



## Capítulo 5: Eventos en JavaScript

1. Eventos en JavaScript
2. Métodos de evento disponibles en JavaScript
3. Eventos *onLoad* y *onUnload*
4. Ejemplo de aplicación en formularios

### 1. Eventos en JavaScript

En JavaScript, la interacción con el usuario se consigue mediante la captura de los eventos que éste produce. Un evento es una acción del usuario ante la cual puede realizarse algún proceso (por ejemplo, el cambio del valor de un formulario, o la pulsación de un enlace).

Los eventos se capturan mediante los manejadores de eventos. El proceso a realizar se programa mediante funciones JavaScript llamadas por los manejadores de eventos.

La siguiente tabla muestra los manejadores de eventos que pueden utilizarse en JavaScript, la versión a partir de la cual están soportados y su significado.

Manejador	Versión	Se produce cuando...
<i>onAbort</i>	1.1	El usuario interrumpe la carga de una imagen
<i>onBlur</i>	1.0	Un elemento de formulario, una ventana o un marco pierden el foco
<i>onChange</i>	1.0 (1.1 para <i>FileUpload</i> )	El valor de un campo de formulario cambia
<i>onClick</i>	1.0	Se hace <i>click</i> en un objeto o formulario
<i>onDbClick</i>	1.2 (no en Mac)	Se hace <i>click</i> doble en un objeto o formulario
<i>onDragDrop</i>	1.2	El usuario arrastra y suelta un objeto en la ventana
<i>onError</i>	1.1	La carga de un documento o imagen produce un error
<i>onFocus</i>	1.1 (1.2 para <i>Layer</i> )	Una ventana, marco o elemento de formulario recibe el foco
<i>onKeyDown</i>	1.2	El usuario pulsa una tecla
<i>onKeyPress</i>	1.2	El usuario mantiene pulsada una tecla
<i>onKeyUp</i>	1.2	El usuario libera una tecla
<i>onLoad</i>	1.0 (1.1 para	El navegador termina la carga

	<i>image</i> )	de una ventana
<i>onMouseDown</i>	1.2	El usuario pulsa un botón del ratón
<i>onMouseMove</i>	1.2	El usuario mueve el puntero
<i>onMouseOut</i>	1.1	El puntero abandona una área o enlace
<i>onMouseOver</i>	1.0 (1.1 para <i>area</i> )	El puntero entra en una área o imagen
<i>onMouseUp</i>	1.2	El usuario libera un botón del ratón
<i>onMove</i>	1.2	Se mueve una ventana o un marco
<i>onReset</i>	1.1	El usuario limpia un formulario
<i>onResize</i>	1.2	Se cambia el tamaño de una ventana o marco
<i>onSelect</i>	1.0	Se selecciona el texto del campo texto o área de texto de un formulario
<i>onSubmit</i>	1.0	El usuario envía un formulario
<i>onUnload</i>	1.0	El usuario abandona una página

Ejemplo de evento:

```
<INPUT TYPE="text" onChange="CompruebaCampo(this)">
```

En este ejemplo, **CompruebaCampo()** es una función JavaScript definida en alguna parte del documento HTML (habitualmente en la cabecera del mismo). El identificador **this** es una palabra propia del lenguaje, y se refiere al objeto desde el cual se efectúa la llamada a la función (en este caso, el campo del formulario).

La siguiente tabla muestra los eventos que pueden utilizarse con los objetos del modelo de objetos JavaScript del Navigator.

