

Window

Introducción:

El modelo BOM (**B**rowser **O**bject **M**odel) o **M**odelo de **O**bjeto de **N**avegador, permite acceder a algunas características del navegador a través del JavaScript.

A diferencia del DOM que es un estándar creado por la W3C, el BOM no es un estándar oficial.

Los navegadores modernos han implementado (al menos) los mismos métodos y propiedades para los siguientes objetos:

Window (ventana), **Screen** (pantalla), **Location** (localización), **History** (historial), **Navigator** (navegador), **PopAlert** (cajas de alerta), **Timing** (temporización).

Objeto Window:

El objeto **window** está soportado por la mayoría de los navegadores, y representa la ventana del navegador.

Todos los objetos globales del JavaScript, funciones y variables automáticamente se convierten en elementos del objeto **window**.

- Las variables globales son propiedades el objeto **window**.
- Las funciones globales son métodos del objeto **window**.
- Cada objeto del documento (del DOM HTML) es una propiedad del objeto **window**

```
window.document.getElementById("id_elemento");
```

Es igual a:

```
document.getElementById("id_elemento");
```

Método Open():

El objeto **window** tiene seis métodos, el primero de ellos es el método **window.open()**, que sirve para abrir una ventana.

Sintaxis:

[referencia =] window.open([URL], [nombre], [opciones], [reemplazar]);

- **URL:** opcional, dirección del documento, si no se especifica abre con *about:blank*.
- **nombre:** opcional, las mismas opciones que el atributo *target* de los enlaces HTML.
 - **_blank:** abre una ventana nueva (por defecto).
 - **_parent:** abre en el marco padre.
 - **_self:** en la ventana actual.
 - **_top:** reemplaza cualquier conjunto de marcos que exista.
 - **nombre:** nombre de la ventana.
- **opciones:** opcional, opciones relacionadas con la ventana:
 - **channelmode:** ventana en modo *teatro*. Por defecto es no. Sólo IE.
 - **fullscreen:** ventana en modo pantalla completa. Por defecto es no. Debe de estar en modo *teatro*. Sólo IE.
 - **height:** altura en pixels, mínimo 100.
 - **left:** desplazamiento horizontal en pixels, sólo valores positivos.
 - **location:** activar o no la barra de navegación. Sólo Opera.
 - **menubar:** activar o no la barra de menús.
 - **resizable:** ventana redimensionable o no. Sólo IE.
 - **scrollbars:** activar o no la barras de desplazamiento. Sólo IE, Firefox y Opera.
 - **status:** activar o no la barra de estado.
 - **titlebar:** activar o no la barra de título.
 - **top:** desplazamiento vertical en pixels, sólo valores positivos.
 - **width:** anchura de la ventana en pixels, mínimo 100.

- **reemplazar**: opcional, si crea una nueva entrada o reemplaza la actual en el historial de navegación:
 - **true**: reemplaza la entrada actual del documento.
 - **false**: crea una nueva entrada.
- **referencia**: referencia a la ventana abierta para enviar contenido

Ejemplos:

```
window.open("pag1.html", "vent1", "menubar=yes,toolbar=no,location=no");  
  
var v2 = window.open("", "vent2", "menubar=1,toolbar=0,location=0");  
  
v2.document.writeln("<p>Párrafo en la ventana 2.</p>");
```

Método close():

El método **window.close()** cierra la ventana. Se usa en conjunción con el método **window.open()**.

Sintaxis: `window.close()` | *referencia.close()*

Ejemplos:

```
var vent1 = window.open("", "pag1", "width=640,height=480");  
  
vent1.document.writeln("<p>¡Hola Mundo!</p>");  
  
function cerrar()  
{  
    vent1.close()  
}
```

Método **moveTo()**:

El método **window.moveTo()** desplaza la ventana en posiciones absolutas con origen de coordenadas la parte superior izquierda de la pantalla.

Sintaxis: `window.moveTo(x, y);`

- **x**: obligatorio, coordenada horizontal a mover la ventana, admite valores negativos.
- **y**: obligatorio, coordenada vertical a mover la ventana, admite valores negativos.

Ejemplo:

```
var miVentana = window.open("", "myWin", "width=720,height=576");
```

```
miVentana.moveTo(100, 200); // Posición absoluta (100, 200)
```

Método **moveBy()**:

El método **window.moveBy()** hace el desplazamiento de la ventana desde su posición actual.

Sintaxis: `window.moveBy(dx, dy);`

- **dx**: obligatorio, desplazamiento horizontal a mover la ventana, admite valores negativos.
- **dy**: obligatorio, desplazamiento vertical a mover la ventana, admite valores negativos.

Ejemplo:

```
var miVentana = window.open("", "myWin", "width=720,height=576");
```

```
miVentana.moveTo(100, 200); // Posición absoluta (100, 200)
```

```
miVentana.moveBy(-30,-60); // Posición absoluta (70, 140)
```

Método `resizeTo()`:

El método **`window.resizeTo()`** ajusta la ventana a unas dimensiones dadas.

