Saturday, January 14, 2023 11:41 AM

Good Morning :))

Promises in Javascript -----

1. It is an object that Promises to run the code in the executor passed to it
2. In this process the code may succeed or fail or be in executing state(PENDING)
3. The promise has two functions

    then = this function registers a callback that will be called ONLY when promise resolves(succeeds)

    catch = this function registers a callback that will be called only when promise is rejected(FAILS)

4. A separate queue is maintained for promises and simple callback functions

    when the main stack becomes empty then the promises queue is executed followed by callback queue

5. We can return a new Promise in **"then"** of first Promise

    We can have a cascading syntax

    new Promise(callback) .then(callback).then(callback).then(…..)

____

Http requests through React -------------------

We can do it in many ways ----

1. XmlHttpRequest
2. fetch API
3. Using axios library
    i. Step 1 = run npm install axios //from the firsapp folder on the terminal

Ex 1---- Accept a name and a job from the User and post it as a JSON to the reqres url

Show the result

Ex2 ---- Write a JSON file that contains the ITEM information that is hardcoded currently in ShoppingCart available_items .

Put the file in the public folder (so that the file is sent to the browser)

Write axios.get(to get the JSON file content)

Use the reponse of this query to set the available_items array

Where should we call axios.get () ? As soon as the page is mounted this should be fetched and the dropdown

Should be populated ?

If class component ---- write in componentDidMount()

If function component ---- write in useEffect(callback ,[])

____

Routing in React -----------------

1. In SPA applications all components are present on the browser
2. WE cant show all components at a time
3. To get a Menu effect where components are rendered only when clicked we will use Routing
4. We can see multiple Route Links on the screen

    These Links are different from traditional anchor tag <a href="" >

The anchor tag refreshes or reloads the page from the server ----whereas Route Links don’t go to the server
They simply render available components on browser outlet

5. Steps for routing ---
    1. npm install react-router-dom } on cmd @ the project folder (firstapp)
    2. Wrap our App component in <BrowserRouter> <App2 /> </BrowserRouter >
    3. Define Routes in the App using <Routes> <Route /> <Route /> <Route /> <Route /> </Routes>

       path is mapped to the component
    4. Use the Route Links and Outlet tag ----------- on the Landing Page

Pass parameter to the Route -------------

Step1 ----- In the Route path mention the placeholder for the route parameter using **:**
<Route path="/url/:route-parameter-name"

Step2 ----- In the <Link to={/url/pass-the-value} tables </Link>

Step3 ----In the component mapped to this path
Use a hook ------- somemap = useParams()
This will return the MAP of all the route params passed to this url
Let value = somemap.route-parameter-name
Use the value

Ex ---- Show all the item_names from shopstock.json in a table with update link beside it
When the update link is clicked the component should display the content of the selected item
Id item_name and unit_cost In text fields