MODULE: 7 (React – Applying Redux)

• What is Redux?

Redux is an open-source JavaScript library used to manage application state. React uses Redux for building the user interface.

• What is Redux Thunk used for?

Thunk is a logical concept in programming where you deal with a function that is primarily used to delay the calculation or evaluation of any operation.

Redux Thunk acts as a middleware that will return you a function instead of an object while calling through the action creators. The returned function receives the dispatch method from the store and then later it is used to dispatch synchronously inside the body of function once the asynchronous actions have been completed. Let's move ahead with Redux Thunk and talk more about its installation and usage.

• What is Pure Component? When to use Pure Component over Component?

PureComponent is similar to Component but it skips re-renders for same props and state. Class components are still supported by React, but we don't recommend using them in new code.

Usage

Skipping unnecessary re-renders for class components

React normally re-renders a component whenever its parent re-renders. As an optimization, you can create a component that React will not re-render when its parent re-renders so long as its new props and state are the same as the old props and state. Class components can opt into this behavior by extending ***PureComponent***:

• What is the second argument that can optionally be passed tosetState and what is its purpose?

The second argument that can optionally be passed to setState is a **callback function** which gets called immediately after the setState is completed and the components get re-rendered.
If you want your program to update the value of a state using setState and then perform certain actions on the updated value of state then you must specify those actions in a function which should be the second argument of the setState. If we would not do so then those actions will be performed on the previous value of state because of **asynchronous** nature of setState.