CSC309 Week 6 Notes

The Virtual DOM:

- We know that the DOM has a tree structure, and that it is somehow painted onto the browser.
- The tree has to be built first, and then properly painted onto the browser and any time we want to edit the DOM, it has to go through a process.
- Using vanilla JS, we changed the DOM manually using the API available through the document object.
- When we do this, there is a process by which the DOM tree structure is altered. Most of the time, this process will be very inefficient. There are smart and efficient ways to do manual DOM manipulation, but this can take time to get right.
- Instead, frameworks like React, Vue, and Angular have an extra layer to their DOM manipulation system.
- They use an in-memory representation of the DOM, known as the **virtual DOM**.
- Changes to the virtual DOM don't need to go through the process of painting on the browser, so it saves resources. This means we can create completely new DOM trees very quickly.
- React will take what it's given from the virtual DOM and then find an efficient way to change the real DOM. When rendering a component, a virtual DOM tree is quickly made in memory. React will then take the Virtual DOM tree and figure out an efficient way of painting the elements in the browser using a reconciliation algorithm.
- The reconciliation algorithm is a smart algorithm for finding differences. It attempts to only change child DOM nodes that are needed. In React, it is helped by things like keys that we add to lists to differentiate the <li> elements.