Manejador de evento	Objetos para los que está definido
<i>onAbort</i>	<i>Image</i>
<i>onBlur</i>	<i>Button, Checkbox, FileUpload, Layer, Password, Radio, Reset, Select, Submit, Text, Textarea, window</i>
<i>onChange</i>	<i>FileUpload, Select, Text, Textarea</i>
<i>onClick</i>	<i>Button, document, Checkbox, Link, Radio, Reset, Submit</i>
<i>onDbClick</i>	<i>document, Link</i>
<i>onDragDrop</i>	<i>window</i>

<i>onError</i>	<i>Image, window</i>
<i>onFocus</i>	<i>Button, Checkbox, FileUpload, Layer, Password, Radio, Reset, Select, Submit, Text, Textarea, window</i>
<i>onKeyDown</i>	<i>document, Image, Link, Textarea</i>
<i>onKeyPress</i>	<i>document, Image, Link, Textarea</i>
<i>onKeyUp</i>	<i>document, Image, Link, Textarea</i>
<i>onLoad</i>	<i>Image, Layer, window</i>
<i>onMouseDown</i>	<i>Button, document, Link</i>
<i>onMouseMove</i>	Ninguno (debe asociarse a uno)
<i>onMouseOut</i>	<i>Layer, Link</i>
<i>onMouseOver</i>	<i>Layer, Link</i>
<i>onMouseUp</i>	<i>Button, document, Link</i>
<i>onMove</i>	<i>window</i>
<i>onReset</i>	<i>Form</i>
<i>onResize</i>	<i>window</i>
<i>onSelect</i>	<i>Text, Textarea</i>
<i>onSubmit</i>	<i>Form</i>
<i>onUnload</i>	<i>window</i>

## 2. Métodos de evento disponibles en JavaScript

Los siguientes métodos de evento pueden utilizarse en JavaScript:

Métodos de evento	Función que realizan
<i>blur()</i>	Elimina el foco del objeto desde el que se llame
<i>click()</i>	Simula la realización de un <i>click</i> sobre el objeto desde el que se llame
<i>focus()</i>	Lleva el foco al objeto desde el que se llame
<i>select()</i>	Selecciona el área de texto del campo desde el que se llame
<i>submit()</i>	Realiza el envío del formulario desde el que se llame

Ejemplo:

```

<HTML>
<HEAD><TITLE>Eventos</TITLE>
<SCRIPT>
<!--
function Reacciona(campo) {
    alert("¡Introduzca un valor!")
    campo.focus()
}
//-->
</SCRIPT></HEAD>
<BODY> <FORM METHOD=POST>
<INPUT TYPE=text NAME=campo onFocus="Reacciona
(this)">
</FORM> </BODY> </HTML>

```

### 3. Eventos *onLoad* y *onUnload*

Se usan como atributos del tag **<BODY>** de HTML.

Ejemplo:

```

<BODY onLoad="Hola()" onUnload="Adios()">

```

La función **Hola()** se ejecutará antes de que se cargue la página y la función **Adios()** al abandonarla.

Ejemplo:

```

<HTML>
<HEAD>
<TITLE>Ejemplo onLoad y onUnload</TITLE>
</HEAD>
<BODY onLoad="alert('¡Bienvenido a mi página!')"
onUnload="alert('¡Vuelva pronto!')">
...
</BODY>
</HTML>

```

En este otro ejemplo se utilizan funciones:

```

<HTML>
<HEAD>
<TITLE>Ejemplo con funciones</TITLE>
<SCRIPT LANGUAGE="JavaScript">
<!--
var name = ""
function Hola() {
    nombre = prompt('Introduzca su nombre:', '')
    alert('¡Hola ' + nombre + '!')
}

function Adios() {
    alert('¡Adios ' + nombre + '!')
}
//-->
</SCRIPT>

```

### 4. Ejemplo de aplicación en formularios

Vamos a ver un ejemplo para un campo de texto:

```

<INPUT TYPE=text NAME="test" onBlur="alert
('¡Gracias!')" onChange="Comprueba(this)">

```

En este ejemplo, simulamos una calculadora interactiva:

```
<HTML>
<HEAD><TITLE>Calculadora interactiva</TITLE>
<SCRIPT LANGUAGE="JavaScript">
<!--
function Calcula(form) {
    form.resultados.value = eval
(form.entrada.value)
}

function CogeExpresion(form) {
    form.entrada.blur()
    form.entrada.value = prompt("Introduce una
expresión matemática válida en JavaScript","")
    Calcula(form)
} //-->
</SCRIPT></HEAD> <BODY> <FORM METHOD=POST>
```

5

6