

Lección 27

HTML5

JavaScript APIs

¿Qué es una API?



¿Qué es una API?

Una **API** (*Application Programming Interface* o Interfaz de Programación de Aplicaciones) es un conjunto de instrucciones que dan acceso a fuentes de datos o servicios, que permiten implementar funciones y procedimientos sin necesidad de programarlas de nuevo, sino que simplemente usándolas.



¿Qué son las
API de HTML5?



API JavaScript HTML5

HTML5 tiene varias API, ¿qué significa esto? que tiene varias **funcionalidades ya programadas** que podemos llegar y utilizar.

Éstas van desde poder dibujar en nuestro documento (*Canvas*) hasta poder recibir coordenadas de geolocalización (*Geolocation*), **sólo con nuestro navegador**.

Pero ¿Qué tiene que ver ésto con JavaScript?

API JavaScript HTML5

JavaScript es la estrella de HTML5, ya que la mayoría de las API funcionan con este lenguaje (y las API corresponden casi al 70% de las novedades de HTML5).

No podemos hablar de HTML5 sin darle importancia a JavaScript, ya que este lenguaje se introdujo de lleno en el *core* de los navegadores y hoy nos ofrece (gracias a los motores de JavaScript) rendimiento y posibilidades similares a los lenguajes de propósito general.

API JavaScript HTML5

En el contexto de un navegador existen muchas utilidades, las cuales podemos invocar por medio de APIs HTML5, no sólo para acceder a los elementos de una página, sino también a sus periféricos.

Para todo ello se han desarrollado APIs en JavaScript, que nos permiten acceder a sistemas de almacenamiento, geolocalización, comunicación con servicios web, a periféricos como la cámara, elementos multimedia, etc.

API JavaScript HTML5

Nota: No todos los navegadores soportan actualmente estas API, en principio las últimas versiones de Firefox, Chrome y Opera no tendremos problemas, pero sí en Internet Explorer y Safari.

File API



API File

La API *file* permite crear aplicaciones que le permiten al usuario **interactuar con archivos de forma local**, es decir, cargar archivos e interactuar con ellos sin la necesidad de subirlos.

Nos puede servir para:

- Crear una vista previa en miniatura de imágenes mientras se envían al servidor.
- Permitir que una app guarde una referencia del archivo mientras el usuario está sin conexión.
- Restringir el tamaño de una subida, etc.

API File

En concreto disponemos de las siguientes funciones:

File: Nos permite acceder a información de un fichero individual como es el nombre, el tamaño, el mimetype o una referencia un manejador del fichero.

FileList: Nos permite las mismas acciones que File pero aplicadas a una lista de ficheros. Se utiliza cuando usamos un `<input type="file" multiple>`.


Blob: Nos permite cortar un fichero por rango de bytes.

API File



← → ↻ localhost:3000 🔍 ☆ ⋮

Create Verify



Choose a file or drag it here.

Last documents registered:


File Hash	Timestamp
e55bde19b226181fb630a2148b3969cac1c3fd6a44ebe464ca5ae6e408694227	2016-08-02 23:35:29 UTC
c5ae227955ac5c00545aeb86fc0860dceb17393bfd426d452a9b66bd040fed1f	2016-08-02 23:29:44 UTC
26172ac84d66d9f184d548d3550e0701f812ea34d0f739e17a054ef680f18a76	2016-08-02 23:29:30 UTC
1739949f52cab4a6501e5cc264e1a1e29a9af4c93197b9467136788c47a2320a	2016-08-02 23:18:04 UTC
c5ae227955ac5c00545aeb86fc0860dceb17393bfd426d452a9b66bd040fed1f	2016-08-02 22:57:14 UTC

Last documents confirmed:

File Hash	Timestamp
e55bde19b226181fb630a2148b3969cac1c3fd6a44ebe464ca5ae6e408694227	2016-08-02 23:40:00 UTC
26172ac84d66d9f184d548d3550e0701f812ea34d0f739e17a054ef680f18a76	2016-08-02 23:30:00 UTC
c5ae227955ac5c00545aeb86fc0860dceb17393bfd426d452a9b66bd040fed1f	2016-08-02 23:30:00 UTC
1739949f52cab4a6501e5cc264e1a1e29a9af4c93197b9467136788c47a2320a	2016-08-02 23:20:00 UTC
1739949f52cab4a6501e5cc264e1a1e29a9af4c93197b9467136788c47a2320a	2016-08-02 23:20:00 UTC

← → ↻ localhost:3000 🔍 ☆ ⋮

Create Verify



Choose a file or drag it here.

