**WEEK 6**

**React**

1. Create a new React Application with the name “myfirstreact”, Run the application to print “welcome to the first session of React” as heading of that page.

A:-

**Step – 1:- install nodejs and npm** [**https://nodejs.org/en/download/**](https://nodejs.org/en/download/)

**Step – 2:- Install Create React App Run this command**

npm install -g create-react-app

**Step-3:- Create a New React App**

npx create-react-app myfirstreact

**Step – 4:- Navigate to the Project Folder**

cd myfirstreact

**Step – 5:- Open in Visual Studio Code**

code .

**Step – 6:- In app.js file paste the below code.**

*import React from 'react';*

*function App() {*

*return (*

*<div>*

*<h1>Welcome to the first session of React</h1>*

*</div>*

*);*

*}*

*export default App;*

**Step 9: Run the React App**

npm start

**Output :- A screenshot of a computer

AI-generated content may be incorrect.**

1. Create a react app for Student Management Portal named StudentApp and create a component named Home which will display the Message “Welcome to the Home page of Student Management Portal”. Create another component named About and display the Message “Welcome to the About page of the Student Management Portal”. Create a third component named Contact and display the Message “Welcome to the Contact page of the Student Management Portal”. Call all the three components.

A:-

**Step 1: Create React App Named StudentApp**

npx create-react-app StudentApp

**Step 2: Open Project in VS Code**

cd StudentApp

code .

**Step 3: Create Components Folder**

Inside VS Code:

* Go to the src folder.
* Right-click and choose **New Folder** → name it Components.

**Step 4: Create Home.js File**

Inside the Components folder:

* Right-click → **New File** → name it Home.js
* Paste this code:

import React from 'react';

function Home() {

return (

<div>

<h1>Welcome to the Home page of Student Management Portal</h1>

</div>

);

}

**Step 5: Create About.js File**

* In the same Components folder → Right-click → New File → name it About.js
* Paste this:

import React from 'react';

function About() {

return (

<div>

<h1>Welcome to the About page of the Student Management Portal</h1>

</div>

);

}

export default About;

**Step 6: Create Contact.js File**

* Right-click → New File → name it Contact.js
* Paste this:

import React from 'react';

function Contact() {

return (

<div>

<h1>Welcome to the Contact page of the Student Management Portal</h1>

</div>

);

}

export default Contact;

**Step 7: Edit App.js**

Go to src/App.js, replace everything with:

import React from 'react';

import Home from './Components/Home';

import About from './Components/About';

import Contact from './Components/Contact';

function App() {

return (

<div>

<Home />

<About />

<Contact />

</div>

);

