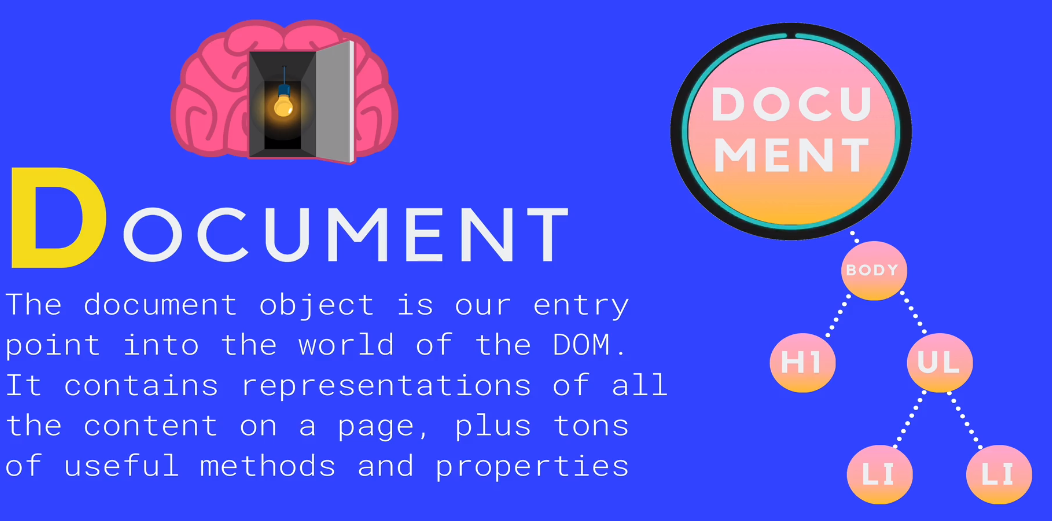
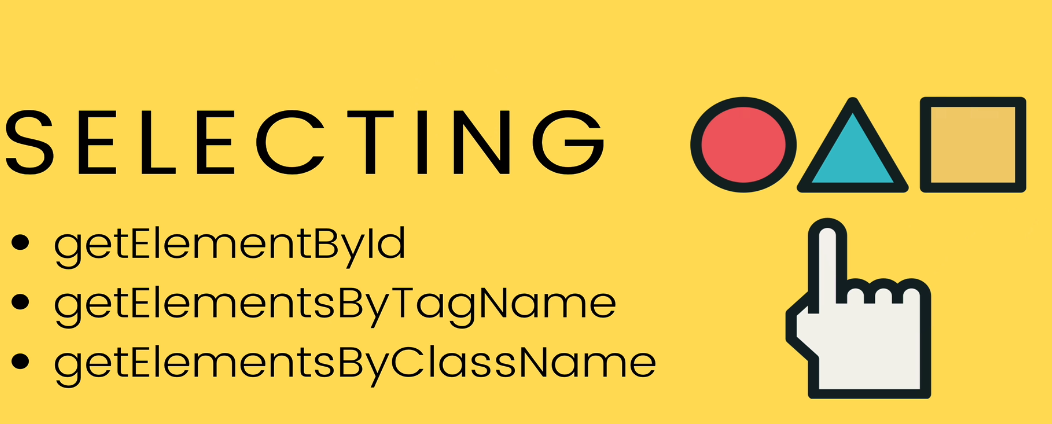
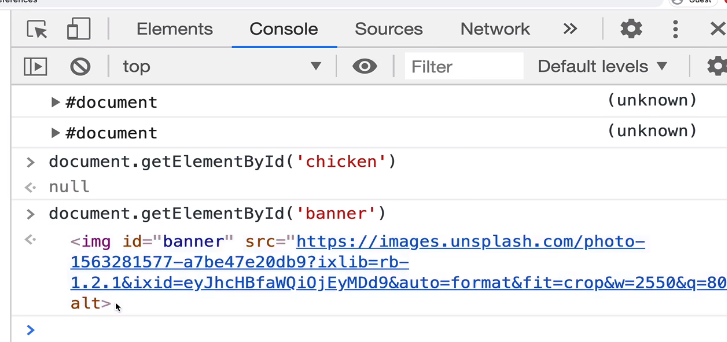
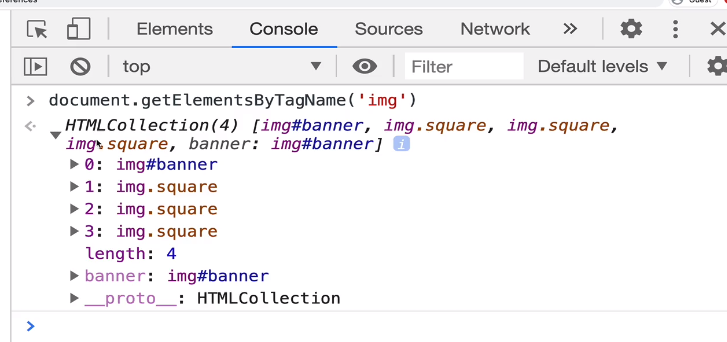
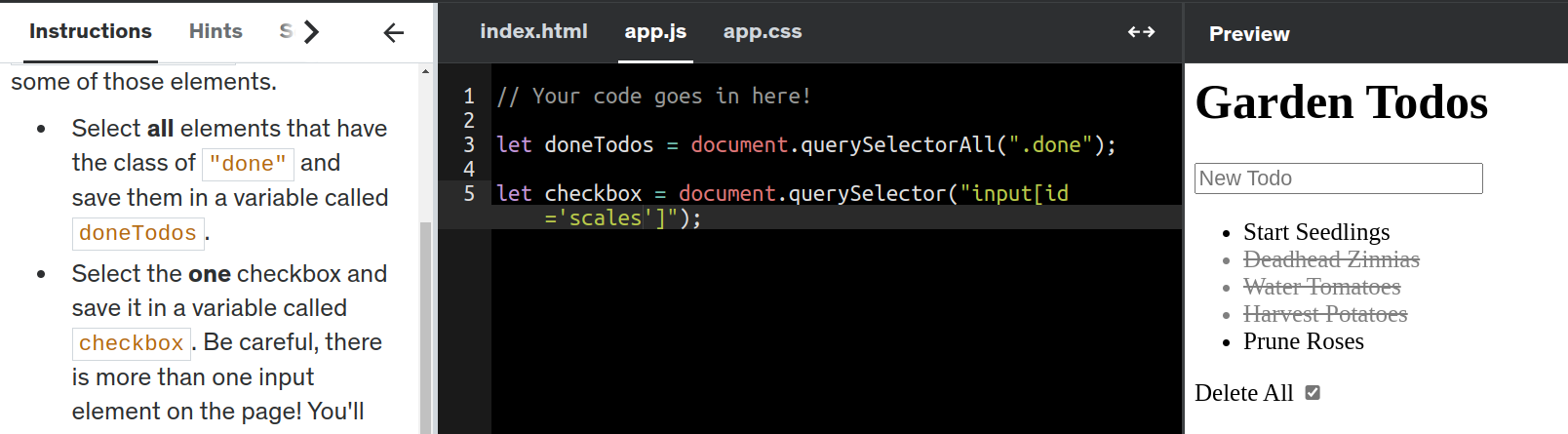
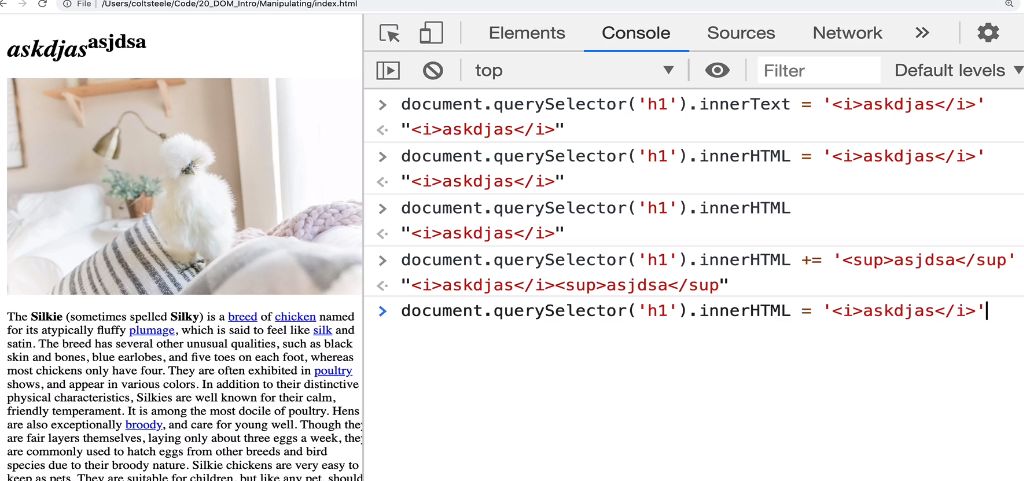
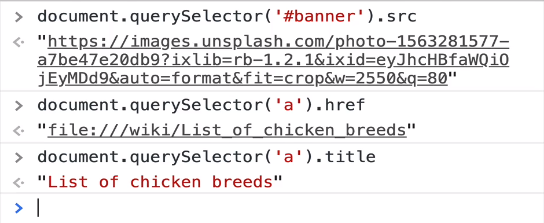
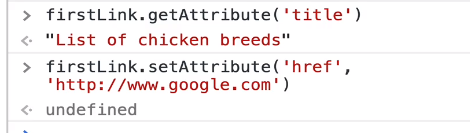
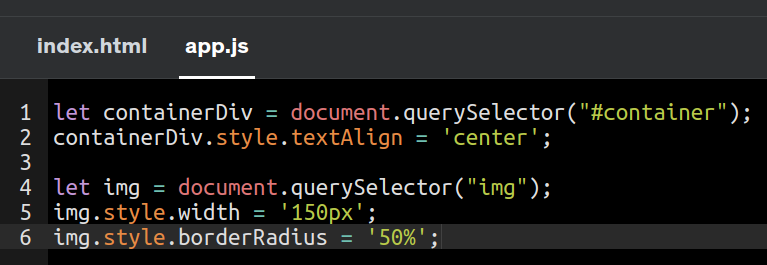
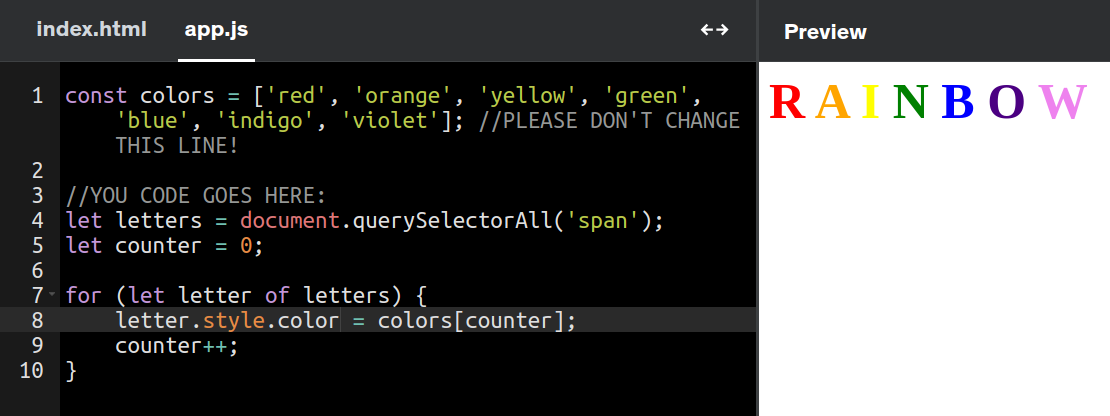
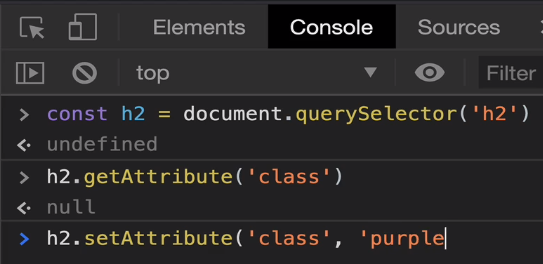
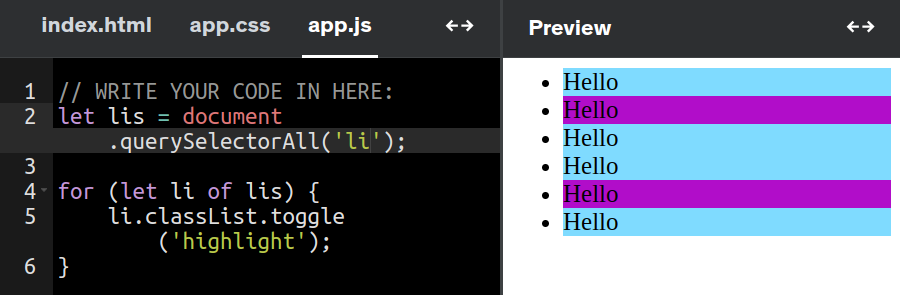
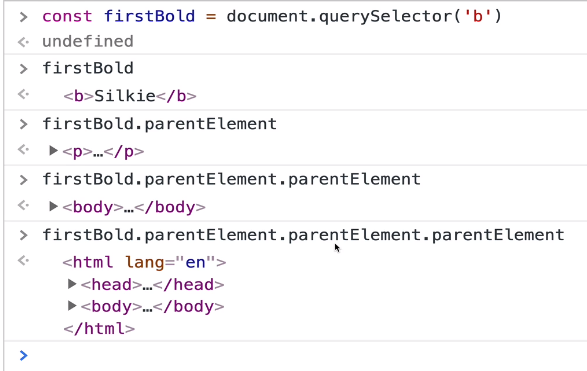
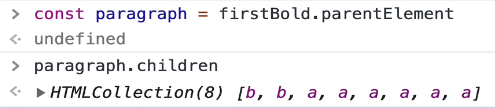
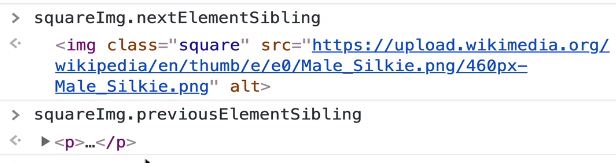
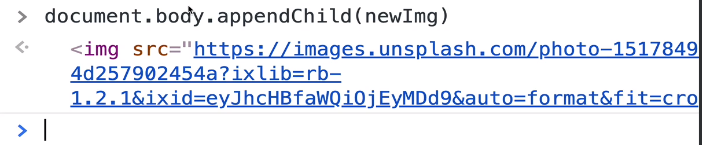
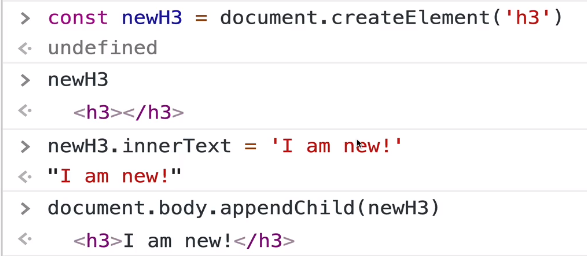
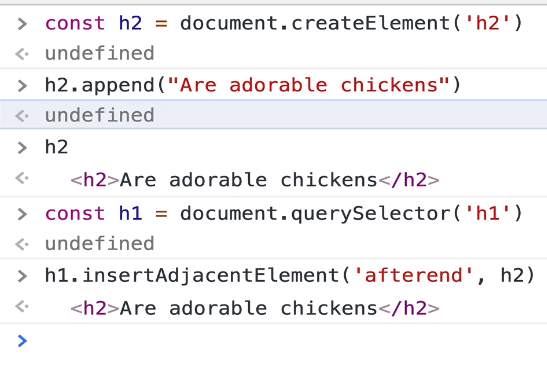
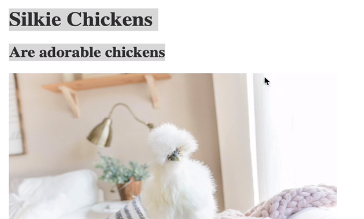
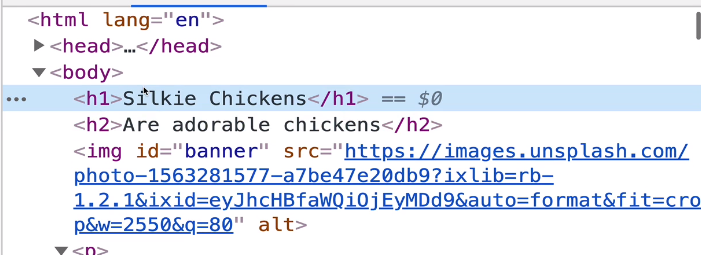
