DOM (DOCUMENT OBJECT MODEL)

" The DOM (Document Object Model) is an API that represents and interacts with any HTML or XML document. The DOM is a document model loaded in the browser and representing the document as a node tree, where each node represents part of the document (e.g. an element, text string, or comment). " - MDN Docs

" The Document Object Model represents the HTML on our page, as a hierarchy of objects "

" The Document Object Model represents the HTML on our page, as a hierarchy of objects "

" With the DOM, we can interact with our HTML via these objects "

Simply put, the DOM allows us to:

Simply put, the DOM allows us to:

- add HTML to our page

Simply put, the DOM allows us to:

- add HTML to our page
- remove HTML from our page

Simply put, the DOM allows us to:

- add HTML to our page
- remove HTML from our page
- change HTML on our page

Simply put, the DOM allows us to:

- add HTML to our page
- remove HTML from our page
- change HTML on our page

The DOM allows us to make our pages more dynamic (it responds in real-time without page loads)

Let's break down this phrase

" Document Object Model "

and examine it more closely

D

DOCUMENT

DOCUMENT

The document just refers to our webpage

o

OBJECT

OBJECT

Everything in the DOM is an object, which means:

OBJECT

Everything in the DOM is an object, which means:

- the <head></head> is an object

OBJECT

Everything in the DOM is an object, which means:

- the <head></head> is an object
- the <body></body> is an object

OBJECT

Everything in the DOM is an object, which means:

- the <head></head> is an object
- the <body></body> is an object
- the <h1></h1> is an object

OBJECT

Everything in the DOM is an object, which means:

- the <head></head> is an object
- the <body></body> is an object
- the <h1></h1> is an object
- the is an object

OBJECT

Everything in the DOM is an object, which means:

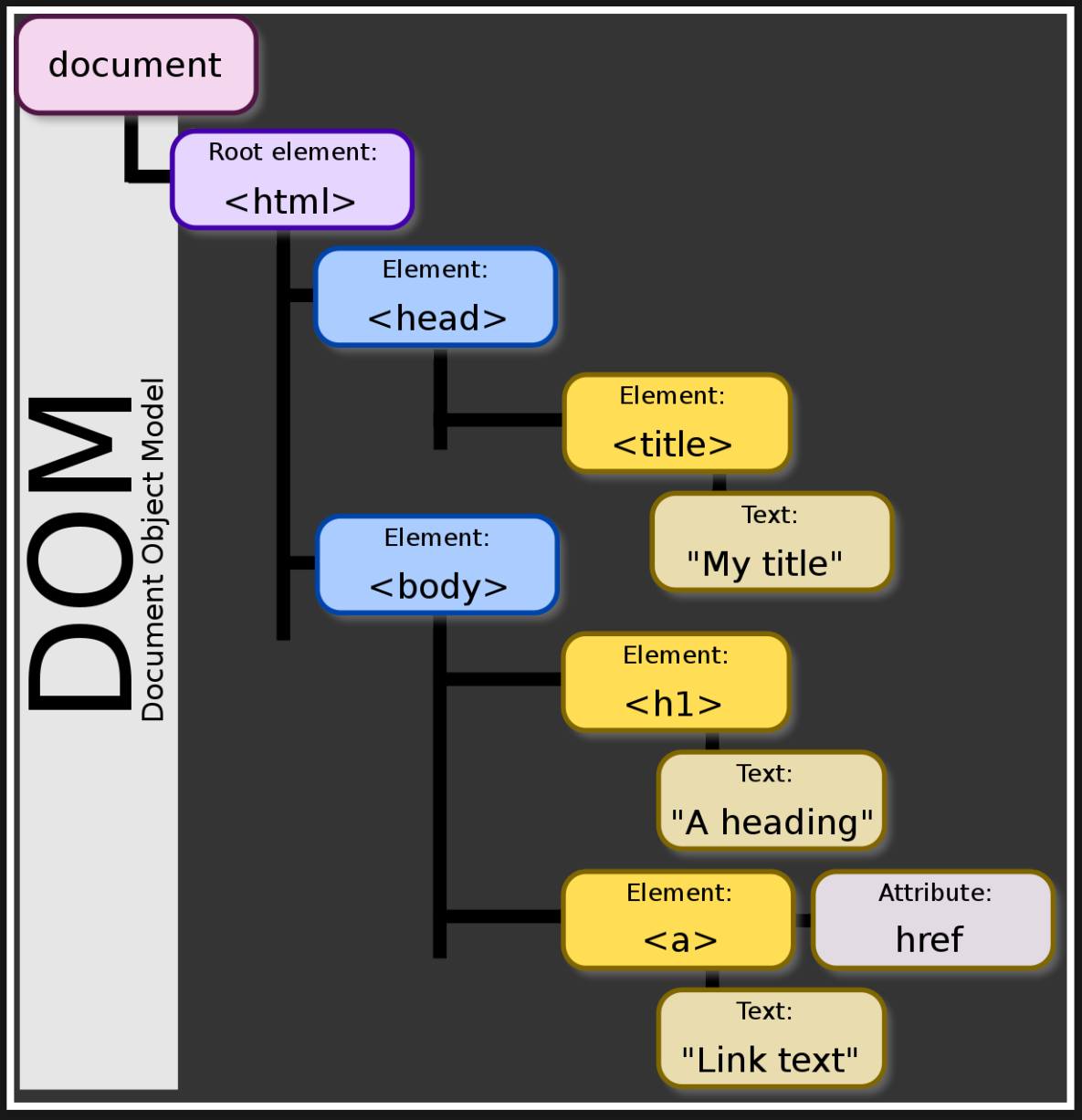
- the <head></head> is an object
- the <body></body> is an object
- the <h1></h1> is an object
- the is an object
- ...in short *all* HTML tags are an object

