# Eventos

**1. ¿Qué son los manejadores de eventos?**

Los manejadores de eventos son especificaciones vinculadas a elementos XHTML que permiten ejecutar un fragmento de código del script cuando ocurren las condiciones especificadas en el mismo.

**2. Define Manejadores como atributos XHTML e indica un ejemplo de codigo**

Los manejadores de eventos como atributos XHTML se trata de atributos especiales que actúan como llamada a una función dentro de un script o, directamente, un código escrito dentro del valor de la propia etiqueta. Un ejemplo sería un evento que sirviese para resaltar un fragmento del texto cambiando el color al pasar el cursor por encima.

<span onmouseover=”changeColor()”>esto es un texto html</span>

**3. Define Manejadores de eventos con la variable this e indica un ejemplo de**

**código**

La variable this, en el contexto de un evento, hace referencia **siempre** a la información de la propia etiqueta que lo ha llamado, que al ser llamada en un evento es definida como un objeto cuyos atributos son los atributos pertenecientes a la etiqueta XHTML. Por ejemplo, si con un evento llamamos a una etiqueta select, podríamos intentar acceder al valor seleccionado de la siguiente forma.

this.value[‘selected\_value’], siendo que los valores del select forman un array de posibilidades (cada una de las posiciones equivale a un option)

**4. Define Manejadores de eventos como funciones externas e indica un ejemplo**

**de código**

Los manejadores de eventos mediante el uso de funciones externas son aquellos que, al ser llamados mediante el atributo XHTML pertinente, hacen una llamada a una función definida en un script. Un ejemplo sería el de añadir una función que se active al pulsar un botón.

<button onclick=”funcionEjemplo()”>¡Soy un botón!</button>

**5. Define Manejadores de eventos semánticos e indica un ejemplo de código.**

Un manejador de eventos con funciones externas funciona con el uso de funciones creadas en un script vinculado al documento de forma externa, en la que se toma el elemento al que vincularle el evento y se le añade una función de escucha a la llamada.

Con esto nos referimos a addEventListener, que ayuda enormemente a mantener despejado el documento XHTML, lo cual es realmente importante, teniendo en cuenta la cantidad de código php o NodeJS que presentará en una página real.

El mismo evento de ejemplo de antes, en forma de comparador léxico.

document.getItemById(‘botonEjemplo’).addEventListener(“mousedown”, (e)=>{funcionEjemplo});

**6. ¿Qué es el Event bubbling?**

El bubbling de eventos es, cuando al activar a un evento ubicado en una etiqueta hija de una etiqueta padre que a su vez también tiene un evento, este es activado por efecto de la herencia inversa. Muy rara vez esto es un efecto deseado, normalmente se trata de reacciones en cadena no deseadas. Esto puede ser prevenido teniendo mucho cuidado o, en caso de que sea inevitable, especificando que el evento NO se propague usando la propiedad del objeto evento .stopPropagation()

**7. ¿Qué es el Event capturing?**

El event capturing es exactamente lo mismo que el event bubbling, con la diferencia de que la propagación del evento no se da entre hijos anidados por herencia inversa, si no por herencia común en cascada de elemento padre a elemento hijo.

**8. ¿Que es el Event Dom?**

Un evento DOM es un evento que se crea y llama mediante las librerías DOM, osease, es lo mismo que un manejador de evento semántico.

**9. Define el objeto event, sus propiedades y metodos**

El objeto event es el objeto que se crea para recoger toda la información referente al evento. Está estrechamente relacionado con “this”, sólo que como ya se dijo, “this” hace referencia al elemento que llama al evento y el objeto event (normalmente llamado “e”) hace referencia únicamente a la información del evento, siendo usado normalmente como parámetro de la función que se activa con el call en caso de ser necesario. Aquí dejo una lista de los métodos del objeto evento, sacados de la librería de DOM.

