16. fetch APIs and DOM Manipulation Revisited

Table of Content

1. Fetch API

The `fetch()` method starts the process of fetching a resource from a server.
The `fetch()` method returns a Promise that resolves to a Response object.

```
const data = await fetch('https://dummyjson.com/products/1');

const jsonData = await data.json();
```

or

```
fetch(url)
    .then(response => {
        // handle the response
    })
    .catch(error => {
        // handle the error
    });
```
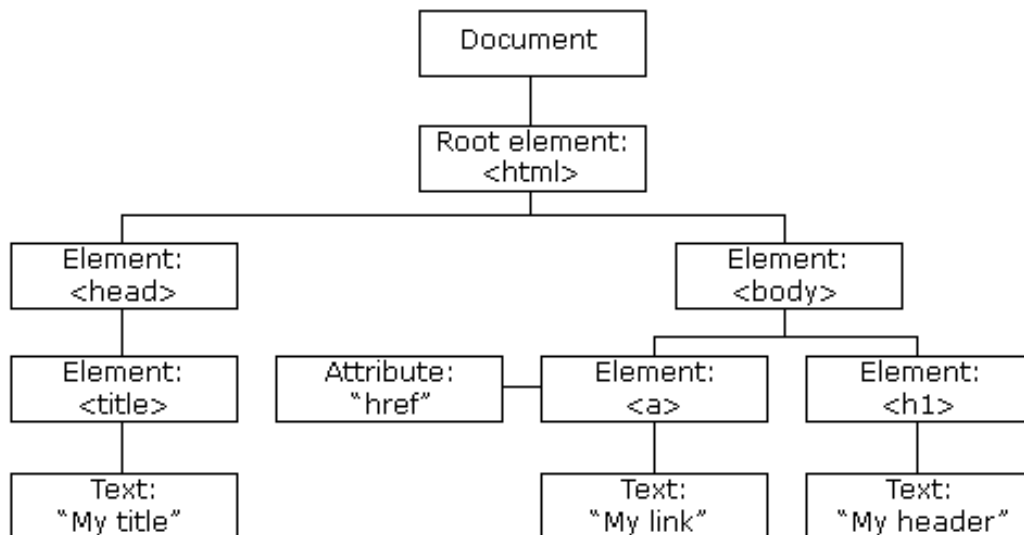
Handling the status codes of the Response

The Response object provides the status code and status text via the status and statusText properties. When a request is successful, the status code is 200 and status

text is OK:

```
async function fetchText() {
    let response = await fetch('/readme.txt');

    console.log(response.status); // 200
    console.log(response.statusText); // OK

    if (response.status === 200) {
        let data = await response.text();
        // handle data
    }
}

fetchText();
```

We also have headers and other options in Fetch API, but we will learn more about them later in this course (while learning NodeJS)

2. DOM Navigation

The nodes in the node tree have a hierarchical relationship to each other.

The terms parent, child, and sibling are used to describe the relationships.

- In a node tree, the top node is called the root (or root node)

- Every node has exactly one parent, except the root (which has no parent)

- A node can have a number of children

- Siblings (brothers or sisters) are nodes with the same parent

▼ **Navigating between Nodes**

You can use the following node properties to navigate between nodes with JavaScript:

- `parentNode`

- `childNodes[ nodenumber ]`

- `firstChild`

- `lastChild`

- `nextSibling`

- `previousSibling`

▼ **Node Values**

```
<title id="demo">DOM Tutorial</title>
```

The element node `<title>` (in the example above) does **not** contain text.

It contains a **text node** with the value "DOM Tutorial".

The value of the text node can be accessed by the node's `innerHTML` property:

```
myTitle = document.getElementById("demo").innerHTML;
```

Accessing the innerHTML property is the same as accessing the `nodeValue` of the first child:

```
myTitle = document.getElementById("demo").firstChild.nodeValue;
```

▼ **Document Body and Element**

There are two special properties that allow access to the full document:

- `document.body` - The body of the document (returns the <body> element)

- `document.documentElement` - The full document (returns the <html> element)

Assignments

1. Build a Random Joke Generator whenever the Next Joke Button is Clicked. Use API: https://icanhazdadjoke.com/slack

2. Add the bottom navigational dots in the Image Slider.

3. Build a List of Quotes along with the author names. Use API: https://dummyjson.com/quotes