Last documents registered:

File Hash	Timestamp
e55bde19b226181fb630a2148b3969cac1c3fd6a44ebe464ca5ae6e408694227	2016-08-02 23:35:29 UTC
c5ae227955ac5c00545aeb86fc0860dceb17393bfd426d452a9b66bd040fed1f	2016-08-02 23:29:44 UTC
26172ac84d66d9f184d548d3550e0701f812ea34d0f739e17a054ef680f18a76	2016-08-02 23:29:30 UTC
1739949f52cab4a6501e5cc264e1a1e29a9af4c93197b9467136788c47a2320a	2016-08-02 23:18:04 UTC
c5ae227955ac5c00545aeb86fc0860dceb17393bfd426d452a9b66bd040fed1f	2016-08-02 22:57:14 UTC

Last documents confirmed:

File Hash	Timestamp
e55bde19b226181fb630a2148b3969cac1c3fd6a44ebe464ca5ae6e408694227	2016-08-02 23:40:00 UTC
26172ac84d66d9f184d548d3550e0701f812ea34d0f739e17a054ef680f18a76	2016-08-02 23:30:00 UTC
c5ae227955ac5c00545aeb86fc0860dceb17393bfd426d452a9b66bd040fed1f	2016-08-02 23:30:00 UTC
1739949f52cab4a6501e5cc264e1a1e29a9af4c93197b9467136788c47a2320a	2016-08-02 23:20:00 UTC
1739949f52cab4a6501e5cc264e1a1e29a9af4c93197b9467136788c47a2320a	2016-08-02 23:20:00 UTC

API File

HTML ▾

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width">
  <title>JS Bin</title>
</head>
<body>
  <input type="file" id="files" name="files[]" multiple /><br>
  <output id="list"></output> <!-- Aquí se verán los resultados -->
</body>
</html>
```

History API



API History

La *API History* permite **controlar el historial del navegador**, modificando la URL pero sin salir de la página actual.

Incorpora técnicas para generar artificialmente URLs por cada paso en el proceso, ofreciendo la posibilidad de retornar a estados previos de la aplicación utilizando procedimientos estándar de navegación.

API History: Métodos

History.back(): Va a la' página **anterior** en el historial de la sesión (es lo mismo que hacer click en el botón atrás del navegador). Equivalente a `history.go (-1)`.

History.forward(): Va a la' página **siguiente** en el historial de la sesión (es lo mismo que hacer click en el botón atrás del navegador). Equivalente a `history.go (1)`.

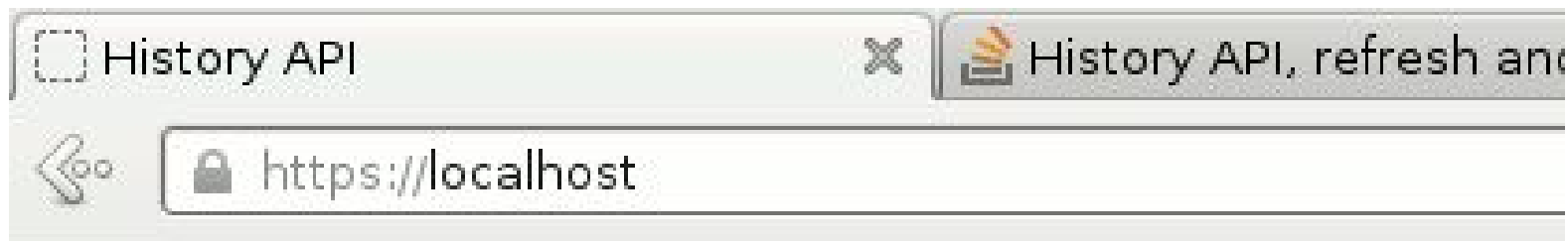
History.go(): Carga una página del historial de la sesión, **identificado por su ubicación con respecto a la página actual**, por ejemplo, -1 para la página anterior o 1 de la página siguiente.

API History: Métodos

History.pushState(): Cambiar directamente la url actual del navegador sin necesidad de refrescar la página. Por razones de seguridad **sólo podrás cambiar el PATH del dominio y no la URL completa.**

History.replaceState(): Cambia la dirección URL en la barra de direcciones con “/ejemplo”. **Este método no manipula la historia del navegador**, sino que simplemente reemplaza la URL actual en la barra de direcciones.

API History



[Page 1](#) [Page 2](#)

Storage API



API Storage

La API de Storage tiene el propósito de **almacenar datos sin utilizar cookies**. Ésta transfiere la responsabilidad por el almacenamiento de datos del servidor al ordenador del usuario.

La API Proporciona dos métodos para almacenar datos en el cliente:

localStorage: Almacena los datos sin fecha de caducidad.

sessionStorage: Almacena los datos para una sola sesión (los datos se pierden cuando la pestaña del navegador se cierra)

API Storage



home.dev/Krasimir/Isbridge/example/

Isbridge example

Send a message: **Subscribe:**

Namespace: Namespace:

Message:

> Subscribed to **bridge** namespace.

home.dev/Krasimir/Isbridge/example/

Isbridge example

Send a message: **Subscribe:**

Namespace: Namespace:

Message:

> Subscribed to **bridge** namespace.

home.dev/Krasimir/Isbridge/example/

Isbridge example

Send a message: **Subscribe:**

Namespace: Namespace:

Message:

> Subscribed to **bridge** namespace.

Geolocation API



API Geolocation

La *API Geolocation* se utiliza para **localizar la posición del usuario** y regresa las coordenadas de longitud y latitud.

Dado que esto puede comprometer la privacidad, la posición no está disponible a menos que el usuario lo apruebe.

API Geolocation: Métodos

Geolocation.getCurrentPosition(): Determina la ubicación actual del dispositivo y devuelve un objeto de posición con los datos.

Geolocation.watchPosition(): Se utiliza para registrar una función de controlador que se llamará automáticamente cada vez que la ubicación del dispositivo cambie.

Geolocation.clearWatch(): Elimina el controlador instalado previamente con watchPosition().

API Geolocation