bubbles Returns whether or not a specific event is a bubbling event

cancelBubble Sets or returns whether the event should propagate up the hierarchy or not

cancelable Returns whether or not an event can have its default action prevented

composed Returns whether the event is composed or not

createEvent() Creates a new event

composedPath() Returns the event's path

currentTarget Returns the element whose event listeners triggered the event

defaultPrevented Returns whether or not the preventDefault() method was called for the event

eventPhase Returns which phase of the event flow is currently being evaluated

isTrusted Returns whether or not an event is trusted

preventDefault() Cancels the event if it is cancelable, meaning that the default action that belongs to the event will not occur

stopImmediatePropagation() Prevents other listeners of the same event from being called

stopPropagation() Prevents further propagation of an event during event flow

target Returns the element that triggered the event

timeStamp Returns the time (in milliseconds relative to the epoch) at which the event was created

type Returns the name of the event

Event Types

These event types belongs to the Event Object:

Event Description

abort The event occurs when the loading of a media is aborted

afterprint The event occurs when a page has started printing

beforeprint The event occurs when a page is about to be printed

beforeunload The event occurs before the document is about to be unloaded

canplay The event occurs when the browser can start playing the media (when it has buffered enough to begin)

canplaythrough The event occurs when the browser can play through the media without stopping for buffering

change The event occurs when the content of a form element, the selection, or the checked state have changed (for <input>, <select>, and <textarea>)

error The event occurs when an error occurs while loading an external file

fullscreenchange The event occurs when an element is displayed in fullscreen mode

fullscreenerror The event occurs when an element can not be displayed in fullscreen mode

input The event occurs when an element gets user input

invalid The event occurs when an element is invalid

load The event occurs when an object has loaded

loadeddata The event occurs when media data is loaded

loadedmetadata The event occurs when meta data (like dimensions and duration) are loaded

message The event occurs when a message is received through the event source

offline The event occurs when the browser starts to work offline

online The event occurs when the browser starts to work online

open The event occurs when a connection with the event source is opened

pause The event occurs when the media is paused either by the user or programmatically

play The event occurs when the media has been started or is no longer paused

playing The event occurs when the media is playing after having been paused or stopped for buffering

progress The event occurs when the browser is in the process of getting the media data (downloading the media)

ratechange The event occurs when the playing speed of the media is changed

resize The event occurs when the document view is resized

reset The event occurs when a form is reset

scroll The event occurs when an element's scrollbar is being scrolled

search The event occurs when the user writes something in a search field (for <input="search">)

seeked The event occurs when the user is finished moving/skipping to a new position in the media

seeking The event occurs when the user starts moving/skipping to a new position in the media

select The event occurs after the user selects some text (for <input> and <textarea>)

show The event occurs when a <menu> element is shown as a context menu

stalled The event occurs when the browser is trying to get media data, but data is not available

submit The event occurs when a form is submitted

suspend The event occurs when the browser is intentionally not getting media data

timeupdate The event occurs when the playing position has changed (like when the user fast forwards to a different point in the media)

toggle The event occurs when the user opens or closes the <details> element

unload The event occurs once a page has unloaded (for <body>)

waiting The event occurs when the media has paused but is expected to resume (like when the media pauses to buffer more data)

