FULLSTACK OPEN NOTES:

* Core Web Elements:
  + HTML file defines structure and elements.
  + CSS file defines the aesthetic qualities of the HTML elements.
  + Javascript(JS) files can re-render the HTML dynamically on the client browser without having to request a new HTML from the server.
  + Node.js backend technology on the server side.
* HTML:
* CSS:
* React Library:
  + JS library
  + Create a new react app:
    - Npx create-react-app <name of app>
  + When using react all contents to be rendered tend to be defined in react rather than HTML, although this is not a rule.
  + Must have an export default call at the bottom.
    - 
  + console.log() will print to the browser console. Great for debugging.
  + React components must be capitalized.
  + In ES6 functions are assigned to variables:
    - Text

      Description automatically generated with medium confidence
  + Code within curly braces is a reference to a variable.
    - Text

      Description automatically generated
    - A picture containing text

      Description automatically generated
  + The previous pieces of HTML-like code are in fact JSX and not HTML. JSX returned by react components is compiled into JS code under the hood. The above compiles to:
    - Text

      Description automatically generated
  + In JSX all tags must be closed. Ie. <br /> in JSX while in HTML it could be <br> since it is an empty tag (line break).
  + The root react component must wrap what it returns in a root tag, which is generally chosen to be an <div> tag or an empty tag <> as seen below. It can also return them as a list (in brackets []) without the outermost tag but this is seen as ugly and overly complex.
    - *Text

      Description automatically generated*
  + You can use a component inside the JSX returned by another component by using the following syntax: (Hello is its own JS function return JSX markup)
    - Text

      Description automatically generated with low confidence
  + Props:
    - Used to pass data to components.
    - It’s a catch all “class instance”-like argument for a component which can have many class parameters within it.
    - Text

      Description automatically generated
  + Helper Functions:
    - Can have internal functions within components defined with the same syntax as a normal component.
    - These internal helper functions have access to the same variables as its parent function.
    - Text

      Description automatically generated
  + Destructuring:
    - The props passed to a component can be split into its member variables. This can be done individually or in a single line using the following syntax.
    - 
    - It can even be done in the component declaration as follows:
    - 
  + Re-Rendering:
    - To update an element after changes have been made to its props call the render method again as follows.
    - Text

      Description automatically generated
    - We can add a delay by using the setInterval method:
    - Text

      Description automatically generated
    - Re-rendering is better performed using stateful components than calling render directly.
  + Stateful Components:
    - Needs to import the ‘useState’ function from react.
    - Text

      Description automatically generated
    - Calls the useState function with an initial value and deconstructs its output into two variables:
    - 
* JS and Node.js
  + JS runtime environment based on Chrome’s V8 JS engine.
  + Files are run by the command:
    - *Node <filename>.js*
  + Variables:
    - Const-> constant variable
    - Let or var-> normal variable
    - Type can change, it is not fixed.
    - *Text

      Description automatically generated with medium confidence*
  + Arrays;
    - Defined with brackets.
    - Even if defined as const its elements can change.
    - **
  + Objects:
    - Object literals:
      * *Text

        Description automatically generated*
      * Can be referenced using the dot notation or brackets:
      * *A screenshot of a computer

        Description automatically generated with medium confidence*
      * You can add properties to an object on the fly using the same notation.
      * Can also have methods declared. **Methods can use the *this* keyword, but it behaves differently when the function is assigned as a reference to another variable so it is recommended to avoid it**.
        + *Text

          Description automatically generated with medium confidence*
  + Functions:
    - Arrow functions
      * See above in React section.
      * *Text

        Description automatically generated*
      * Or (if it’s a one liner)
      * **
    - Function notation:
      * *Graphical user interface

        Description automatically generated*
  + Classes:
    - JS has classes as of the latest version ES6 but they are not used very often.