

DOM

JavaScript

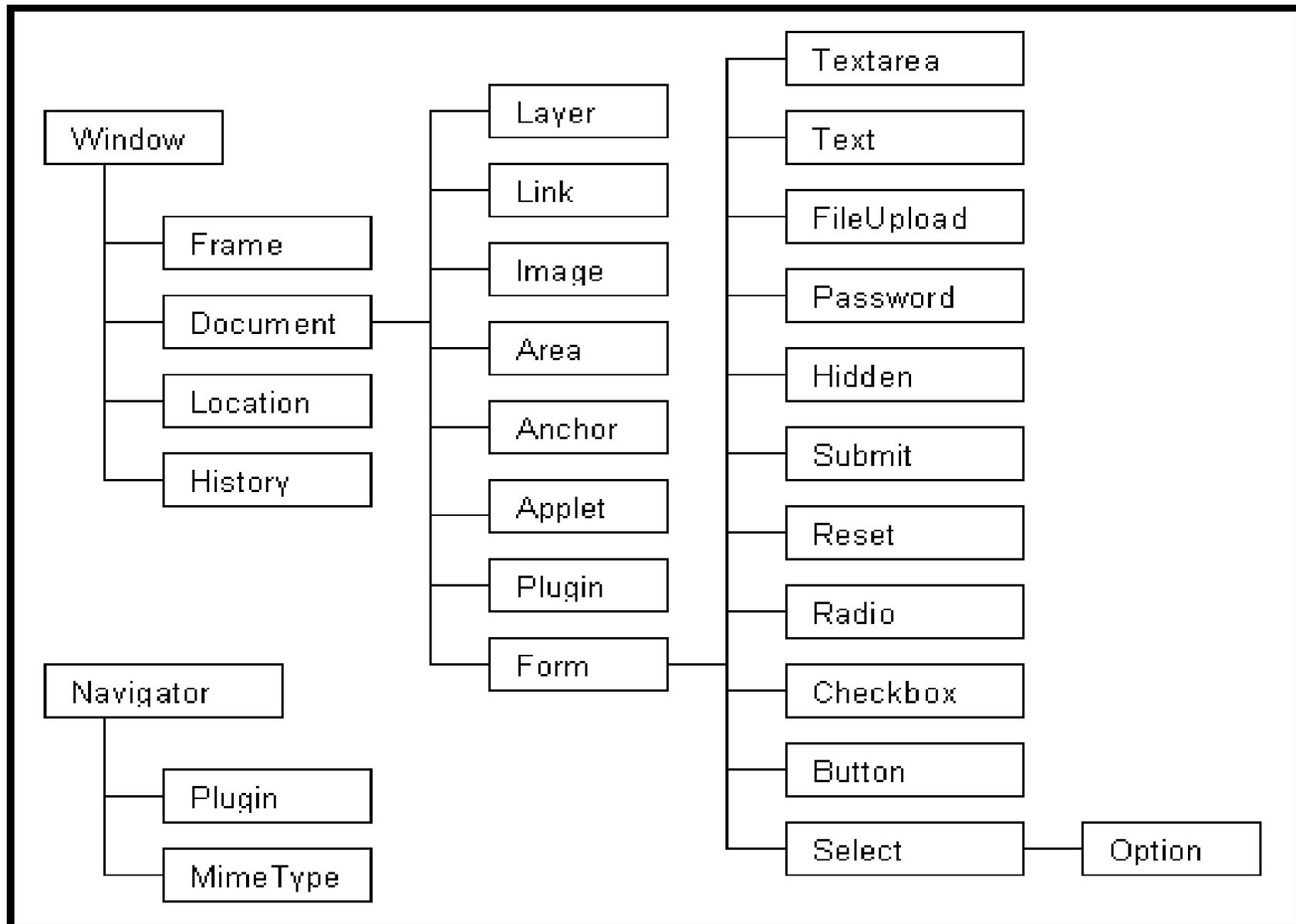
Jerarquía de Objetos del Navegador

- **Son los objetos disponibles en Javascript para controlar cualquier elemento en una página web.**
- **Javascript trabaja con esta jerarquía de objetos.**
- **Con ella se puede controlar cualquier elemento de la página.**
- **Es crucial conocerla bien para poder controlar perfectamente las páginas web.**

