

Eventos JQUERY

Simple

// Al pulsar en un elemento p se muestra en consola un mensaje.

```
$( "p" ).on( "click", function() {  
    console.log( "<p> fue pulsado" );  
});
```

Evento Blur.

El evento Blur ocurre cuando se pierde el foco en un input.

```
<!DOCTYPE html>  
<html>  
<head>  
<script src="//code.jquery.com/jquery-1.11.1.min.js"></script>  
  <meta charset="utf-8">  
  <title>jQuery: Attach a function to the blur event</title>  
</head>  
<body>  
  <form>  
    <input id="field1" type="text" value="Field 1">  
    <input id="field2" type="text" value="Field 2">  
  </form>  
  
  <p>Write something in the input Field 1, and then click outside the field to check  
  lose focus.</p>
```

```
$( "#field1" ).blur(function() {  
    alert( "Lose focus from Field1" );  
});  
</body>  
</html>
```

Varios eventos

// Cuando un usuario se situe sobre cualquier div o lo abandone se muestra un mensaje

```
// we expect a console message bind to multiple events  
$( "div" ).on( "mouseenter mouseleave", function() {  
    console.log( "mouse hovered over or left a div" );  
});
```

Varios eventos, acciones en función del evento

```
$( "div" ).on({  
    mouseenter: function() {  
        console.log( "hovered over a div" );  
    },  
    mouseleave: function() {  
        console.log( "mouse left a div" );  
    },  
    click: function() {  
        console.log( "clicked on a div" );  
    }  
});
```

Aplicar un funcion a un evento

```
$( "div" ).on( "click", function( event ) {  
    console.log( "event object:" );  
    console.dir( event );  
});
```

Paso de parámetros a un evento

```
$( "p" ).on( "click", {  
    foo: "bar"  
}, function( event ) {  
    console.log( "event data: " + event.data.foo + " (should be 'bar')" );  
});
```

Eventos que se ejecutan una sola vez

```
$( "p" ).one( "click", function() {  
    console.log( "You just clicked this for the first time!" );  
    $( this ).click(function() {  
        console.log( "You have clicked this before!" );  
    });  
});
```

Even Handling. Prevent. (Evitar).

```
$( "a" ).click(function( event ) {  
    var elem = $( this );  
    if ( elem.attr( "href" ).match( "evil" ) ) {  
        event.preventDefault();  
        elem.addClass( "evil" );  
    }  
});
```

Ejemplo:

```
<!DOCTYPE html>  
<html>  
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  
<script>  
$(document).ready(function(){  
    $("a").click(function(event){  
        event.preventDefault();  
    });  
});  
</script>  
</head>  
<body>  
  
<a href="http://w3schools.com/">Go to W3Schools.com</a>  
  
<p>The preventDefault() method will prevent the link above from following the URL.</p>  
  
</body>  
</html>
```

Listado de eventos

1) Eventos del ratón

A continuación podemos ver una lista de los eventos que se pueden definir en jQuery que tienen que ver con el ratón. Es decir, cómo definir eventos cuando el usuario realiza diferentes acciones con el ratón sobre los elementos de la página.

`click()`

Sirve para generar un evento cuando se produce un clic en un elemento de la página.

`dblclick()`

Para generar un evento cuando se produce un doble clic sobre un elemento.

`hover()`

Esta función en realidad sirve para manejar dos eventos, cuando el ratón entra y sale de encima de un elemento. Por tanto espera recibir dos funciones en vez de una que se envía a la mayoría de los eventos.

`mousedown()`

Para generar un evento cuando el usuario hace clic, en el momento que presiona el botón e independientemente de si lo suelta o no. Sirve tanto para el botón derecho como el izquierdo del ratón.

`mouseup()`

Para generar un evento cuando el usuario ha hecho clic y luego suelta un botón del ratón. El evento `mouseup` se produce sólo en el momento de soltar el botón.

`mouseenter()`

Este evento se produce al situar el ratón encima de un elemento de la página.

`mouseleave()`

Este se desata cuando el ratón sale de encima de un elemento de la página.

`mousemove()`

Evento que se produce al mover el ratón sobre un elemento de la página.

`mouseout()`

Este evento sirve para lo mismo que el evento `mouseout` de JavaScript. Se desata cuando el usuario sale con el ratón de la superficie de un elemento.

`mouseover()`

Sirve para lo mismo que el evento `mouseover` de Javascript. Se produce cuando el ratón está sobre un elemento, pero tiene como particularidad que puede producirse varias veces mientras se mueve el ratón sobre el elemento, sin necesidad de haber salido.

`toggle()`

Sirve para indicar dos o más funciones para ejecutar cosas cuando el usuario realiza clics, con la particularidad que esas funciones se van alternando a medida que el usuario hace clics.

2) Eventos relacionados con el teclado

A continuación se muestran los eventos que pueden modelizarse como respuesta a la pulsación de teclas del teclado.

`keydown()`

Este evento se produce en el momento que se presiona una tecla del teclado, independientemente de si se libera la presión o se mantiene. Se produce una única vez en el momento exacto de la presión.

`keypress()`

Este evento ocurre cuando se digita un carácter, o se presiona otro tipo de tecla. Es como el evento `keypress` de Javascript, por lo que se entiende que `keypress()` se ejecuta una vez, como respuesta a una pulsación e inmediata liberación de la tecla, o varias veces si se pulsa una tecla y se mantiene pulsada.

`keyup()`

El evento `keyup` se ejecuta en el momento de liberar una tecla, es decir, al dejar de presionar una tecla que teníamos pulsada.

Nota: a través del objeto evento, que reciben las funciones que indiquemos como parámetro de estos métodos, podemos saber qué tecla se está pulsando, aparte de otras muchas informaciones.

3) Eventos combinados teclado o ratón

Ahora mostramos varios eventos que pueden producirse tanto por el ratón como por el teclado, es decir, como resultado de una acción con el ratón o como resultado de presionar teclas en el teclado.

`focusin()`

Evento que se produce cuando el elemento gana el foco de la aplicación, que puede producirse al hacer clic sobre un elemento o al presionar el tabulador y situar el foco en ese elemento.

`focusout()`

Ocurre cuando el elemento pierde el foco de la aplicación, que puede

ocurrir cuando el foco está en ese elemento y pulsamos el tabulador, o nos movemos a otro elemento con el ratón.

focus()

Sirve para definir acciones cuando se produce el evento focus de Javascript, cuando el elemento gana el foco de la aplicación.

Más información en: <https://api.jquery.com/category/events/>