



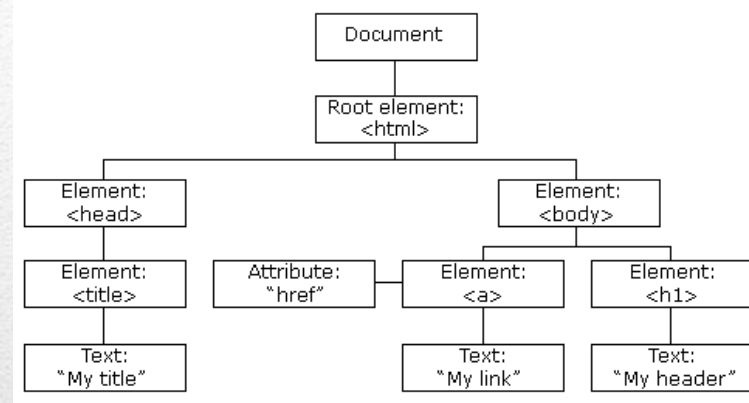
DOM

Manipulation

Document Object Model

Document Object Model

- Every time a website is loaded, the browser creates a **document object model** of the page. It stacks the objects in a hierarchy.



- DOM is a fancy term for “everything on your website”. This includes images, divs, paragraphs, headers, links, tables, forms, etc.
 - **"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."**
 - If you took CIT 160- we used some dom manipulations in our program code.
 - Example: `document.getElementById(idname);` allows us to get information from an element id on the page.
-

HTML DOM

Finding HTML ELEMENTS

Method	Description
<code>document.getElementById()</code>	Find an element by element id
<code>document.getElementsByTagName()</code>	Find elements by tag name
<code>document.getElementsByClassName()</code>	Find elements by class name

Changing HTML Elements

Method	Description
<code>element.innerHTML=</code>	Change the inner HTML of an element
<code>element.attribute=</code>	Change the attribute of an HTML element
<code>element.setAttribute(attribute,value)</code>	Change the attribute of an HTML element
<code>element.style.property=</code>	Change the style of an HTML element

HTML DOM

Adding or Deleting Elements

Method	Description
<code>document.createElement()</code>	Create an HTML element
<code>document.removeChild()</code>	Remove an HTML element
<code>document.appendChild()</code>	Add an HTML element
<code>document.replaceChild()</code>	Replace an HTML element
<code>document.write(<i>text</i>)</code>	Write into the HTML output stream

Adding Event Handlers

Method	Description
<code>document.getElementById(<i>id</i>).onclick=function(<i>code</i>)</code>	Adding event handler code to an onclick event

HTML DOM

Finding and returning HTML objects in the document

Property	Description	DOM
document.anchors	Returns all <a> elements that have a name attribute	1
document.applets	Returns all <applet> elements (Deprecated in HTML5)	1
document.baseURI	Returns the absolute base URI of the document	3
document.body	Returns the <body> element	1
document.cookie	Returns the document's cookie	1
document.doctype	Returns the document's doctype	3
document.documentElement	Returns the <html> element	3
document.documentMode	Returns the mode used by the browser	3
document.documentURI	Returns the URI of the document	3
document.domain	Returns the domain name of the document server	1
document.domConfig	Obsolete. Returns the DOM configuration	3

See the full list at [w3schools.org](https://www.w3schools.org)

DOM Events

There are all kinds of events you can create with elements on your page.

Types of Events

Click	onclick, ondblclick, oncontextmenu, click
Touch	touchstart, touchmove, touchend, touchenter, touchleave
Keyboard	keydown, keypress, keyup
Mouseover	mousedown, mousemove, mouseup, mouseover, mouseout, mousewheel
Resize	resize
Loading	onload, load

Setting an Event Handler

If you want your code to respond to an event, it needs to register an eventhandler.

```
addEventListener(event,fxn);
```

It takes in 2 parameters

1. Name of the event
2. Name of the event to listen for

Syntax:

```
functionToRun() {  
  // code to execute  
}  
element.addEventListener('event name', functionToRun);
```

Example Code: Events and Event Handler

```
<html>
<head></head>
<body>
  <p id="daisies">I <b>love</b> flowers.</p>

  <script type="text/javascript">
    // define a function that can be called
    function doSomething() {
      alert("Especially Daisies!");
    }

    // grab the element you want to attach the event to
    var flowers = document.getElementById("daisies");

    // attach the function for chosen event
    flowers.addEventListener("click", doSomething);
  </script>
</body>
</html>
```

Output:

I **love** flowers.

(When clicked, it will alert)

Especially Daisies!

DOM Nodes

What is a node?

Anything in the HTML is considered a node. All the html is considered a node, the elements on the page is considered a node, text in a paragraph is a node, even a comment is considered a node.

Nodes are seen in a hierarchy with parent/child relationships.

They can be accessed by:

- parentNode
- childNodes[*nodenumber*]
- firstChild
- lastChild
- nextSibling
- previousSibling

Text is not considered inside a paragraph. There is a paragraphnode, that has a textnode inside it that has text in it.

DOM Root Nodes

DOM Root Nodes

You can access the whole document with:

- `document.body` - The body of the document
- `document.documentElement` - The full document

```
<!DOCTYPE html>
<html>
<body>
<p>Learning is fun!</p>
<div>
<p>It is easy to output code from the body of a document.</p>
<p>This example demonstrates the
<b>document.body</b> property.</p>
</div>
<script>
alert(document.body.innerHTML);
</script>
</body>
</html>
```

Output: (It will alert everything between the body tags.)

```
<p>Learning is fun!</p>
<div>
<p>It is easy to output code from the body
of a document.</p>
<p>This example demonstrates the
<b>document.body</b> property.</p>
</div>
<script>
alert(document.body.innerHTML);
</script>
```


Changing CSS with DOM

You can change CSS styling using DOM.

`document.getElementById(id).style.property=new style`

First you call the element by the id.

We are changing the style.

Color is what we are changing

The new color will be blue

```
document.getElementById(id).  
style.  
.color=  
"blue"
```



Code Examples: This code creates a new `<p>` element:

```
var para = document.createElement("p");
```

To add text to the `<p>` element, you must create a text node first. This code creates a text node:

```
var node = document.createTextNode("Hi. This sentence was created with dom manipulation.");
```

Then you must append the text node to the `<p>` element:

```
para.appendChild(node);
```

Finally you must append the new element to an existing element.

This code finds an existing element:

```
var element = document.getElementById("div1");
```

This code appends the new element to the existing element:

```
element.appendChild(para);
```

Below is the source code to add a new paragraph through DOM Manipulation. The first section is the html code. The highlighted javascript is adding to that code.

```
<div id="div1">  
<p id="p1">This paragraph is the first paragraph in a div.>  
<p id="p2">This is another paragraph.</p>  
</div>
```

```
<script>  
var para = document.createElement("p");  
var node = document.createTextNode("This is a new  
paragraph created with dom manipulation .");  
para.appendChild(node);  
  
var element = document.getElementById("div1");  
element.appendChild(para);  
</script>
```

The “p” is the element that is created.

The create text node creates text.

The append child adds the text to the paragraph.

Resources:

W3schools.com

http://www.w3schools.com/js/js_htmlDOM.asp

GitHub

<https://github.com/processing/p5.js/wiki/Intro-to-DOM-manipulation-and-events>

Call Me Nick

<http://callmenick.com/post/basics-javascript-dom-manipulation>

Mozilla Developer Network

https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model