JavaScript - Jerarquía de Objetos

- El navegador Presenta una jerarquía en una relación “contenedor – contenido”
- En JavaScript podemos considerar a los objetos como colecciones de:
 - Propiedades: Características que lo definen
 - Métodos: implementan las cosas que se pueden hacer.
- Admite tres tipos de objetos:
 - Los objetos del navegador: **window, frame, location, history y navigator.**
 - Los objetos predefinidos: **Array, Boolean, Date, Function, Math, Number, Object y String.**
 - Los objetos creados por el usuario

JavaScript - Jerarquía de Objetos



JavaScript - Jerarquía de Objetos

El Objeto Windows

- **Es el objeto principal en la jerarquía**
- **contiene las propiedades y métodos para controlar la ventana del navegador**
- **Hace referencia a la ventana actual.**
- **De él dependen todos los demás objetos de la jerarquía**

JavaScript - Jerarquía de Objetos

El Objeto Windows

Propiedades

closed	location	self
defaultStatus	locationbar	scrollbars
document	menubar	status
Frame	name	statusbar
frames array	opener	toolbar
history	outerHeight	top
innerHeight	outerWidth	Window
innerWidth	parent	
length	personalbar	

JavaScript - Jerarquía de Objetos

El Objeto Windows – Propiedades – I

- **closed:** Es un booleano. Indica si la ventana está cerrada (true) o no (false)
- **defaultStatus:** Texto que se escribe por defecto en la barra de estado del navegador.
- **document:** Objeto. Contiene la página web que se está mostrando.
- **Frame:** Un objeto frame de una página web. Se accede por su nombre.
- **frames array:** Vector que contiene todos los frames de la página. Se accede por su índice a partir de 0. Su orden se asigna según se definen en el documento HTML.
- **history:** Objeto. Historial de páginas visitadas.

JavaScript - Jerarquía de Objetos

El Objeto Windows – Propiedades – II

- **innerHeight**: Tamaño en pixels del espacio donde se visualiza la página, en vertical.
- **innerWidth**: Tamaño en pixels del espacio donde se visualiza la página, en horizontal.
- **length**: Numero de frames de la ventana.
- **location**: La URL del documento que se está visualizando. Se puede cambiar su valor para ir a otra página. Ver también la propiedad location del objeto **document**
- **locationbar**: Objeto. Barra de direcciones de la ventana.
- **menubar**: Objeto. Barra de menús de la ventana.
- **name**: Nombre de la ventana o del frame actual. Se asigna cuando se abre una nueva ventana.

JavaScript - Jerarquía de Objetos

El Objeto Windows – Propiedades – III

- **opener:** Hace referencia a la ventana del navegador que abrió la ventana activa.
- **outerHeight:** Tamaño en pixels del espacio de toda la ventana, en vertical. Incluye las barras de desplazamiento, botones, etc.
- **outerWidth:** Tamaño en pixels del espacio de toda la ventana, en horizontal. Incluye las barras de desplazamiento.
- **parent:** Hace referencia a la ventana donde está situado el frame actual.
- **personalbar:** Objeto barra personal del navegador.
- **self:** Ventana o frame actual.
- **scrollbars:** Objeto de las barras de desplazamiento de la ventana.

JavaScript - Jerarquía de Objetos

El Objeto Windows – Propiedades - IV

- **status:** Texto de la barra de estado.
- **statusbar:** Objeto barra de estado del navegador.
- **toolbar:** Objeto. Barra de herramientas.
- **top:** Hace referencia a la ventana donde está situada el frame Actual. Como la propiedad parent.
- **window:** Hace referencia a la ventana actual, igual que la propiedad self.

