| | |
|---|---|
| **Time** | ~ 5 minutes read |
| **Learning Goals** | • Understand the document object model (DOM) |

We will now take a short break from coding and get on with some reading. Here we will be introducing the Document Object Model, or more commonly referred to as the DOM.

The DOM is a W3C (World Wide Web Consortium) standard for accessing documents and has the following definition:

*"The W3C Document Object Model (DOM) is a platform and language*-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document.

Basically, when your browser loads a webpage, a document object model which takes the form of a tree of objects is created. The tree given in w3schools was formed from the code as below:

```
<!DOCTYPE html>
<html>

<head>
  <title>My title</title>
</head>

<body>
 <a href="">My link</a>
 <h1>My header</h1>
</body>

</html>
```

This tree structure provides a standard for languages such as JavaScript to manipulate the behaviour of the HTML elements and attributes, and CSS styles. With this standard,

traversing/moving through the elements is as simple as understanding the relationship between the nodes in a family tree. Remember the CSS Diner game you played previously? There were certain levels where you needed to traverse down through a parent to get to its child, or move sideways (sibling relationships) - that's your family tree! With more practice and experimentation on this, you will slowly get used to the relationships between the elements and be on your way to be a front-end jedi!

In the next two challenges, you will be learning how to scrape websites. The concept of the tree structure will be helpful in identifying the required selectors for your scraping purposes. Now go and have some hacking fun!

***Important note:*** notice how we have indentations for tags between `<head> ... </head>` and `<body> ... </body` ? The indentations help us identify which are parent nodes, child nodes, and siblings. It gives you a better picture of the tree structure, and helps you identify which selectors to use for your styling and scripting. So from now on, start indenting your code accordingly!