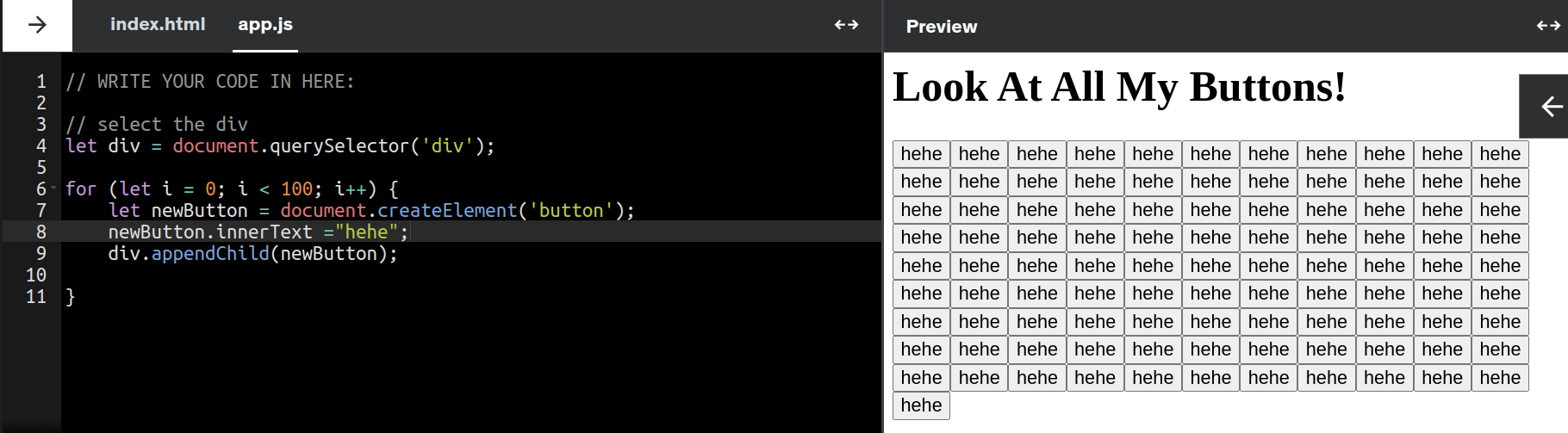
Web Dev Notes

# Section 24 – Intro to DOM

* **Document Object**
  + This represents the page itself. It is a JS object
  + 
  + HTML and CSS components are converted into JS objects
* **SELECTING** methods are used to choose an element in your document
  + 
  + Example of **getElementById**
    - 
  + Example of **getElementByTagName**
    - Rebecca Donaldson
* **SELECTING Modernized:**
  + **QuerySelector()**
    - 
      * Just gives us the first matching element
  + **QuerySelectorAll()**
    - Returns a **collection** of matching elements
      * 
* **DOM MANIPULATION**
  + Methods and Properties of DOM
    - 
    - **Example of innerText, innerHTML and textContent**
      * 
      * let updatedText = document.querySelector("h1 span").innerHTML = "<b>Disgusting<b>"