Ejemplo

JavaScript - Jerarquía de Objetos

El Objeto Windows

Métodos

- | | | |
|---|---|--|
| <ul style="list-style-type: none">• alert()• back()• blur()• captureEvents()• clearInterval()• clearTimeout()• close()• confirm(texto)• find() | <ul style="list-style-type: none">• focus()• forward()• home()• moveBy()• moveTo()• open()• print()• prompt()• releaseEvents() | <ul style="list-style-type: none">• resizeBy()• resizeTo()• routeEvent()• scroll()• scrollBy()• scrollTo()• setInterval()• setTimeout()• stop() |
|---|---|--|

JavaScript - Jerarquía de Objetos

El Objeto Windows – Métodos I

- **alert(mensaje):** Presenta una ventana de alerta donde se puede leer el texto que recibe por parámetro
- **back():** Vuelve una página atrás en el historial
- **blur():** Quita el foco de la ventana actual.
- **captureEvents(eventos):** Captura los eventos que se indiquen por parámetro
- **clearInterval():** Elimina la ejecución de sentencias asociadas a un intervalo indicadas con el método setInterval().
- **clearTimeout():** Elimina la ejecución de sentencias asociadas a un tiempo de espera indicadas con el método setTimeout().
- **close():** Cierra la ventana.

JavaScript - Jerarquía de Objetos

El Objeto Windows – Métodos II

- **confirm(mensaje):** Muestra una ventana de confirmación con el mensaje “mensaje” y botones de aceptación o rechazo.
- **find():** Muestra una ventana de búsqueda.
- **focus():** Coloca el foco de la aplicación en la ventana.
- **forward():** Va una página adelante en el historial.
- **home():** Va a la página de inicio que haya configurada en el explorador.
- **moveBy(pixelsX, pixelsY):** Mueve la ventana del navegador los pixels que se indican por parámetro hacia la derecha y abajo.
- **moveTo(pixelsX, pixelsY):** Mueve la ventana del navegador a la posición indicada en las coordenadas que recibe por parámetro.

JavaScript - Jerarquía de Objetos

El Objeto Windows – Métodos III

- **open(URL, nombre, propiedades)**. Abre *URL* en la ventana *nombre* (si no existe, abrirá una nueva) con las *propiedades* especificadas. Las propiedades posibles son:

–toolbar = [yes|no|1|0].

–location = [yes|no|1|0].

–directories = [yes|no|1|0].

–status = [yes|no|1|0].

–menubar = [yes|no|1|0].

–scrollbars = [yes|no|1|0].

–resizable = [yes|no|1|0].

–width = px. Ancho de la ventana

–outerWidth = px. Ancho *total* de la ventana en pixels.

–outerHeight = px. Alto *total* de la ventana en pixels.

–left = px. Distancia en pixels desde el lado izquierdo de la pantalla a la que se debe colocar la ventana.

–top = px. Distancia en pixels desde el lado superior de la pantalla a la que se debe colocar la ventana.

JavaScript - Jerarquía de Objetos

El Objeto Windows – Métodos IV

- **print():** Equivale al botón de imprimir del navegador.
- **prompt(mensaje, respuesta_por_defecto):** Muestra un cuadro de diálogo con una caja de texto en la cual se puede escribir una respuesta a la pregunta en 'mensaje'. *respuesta_por_defecto* es opcional, y mostrará la respuesta por defecto indicada al abrirse el cuadro de diálogo. Retorna la cadena introducida.
- **releaseEvents(eventos):** Deja de capturar eventos del tipo que se indique por parámetro.
- **resizeBy(pixelsAncho,pixelsAlto):** Redimensiona el tamaño de la ventana. Añade a su tamaño los valores indicados en los parámetros. Admite valores negativos si se desea reducir la ventana.

