Extra notes - HTML - DOM Intro

- Dr Nick Hayward

A brief introduction to HTML's *document object model*, or **DOM**.

Contents

Intro

The *document object model*, or **DOM**, is the core structure for a HTML document. It also informs how we access and manipulate a document using selectors in CSS and JavaScript.

Understanding this structure is pretty straightforward, but nonetheless crucial to working with HTML, CSS, and JavaScript.
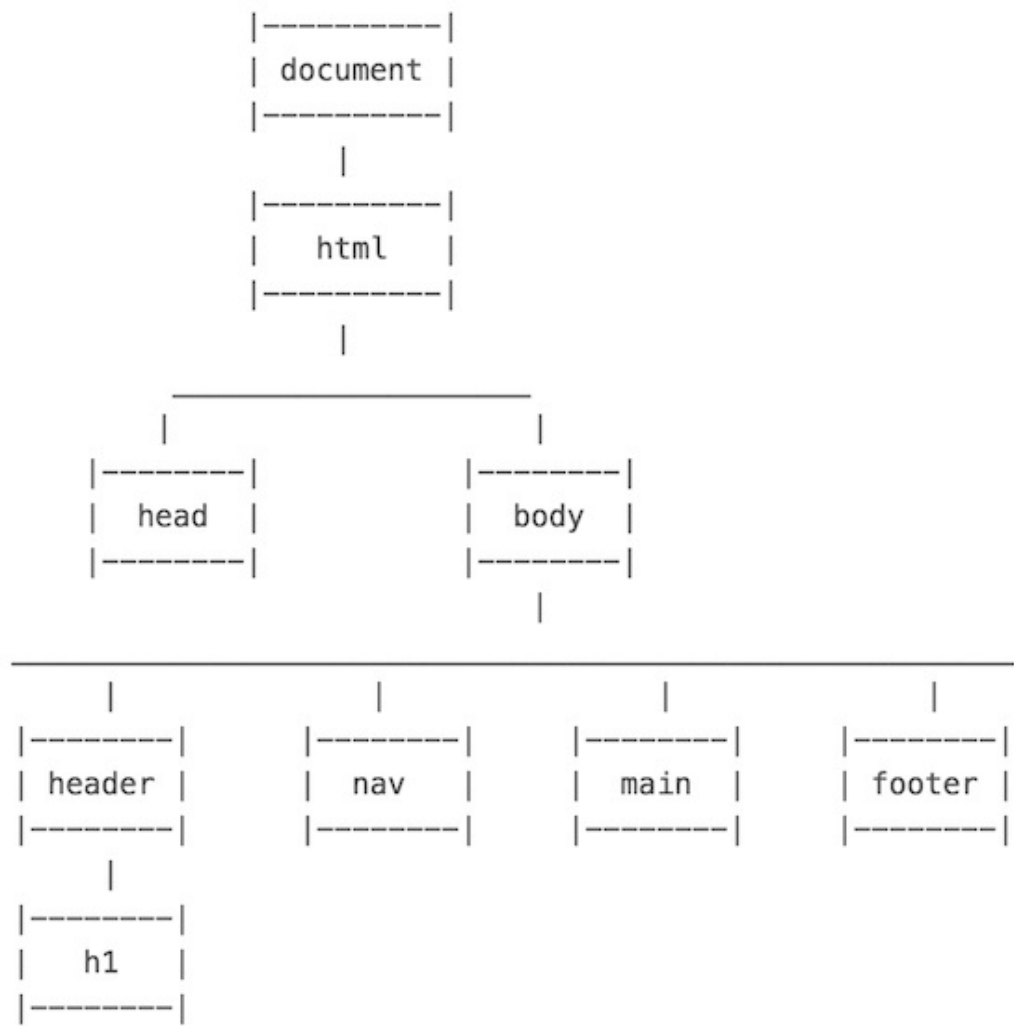
What is DOM?

The DOM presents a platform and language independent way to access and manipulate the underlying structure of a HTML document. It is structured as a representation of a tree data structure, and its manipulation follows this same, standard principle.

This DOM tree is constructed using a set of nodes, and this tree is designed as a hierarchical representation of the underlying document. As we find with an XML document, for example, each node on our tree is an element within our HTML document. The inherent hierarchical order originates with the **root** element, at the top of our **tree**, and descends down following lineage from node to node.

Each node is a child to its parent, and we can often find many siblings per node as well. In effect, we may consider the children as simple descendants of the parent, grandparent, and so on. Each node may also include many children, grandchildren, &c.

A given path along a tree is known as a **branch**, and each node is sometimes referred to as a **leaf**. You'll also notice with a DOM tree, as with tree data structures in general, the root is at the top of the tree.