**10. ¿Qué tipos de eventos existen segun el DOM? indica cada uno de los grupos**

**existentes y ejemplos de cada grupo.**

Lista de tipo de eventos DOM según W3School

| [abort](https://www.w3schools.com/jsref/event_onabort_media.asp) | The event occurs when the loading of a media is aborted | [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp), [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| --- | --- | --- |
| [afterprint](https://www.w3schools.com/jsref/event_onafterprint.asp) | The event occurs when a page has started printing, or if the print dialogue box has been closed | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [animationend](https://www.w3schools.com/jsref/event_animationend.asp) | The event occurs when a CSS animation has completed | [AnimationEvent](https://www.w3schools.com/jsref/obj_animationevent.asp) |
| [animationiteration](https://www.w3schools.com/jsref/event_animationiteration.asp) | The event occurs when a CSS animation is repeated | [AnimationEvent](https://www.w3schools.com/jsref/obj_animationevent.asp) |
| [animationstart](https://www.w3schools.com/jsref/event_animationstart.asp) | The event occurs when a CSS animation has started | [AnimationEvent](https://www.w3schools.com/jsref/obj_animationevent.asp) |
| [beforeprint](https://www.w3schools.com/jsref/event_onbeforeprint.asp) | The event occurs when a page is about to be printed | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [beforeunload](https://www.w3schools.com/jsref/event_onbeforeunload.asp) | The event occurs before the document is about to be unloaded | [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp), [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [blur](https://www.w3schools.com/jsref/event_onblur.asp) | The event occurs when an element loses focus | [FocusEvent](https://www.w3schools.com/jsref/obj_focusevent.asp) |
| [canplay](https://www.w3schools.com/jsref/event_oncanplay.asp) | The event occurs when the browser can start playing the media (when it has buffered enough to begin) | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [canplaythrough](https://www.w3schools.com/jsref/event_oncanplaythrough.asp) | The event occurs when the browser can play through the media without stopping for buffering | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [change](https://www.w3schools.com/jsref/event_onchange.asp) | The event occurs when the content of a form element, the selection, or the checked state have changed (for <input>, <select>, and <textarea>) | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [click](https://www.w3schools.com/jsref/event_onclick.asp) | The event occurs when the user clicks on an element | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [contextmenu](https://www.w3schools.com/jsref/event_oncontextmenu.asp) | The event occurs when the user right-clicks on an element to open a context menu | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [copy](https://www.w3schools.com/jsref/event_oncopy.asp) | The event occurs when the user copies the content of an element | [ClipboardEvent](https://www.w3schools.com/jsref/obj_clipboardevent.asp) |
| [cut](https://www.w3schools.com/jsref/event_oncut.asp) | The event occurs when the user cuts the content of an element | [ClipboardEvent](https://www.w3schools.com/jsref/obj_clipboardevent.asp) |
| [dblclick](https://www.w3schools.com/jsref/event_ondblclick.asp) | The event occurs when the user double-clicks on an element | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [drag](https://www.w3schools.com/jsref/event_ondrag.asp) | The event occurs when an element is being dragged | [DragEvent](https://www.w3schools.com/jsref/obj_dragevent.asp) |
| [dragend](https://www.w3schools.com/jsref/event_ondragend.asp) | The event occurs when the user has finished dragging an element | [DragEvent](https://www.w3schools.com/jsref/obj_dragevent.asp) |
| [dragenter](https://www.w3schools.com/jsref/event_ondragenter.asp) | The event occurs when the dragged element enters the drop target | [DragEvent](https://www.w3schools.com/jsref/obj_dragevent.asp) |
| [dragleave](https://www.w3schools.com/jsref/event_ondragleave.asp) | The event occurs when the dragged element leaves the drop target | [DragEvent](https://www.w3schools.com/jsref/obj_dragevent.asp) |
| [dragover](https://www.w3schools.com/jsref/event_ondragover.asp) | The event occurs when the dragged element is over the drop target | [DragEvent](https://www.w3schools.com/jsref/obj_dragevent.asp) |
| [dragstart](https://www.w3schools.com/jsref/event_ondragstart.asp) | The event occurs when the user starts to drag an element | [DragEvent](https://www.w3schools.com/jsref/obj_dragevent.asp) |
| [drop](https://www.w3schools.com/jsref/event_ondrop.asp) | The event occurs when the dragged element is dropped on the drop target | [DragEvent](https://www.w3schools.com/jsref/obj_dragevent.asp) |
| [durationchange](https://www.w3schools.com/jsref/event_ondurationchange.asp) | The event occurs when the duration of the media is changed | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [ended](https://www.w3schools.com/jsref/event_onended.asp) | The event occurs when the media has reach the end (useful for messages like "thanks for listening") | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [error](https://www.w3schools.com/jsref/event_onerror.asp) | The event occurs when an error occurs while loading an external file | [ProgressEvent](https://www.w3schools.com/jsref/obj_progressevent.asp), [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp), [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [focus](https://www.w3schools.com/jsref/event_onfocus.asp) | The event occurs when an element gets focus | [FocusEvent](https://www.w3schools.com/jsref/obj_focusevent.asp) |
