**JavaScript Notes**

**The HTML DOM (Document Object Model)**

* When a web page is loaded, the browser creates a **D**ocument **O**bject **M**odel of the page.
* The **HTML DOM** model is constructed as a tree of **Objects**:



* With the object model, JavaScript gets all the power it needs to create dynamic HTML:
  + JavaScript can change all the HTML elements in the page
  + JavaScript can change all the HTML attributes in the page
  + JavaScript can change all the CSS styles in the page
  + JavaScript can remove existing HTML elements and attributes
  + JavaScript can add new HTML elements and attributes
  + JavaScript can react to all existing HTML events in the page
  + JavaScript can create new HTML events in the page
* HTML DOM methods are **actions** you can perform (on HTML Elements).
* HTML DOM properties are **values** (of HTML Elements) that you can set or change.

**Methods:**

* getElementById Method - Access an HTML element is to use the id of the element.
* innerHTML Property - get the content of an element is by using the innerHTML property. The innerHTML property is useful for getting or replacing the content of HTML elements.
* **Finding HTML Elements:**
  + document.getElementById(*id*)
  + document.getElementsByTagName(*name*)
  + document.getElementsByClassName(*name*)
* **Changing HTML Elements:**
  + *element*.innerHTML =  *new html content*
  + *element*.*attribute = new value*
  + *element*.style.*property = new style*
  + *element*.setAttribute*(attribute, value)*
* **Adding & Deleting Elements:**
  + *document.createElement(*element*)*
  + *document.removeChild(*element*)*
  + *document.appendChild(*element*)*
  + *document.replaceChild(*new, old*)*
  + *document.write(*text*)*

***JavaScript HTML DOM Events***

* HTML DOM allows JavaScript to react to HTML events:
* Reacting to Events
  + A JavaScript can be executed when an event occurs, like when a user clicks on an HTML element.
  + To execute code when a user clicks on an element, add JavaScript code to an HTML event attribute:
    - **onclick=*JavaScript***
  + Examples of HTML events:
    - When a user clicks the mouse
    - When a web page has loaded
    - When an image has been loaded
    - When the mouse moves over an element
    - When an input field is changed
    - When an HTML form is submitted
    - When a user strokes a key
    - Example - <button onclick="displayDate()">Try it</button>
  + addEventListener() method
    - document.getElementById("myBtn").addEventListener("click", displayDate);
    - The addEventListener() method attaches an event handler to the specified element.
    - You can add many event handlers to one element.
    - You can add many event handlers of the same type to one element, i.e two "click" events.
    - You can add event listeners to any DOM object not only HTML elements. i.e the window object.
    - The addEventListener() method makes it easier to control how the event reacts to bubbling.
    - When using the addEventListener() method, the JavaScript is separated from the HTML markup, for better readability and allows you to add event listeners even when you do not control the HTML markup.
    - You can easily remove an event listener by using the removeEventListener() method.
  + element.addEventListener("click", function(){ alert("Hello World!"); });