

TEMA 9: jQuery

d. EVENTOS

1. Listado de eventos en jQuery

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

1. Listado de eventos en jQuery

event.preventDefault()	Evita la acción predeterminada del evento
event.relatedTarget	Devuelve el elemento entrado o salido en el movimiento del ratón.
event.result	Contiene el ultimo / anterior valor devuelto por un manejador de eventos disparado por el evento especificado
event.stopImmediatePropagation()	Evita que se llamen a otros controladores de eventos
event.stopPropagation()	Evita que el evento burbujee el árbol DOM, evitando que los manejadores de origen sean notificados del evento.
event.target	Devuelve el elemento DOM que ha desencadenado el evento.
event.timeStamp	Devuelve el número de milisegundos desde el 1 de enero de 1970, cuando se desencadena el evento
event.type	Devuelve que tipo de evento se ha activado
event.which	Devuelve la tecla del teclado o el botón del ratón para el evento
focus()	Conecta / activa el evento de enfoque
focusin()	Conecta un controlador de eventos al evento focus
focusout()	Conecta un controlador de eventos al evento de enfoque
hover()	Conecta dos manejadores de eventos al evento hover
keydown()	Conecta / activa el evento keydown
keypress()	Conecta / activa el evento de pulsaciones de teclas
keyup()	Conecta / activa el evento keyup

1. Listado de eventos en jQuery

<code>mousedown()</code>	Conecta / activa el evento mousedown
<code>mouseenter()</code>	Conecta / activa el evento mouseenter
<code>mouseleave()</code>	Adjunta / activa el evento mouseleave
<code>mousemove()</code>	Adjunta / activa el evento mousemove
<code>mouseout()</code>	Conecta / activa el evento mouseout
<code>mouseover()</code>	Conecta / activa el evento mouseover
<code>mouseup()</code>	Adjunta / desencadena el evento mouseup
<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

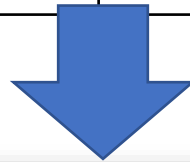
2. Crear eventos en jQuery

- La forma de crear eventos de jQuery es similar a como se hace en JavaScript con el método **addEventListener**.
- Partimos del siguiente código HTML:

```
<input type="text" name="nombre" id="nombre">  
<input type="submit" value="Enviar" id="enviar">
```

2. Crear eventos en jQuery

JavaScript	jQuery
<pre>let enviar=document.querySelector("#enviar"); enviar.addEventListener("click",function () { let nombre=document.querySelector("#nombre").value; alert("hola "+nombre); });</pre>	<pre>\$(document).ready(function () { let enviar=\$("#enviar"); \$(enviar).click(function () { let nombre=\$("#nombre").val(); alert("hola "+nombre); }); });</pre>



juan

Esta página dice
hola juan

3. Preventdefault en formularios

- Es exactamente igual que en cuando se vió en Eventos de JavaScript puro (Vanilla).
- Vamos a realizar un ejemplo con un campo de tipo **text** y si la longitud del campo es menor que 3 que no deje seguir con el envío de dicho formulario (cuidado que es un evento de formulario, no de botón).

```
<form action="" method="post" id="formulario">  
  <p><input type="text" name="nombre" id="nombre"></p>  
  <p><input type="submit" value="Enviar"></p>  
</form>
```

3. Preventdefault en formularios

JavaScript

```
let formulario = document.querySelector("#formulario");
formulario.addEventListener("submit", (portero) => {
  let nombre = document.querySelector("#nombre").value;
  if ((nombre.length) < 3) {
    portero.preventDefault();
    alert("N000000000");
  }
});
```

jQuery

```
let formulario = $("#formulario");
$(formulario).submit(function (portero) {
  let nombre = $("#nombre").val();
  if ((nombre.length) < 3) {
    portero.preventDefault();
    alert("N000000000");
  }
});
```


3. Preventdefault en formularios

