

JAVASCRIPT

Eventos

¿QUÉ ES UN EVENTO?

Los eventos son cosas que suceden sobre los elementos
HTML

JAVASCRIPT PERMITE REACCIONAR ANTES ESTOS EVENTOS

Por ejemplo si se modifica un input, o se hace click sobre un botón.

EJEMPLOS DE EVENTOS

- onclick
- onchange
- onload
- onmouseover
- onkeydown

OBTRUSION

JAVASCRIPT PUEDE AGREGARSE DE FORMA OBTRUSIVA O NO- OBTRUSIVA

O sea dentro del mismo HTML o dentro de javascript

¿POR QUÉ ES IMPORTANTE LA DIFERENCIA?

Las dos son opciones válidas, pero el javascript no obtrusivo
nos hace más fácil reutilizar nuestro código

EJEMPLO OBTRUSIVO

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title> heey </title>
  </head>
  <body>
    <p id="p1" onclick="alert('hola')"> Obtrusivo </p>
  </body>
</html>
```


EJEMPLO MENOS OBTRUSIVO

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title> heey </title>
  </head>
  <body>
    <p id="p2" onclick="handle_click()"> Menos Obtrusivo </p>

    <script type="text/javascript">
      function handle_click(){
        alert('hola');
      }
    </script>

  </body>
</html>
```

NO OBTRUSIVO

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title> heey </title>
  </head>
  <body>
    <p id="p3"> No Obtrusivo </p>
    <script type="text/javascript">
      document.getElementById("p3").addEventListener("click",
        function(){
          alert("hola");
        }
      );
    </script>
  </body>
</html>
```

EL OBJETO EVENTO

```
$( document ).on( 'click', function( event ) {  
    console.log( event.type );    // The event type, eg. "click"  
    console.log( event.which );  // The button or key that was pressed.  
    console.log( event.target ); // The originating element.  
    console.log( event.pageX );  // The document mouse X coordinate.  
    console.log( event.pageY );  // The document mouse Y coordinate.  
    event.preventDefault();      // Prevent the default action of the event (e.g. following a  
link).  
    event.stopPropagation();     // Stop the event from propagating up to other elements.  
});
```

PREVENT DEFAULT

Previene el comportamiento por defecto de una acción en el navegador

EJEMPLOS

- Click en un link
- Click derecho (menú contextual)
- Envío de un formulario

DESACTIVANDO EL MENÚ CONTEXTUAL

```
$(document).on("contextmenu", function(e){  
    e.preventDefault();  
});
```

PODRÍAMOS HACER LO MISMO PERO SÓLO PARA LAS IMÁGENES

```
$("#img").on("contextmenu", function(e){  
    e.preventDefault();  
});
```

DESACTIVANDO LOS LINKS

```
$("#a").on("click", function(e){  
    e.preventDefault();  
});
```


AGREGANDO EVENTOS DINÁMICAMENTE

CON ONCLICK

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
    <script src="https://code.jquery.com/jquery-1.12.4.js">
    </script>
  </head>
  <body>
    <button id="button" value="+"> + </button>
    <p class="par" onclick="alert('hola')"> hola </p>
  </body>
  <script type="text/javascript">
    $("#button").click(function(){
      $("body").append('<p class="par" onclick="alert(\'hola\')"> hola </p>')
    });
  </script>
</html>
```

DE FORMA NO OBTRUSIVA

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
    <script src="https://code.jquery.com/jquery-1.12.4.js">
    </script>
  </head>
  <body>
    <button id="button" value="+"> + </button>
    <div id="selectable">
      <p class="par"> hola </p>
    </div>

  </body>
  <script type="text/javascript">
    $("#button").click(function(){
      $("#selectable").append('<p class="par"> hola </p>')
    });