| [focusin](https://www.w3schools.com/jsref/event_onfocusin.asp) | The event occurs when an element is about to get focus | [FocusEvent](https://www.w3schools.com/jsref/obj_focusevent.asp) |
| [focusout](https://www.w3schools.com/jsref/event_onfocusout.asp) | The event occurs when an element is about to lose focus | [FocusEvent](https://www.w3schools.com/jsref/obj_focusevent.asp) |
| [fullscreenchange](https://www.w3schools.com/jsref/event_fullscreenchange.asp) | The event occurs when an element is displayed in fullscreen mode | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [fullscreenerror](https://www.w3schools.com/jsref/event_fullscreenerror.asp) | The event occurs when an element can not be displayed in fullscreen mode | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [hashchange](https://www.w3schools.com/jsref/event_onhashchange.asp) | The event occurs when there has been changes to the anchor part of a URL | [HashChangeEvent](https://www.w3schools.com/jsref/obj_hashchangeevent.asp) |
| [input](https://www.w3schools.com/jsref/event_oninput.asp) | The event occurs when an element gets user input | [InputEvent](https://www.w3schools.com/jsref/obj_inputevent.asp), [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [invalid](https://www.w3schools.com/jsref/event_oninvalid.asp) | The event occurs when an element is invalid | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [keydown](https://www.w3schools.com/jsref/event_onkeydown.asp) | The event occurs when the user is pressing a key | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| [keypress](https://www.w3schools.com/jsref/event_onkeypress.asp) | The event occurs when the user presses a key | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| [keyup](https://www.w3schools.com/jsref/event_onkeyup.asp) | The event occurs when the user releases a key | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| [load](https://www.w3schools.com/jsref/event_onload.asp) | The event occurs when an object has loaded | [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp), [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [loadeddata](https://www.w3schools.com/jsref/event_onloadeddata.asp) | The event occurs when media data is loaded | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [loadedmetadata](https://www.w3schools.com/jsref/event_onloadedmetadata.asp) | The event occurs when meta data (like dimensions and duration) are loaded | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [loadstart](https://www.w3schools.com/jsref/event_onloadstart.asp) | The event occurs when the browser starts looking for the specified media | [ProgressEvent](https://www.w3schools.com/jsref/obj_progressevent.asp) |
| [message](https://www.w3schools.com/jsref/event_onmessage_sse.asp) | The event occurs when a message is received through the event source | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [mousedown](https://www.w3schools.com/jsref/event_onmousedown.asp) | The event occurs when the user presses a mouse button over an element | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [mouseenter](https://www.w3schools.com/jsref/event_onmouseenter.asp) | The event occurs when the pointer is moved onto an element | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [mouseleave](https://www.w3schools.com/jsref/event_onmouseleave.asp) | The event occurs when the pointer is moved out of an element | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [mousemove](https://www.w3schools.com/jsref/event_onmousemove.asp) | The event occurs when the pointer is moving while it is over an element | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [mouseover](https://www.w3schools.com/jsref/event_onmouseover.asp) | The event occurs when the pointer is moved onto an element, or onto one of its children | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [mouseout](https://www.w3schools.com/jsref/event_onmouseout.asp) | The event occurs when a user moves the mouse pointer out of an element, or out of one of its children | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [mouseup](https://www.w3schools.com/jsref/event_onmouseup.asp) | The event occurs when a user releases a mouse button over an element | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| mousewheel | Deprecated. Use the [wheel](https://www.w3schools.com/jsref/event_onwheel.asp) event instead | [WheelEvent](https://www.w3schools.com/jsref/obj_wheelevent.asp) |
| [offline](https://www.w3schools.com/jsref/event_onoffline.asp) | The event occurs when the browser starts to work offline | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [online](https://www.w3schools.com/jsref/event_ononline.asp) | The event occurs when the browser starts to work online | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [open](https://www.w3schools.com/jsref/event_onopen_sse.asp) | The event occurs when a connection with the event source is opened | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [pagehide](https://www.w3schools.com/jsref/event_onpagehide.asp) | The event occurs when the user navigates away from a webpage | [PageTransitionEvent](https://www.w3schools.com/jsref/obj_pagetransitionevent.asp) |
| [pageshow](https://www.w3schools.com/jsref/event_onpageshow.asp) | The event occurs when the user navigates to a webpage | [PageTransitionEvent](https://www.w3schools.com/jsref/obj_pagetransitionevent.asp) |
| [paste](https://www.w3schools.com/jsref/event_onpaste.asp) | The event occurs when the user pastes some content in an element | [ClipboardEvent](https://www.w3schools.com/jsref/obj_clipboardevent.asp) |
