**REACT-REDUX:**

**PROVIDER**

\_Has Provider component and the connect() method

Q: What does the connect() method do?

A: Allows me to specify which components should receive which data from the store

Q: What does Provider do?

A: Makes connect() work properly

+) wrap the entire app so that all the subcomponents who need to can receive data rom the store directly

\_ Add import { Provider } from react-redux to wrap our main root component inside of provider

Q: What does Provider do?

A:

+) Provider can store the store so all components can have access to the store. It passes data through the component tree without having to pass the props down manually at every level

++) Also takes store as a prop, then sets the store context, passing it down to all subcomponents

Q: What’s currying?

A: a dynamic technique that lets you give some data to a function but wait until some later point to provide that function everything it needs

Ex:

Instead of writing function like this:

function plate(vegetables, fruit) {

return `I ate a plate of ${vegetables} and ${fruit}!`;

}

plate('corn', 'apples');

We can also write it like this so that the plate can get the fruit later:

function plate(vegetables) {

return function fruitFunc (fruit) {

return `I ate a plate of ${vegetables} and ${fruit}!`;

}

}

const sentence = plate('corn')('apples');

// ` `, not “ “

ALSO CALLED A **PARTIAL APPLIED FUNCTION**

**ANOTHER EXAMPLE:**

function iceCreamOrder(name) {

return function flavorPicker (flavor) {

return function scoops (numScoops) {

return `${name} ordered ${numScoops} scoops of ${flavor} ice cream!`;

};

};

}

const func = iceCreamOrder('Bryan')('jasmine')(5)

The general rule is that number of functions returned is one less than the number of functions called. For example, if you see three function calls, you need to return two functions.

**Another one:**

/\* Write a Curried Function

\*

\* Create a function called "houseBuilder" that should:

\* - Accept the number of stories (floors)

\* - Return a function

\*

\* The returned function should:

\* - Accept the color of the house

\* - Return a string in the following format:

\* "building a <numOfStories>-story, <color> house"

\*

\* Example:

\* const response = houseBuilder(3)('blue');

\* console.log(response); // building a 3-story, blue house

\*/

function houseBuilder(floors){

return function Color(color){

return `building a ${floors}-story, ${color} house`

}

}

**CONNECT():**

\_returns a curried function

\_We can pass specific parts of our stored state as well as any action dispatches to our component to access as props