**REACT SCHEME TUTORIAL REPORT**

FINDINGS

* React is a JavaScript library used for building user interfaces. It uses component-based structure that may have its own logic and template, where you can create reusable UI components. These components are like building blocks you can combine to create whatever complex interface you desire.
* Functions in React start with capital letters.
* In React, I will work with JSX, which is a syntax extension that allows me to write in HTML like code within my JavaScript files.
* React has a concept called “props”. They are like JavaScript objects that makes code reusable. Props are used to pass data from a parent component to its child component. It enables customization of components based on the data they receive.
* {} are used to take in dynamic values.
* To manage my components, React provides some hook features.
* The useState hook allows me to add state to my functional components. With useState, I can declare a state variable and a function to update that variable. This helps me manage and update the state within my component.
* The useEffect hook is used to control what happens depending on the stage of life cycle my components are. It is used to perform side effects in my components. Side effects can include things like fetching data from an API, subscribing to events, or manipulating the DOM.
* The empty dependency array is used to prevent display while updating.
* Fetching data. Preferably Axios.

SETBACKS

* Couldn’t create scenarios properly in my head to implement some findings and was not able to solve most of the YouTube exercises on my own.
* useContext hook. Proper State management and interconnecting multi pages.
* React Router DOM
* Have not completed the weekend part of the scheme.

OVERALL

* Interesting usage and fascinating flexibility, I believe with time I’d be able to utilize its powerful functionalities properly.

10-03-2024

Wisdom Lotachukwu Ngaloru.