| [pause](https://www.w3schools.com/jsref/event_onpause.asp) | The event occurs when the media is paused either by the user or programmatically | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [play](https://www.w3schools.com/jsref/event_onplay.asp) | The event occurs when the media has been started or is no longer paused | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [playing](https://www.w3schools.com/jsref/event_onplaying.asp) | The event occurs when the media is playing after having been paused or stopped for buffering | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| popstate | The event occurs when the window's history changes | [PopStateEvent](https://www.w3schools.com/jsref/obj_popstateevent.asp) |
| [progress](https://www.w3schools.com/jsref/event_onprogress.asp) | The event occurs when the browser is in the process of getting the media data (downloading the media) | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [ratechange](https://www.w3schools.com/jsref/event_onratechange.asp) | The event occurs when the playing speed of the media is changed | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [resize](https://www.w3schools.com/jsref/event_onresize.asp) | The event occurs when the document view is resized | [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp), [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [reset](https://www.w3schools.com/jsref/event_onreset.asp) | The event occurs when a form is reset | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [scroll](https://www.w3schools.com/jsref/event_onscroll.asp) | The event occurs when an element's scrollbar is being scrolled | [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp), [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [search](https://www.w3schools.com/jsref/event_onsearch.asp) | The event occurs when the user writes something in a search field (for <input="search">) | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [seeked](https://www.w3schools.com/jsref/event_onseeked.asp) | The event occurs when the user is finished moving/skipping to a new position in the media | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [seeking](https://www.w3schools.com/jsref/event_onseeking.asp) | The event occurs when the user starts moving/skipping to a new position in the media | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [select](https://www.w3schools.com/jsref/event_onselect.asp) | The event occurs after the user selects some text (for <input> and <textarea>) | [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp), [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [show](https://www.w3schools.com/jsref/event_onshow.asp) | The event occurs when a <menu> element is shown as a context menu | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [stalled](https://www.w3schools.com/jsref/event_onstalled.asp) | The event occurs when the browser is trying to get media data, but data is not available | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| storage | The event occurs when a Web Storage area is updated | [StorageEvent](https://www.w3schools.com/jsref/obj_storageevent.asp) |
| [submit](https://www.w3schools.com/jsref/event_onsubmit.asp) | The event occurs when a form is submitted | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [suspend](https://www.w3schools.com/jsref/event_onsuspend.asp) | The event occurs when the browser is intentionally not getting media data | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [timeupdate](https://www.w3schools.com/jsref/event_ontimeupdate.asp) | The event occurs when the playing position has changed (like when the user fast forwards to a different point in the media) | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [toggle](https://www.w3schools.com/jsref/event_ontoggle.asp) | The event occurs when the user opens or closes the <details> element | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [touchcancel](https://www.w3schools.com/jsref/event_touchcancel.asp) | The event occurs when the touch is interrupted | [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [touchend](https://www.w3schools.com/jsref/event_touchend.asp) | The event occurs when a finger is removed from a touch screen | [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [touchmove](https://www.w3schools.com/jsref/event_touchmove.asp) | The event occurs when a finger is dragged across the screen | [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [touchstart](https://www.w3schools.com/jsref/event_touchstart.asp) | The event occurs when a finger is placed on a touch screen | [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [transitionend](https://www.w3schools.com/jsref/event_transitionend.asp) | The event occurs when a CSS transition has completed | [TransitionEvent](https://www.w3schools.com/jsref/obj_transitionevent.asp) |
| [unload](https://www.w3schools.com/jsref/event_onunload.asp) | The event occurs once a page has unloaded (for <body>) | [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp), [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [volumechange](https://www.w3schools.com/jsref/event_onvolumechange.asp) | The event occurs when the volume of the media has changed (includes setting the volume to "mute") | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [waiting](https://www.w3schools.com/jsref/event_onwaiting.asp) | The event occurs when the media has paused but is expected to resume (like when the media pauses to buffer more data) | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [wheel](https://www.w3schools.com/jsref/event_onwheel.asp) | The event occurs when the mouse wheel rolls up or down over an element | [WheelEvent](https://www.w3schools.com/jsref/obj_wheelevent.asp) |

