



# BOM - Objetos del navegador. History

- 
- C.F.G.S. DAW
  - 6 horas semanales
  - Mes aprox. de impartición: Nov
  - iPasen - [cjaedia071@g.educaand.es](mailto:cjaedia071@g.educaand.es)

Carmelo José Jaén Díaz

# Índice



Objetivo

Glosario

Interacción persona - ordenador

Objetivos

Características. Usable.

Características. Visual.

Características. Educativo y actualizado.

# OBJETIVO

---



- Identificar el concepto de BOM con los objetos que lo componen y saber utilizarlo en el desarrollo de aplicaciones web.

# GLOSARIO

---



**BOM (Browser Object Model).** Objeto en el que se representa el navegador, no solo el documento.

Mediante el BOM se puede acceder a datos como el historial de navegación, dimensiones de la ventana, etcétera.

**DOM.** Plataforma e interfaz de un documento que permite a los scripts y programas acceder y modificar dinámicamente su contenido, estructura y estilo.

**Tag.** Término en inglés que significa “etiqueta” y hace referencia a una palabra clave que sirve para describir un documento.

# INTRODUCCIÓN

---



Con el objeto *history* podemos manipular el historial de sesión del navegador con facilidad: es decir, podemos consultar las **URL** de las páginas que ha visitado un usuario en el navegador.

En cuanto a propiedades, estudiaremos:

- *length*, que como os podéis imaginar, almacena el número de páginas que contiene nuestro historial.

Y, por supuesto, los métodos, que pueden ser de gran utilidad:

- *back()*: carga la **URL** anterior del historial (es decir, sería como hacer “atrás” en el navegador).
- *forward()*: lo inverso al caso anterior: carga la **URL** siguiente del historial (si la hay).
- *go()*: va a una página en concreto del historial que pasamos por parámetro.

# INTRODUCCIÓN

---



En definitiva los métodos y propiedad de *history* nos van a servir para movernos por el historial utilizando Javascript sin necesidad de interactuar con el propio navegador.

Recuerda que los navegadores tienen dos botones: uno que carga la web anterior y otro que carga la siguiente web que se ha tecleado a la actual (la mayoría de las veces, el usuario está en la más reciente, salvo que se haya navegado hacia atrás).

# CARACTERÍSTICAS

---



El objeto *history* guarda información de las URLs visitadas por el usuario dentro de una ventana del explorador, es decir, no nos referimos al historial completo de todo el navegador sino de la página en que nos encontramos en ese momento.

No hay un estándar público pero la mayoría de los navegadores soportan este objeto

# PROPIEDADES DE *history*

## *length*



*//length: número de URLs en el historial de la página.*

```
alert("URLs del historial: "+history.length);
```



# MÉTODOS DEL OBJETO *history*

---



*//back: carga la url anterior en el historial.*

```
function atras(){  
    history.back();  
}
```

*//forward: cargar la url siguiente en el historial. Solo podemos usarlo si hemos avanzado hacia adelante y //ahora estamos atrás. 0 si hemos cargado una nueva página con el método assign() del objeto location.*

```
function adelante(){  
    history.forward();  
}
```

# MÉTODOS DEL OBJETO *history*

---



*//go(<numero|url>): va a una página concreta del historial. Si indicamos un número positivo va hacia delante el número especificado de páginas; si indicamos un número negativo va hacia atrás el número especificado de páginas; si indicamos una url o parte de ella va a la url en concreto.*

```
function ir(){  
    let numero = prompt("Indica un número para moverte en el  
historial");  
    history.go(numero);  
}
```

# MÉTODOS DEL OBJETO *history*

---



A continuación, para ir testeando el comportamiento de estos tres métodos, en un documento HTML vinculamos el fichero JS dónde estamos implementando este código y creamos tres botones en el `<body>` que llamen a las funciones definidas anteriormente.

```
<button onclick="atras()">Atrás</button>  
<button onclick="adelante()">Adelante</button>  
<button onclick="ir()">Ir a una página</button>
```

# MÉTODOS DEL OBJETO *history*



La salida del código anterior puede verse en la imagen adjunta:

***Nota:*** Estos métodos están limitados a que existan páginas previamente cargadas en el historial.

***Nota II:*** Como se aprecia en la figura, realizan la misma función que los botones del navegador “Retroceder” y “Avanzar”.

