	Adjunta/desencadena el evento <code>mouseup</code> .
<code>off()</code>	Elimina los manejadores de eventos conectados con el método <code>on()</code> .
<code>on()</code>	Agrega los controladores de eventos a elementos.
<code>one()</code>	Agrega uno o más controladores de eventos a elementos seleccionados. Este controlador se lo se puede activar una vez por elemento.
<code>\$.proxy()</code>	Toma una función existente y devuelve una nueva con un contexto particular.
<code>ready()</code>	Especifica una función a ejecutar cuando el DOM está completamente cargado.
<code>resize()</code>	Adjunta/activa el evento de cambio de tamaño.
<code>scroll()</code>	Conecta/activa el evento de desplazamiento
<code>select()</code>	Conecta/activa el evento seleccionado.
<code>submit()</code>	Adjunta/activa el evento de envío.
<code>trigger()</code>	Activa todos los eventos vinculados a los elementos seleccionados.
<code>triggerHandler()</code>	Dispara todas las funciones enlazadas a un evento especificado para los elementos seleccionados.

Los siguientes selectores de JQuery permiten seleccionar y especificar secciones de una página usando identificadores (ID), clases o elementos (etiquetas). Se pueden encadenar para seleccionar descendientes o sus atributos. El uso de los selectores es una gran ventaja de JQuery sobre JavaScript.

Selector	Ejemplo	Selecciona
<code>*</code>	<code>\$("*")</code>	Todos los elementos.
<code>#id</code>	<code>\$("#lastname")</code>	El elemento con <code>id = "lastname"</code>
<code>.class</code>	<code>\$(".intro")</code>	Todos los elementos con <code>clase = "intro"</code>
<code>.class, .class</code>	<code>\$(".intro, .demo")</code>	Todos los elementos con la clase <code>"intro"</code> o

		"demo"
element	\$( "p" )	Todos los elementos <p>
el1,el2,el3	\$( "h1,div,p" )	Todos los elementos <h1>, <div> y <p>
:first	\$( "p:first" )	El primer elemento <p>
:last	\$( "p:last" )	El último <p> elemento.
:even	\$( "tr:even" )	Todos los elementos <tr>
:odd	\$( "tr:odd" )	Todos los elementos impares <tr>
:first-child	\$( "p:first-child" )	Todos los <p> elementos que son el primer hijo de su padre.
:first-of-type	\$( "p:first-of-type" )	Todos los <p> elementos que son el primer <p> elemento de su padre.
:last-child	\$( "p:last-child" )	Todos los <p> elementos que son el último hijo de su padre.
:last-of-type	\$( "p:last-of-type" )	Todos los <p> elementos que son el último <p> elemento de su padre.
:nth-child(n)	\$( "p:nth-child(2) " )	Todos los <p> elementos que son el segundo hijo de su padre.
:nth-last-child(n)	\$( "p:nth-last-child(2) " )	Todos los <p> elementos que son el segundo hijo de su padre, contando desde el último hijo.
:nth-of-type(n)	\$( "p:nth-of-type(2) " )	Todos los <p> elementos que son el segundo <p> elemento de su padre.
:nth-last-of-type(n)	\$( "p:nth-last-of-type(2) " )	Todos los <p> elementos que son el segundo <p> elemento de su padre, contando desde el último niño.
:only-child	\$( "p:only-child" )	Todos los <p> elementos que son el único hijo de su padre.
:only-of-type	\$( "p:only-of-type" )	Todos los <p> elementos que son el único hijo, de su tipo, de su padre.
parent > child	\$( "div > p" )	Todos los <p> elementos que son hijos directos de un elemento <div>
parent descendant	\$( "div p" )	Todos los <p> elementos que son descendientes de un elemento <div>
element + next	\$( "div + p" )	El elemento <p> que está junto a cada elemento <div>
element ~ siblings	\$( "div ~ p" )	Todos los <p> elementos que son hermanos de un elemento <div>
:eq(index)	\$( "ul li:eq(3) " )	El cuarto elemento de una lista (el índice empieza en 0).
:gt(no)	\$( "ul li:gt(3) " )	Lista de elementos con un índice superior a 3.
:lt(no)	\$( "ul li:lt(3) " )	Lista de elementos con un índice inferior a 3.
:not(selector)	\$( "input:not(:empty) " )	Todos los elementos de entrada que no están vacíos.
:header	\$( ":header" )	Todos los elementos de encabezado <h1>, <h2> ...
:animated	\$( ":animated" )	Todos los elementos animados.
:focus	\$( ":focus" )	El elemento que actualmente tiene foco.
:contains(text)	\$( ":contains('Hello') " )	Todos los elementos que contienen el texto "Hello"
:has(selector)	\$( "div:has(p) " )	Todos los elementos <div> que tienen un elemento <p>
:empty	\$( ":empty" )	Todos los elementos que están vacíos.