## HTML DOM Event Properties and Methods

| **Property/Method** | **Description** | | **Belongs To** |
| --- | --- | --- | --- |
| [altKey](https://www.w3schools.com/jsref/event_altkey.asp) | Returns whether the "ALT" key was pressed when the mouse event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [altKey](https://www.w3schools.com/jsref/event_key_altkey.asp) | Returns whether the "ALT" key was pressed when the key event was triggered | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp), [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [animationName](https://www.w3schools.com/jsref/event_animation_animationName.asp) | Returns the name of the animation | | [AnimationEvent](https://www.w3schools.com/jsref/obj_animationevent.asp) |
| [bubbles](https://www.w3schools.com/jsref/event_bubbles.asp) | Returns whether or not a specific event is a bubbling event | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [button](https://www.w3schools.com/jsref/event_button.asp) | Returns which mouse button was pressed when the mouse event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [buttons](https://www.w3schools.com/jsref/event_buttons.asp) | Returns which mouse buttons were pressed when the mouse event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [cancelable](https://www.w3schools.com/jsref/event_cancelable.asp) | Returns whether or not an event can have its default action prevented | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [charCode](https://www.w3schools.com/jsref/event_key_charcode.asp) | Returns the Unicode character code of the key that triggered the onkeypress event | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| changeTouches | Returns a list of all the touch objects whose state changed between the previous touch and this touch | | [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [clientX](https://www.w3schools.com/jsref/event_clientx.asp) | Returns the horizontal coordinate of the mouse pointer, relative to the current window, when the mouse event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp), [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [clientY](https://www.w3schools.com/jsref/event_clienty.asp) | Returns the vertical coordinate of the mouse pointer, relative to the current window, when the mouse event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp), [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| clipboardData | Returns an object containing the data affected by the clipboard operation | | [ClipboardData](https://www.w3schools.com/jsref/obj_clipboardevent.asp) |
| [code](https://www.w3schools.com/jsref/event_key_code.asp) | Returns the code of the key that triggered the event | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| composed | Returns whether the event is composed or not | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [ctrlKey](https://www.w3schools.com/jsref/event_ctrlkey.asp) | Returns whether the "CTRL" key was pressed when the mouse event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [ctrlKey](https://www.w3schools.com/jsref/event_key_ctrlkey.asp) | Returns whether the "CTRL" key was pressed when the key event was triggered | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp), [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [currentTarget](https://www.w3schools.com/jsref/event_currenttarget.asp) | Returns the element whose event listeners triggered the event | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [data](https://www.w3schools.com/jsref/event_inputevent_data.asp) | Returns the inserted characters | | [InputEvent](https://www.w3schools.com/jsref/obj_inputevent.asp) |
| dataTransfer | Returns an object containing the data being dragged/dropped, or inserted/deleted | | [DragEvent](https://www.w3schools.com/jsref/obj_dragevent.asp), [InputEvent](https://www.w3schools.com/jsref/obj_inputevent.asp) |
| [defaultPrevented](https://www.w3schools.com/jsref/event_defaultprevented.asp) | Returns whether or not the preventDefault() method was called for the event | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [deltaX](https://www.w3schools.com/jsref/event_wheel_deltax.asp) | Returns the horizontal scroll amount of a mouse wheel (x-axis) | | [WheelEvent](https://www.w3schools.com/jsref/obj_wheelevent.asp) |
| [deltaY](https://www.w3schools.com/jsref/event_wheel_deltay.asp) | Returns the vertical scroll amount of a mouse wheel (y-axis) | | [WheelEvent](https://www.w3schools.com/jsref/obj_wheelevent.asp) |
| [deltaZ](https://www.w3schools.com/jsref/event_wheel_deltaz.asp) | Returns the scroll amount of a mouse wheel for the z-axis | | [WheelEvent](https://www.w3schools.com/jsref/obj_wheelevent.asp) |
| [deltaMode](https://www.w3schools.com/jsref/event_wheel_deltamode.asp) | Returns a number that represents the unit of measurements for delta values (pixels, lines or pages) | | [WheelEvent](https://www.w3schools.com/jsref/obj_wheelevent.asp) |
| [detail](https://www.w3schools.com/jsref/event_detail.asp) | Returns a number that indicates how many times the mouse was clicked | | [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp) |
