# Manipulating the DOM

## Updating Content

### InnerHTML

To change content for an element, we can use the InnerHTML.

For example:

document.getElementsByTagName('li')[0].innerHTML = '<a href="http://allrecipes.com/recipe/17500/corn-tortillas/">Corn Tortillas</a>';

Or we can store it as a variable for use later on.

var firstListItem = document.getElementsByTagName('li')[0].innerHTML;

### Text Content

This allows us to get and set the text Content for an element.

document.getElementById('important').textContent = 'Check!';

### InnerHTML Vs Text Content

When we are using Text Content anything we use will be displayed as text.

InnerHTML works like a .html file, where any statement inside will be executed like a HTML code.

## Adding Elements

You can create a new element in DOM also,

document.createElement(‘<tag name>’)

var newListItem = document.createElement('li');

newListItem.textContent = ‘Jalapenos’;

document.getElement**s**ByTagName(‘ul’)[0].appendChild(newListItem);

## Getting and Setting Attributes

We can change the class by using the className property. This will apply the styles in our CSS associated with that particular class.

document.getElementById('important').className = 'complete';

Also we can set and remove attributes by using **setAttribute()** and **removeAttribute();**

document.getElementsByTagName(‘a’)[0].setAttribute(‘href’,’http://newurl.com’)

document.getElementsByTagName(‘a’)[0].removeAttribute(‘id’);

What are attributes?

<img src=”http://image.com” alt text = “image” height = “500”>

alt text and height are attributes of the img tag.

## Exercise Reflections

function runFunction() {

for (let i = 1; i < 4; i++) {

var boxElement = document.createElement('div');

boxElement.className = 'box-'+ i;

document.body.appendChild(boxElement);

}

}

* document.createElement('div') creates a DOM Node. ‘ **NOT** document.body.createElement..
* document.body.appendChild(boxElement); **NOT** document.getElementByTagName…..