

Week 5 Day 2

React Continued



React: Function Components

```
export const Courses:React.FC = () => {

  const [courses, setCourses] = useState<Course[]>([]);

  useEffect(() => {
    setCourses([
      {
        courseId: 1,
        subject: "Reading",
        courseNumber: 100,
        courseName: "Intro to Reading",
        teacher: "NA"
      }
    ]);
    console.log(courses);
  }, [courses.length]);

  const addCourse = (course:Course) => {
    setCourses([...courses, course]);
  }

  return (
    <div className="courses">
      <NewCourse newCourse={addCourse} currentId={courses.length}></NewCourse>
    </div>
  )
}
```

- Used to store and mutate state in a function component
 - Declare the state field in mutator with the useState hook
 - `const [state, setState] = useState()`
- useState hook allows for any type of state data

- Used to perform logic after event/changes to the component
 - Takes in a callback function which specifies the logic
 - Tell it to watch for changes to different states using the dependency array
 - Perform logic “once” at the beginning with an empty array
- `useEffect(callback, [dependencies])`

- useContext
 - Used to access values stored in the ContextAPI
- useReducer
 - Advanced component state manager
 - Stores state value, and reducer functions
 - Reducer functions describe different ways the state should be modified



Create First Function Component



- Register the handler on the elements themselves
- JSX uses case sensitive camelCase for registering event handlers
 - onClick, onChange, ...
- Uses curly braces to pass a reference to the actual handler function
- Like the virtual DOM used, React uses Synthetic events

React: Conditional Rendering and Rendering List Elements

- React allows you to conditionally render with if/else and ternary
 - Depending on the expression, something different will be rendered
- Using data binding, react allows you to render multiple components from a list
 - Typically done with `array.map()`
 - Must provide a unique key for each list entry
 - Not recommended to use the index for the key



Events, Conditionals, and Mapping



- Gives users experience of navigating to different pages
- React Router is the most basic way of adding routing
- React Router DOM provides expanded functionality on React Router
 - `npm i react-router-dom`

- React Router Dom provides the popular `BrowserRouter`
 - Uses the HTML history API to keep the UI in sync with the page URL

INSERT SCREENSHOT HERE



Add Navigation to our App



- Make API calls with Axios
 - `npm i axios`
- Axios is a promised based HTTP client
 - Supported by MOST browsers
 - Intercepts and transfers requests and response data
 - Automatically parses JSON
 - Request cancelation
 - Better error handling
 - Built in cross site request forgery protection

- Fetch is built into the browser vs Axios was built to work better on the server side
- Axios automatically stringifies and parses JSON
- Fetch cannot automatically timeout or interrupt HTTP requests



Make API calls with React



- Some applications need to share state across multiple components
- Possible solutions to this include
 1. Lifting State
 2. React Context API
 3. Flux Design Pattern/Redux

- The simplest way to combat one way data flow
 - However, it can become complicated and hard to follow as it is scaled up
- Done by passing a function to a child as a prop
 - When the child wants to update the state of the parent, it calls the method passed as a prop
 - That method then modifies the state of the parent component

- ContextAPI is Reacts built in global state management system
 - Helps alleviate state lifting and prop drilling
- “Context provides a way to pass data through the component tree without having to pass props down manually at every level” – React Docs

- Setup a global “context”/state
 - Setup the context with the `React.createContext` method
- Share that context by wrapping all components that need that state in a provider
- Consume the context inside of components with the `useContext()` hook
- Context API is great for small scale applications with small amounts of global state



React Context API