| [elapsedTime](https://www.w3schools.com/jsref/event_animation_elapsedtime.asp) | Returns the number of seconds an animation has been running | | [AnimationEvent](https://www.w3schools.com/jsref/obj_animationevent.asp) |
| [elapsedTime](https://www.w3schools.com/jsref/event_transition_elapsedtime.asp) | Returns the number of seconds a transition has been running | |  |
| [eventPhase](https://www.w3schools.com/jsref/event_eventphase.asp) | Returns which phase of the event flow is currently being evaluated | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| getTargetRanges() | Returns an array containing target ranges that will be affected by the insertion/deletion | | [InputEvent](https://www.w3schools.com/jsref/obj_inputevent.asp) |
| [getModifierState()](https://www.w3schools.com/jsref/event_mouse_getmodifierstate.asp) | Returns an array containing target ranges that will be affected by the insertion/deletion | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [inputType](https://www.w3schools.com/jsref/event_inputevent_inputtype.asp) | Returns the type of the change (i.e "inserting" or "deleting") | | [InputEvent](https://www.w3schools.com/jsref/obj_inputevent.asp) |
| isComposing | Returns whether the state of the event is composing or not | | [InputEvent](https://www.w3schools.com/jsref/obj_inputevent.asp), [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| [isTrusted](https://www.w3schools.com/jsref/event_istrusted.asp) | Returns whether or not an event is trusted | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [key](https://www.w3schools.com/jsref/event_key_key.asp) | Returns the key value of the key represented by the event | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| key | Returns the key of the changed storage item | | [StorageEvent](https://www.w3schools.com/jsref/obj_storageevent.asp) |
| [keyCode](https://www.w3schools.com/jsref/event_key_keycode.asp) | Returns the Unicode character code of the key that triggered the onkeypress event, or the Unicode key code of the key that triggered the onkeydown or onkeyup event | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| [location](https://www.w3schools.com/jsref/event_key_location.asp) | Returns the location of a key on the keyboard or device | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| lengthComputable | Returns whether the length of the progress can be computable or not | | [ProgressEvent](https://www.w3schools.com/jsref/obj_progressevent.asp) |
| loaded | Returns how much work has been loaded | | [ProgressEvent](https://www.w3schools.com/jsref/obj_progressevent.asp) |
| [metaKey](https://www.w3schools.com/jsref/event_metakey.asp) | Returns whether the "META" key was pressed when an event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [metaKey](https://www.w3schools.com/jsref/event_key_metakey.asp) | Returns whether the "meta" key was pressed when the key event was triggered | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp), [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| MovementX | Returns the horizontal coordinate of the mouse pointer relative to the position of the last mousemove event | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| MovementY | Returns the vertical coordinate of the mouse pointer relative to the position of the last mousemove event | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| newValue | Returns the new value of the changed storage item | | [StorageEvent](https://www.w3schools.com/jsref/obj_storageevent.asp) |
| [newURL](https://www.w3schools.com/jsref/event_hashchange_newurl.asp) | Returns the URL of the document, after the hash has been changed | | [HasChangeEvent](https://www.w3schools.com/jsref/obj_hashchangeevent.asp) |
| offsetX | Returns the horizontal coordinate of the mouse pointer relative to the position of the edge of the target element | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| offsetY | Returns the vertical coordinate of the mouse pointer relative to the position of the edge of the target element | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| oldValue | Returns the old value of the changed storage item | | [StorageEvent](https://www.w3schools.com/jsref/obj_storageevent.asp) |
| [oldURL](https://www.w3schools.com/jsref/event_hashchange_oldurl.asp) | Returns the URL of the document, before the hash was changed | | [HasChangeEvent](https://www.w3schools.com/jsref/obj_hashchangeevent.asp) |
| onemptied | The event occurs when something bad happens and the media file is suddenly unavailable (like unexpectedly disconnects) | |  |
| [pageX](https://www.w3schools.com/jsref/event_pagex.asp) | Returns the horizontal coordinate of the mouse pointer, relative to the document, when the mouse event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [pageY](https://www.w3schools.com/jsref/event_pagey.asp) | Returns the vertical coordinate of the mouse pointer, relative to the document, when the mouse event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [persisted](https://www.w3schools.com/jsref/event_pagetransition_persisted.asp) | Returns whether the webpage was cached by the browser | | [PageTransitionEvent](https://www.w3schools.com/jsref/obj_pagetransitionevent.asp) |
