

## Week 09 – Chapter 14 Notes – HTML5 APIs

### HTML5

-caniuse.com – website that shows what is currently available on what platform

-interesting that this is the only version I've ever used.

### The data-Attribute

Attributes ignored by the browsers. Almost like secret, so you can reference them in JS. Used like id's inside a tag

data-name = 'Elise'

data-kids = '1.75'

```
<div id='hero' data-powers='flight superSpeed'>
  Superman
</div>
JAVASCRIPT TO ACCESS
const superman = document.getElementById('hero');
const powers = superman.dataset.powers;
<< 'flight superSpeed'
```

**HTML5 APIs** – can give us access to hardware, cameras, batteries, geolocation, graphics card

HTML5 Web Storage (key-value pairs)

-similar to using cookies but has more storage capacity

-has some key differences – more storage, no auto expire, multiple windows as long as same domain

localStorage.setItem('name', 'Elise Halterman'); localStorage.getItem('name'); (also has dot notation)

Geolocation-

-permission has to be granted by user.

-Navigator Property *navigator.geolocation.getCurrentPosition(youAreHere);*

-position object has a coord property that shows latitude and longitude.

-other properties of position are : speed, altitude, heading, timestamp, accuracy,

-watchPosition() of geolocation returns updated location

Web Workers – allows additional thread to run in background.

*Const worker = new Worker('file.js');* downloads file asynchronously and will start when downloaded

-when written Chrome didn't allow local files to be called from Worker() function

- example of practical use is if there's a big math calculation that would stop the page while it calculated. Now it can be a separate file that is run asynchronously and sent back

Shared Web Workers – can have multiple workers on the same script in the same worker object

Service Workers – runs in the background and can also intercept network requests

Websockets –

Allows two way communication with the server instead of having to just send requests and wait for response

```
- Use wss:// instead of http://
- const URL = 'wss://echo.websocket.org/';
- const outputDiv = document.getElementById('output');
- const form = document.forms[0];
- const connection = new WebSocket(URL);
- connection.addEventListener('open', () => {
-   output('CONNECTED');
- }, false);
- function output(message) {
-   const para = document.createElement('p');
-   para.innerHTML = message;
-   outputDiv.appendChild(para);
- }
```

Notifications –

-sends notifications, usually a pop-up

```
if(window.Notification) {
  Notification.requestPermission()
    .then((permission) => {
      if(Notification.permission === 'granted') {
        new Notification('Hello JavaScript!');
      }
    });
}
```

Can react to events like click, close, show.

-Multimedia –

HTML5 introduced <audio> and <video> tags to help prevent having to use things like flash

Use src attribute to point to the file.

Have methods and properties – play(), pause(), volume, muted, currentTime, playbackRate, loop, duration,

**Other APIs**

Canvas, see notes from other chapter. All of this is really basic and there is so much potential to what can be done with canvas.

Shims & Polyfill – used interchangeably. These are libraries of code that allow you to use APIs normally.