



## C.F.G.S.: DESARROLLO DE APLICACIONES WEB

### Módulo: DESARROLLO WEB EN ENTORNO CLIENTE

## 03 GESTIÓN DE EVENTOS

Los eventos sirven para que el usuario interaccione con la página. A medida que se producen podemos programar acciones en respuesta a éstos.

### Eventos provocados por el ratón

Evento	Descripción
click	Pulsar sobre el botón izquierdo del ratón
dblclick	Doble click sobre el botón izquierdo del ratón
mousedown	Pulsar (sin soltar) un botón del ratón
mousemove	Mover el puntero del ratón dentro de un elemento
mouseleave mouseout	El ratón está dentro de un elemento y <i>sale</i> del elemento
mouseenter mouseover	Al revés que el anterior, el puntero del ratón " <i>entra</i> " en el elemento (pasa por encima del elemento)
mouseup	Soltar el botón que estaba pulsado en el ratón

### Eventos disparados en formularios

Evento	Descripción
blur	Cuando un elemento pierde el foco
change	Cuando un elemento que se ha modificado pierde el foco
focus	Cuando un elemento obtiene el foco
select	Seleccionar un texto dentro de un input de tipo text o dentro de un textarea)
submit	Enviar el formulario

### Eventos provocados por el teclado

keydown	Pulsar una tecla del teclado (sin soltar)
keypress	Pulsar una tecla de carácter alfanumérico
keyup	Soltar una tecla pulsada

**Eventos disparados por la ventana**

resize	Se ha modificado el tamaño de la ventana del navegador
scroll	Al utilizar el scroll

[http://www.w3schools.com/jquery/jquery\\_events.asp](http://www.w3schools.com/jquery/jquery_events.asp)

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

Los eventos pueden tener asociados oyentes de eventos: funciones que se ejecutan en respuesta a la acción detectados por el navegador. Estas funciones reciben como argumento un objeto de tipo evento con sus propiedades.

**Objeto Evento**

Objeto que recibe la función oyente (controladora). Se utiliza muy a menudo para prevenir la acción predeterminada del evento a través del método *preventDefault*. Sin embargo, también contiene varias propiedades y métodos útiles:

- `pageX`, `pageY`: La posición del puntero del ratón en el momento que el evento ocurrió, relativo a las zonas superiores e izquierda de la página.
- `type`: El tipo de evento (por ejemplo "click").
- `which`: El botón o tecla presionada.
- `data`: Alguna información pasada cuando el evento es ejecutado.
- `target`: referencia al elemento DOM que inicializó el evento.
- `preventDefault()`: Cancela la acción predeterminada del evento (por ejemplo: seguir un enlace).

```
$('#a').click( function(evento){
    evento.preventDefault();
})
```

También tenemos acceso al objeto jQuery que disparó el evento con `$(this)`

**Ejemplo 01.html**

Cuando estamos pulsando una tecla podemos utilizar propiedades propias del evento `keypress`

`evento.keyCode`      13 corresponde al intro  
corresponde al código ASCII de la letra tecleada

`evento.key`      tecla pulsada

**Ejemplo 02.html**

- **Método on:**

**on**(events,[selector],[data],handler)

Donde

**events** son los eventos que se asocian al conjunto de elementos seleccionados. Se pueden asociar más de un evento separándolos con espacios. Por ejemplo: 'click', 'click blur'.

**handler** función a realizar después de que el evento se dispare.

El método on se puede vincular a múltiples eventos

```
$('#tr').on({'click': function() {  
    $(this).css("background-color", "gray");  
},  
    'mouseleave': function() {  
    $(this).css("background-color", "white");  
}});
```

- **Método off**

Es el opuesto al método on. Se puede **desvincular** un evento:

off([evento],[selector],[funciónOyente]).

```
$("#p").off("click");
```

- **Método one**

El controlador sólo se ejecuta la primera vez que se dispara el evento

```
$('#h2').one('click',function(){  
    alert("Una y no más");  
});
```

- **Método hover()**

permite pasar una o dos funciones que se ejecutarán cuando los eventos mouseenter y mouseleave ocurran en el elemento seleccionado. Si se pasa una sola función, está será ejecutada en ambos eventos; en cambio si se pasan dos, la primera será ejecutada cuando ocurra el evento **mouseenter**, mientras que la segunda será ejecutada cuando ocurra **mouseleave**.

```
$("#div").hover( function(){  
    $(this).html("ENTRO");  
    function() {  
    $(this).html("SALGO");  
    });
```

***Ejemplo 03.html***

Resumen de todo el pdf: ***Ejercicio 1 de la hoja de ejercicios***