| [preventDefault()](https://www.w3schools.com/jsref/event_preventdefault.asp) | Cancels the event if it is cancelable, meaning that the default action that belongs to the event will not occur | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [propertyName](https://www.w3schools.com/jsref/event_transition_propertyName.asp) | Returns the name of the CSS property associated with the animation or transition | | [AnimationEvent](https://www.w3schools.com/jsref/obj_animationevent.asp), [TransitionEvent](https://www.w3schools.com/jsref/obj_transitionevent.asp) |
| pseudoElement | Returns the name of the pseudo-element of the animation or transition | | [AnimationEvent](https://www.w3schools.com/jsref/obj_animationevent.asp), [TransitionEvent](https://www.w3schools.com/jsref/obj_transitionevent.asp) |
| region |  | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [relatedTarget](https://www.w3schools.com/jsref/event_relatedtarget.asp) | Returns the element related to the element that triggered the mouse event | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [relatedTarget](https://www.w3schools.com/jsref/event_focus_relatedtarget.asp) | Returns the element related to the element that triggered the event | | [FocusEvent](https://www.w3schools.com/jsref/obj_focusevent.asp) |
| repeat | Returns whether a key is being hold down repeatedly, or not | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| [screenX](https://www.w3schools.com/jsref/event_screenx.asp) | Returns the horizontal coordinate of the mouse pointer, relative to the screen, when an event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [screenY](https://www.w3schools.com/jsref/event_screeny.asp) | Returns the vertical coordinate of the mouse pointer, relative to the screen, when an event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [shiftKey](https://www.w3schools.com/jsref/event_shiftkey.asp) | Returns whether the "SHIFT" key was pressed when an event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [shiftKey](https://www.w3schools.com/jsref/event_key_shiftkey.asp) | Returns whether the "SHIFT" key was pressed when the key event was triggered | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp), [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| state | Returns an object containing a copy of the history entries | | [PopStateEvent](https://www.w3schools.com/jsref/obj_popstateevent.asp) |
| [stopImmediatePropagation()](https://www.w3schools.com/jsref/event_stopimmediatepropagation.asp) | | Prevents other listeners of the same event from being called | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [stopPropagation()](https://www.w3schools.com/jsref/event_stoppropagation.asp) | Prevents further propagation of an event during event flow | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| storageArea | Returns an object representing the affected storage object | | [StorageEvent](https://www.w3schools.com/jsref/obj_storageevent.asp) |
| [target](https://www.w3schools.com/jsref/event_target.asp) | Returns the element that triggered the event | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| [targetTouches](https://www.w3schools.com/jsref/event_touch_targettouches.asp) | Returns a list of all the touch objects that are in contact with the surface and where the touchstart event occured on the same target element as the current target element | | [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [timeStamp](https://www.w3schools.com/jsref/event_timestamp.asp) | Returns the time (in milliseconds relative to the epoch) at which the event was created | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| total | Returns the total amount of work that will be loaded | | [ProgressEvent](https://www.w3schools.com/jsref/obj_progressevent.asp) |
| [touches](https://www.w3schools.com/jsref/event_touch_touches.asp) | Returns a list of all the touch objects that are currently in contact with the surface | | [TouchEvent](https://www.w3schools.com/jsref/obj_touchevent.asp) |
| [transitionend](https://www.w3schools.com/jsref/event_transitionend.asp) | The event occurs when a CSS transition has completed | | [TransitionEvent](https://www.w3schools.com/jsref/obj_transitionevent.asp) |
| [type](https://www.w3schools.com/jsref/event_type.asp) | Returns the name of the event | | [Event](https://www.w3schools.com/jsref/obj_event.asp) |
| url | Returns the URL of the changed item's document | | [StorageEvent](https://www.w3schools.com/jsref/obj_storageevent.asp) |
| [which](https://www.w3schools.com/jsref/event_which.asp) | Returns which mouse button was pressed when the mouse event was triggered | | [MouseEvent](https://www.w3schools.com/jsref/obj_mouseevent.asp) |
| [which](https://www.w3schools.com/jsref/event_key_which.asp) | Returns the Unicode character code of the key that triggered the onkeypress event, or the Unicode key code of the key that triggered the onkeydown or onkeyup event | | [KeyboardEvent](https://www.w3schools.com/jsref/obj_keyboardevent.asp) |
| [view](https://www.w3schools.com/jsref/event_view.asp) | Returns a reference to the Window object where the event occurred | | [UiEvent](https://www.w3schools.com/jsref/obj_uievent.asp) |