Document Object Model

Document Object Model

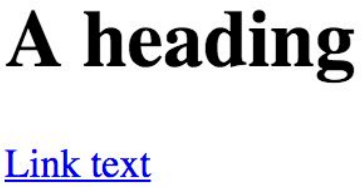
- In this section we will learn about the Document Object Model (DOM)
- The DOM will allow us to interface our Javascript code to interact with HTML and CSS

Document Object Model

- Browsers will construct the DOM, which basically means storing all the HTML tags as Javascript objects.
- Let’s see a simple example…

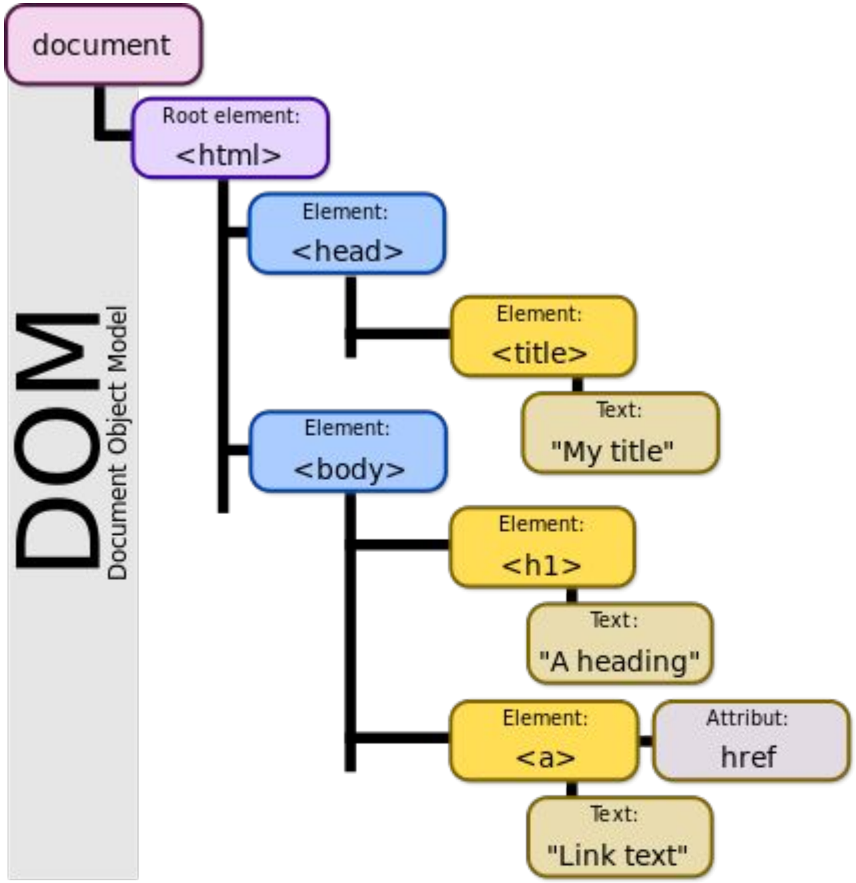
Document Object Model

```
<! DOCTYPE html>
<html>
     <head>
          <title> My Title </title>
     </head>
<body>
     <h1>A heading </h1>
     <a href= “websiteLink”>Link text</a>
</body>
```



A heading

Link text



Document Object Model

- We can see the DOM of any website.
- Go to a website and in the console type:
    - document
- That will return the HTML text of the page. To see the actual objects use:
    - console.dir(document)

Document Object Model

- This DOM will allow us to use Javascript to interact with the web page.
- The DOM is enormous, most developers won’t use all the properties.
- We will cover the common objects used, but be prepared for the unknown!

Document Object Model

- Let’s explore this in the browser!
- Afterwards in the next lecture we will go through an example of using Javascript with the DOM.

Part 1 -DOM Interaction

Using Javascript to interact with DOM elements!

Document Object Model

- In this lecture we will begin to see examples of how to grab HTML elements from the DOM.
- The HTML elements are properties of the DOM

Document Object Model

- We’ll cover how to grab large groups of elements, like the entire body or head of the HTML
- And then focus on grabbing specific HTML items, like classes or ids.

Document Object Model

- Here are some important document attributes:
  - `document.URL`
  - `document.body`
  - `document.head`
  - `document.links`

Document Object Model

- There are many methods for grabbing elements from the DOM:
  - `document.getElementById()`
  - `document.getElementByClassName()`
  - `document.getElementsByTagName()`
  - `document.querySelector()`
  - `document.querySelectorAll()`

Document Object Model

- Check out the relevant files:
  - Part1_MainPage.html
  - Part1_Color_Changer.js
- We will start by disconnecting the html file and exploring it a bit manually, then we will code out a complex example.

Document Object Model

- Let’s explore these various methods!

Document Object Model

- Once you have grabbed an element, you can interact with it!
    - `myvariable.style.color (Many CSS options)`
    - `myvariable.textContent`
    - `myvariable.innerHTML`
    - `myvariable.getAttribute()`
    - `myvariable.setAttribute()`

Document Object Model

- Let’s explore these various methods!

Part 2 - Content Interaction

Document Object Model

Document Object Model

- In this lecture we will see more examples of how to interact with the HTML from the DOM.
- We will show how to change text, HTML code, and attributes.

Document Object Model

- Relevant files are:
  - Part2_Content.html
  - Part2_Interact.js
- Let’s get started!

Part 3 - DOM Events

Using Javascript to trigger on events!

Document Object Model

- We don’t always want to have to specify beforehand how to interact with the DOM
- Many times we only want the interaction to occur on a particular event, such as a click or a hover.

Document Object Model

- We achieve this by adding an Event Listener
- The javascript will be “listening” for an event to occur and then execute a function when it happens.
- Let see some example code!

Document Object Model

- Listening for an event looks like this:
  - `myvariable.addEventListener(event,func);`
- An example:

```
var head = document.querySelector(‘h1’);

head.addEventListener(“click”,changeColor);
```

Document Object Model

- There are many, many possible events!
  - Clicks
  - Hovers
  - Double Clicks
  - Drags
  - Much More!
- https://developer.mozilla.org/en-US/docs/Web/Events

Document Object Model

- Let’s explore these events!

Part 4 - Game Project

Document Object Model

Document Object Model

- It is time to get some practice with the DOM!
- We will be creating a very simple tic tac toe game interface.
- You have two options on the approach for this project.

Document Object Model

- First Option
  - Try to replicate the game completely on your own.
- Second Option
  - Follow along with the “solution” lecture for a code-along session.

Document Object Model

- I recommend you try it at least once on your own to get some practice of seeing something and then trying to replicate it on your own, a great skill to have!
- Let’s start by seeing what the final game looks like!

Part 4 - Game Project Solutions

Document Object Model

PIERIAN DATA