**Describiendo algunas Web APIs**

Existe un largo número de Web APIs especificadas por la W3C, de las cuales algunas ya están completamente implementadas en los navegadores y otras solo lo están parcialmente o no han sido implementadas aún. El objetivo de esta lectura es que exploremos brevemente algunas de ellas, con la finalidad de que conozcas el poder de los navegadores y de JavaScript para construir soluciones de software complejas.

| **Tipo de Web API** | **Descripción** |
| --- | --- |
| **Cookies** | Instrucciones para permitir almacenar y recuperar cookies. Las cookies son archivos que permiten guardar información para su posterior recuperación. Su capacidad es limitada. |
| **Manejo de video** | Manipulación de objetos de video incluyendo la edición del mismo. |
| **Manejo de audio** | Manipulación de objetos de audio incluyendo la edición del mismo. |
| **Canvas** | Proporciona un cuadrado para que dibujemos líneas, formas geométricas, en sí funciones similares a las disponibles en un programa para dibujar. |
| **Manejo de base de datos** | Hay varias implementaciones de bases de datos, entre ellas el Local Storage. Pero se pueden construir bases de datos más grandes, jerárquicas o equivalentes a SQL. Por ejemplo, podemos tener una aplicación con una base de datos creada y gestionada con SQL. Representan la evolución de los cookies para el almacenamiento de información. |
| **Criptografía** | Entre la opciones de seguridad tenemos las APIs que proporcionan opciones para cifrar datos. |
| **Tiempo de usuario** | Están relacionadas con el tiempo del usuario, tal como cuánto tiempo un usuario permanece en un elemento o cuánto permanece en el navegador. |
| **Portapapeles** | Hay instrucciones para obtener información sobre el portapapeles, tal como guardar elementos en el portapapeles o recuperar información almacenada en el mismo. |
| **Captura de pantallas** | Permiten grabar la pantalla. |
| **Errores de red** | Incluyen instrucciones para manejar errores de la red. Por ejemplo, si la red tiene algún problema podemos acceder a esos errores. |
| **Fuera de línea** | Hay instrucciones que permiten detectar que la aplicación que estamos usando está fuera de línea, permitiendo que se programen acciones o notificaciones basadas en este evento. Pero también podríamos combinar estas instrucciones con el Local Storage, para guardar información de nuestra página en esa base de datos local. Así, si la aplicación está fuera de línea, podemos buscar la misma en dicha base de datos local, dándole al usuario la sensación que está en línea o que simplemente la aplicación funciona fuera de línea. |
| **Pagos** | Permiten la gestión de los pagos por vía de una aplicación Web. |
| **Dispositivo** | Conjunto de instrucciones que nos permite conocer en qué tipo de dispositivo está corriendo el navegador. |
| **Batería** | Conjunto de instrucciones que nos permite saber cuánto queda de batería en nuestro dispositivo. |
| **Sensores** | En el caso de dispositivos móviles, tenemos instrucciones que manejan los sensores y elementos propios de éstos, tales como los eventos de touch, sensor de luz, acelerómetro y giroscopio. |
| **Vibración** | Conjunto de instrucciones que nos permite acceder a la vibración para enviar este tipo de acción al dispositivo móvil. |
| **Mensajería** | Conjunto de instrucciones que nos permite acceder a la mensajería, para enviar o recibir mensajes de textos. |
| **Telefonía** | Conjunto de instrucciones que nos permite realizar llamadas telefónicas y otras funciones asociadas a la telefonía. |
| **Contactos** | Conjunto de instrucciones que nos permite acceder a los contactos. |
| **Calendario** | Conjunto de instrucciones que nos permite acceder al calendario. |

Cabe destacar, que muchas de las Web APIs no se ejecutan de una vez sino que, cuando las invocamos, el navegador nos pide permiso, solicitándonos el acceso correspondiente.

Recuerda que la lista anterior es solo un pequeño resumen de las Web APIs que existen actualmente. Esperamos que intentes usar algunas de ellas y sobre todo durante el desarrollo del proyecto, demostrando así tus habilidades de programación en JavaScript.

localStorage.setItem('clave', valor); // para guardar

localStorage.getItem('clave'); // para recuperar