M

MODEL

MODEL

The model refers to the relationship between all the HTML elements on our page


document
Root element:
<html>
Element:
<head>
Element:
<title>
Text:
"My title"
Element:
<body>
Element:
<h1>
Text:
"A heading"
Element:
<a>
Attribute:
href
Text:
"Link text"
DOM
Document Object Model

Communication betweem our

- DOM Methods
- DOM Events
- DOM Properties

DOM METHODS

DOM METHODS

There are many methods to choose from, but we will focus on only a few

Methods to create new HTML objects

- document.createElement()
- *element*.appendChild()

Methods to create new HTML objects

- document.createElement()
- *element*.appendChild()

Methods to remove objects from the DOM

- *element*.remove()

Methods to get objects from the DOM

- document.querySelector()
- document.querySelectorAll()

Methods to get objects from the DOM

- document.querySelector()
- document.querySelectorAll()

Methods to change objects

- *element*.setAttribute()
- *element*.removeAttribute()
- *element*.style()

DOM EVENTS

DOM EVENTS

Events notify us of important changes in the DOM environment

DOM EVENTS

Events notify us of important changes in the DOM environment

For example, we can use a "mouse" event to notify us if the user clicked on a button

Further reading:

MDN Event Reference

DOM PROPERTIES

The DOM also gives our objects properties (values) which we can read or set. For example:

- *element*.clientHeight
- *element*.clientWidth
- *element*.innerText

For example, if we want to know the dimensions of the tag
<body></body>

For example, if we want to know the dimensions of the tag

`<body></body>`

We navigate to the `body` object from our root `document`

For example, if we want to know the dimensions of the tag

`<body></body>`

We navigate to the `body` object from our root `document`

And finally access the `clientHeight` or `clientWidth` properties

For example, if we want to know the dimensions of the tag
`<body></body>`

We navigate to the `body` object from our root `document`

And finally access the `clientHeight` or `clientWidth` properties

```
1 const height = document.body.clientHeight;
```

LAST THOUGHTS ABOUT THE DOM

LAST THOUGHTS ABOUT THE DOM

- We can think of the DOM as a bridge between our HMTL and our JavaScript

LAST THOUGHTS ABOUT THE DOM

- We can think of the DOM as a bridge between our HMTL and our JavaScript
- It is a collection of objects

LAST THOUGHTS ABOUT THE DOM

- We can think of the DOM as a bridge between our HMTL and our JavaScript
- It is a collection of objects
- These objects are a representation of the HTML tags (HTML elements) on our page

LAST THOUGHTS ABOUT THE DOM

- We can think of the DOM as a bridge between our HMTL and our JavaScript
- It is a collection of objects
- These objects are a representation of the HTML tags (HTML elements) on our page
- Each HTML tag is an object in the DOM

LAST THOUGHTS ABOUT THE DOM

- We can think of the DOM as a bridge between our HMTL and our JavaScript
- It is a collection of objects
- These objects are a representation of the HTML tags (HTML elements) on our page
- Each HTML tag is an object in the DOM
- DOM also represents the hierarchy (parent child relationships) between those objects

Now go build something!