* **ATTRIBUTES**
  + Example of *src* attribute
    - 
  + you can also use **getAttribute()** or **setAttribute()**
    - 
      * **Example**:
        + let source = document.querySelector('img');
        + source.setAttribute('src', 'https://devsprouthosting.com/images/chicken.jpg');
        + source.setAttribute('alt', 'chicken');
* **CHANGING STYLES**
  + <html tag>.**style**
    - **ex.** h1.style
    - Contains ALL props of CSS that a HTML tag can use
    - This style object DOESN’T contain the styles coming from a css style sheet (**ex**. app.css)
    - Instead, the style object is checking the ‘inline’ style settings in your HTML tag
      * **ex**. <h1 style=”color": blue”>blah</h1>
    - You can also use **window.getComputedStyle(h1)** to get all the css values per tag
    - 
    - 
* **CLASSLIST**
  + **Option 1**: You can retrieve the class property of an element and then set it’s attribute/prop to a specific value
    - ex. 
  + **BETTER option**: Use .classList to retrieve all classes under an element
    - 
      * Using .**add** or .**remove** would change the class value in the tag
      * You can also use .**toggle** as it does those 2 operations
        + 
        + 
* **Traversing the DOM**
  + How to search through elements in the DOM from BOTTOM to TOP )
    - .parentElement
      * 
  + You can also search to see ALL the children underneath an element
    - * .children
      * 
  + You can also search by adjacent sibling (.nextElementSibling)
    - 
* **AppendChild**
  + **Steps**:
    - Create new elements and adding it to other elements
      * const newImg = document.createElement(‘<tag>’)
        + Ex. newImg.src = ‘<image location>’
    - AppendChild to parent tag
      * **Ex**. Append to body tag
        + 
      * **Ex2**. Append a new h3 to body
        + 
* **Append –** Similar, to appendChild but more flexible!
  + ParentNode.append() --> adds to end of node
    - You could append additional text into/inside a <p> tag
      * 
    - You could append multiple tags
  + .**prepend**() --> adds to beginning as first child of element
* .**insertAdjacentElement** --> Insert between elements
  +  **=** 
  + Inserted as next sibling, not as a child
    - 
  + Exercise: Append 100 buttons to div as a child
    - 
* **.removeChild**()
  + Removes the child from an element, not the element itself
    - 
    - 
* .**Remove**() --> will remove the element it is called on
  + 