JavaScript - Jerarquía de Objetos

El Objeto Windows – Métodos V

- **resizeTo(pixelsAncho, pixelsAlto):** Redimensiona la ventana del navegador para que ocupe el espacio en pixels que se indica en los parámetros.
- **routeEvent()** Enruta un evento por la jerarquía de eventos.
- **scroll(pixelsX,pixelsY):** Hace un scroll de la ventana hacia la coordenada indicada por parámetro.
- **scrollBy(pixelsX,pixelsY):** Hace un scroll del contenido de la ventana relativo a la posición actual.
- **scrollTo(pixelsX,pixelsY):** Hace un scroll de la ventana a la posición indicada por el parámetro. Este método se tiene que utilizar en lugar de scroll.

JavaScript - Jerarquía de Objetos

El Objeto Windows – Métodos VI

- **setInterval():** Define un script para que sea ejecutado indefinidamente en cada intervalo de tiempo.
- **setTimeout(sentencia,milisegundos):** Define un script para que sea ejecutado una vez después de un tiempo de espera determinado.
- **stop():** Como pulsar el botón de stop de la ventana del navegador.

[Ejemplo](#)

JavaScript - Jerarquía de Objetos

El Objeto Document

- Con el objeto *document* se controla la página web y todos los elementos que contiene.
- Es el que tiene el contenido de toda la página que se está visualizando.
- Gracias a este objeto se puede añadir o hacer cambios dinámicamente al contenido de la página.

JavaScript - Jerarquía de Objetos

El Objeto Document – Propiedades

Propiedades

- | | | |
|---|--|--|
| <ul style="list-style-type: none">▪ alinkColor▪ anchor▪ anchors array▪ applet▪ applets array▪ area▪ bgColor▪ classes▪ cookie▪ domain | <ul style="list-style-type: none">▪ embed▪ embeds array▪ fgColor▪ <u>form</u>▪ forms array▪ ids▪ image▪ images array▪ lastModified▪ linkColor | <ul style="list-style-type: none">▪ link▪ links array▪ Location▪ referrer▪ tags▪ title▪ URL▪ vlinkColor |
|---|--|--|

JavaScript - Jerarquía de Objetos

El Objeto Document – Propiedades I

- **alinkColor:** Color de los enlaces activos
- **Anchor:** Ancla de la página. Se consigue con la etiqueta . Se accede por su nombre.
- **anchors array:** Array de las anclas del documento.
- **Applet:** Applet de la página. Se accede por su nombre.
- **applets array:** Array con todos los applets de la página.
- **Area:** Etiqueta <AREA>, vinculada a los mapas de imágenes. Se accede por su nombre
- **bgColor:** Color de fondo del documento.
- **classes:** Clases definidas en la declaración de estilos CSS.

JavaScript - Jerarquía de Objetos

El Objeto Document – Propiedades II

- **cookie:** Una cookie
- **domain:** Nombre del dominio del servidor de la página.
- **Embed:** Elemento de la pagina incrustado con la etiqueta `<EMBED>`. Se accede por su nombre.
- **embeds array:** Array con todos los elementos de la página incrustados con `<EMBED>`.
- **fgColor:** Color del texto. Para ver los cambios hay que reescribir la página.
- **Form:** Formulario de la página. Se accede por su nombre.
- **forms array:** Array con todos los formularios de la página.
- **ids:** Para acceder a estilos CSS.

JavaScript - Jerarquía de Objetos

El Objeto Document – Propiedades III

- **Image:** Imagen de la página web. Se accede por su nombre.
- **images array:** Array con las imágenes de la página
- **lastModified:** Fecha de la última modificación del documento.
- **linkColor:** Color de los enlaces.
- **Link:** Enlace de la página. Se accede por su nombre.
- **links array:** Array con los enlaces de la página.
- **Location:** URL del documento que se está visualizando. Es de solo lectura.
- **Referrer:** Página de la que viene el usuario.

JavaScript - Jerarquía de Objetos

El Objeto Document – Propiedades IV

- **tags:** Estilos definidos a las etiquetas de HTML en la página web.
- **title:** El título de la página.
- **URL:** Idem a location, pero es aconsejable utilizar location ya que URL no existe en todos los navegadores.
- **vlinkColor:** Color de los enlaces visitados.