:parent	\$(":parent")	Todos los elementos que son los padres de otro elemento.
:hidden	\$("p:hidden")	Todos los elementos <p> ocultos.
:visible	\$("table:visible")	Todas las tablas visibles.
:root	\$(":root")	El elemento raíz del documento.
:lang(language)	\$("p:lang(de)")	Todos los <p> elementos con un valor de atributo lang que comienza con "de"
[attribute]	\$("[href]")	Todos los elementos con un atributo href
[attribute=value]	\$("[href='default.htm']")	Todos los elementos con un valor de atributo href igual a "default.htm"
[attribute!=value]	\$("[href!='default.htm']")	Todos los elementos con un valor de atributo href no igual a "default.htm"
[attribute\$=value]	\$("[href\$='.jpg']")	Todos los elementos con un valor de atributo href que termina en ".jpg"
[attribute =value]	\$("[title ='Tomorrow']")	Todos los elementos con un valor de atributo de título igual a 'Tomorrow' o comenzando con 'Tomorrow' seguido de un guión.
[attribute^=value]	\$("[title^='Tom']")	Todos los elementos con un valor de atributo de título comenzando por "Tom"
[attribute~=value]	\$("[title~='hello']")	Todos los elementos con un valor de atributo de título que contiene la palabra específica "hello"
[attribute*=value]	\$("[title*='hello']")	Todos los elementos con un valor de atributo de título que contiene la palabra "hello"
:input	\$(":input")	Todos los elementos de entrada.
:text	\$(":text")	Todos los elementos de entrada con type = "text"
:password	\$(":password")	Todos los elementos de entrada con type = "password"
:radio	\$(":radio")	Todos los elementos de entrada con type = "radio"
:checkbox	\$(":checkbox")	Todos los elementos de entrada con type = "checkbox"
:submit	\$(":submit")	Todos los elementos de entrada con type = "submit"
:reset	\$(":reset")	Todos los elementos de entrada con type = "reset"
:button	\$(":button")	Todos los elementos de entrada con type = "button"
:image	\$(":image")	Todos los elementos de entrada con type = "image"
:file	\$(":file")	Todos los elementos de entrada con type = "file"
:enabled	\$(":enabled")	Todos los elementos de entrada habilitados.
:disabled	\$(":disabled")	Todos los elementos de entrada inhabilitados.
:selected	\$(":selected")	Todos los elementos de entrada seleccionados.
:checked	\$(":checked")	Todos los elementos de entrada seleccionados.

Los siguientes son algunos ejemplos del uso de los selectores con JQuery.

1. Seleccionar un área del documento con ID sección:

Dr. Alejandro S. Cifuentes A.

Con JavaScript: `document.getElementById("seccion");`  
Con JQuery: `$("#seccion");`

2. Seleccionar todos los enlaces en la página:

Con JavaScript: `document.getElementsByTagName("a");`  
Con JQuery: `$("a");`

3. Seleccionar con JQuery todos los enlaces en el área sección:

`$("#seccion, a")`

4. Seleccionar con JQuery todos los enlaces que contengan el texto "más información":

`$("a:contains('más información')")`

5. Seleccionar con JQuery todos los enlaces con el atributo Nofollow:

`$("a[@ref='nofollow']")`;

## EJERCICIOS.

1. Ocultar bloque

```
$("#c1").click(function() {  
    $("#cont1").hide(1500);  
});
```

2. Mostrar bloque

```
$("#c2").click(function() {  
    $("#cont1").show("slow");  
});
```

3. Ocultar/Mostrar bloque

```
$("#c1b").click(function() {  
    $("#cont1").toggle(1500);  
}, function() {  
    $("#cont1").toggle(1500);  
});
```

4. Cambiar tamaño

```
$("#c3").click(function() {  
    $("#cont1").animate({fontSize:'2.4em',width:400,padding:24}, 2500);  
});
```

5. Ocultar con FadeOut

```
$("#c4").click(function() {  
    $("#cont1").fadeToggle(3000);  
});
```

6. Mostrar con FadeIn

```
$("#c5").click(function() {  
    $("#cont1").fadeIn(3000);  
});
```

7. Mover

```
$("#c6").click(function() {  
    $("#cont1").animate({opacity: "0.1", left:  
    "+=400",fontSize:'1em',width: "200"}, 1200)
```

```

        .animate({opacity: "0.4", top: "+=160", height: "20", width:
"80",fontSize:'0.5em'}, "slow")
        .animate({opacity: "1", left: "0", width: "260"}, "slow")
        .animate({top: "0",width: "260",fontSize:'1.2em'}, "fast")
        .slideUp()
        .slideDown(1800)
        return false;

});

```

#### 8. Cambiar estilo CSS

```

$("#c7").click(function() {
    $("#cont1").css({'border':'4px solid #b7e5ff','color':'#cc3333','font-
weight':'bold','background-color':'#ffffff'});
});

```

#### 9. Tocar para ocultarme

```

$("#cont2").click(function() {
    $(this).hide().delay(1500).show(1500);
});

```

#### 10. Fondo amarillo

```

$(".yellow").hover(
    function() { $(this).addClass("hover"); },
    function() { $(this).removeClass("hover"); }
);

```

#### 11. Contador de caracteres

```

$('textarea').keyup(updateCount);
$('textarea').keydown(updateCount);
function updateCount() {
    var cs = $(this).val().length;
    $('#characters').text(cs);
}

```

#### 12. Cargar contenido externo

```

$("#c8").click(function() {
    $("#noticias").load("../facebook/problemas-simbolos.php");
});

```

#### 13. Ocultar

```

$("#c9").click(function() {
    $("#noticias").hide(1500);
});

```

#### 14. GET (Ver código fuente)

```

$("#c10").click(function() {
    $.get('https://norfipc.com/codigos/jquery-ejemplos-practicos-usar-paginas-
web.php', function(data, status){
        alert("Data: " + data + "\nStatus: " + status);
    });
});

```

#### 15. Versión de JQuery

```

$("#c11").click(function() {

```



```
var version=$().jquery;  
alert("Estas usando la versión: " +version+" de JQuery");  
});
```