Sintaxis: `window.resizeTo(w, h);`

Ejemplo:

```
var miVentana = window.open("", "ventana", "width=512,height=384");  
miVentana.document.writeln("<p>Ventana de 512x384 pixels</p>");  
miVentana.resizeTo(640, 480);  
miVentana.document.writeln("<p>Ventana de 640x480 pixels</p>");
```

Método `resizeBy()`:

El método **`window.resizeBy()`** redimensiona una ventana a unos valores relativos.

Sintaxis: `window.resizeBy(dw, dh);`

Ejemplo:

```
var miVentana = window.open("", "ventana", "width=512,height=384");  
miVentana.document.writeln("<p>Ventana de 512x384 pixels</p>");  
miVentana.resizeTo(640, 480);  
miVentana.document.writeln("<p>Ventana de 640x480 pixels</p>");  
miVentana.resizeBy(160, 120);  
miVentana.document.writeln("<p>Ventana de 800x600 pixels</p>");
```

Objeto Location:

El objeto window.location tiene información sobre la URL de la página actual, se puede escribir sin el window y tiene las siguiente propiedades:

Propiedad	Descripción
hash	Devuelve o asigna la porción del ancla de la URL
host	Devuelve o asigna el nombre del host y el número de puerto de la URL
hostname	Devuelve o asigna el nombre del host de la URL
href	Devuelve o asigna la URL completa
origin	Devuelve el protocolo, el nombre de host y el número de puerto de una URL
pathname	Devuelve o asigna el camino completo de una URL
port	Devuelve o asigna el número de puerto de una URL
protocol	Devuelve o asigna el protocolo de la URL
search	Devuelve o asigna la porción de los parámetros de consulta de la URL (GET)

Dispone de los siguientes métodos:

Método	Descripción
<code>assign()</code>	Carga un nuevo documento
<code>reload()</code>	Recarga el documento actual
<code>replace()</code>	Reemplaza el documento actual por uno nuevo

Ejemplos:

<script>

`document.write(window.location.href);` // Escribir la URL del documento

**`document.writeln("
");`**

`document.write(location.pathname);` // Escribir el camino

**`document.writeln("
");`**

`document.write(location.protocol);` // Escribir el protocolo

**`document.writeln("
");`**

`location.reload();` // Recargar la página

`location.assign("http://www.fundaciocim.org/");`

`location.replace("http://www.google.es/");`

</script>

Objeto History:

El objeto `window.history` contiene el historial de navegación del navegador. Como el objeto `location`, se puede escribir sin `window` y contiene la siguiente propiedad:

Propiedad	Descripción
<code>length</code>	Devuelve el número de URLs del historial

Y los siguientes métodos:

Método	Descripción
<code>back()</code>	Carga la URL anterior del historial
<code>forward()</code>	Carga la siguiente URL del historial
<code>go()</code>	Carga una URL específica del historial

Ejemplos:

```
<script>  
  
    document.write("Número total de URLs: " + history.length);  
  
    document.writeln("<br>");  
  
    history.back(); // Equivale a pulsar el botón retroceder  
  
    history.forward(); // Equivale a pulsar el botón de avanzar  
  
    history.go(-3); // Ir a la URL de tres posiciones anteriores.  
  
</script>
```


Objeto Navigator:

El objeto `window.navigator` contiene información acerca del propio navegador. Se puede usar sin escribir `window` y tiene las siguientes propiedades:

Propiedad	Descripción
<code>appCodeName</code>	Devuelve el código de nombre del navegador
<code>appName</code>	Devuelve el nombre del navegador
<code>appVersion</code>	Devuelve la información de versión del navegador
<code>cookieEnabled</code>	Indica si están activas las cookies del navegador
<code>language</code>	Devuelve el idioma predefinido del navegador
<code>onLine</code>	Indica si el navegador está en línea
<code>platform</code>	Devuelve información acerca de a qué plataforma está compilado
<code>product</code>	Devuelve el nombre del motor del navegador
<code>userAgent</code>	Devuelve la cabecera user-agent que envía el navegador al servidor para identificarse

Y el siguiente método:

Método	Descripción
<code>javaEnabled()</code>	Indica si está o no activa la máquina virtual de Java

Ejemplos:

<script>

```
document.writeln("Código: " + navigator.appCodeName + "<br>");  
  
document.writeln("Nombre: " + navigator.appName + "<br>");  
  
document.writeln("Versión: " + navigator.appVersion + "<br>");  
  
document.writeln("¿Cookies?: " + navigator.cookieEnabled + "<br>");  
  
document.writeln("Idioma: " + navigator.language + "<br>");  
  
document.writeln("¿En línea?: " + navigator.onLine + "<br>");  
  
document.writeln("Plataforma: " + navigator.platform + "<br>");  
  
document.writeln("Producto: " + navigator.product + "<br>");  
  
document.writeln("Agente: " + navigator.userAgent + "<br>");  
  
document.writeln("¿Java?: " + navigator.javaEnabled() + "<br>");
```

</script>

Acceso a elementos de la ventana padre:

Para acceder a los elementos de la ventana padre mediante el objeto documento se utiliza la propiedad **opener**. Su sintaxis sería la siguiente:

Ejemplos:

```
window.opener.document.getElementById('Title').value;
```