[Ejemplo](#)

JavaScript - Jerarquía de Objetos

El Objeto Document – Métodos

- Los métodos de document cumplen principalmente dos funcionalidades:
 - Control de los documentos (escribir, abrirlos y cerrarlos).
 - Control de los eventos en el documento

JavaScript - Jerarquía de Objetos

El Objeto Document – Métodos I

- **captureEvents(evento):** Para capturar los eventos que ocurran en la página web. Recibe como parámetro el evento que se desea capturar.
- **close():** Cierra el flujo del documento.
- **contextual():** Ofrece una línea de control de los estilos de la página. En el caso que se desee especificarlos con Javascript.
- **getSelection():** Devuelve un string que contiene el texto que seleccionado.
- **handleEvent():** Invoca el manejador de eventos del elemento especificado.
- **open():** Abre el flujo del documento.

JavaScript - Jerarquía de Objetos

El Objeto Document – Métodos II

- **releaseEvents():** Libera los eventos capturados del tipo que se especifique, enviándolos a los objetos siguientes en la jerarquía.
- **routeEvent():** Pasa un evento capturado a través de la jerarquía de eventos habitual.
- **write():** Escribe dentro de la página web. Podemos escribir etiquetas HTML y texto normal.
- **writeln():** Escribe igual que el método write(), aunque coloca un salto de línea al final.

JavaScript - Jerarquía de Objetos

El Objeto Form

- Depende del objeto document en la jerarquía de objetos
- Es el contenedor de todos los elementos de un formulario.
- Los formularios se agrupan en un array dentro de document.
- Cada elemento de este array es un objeto de tipo form.

JavaScript - Jerarquía de Objetos

El Objeto Form

- Para acceder a un formulario desde el objeto `document` podemos hacerlo de dos formas.
 - A partir de su nombre, asignado con el atributo `NAME` de `HTML`.

`document.nombreFormulario`

- Mediante la matriz de formularios del objeto `document`, con el índice del formulario al que queremos acceder.

`document.forms[i]`

JavaScript - Jerarquía de Objetos

El Objeto Form

- Para acceder a los elementos de un formulario que depende del objeto Forms podemos hacerlo de dos formas.

- A partir del nombre del objeto asignado con el atributo NAME de HTML.

`document.nombreFormulario.nombreCampo`

- Mediante la matriz de elementos del objeto form, con el índice del elemento al que queremos acceder.

`document.nombreFormulario.elements[i]`

[Ejemplo](#)

JavaScript - Jerarquía de Objetos

El Objeto Form - Propiedades

- **action:** Acción a realizar cuando se envía el formulario. Se corresponde con el atributo ACTION del formulario.
- **elements array:** Matriz de elementos que contiene cada uno de los campos del formulario.
- **encoding:** Tipo de codificación del formulario
- **length:** Número de campos del formulario.
- **method:** Método por el se envía la información. Se corresponde con el atributo METHOD del formulario.
- **Name:** Nombre del formulario. Se corresponde con el atributo NAME del formulario.
- **target:** Ventana o frame en la que está dirigido el formulario. Cuando se envía, se actualizará la ventana o frame indicado. Se corresponde con el atributo target del formulario.

JavaScript - Jerarquía de Objetos

El Objeto Form - Métodos

- **submit():** Hace que el formulario se envíe, aunque no se haya pulsado el botón de submit.
- **reset():** Reinicializa todos los campos del formulario, como si se hubiese pulsado el botón de reset.