    $("#selectable").on("click", ".par", function(){
      alert("hola");
    })
  </script>
```

EL PROBLEMA

Los objetos HTML agregados dinámicamente no tienen los eventos asociados

EL PROBLEMA

objeto con evento

EL PROBLEMA

objeto con evento

objeto sin evento

EL PROBLEMA

objeto con evento

objeto sin evento

objeto sin evento

LA SOLUCIÓN PASA POR DELEGAR EL EVENTO

**Padre que delega
el evento**

objeto

LA SOLUCIÓN PASA POR DELEGAR EL EVENTO

**Padre que delega
el evento**

objeto

objeto

LA SOLUCIÓN PASA POR DELEGAR EL EVENTO

**Padre que delega
el evento**

objeto

objeto

objeto

SOLUCIÓN

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
    <script src="https://code.jquery.com/jquery-1.12.4.js">
    </script>
  </head>
  <body>
    <button id="button" value="+"> + </button>
    <div id="selectable">
      <p class="par"> hola </p>
    </div>

  </body>
  <script type="text/javascript">
    $("#button").click(function(){
      $("#selectable").append('<p class="par"> hola </p>')
    });

    $("#selectable").on("click", ".par", function(){
      alert("hola");
    })

  </script>
</html>

</html>
```



PROPAGACIÓN

bubbling

EL PROBLEMA

A veces el mismo evento es posible que se registre sobre múltiples elementos de forma **simultanea**

EJEMPLO

Un div que cambia de color con el click, y tiene dentro un link que permite eliminarlo



**CREEMOS UN EJEMPLO BÁSICO
DONDE PODAMOS VISUALIZAR
EL PROBLEMA**

VAMOS A OCUPAR ESTE ESTILO PARA EL EJEMPLO

```
#outer-div{  
  border: solid 1px black;  
  padding: 20px;  
  width: 400px;  
  height: 300px;  
  margin: auto;  
}  
#inner-div{  
  height: 50%;  
  width: 50%;  
  border: solid 1px red;  
  padding: 20px;  
  margin: auto;  
  text-align: center;  
}
```


BUBBLING

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
    <link rel="stylesheet" href="style.css">
    <script src="https://code.jquery.com/jquery-1.12.4.js"> </script>
  </head>
  <body>
    <div id="outer-div">
      <div id="inner-div">
        <button id="button">Click me</button>
      </div>
    </div>
    <script type="text/javascript">
      $("#outer-div").click(function(){
        alert("outer");
      })

      $("#inner-div").click(function(){
        alert("inner");
      })

      $("#button").click(function(){
        alert("button");
      })
    </script>
  </body>
</html>
```

**HAGAMOS CLICK EN EL
BOTÓN**

DETENIENDO LA PROPAGACIÓN

`event.stopPropagation`

SOLUCIÓN

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
    <link rel="stylesheet" href="style.css">
    <script src="https://code.jquery.com/jquery-1.12.4.js"> </script>
  </head>
  <body>
    <div id="outer-div">
      <div id="inner-div">
        <button id="button">Click me</button>
      </div>
    </div>
    <script type="text/javascript">
      $("#outer-div").click(function(){
        alert("outer");
      })

      $("#inner-div").click(function(){
        alert("inner");
      })

      $("#button").click(function(event){
        alert("button");
        event.stopPropagation();
      })

    </script>
  </body>
</html>
```

¿SI LOS EVENTOS SE PROPAGAN
COMO PUEDO DISTINGUIR A UNO
EN ESPECÍFICO?

¿SI LOS EVENTOS SE PROPAGAN COMO PUEDO DISTINGUIR A UNO EN ESPECÍFICO?

Con target



TARGET

Target es el elemento que disparó el evento

TARGET

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title> heey </title>
  </head>
  <body>
    <p id="par"> Target </p>
    <script type="text/javascript">
      document.getElementById("par").addEventListener("click",
        function(e){
          console.log(e.target);
        }
      );
    </script>
  </body>
</html>
```



```
$("#inner-div").click(function(e){  
    if (this == e.target){  
        console.log("caso especial");  
    }  
});
```

EL PROBLEMA DE `$(THIS)` Y LA DELEGACIÓN

```
$("body").click(".pin", function(){  
    $(this).parent().fadeOut(400)  
})
```