Image - an example DOM

```
                        |----------|
                        | document |
                        |----------|
                             |
                        |----------|
                        |   html   |
                        |----------|
                             |
               _____
              |                            |
          |--------|                   |--------|
          |  head  |                   |  body  |
          |--------|                   |--------|
                                            |
         _____
        |                  |                 |                 |
   |--------|          |--------|         |--------|        |--------|
   | header |          |  nav   |         |  main  |        | footer |
   |--------|          |--------|         |--------|        |--------|
        |
   |--------|
   |   h1   |
   |--------|
```

Some useful elements

The following is a brief table of some of the common elements we may use to build a HTML DOM, and hence our HTML document. There is a mix of HTML4 and HTML5 elements, which helps us create many different constructs within our HTML pages. These also act as a type of selector, which we may also consider relative to CSS and JavaScript.

There are many other HTML elements, which we'll come across as we start to build our test apps.

element tag	usage & description
`<html>`	container element for a HTML document
`<head>`	contains metadata and document information
`<body>`	contains main content rendered as the HTML document
`<header>`	page header...
`<nav>`	navigation, stores and defines a set of links for internal or external navigation
`<main>`	defined primary content area of document
`<footer>`	page footer...
`<section>`	a section of a page or document
`<article>`	suitable for organising and containing independent content
`<aside>`	defines content aside from the content which contains this element
`<figure>`	logical grouping of image and caption
`<img>`	image - can be local or remote using url in `src` attribute
`<figcaption>`	image caption
`<h1>, <h2>...`	headings from 1 to 6 (1 = largest)
`<a>`	anchor - link to another anchor, document, site...
`<p>`	paragraph
`<ul>, <ol>, <dl>`	unordered, ordered, definition lists
`<li>`	list item, used with `<ul>, <ol>...`
`<dt>`	definition term, used with `<dl>`
`<dd>`	definition description, used with `<dl>`

`<table>`	standard table with rows, columns...
`<tr>` >	table row, used with `<table>`
`<th>`	table heading, used with `<table>` and child to `<tr>`
`<td>`	table cell, used with `<table>` and child to `<tr>`
`<div>`	non-semantic container for content, similar concept to `<section>`
`<span>`	group inline elements in a HTML document
`<canvas>`	HTML5 element for drawing on the HTML page
`<video>`	HTML5 element for embedding video playback
`<audio>`	HTML5 element for embedding audio playback

DOM basics - an example

The following example is a basic `index.html` page for a test app. Our web applications may start with this type of HTML page, which is saved as a `.html` or `.htm` document at the root of our application's directory structure. As the browser loads our web app, it will be looking for this page, which acts as the kickstart for loading and rendering the application. We can customise the structure of this page using HTML, and then add JavaScript and CSS for further DOM manipulation and styling.

You'll also notice that there are a few consistent elements to this underlying structure, which we may use with all of our initial web app templates. These include the HTML elements, `<html>`, `<head>`, and `<body>`.

As we start to build our HTML5 apps, we may also regularly add elements for `<header>`, `<main>`, and `<footer>`, plus lots of other elements suitable for the structure of the app we're building.

```
<!DOCTYPE html>
<html>
  <head>
    <base href="media/images/">
    <meta charset="UTF-8">
    <!-- demo -->
    <title>Demo</title>
  </head>
  <body>
    <header>
      <h1>Ancient Egypt</h1>
    </header>
    <nav>...</nav>
    <main>
      <section>
        <p>
          Welcome to the Ancient Egypt information site.
        </p>
        <figure>
          <img src="philae-demo2.jpg" alt="philae temple" width="333px"
          height="200px">
          <figcaption>Ptolemaic temple at Philae, Egypt</figcaption>
        </figure>
      </section>
      <aside>
        Temple at Philae in Egypt is Ptolemaic era of Egyptian history.
      </aside>
    </main>
    <footer>
      foot of the page...
    </footer>
  </body>
</html>
```

So, the above code is an example breakdown of an initial HTML5 page. We may define this page as follows,

- basic `index.html` page for loading web apps
- app will start with the `index.html` document
  - html pages saved as `.html` or `.htm`
  - `.html` more common...

- `index.html` acts as a kickstart
  - for loading and rendering the app
  - loads other app resources - CSS, JS...

- consistent elements in the HTML DOM
  - `<html>`, `<head>`, and `<body>`

- HTML5 apps will add
  - `<header>`, `<main>`, and `<footer>` (when required)
  - many other elements for building the app...

References

- MDN
  - [HTML developer guide](#)

- W3 Schools
  - [W3Schools - DOM Image](#)