[Ejemplo](#)

JavaScript - Jerarquía de Objetos

Propiedades y Métodos de los Elementos de un Objeto Form

Campo	Propiedades	Métodos
Text	defaultValue – form – name – type – value	blur() – focus() – select()
Password Hidden	form – name – type – value	
Checkbox	checked – defaultChecked – value	click() – blur () – focus()
Radiobutton	checked – defaultChecked – value – length	click() – blur () – focus()

JavaScript - Jerarquía de Objetos

Propiedades y Métodos de los Elementos de un Objeto Form

Campo	Propiedades	Métodos
Select	length – option – options – selectedIndex	blur() – focus()
Option	defaultSelected – index – selected – text – value	No tienen métodos
Textarea	defaultValue – value – form – name – type	blur() – focus() – select()

[Ejemplo](#)

JavaScript - Jerarquía de Objetos

Manejo de Eventos con JavaScript

- **Los eventos son la manera que provee Javascript para controlar las acciones de los visitantes y definir el comportamiento de la página cuando se producen.**
- **Para definir las acciones a realizar al producirse un evento se utilizan los manejadores de eventos.**
- **El manejador de eventos se coloca en la etiqueta HTML del elemento que debe responderle,**

JavaScript - Jerarquía de Objetos

Manejo de Eventos con JavaScript

Sintaxis:

```
<etiqueta_HTML manejador_de_Evento="sentencias_javascript">
```

Ejemplo:

```
<INPUT name="b1" type=button value="Pulsar Aqui"  
onclick="b1.value='Pulsado'">
```

[Ejemplo](#)

JavaScript - Jerarquía de Objetos

Manejo de Eventos con JavaScript

Manejador	Descripción
onabort	Se produce cuando se detiene la carga de una imagen
onblur	Se produce cuando un elemento pierde el foco de la aplicación.
onchange	Se produce cuando cambia el estado de un elemento de un formulario
onclick	Se produce cuando se da una pulsación o clic al botón del ratón sobre un elemento de la página

JavaScript - Jerarquía de Objetos

Manejo de Eventos con JavaScript

Manejador	Descripción
ondragdrop	Se produce cuando se suelta algo que se había arrastrado sobre la página.
onerror	Se produce cuando no se puede cargar un documento o una imagen y esta queda rota.
onfocus	Se produce cuando un elemento de la página o la ventana ganan el foco de la aplicación.
onkeydown	Este evento se produce cuando se presiona una tecla, independientemente que la suelte o no.

JavaScript - Jerarquía de Objetos

Manejo de Eventos con JavaScript

Manejador	Descripción
onkeypress	Se produce cuando una tecla se deja pulsada un tiempo determinado. Antes de este evento se produce un onkeydown
onkeyup	Se produce cuando se deja de apretar una tecla. Se produce en el momento que se libera la tecla.
onload	Se produce cuando la página, o las imágenes, han terminado de cargarse.

JavaScript - Jerarquía de Objetos

Manejo de Eventos con JavaScript

Manejador	Descripción
onmousedown	Se produce cuando se pulsa sobre un elemento de la página el botón del mouse. Se produce en el momento de pulsar el botón, se suelte o no.
onmousemove	Se produce cuando el ratón se mueve por la página.
onmouseout	Se produce cuando el puntero del ratón sale del área ocupada por un elemento de la página.
onmouseover	Se produce cuando el puntero del ratón entra en el área ocupada por un elemento de la página.

JavaScript - Jerarquía de Objetos

Manejo de Eventos con JavaScript

Manejador	Descripción
onmouseup	Se produce en el momento que se suelta el botón del ratón, que previamente había pulsado.
onmove	Se produce cuando se mueve la ventana del navegador, o un frame.
onresize	Se produce cuando se redimensiona la ventana del navegador o un frame
onreset	Está asociado a los formularios. Se produce cuando se hace clic en el botón reset del formulario.

JavaScript - Jerarquía de Objetos

Manejo de Eventos con JavaScript

Manejador	Descripción
onselect	Se produce cuando se realiza la selección de un elemento de un formulario.
onsubmit	Se produce cuando se presiona el botón de enviar de un formulario. Se ejecuta antes del envío propiamente dicho.
onunload	Se produce al abandonar una página, ya sea porque se pulse sobre un enlace que nos lleve a otra página o porque se cierre la ventana del navegador.

[Ejemplo](#)