Lecture 5 – Lets get hooked – Theory

Noted. Here are the points you mentioned:

1. **Best Practice in Industry**: It's a common practice to separate files for separate apps. This helps in maintaining the codebase and makes it easier to manage as the project grows.
2. **Current File Structure**: Currently, all files are at the root level.
3. **React Project Structure**: The main code for the React project is in the `src` folder.

First we put every component in component folder.

Once you do this you server will thouwo error that , for example if you put header inside component it will throw header not defined error.

So you need to import header.js from the components but before that inside header.js you need to export the component only then you can impoort it.

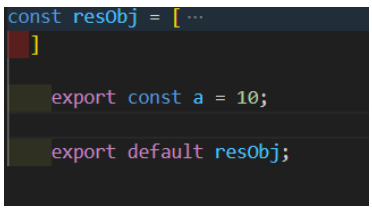
4. **Export Default**: In JavaScript, when you use `export default` in a module, you are indicating that this is the primary thing you want to export from that module. The `default` export is used when you want to provide a single, "default" value, function, or object that can be easily imported without the need for curly braces {} during the import.
5. **Importing the Default Export**: After using `export default` in a module (for example, `Header`), you can import it in another module (like `App.js`) using the `import` statement.

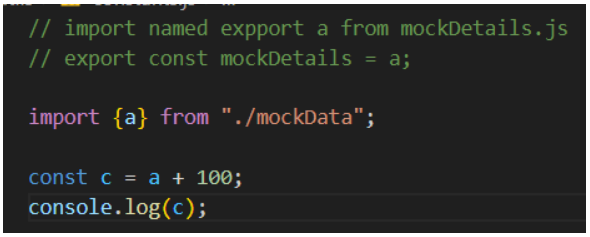
There are basically 2 types of export.

Default export and named export.

We cant use 2 default export in a single file.

We can use named export by simply putting export behind anything we want to return.





We need to call named export with {} while importing

Suppose if we want return multiple things in a single file then we use named export.

----------par1 done----------

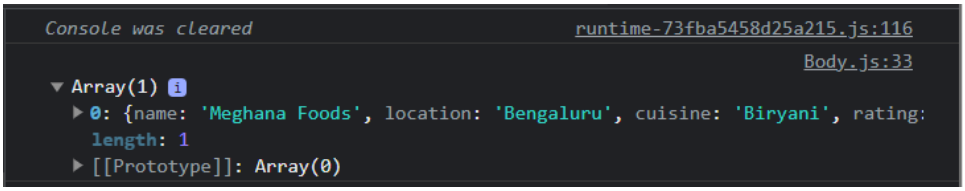
Now we will make our website dynamic.

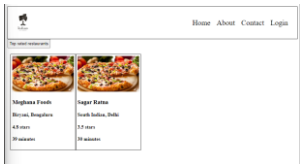
Lets create a button and if we click on it it should show top rated restaurant.

```
return (
  <div className="body">
    <div className="filter">
      <button className="filter-btn" onClick={()=>{
        // i want to write a filter logic to get the top rated res
        listOfRestaurants = listOfRestaurants.filter((ele)=>{
          return ele.rating > 4.0;
        }
    );
```

As we can see I have created an onclick event that gives me only those restaurants that are on rating aove 4.0.

But the sme this is not reflecting in ui ??





We see in console only one obj present but ui is not updating. To do that we have to manually write the logic write. This is where react plays its card.

The above thing was done with normal js variables . It is not in sync with data.

We are trying to achive something that is in sync with ui and data layers. Here react hooks come into picture.

Lets create a superpower variable.

What are hooks? It's a normal js function but It comes with some superpowers given by react.

It's a utility function.

All these uility functons are written inside react code which has been downloaded in node modules.

2 important hooks.

UseState() - use to generate superpowerful state variables in react. Need to import useState from react.(named import)

It maintains the state of the component.

I am creating a local state variable.

```
const [listOfRestaurants] = useState([] , )
```

This is it I created a state variable and passed empty array as an initiAL value. Nut I have to import it in the file.

Remember to create it inside an array.

```
const [listOfRestaurants , setListofRestaurant] = useState([{
    "name": "Meghana Foods",
    "location": "Bengaluru",
    "cuisine": "Biryani",
    "rating": 4.8,
    "deliveryTime": 39,
  }] , )
```

As we can see it takes one more parameter , called “set method”. That’s why we put it inside array

The `useState` hook in React is a function that allows you to add state to functional components. It takes an initial state as an argument and returns an array of two elements:

6. The current state value.

7. A function to update the state.

```
<div className="body">
  <div className="filter">
      <button className="filter-btn" onClick={()=>{
          // local state variable with use state

          // i want to write a filter logic to get the top rated res
          filteredRestaurant = listOfRestaurants.filter((ele)=>{
              return ele.rating > 4.0;
          }
  );
  setListofRestaurant(filteredRestaurant)
```

Here look closely I have created a filterRestaurant variale and passed it inside set method so that it will automatically update my listofrestaurans and update the ui while rendering.

WHENEVER A STATE VARIABLE CHANGES REACT WILL RERENDER THE COMPONENT.

React uses something called as reconciliation algorithm. And this is also known as react fiber

We will take res coard example.

Previously it contains 4 res cards , on clicking the button we care updating only those with 4.0 stars or above and it will show only 2 cards.

How is this happening.

We already kmow that the react elements are kind of js objects.

So what react does is it checks the difference between 2 objects that are virtual doms.

It check whats the difference between them and then it actually updates the actual dom so its way faster than comaring the actual html and updating.

Whenevr a change in state variable , react will check the change in virtual doms and updates the actual dom.

This is done by react reconsilation algorithm or recently temed as react fiber.