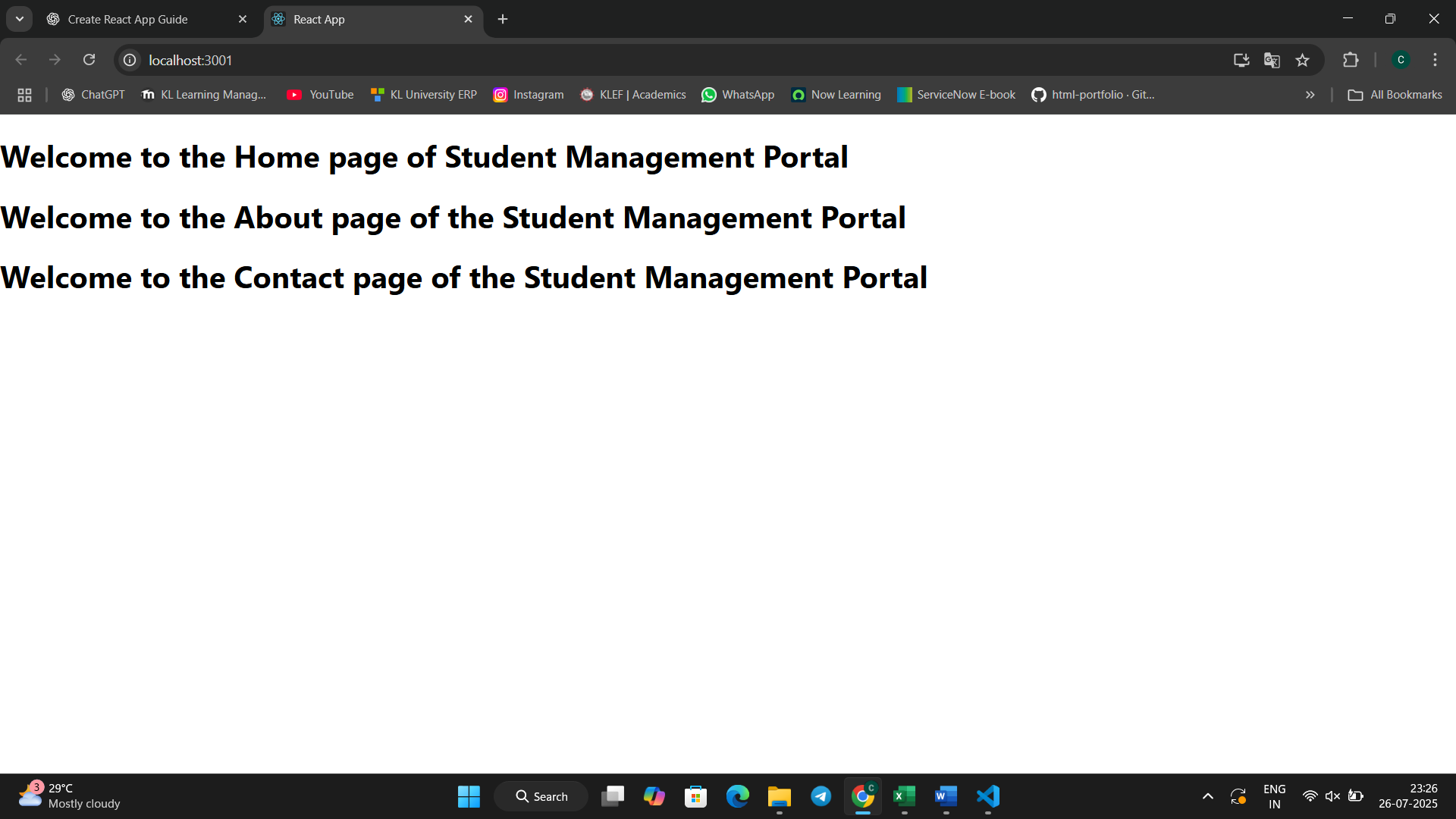
}

export default App;

**Step 8: Run the React App**

In VS Code terminal (or CMD), make sure you're inside StudentApp folder:

npm start

**Output:- **

1. Create a react app for Student Management Portal named scorecalculatorapp and create a function component named “CalculateScore” which will accept Name, School, Total and goal in order to calculate the average score of a student and display the same.

**A:-**

**Step 1: Create React App**

npx create-react-app scorecalculatorapp

**Step 2: Open the Project in VS Code**

cd scorecalculatorapp

code .

**Step 3: Create Components Folder and CalculateScore.js**

1. Go to src folder in VS Code.
2. Right-click → New Folder → name it Components.
3. Right-click on Components → New File → name it CalculateScore.js.
4. Paste the following code:

import React, { useState } from 'react';

import '../Stylesheets/mystyle.css';

function CalculateScore() {

const [score, setScore] = useState(0);

const handleAddScore = () => {

setScore(score + 1);

};

const handleSubtractScore = () => {

setScore(score - 1);

};

return (

<div className="score-container">

<h2>Score Calculator</h2>

<p className="score">Current Score: {score}</p>

<button onClick={handleAddScore}>Add</button>

<button onClick={handleSubtractScore}>Subtract</button>

</div>

);

}

export default CalculateScore;

**Step 4: Create Stylesheets Folder and mystyle.css**

1. Go to src folder.
2. Right-click → New Folder → name it Stylesheets.
3. Right-click on Stylesheets → New File → name it mystyle.css.
4. Paste the following CSS:

.score-container {

text-align: center;

margin-top: 50px;

}

.score {

font-size: 24px;

font-weight: bold;

color: green;

}

button {

margin: 10px;

padding: 10px 20px;

font-size: 16px;

}

import React from 'react';

import CalculateScore from './Components/CalculateScore';

function App() {

return (

<div className="App">

<CalculateScore />

</div>

);

}

export default App;

**✅ Step 8: Run the App**

npm start

Output:- A screenshot of a computer

AI-generated content may be incorrect.

4. Implement componentDidMount() hook. Implementing componentDidCatch() life cycle hook.

**Step – 1:- Create the React App**

npx create-react-app blogapp

cd blogapp

code .

**Step – 2:- Create Post.js (Model Component)**

Inside src/, create a file named Post.js and paste:

import React from 'react';

class Post extends React.Component {

render() {

return (

<div>

<h2>{this.props.title}</h2>

<p>{this.props.body}</p>

</div>

);

}

}

export default Post;

**Step – 4:- Create Posts.js (Main Logic Component)**

import React from 'react';

import Post from './Post';

class Posts extends React.Component {

constructor(props) {

super(props);

this.state = {

posts: [],

hasError: false

};

}

loadPosts() {

fetch("https://jsonplaceholder.typicode.com/posts")

.then(response => response.json())

.then(data => this.setState({ posts: data }))

.catch(error => this.setState({ hasError: true }));

}

componentDidMount() {

this.loadPosts();

}

componentDidCatch(error, info) {

alert("An error occurred while rendering the component.");

console.error("Error:", error, info);

}

render() {

return (

<div>

<h1>Blog Posts</h1>

{

this.state.posts.map(post => (

<Post key={post.id} title={post.title} body={post.body} />

))

}

</div>

);

}

}

export default Posts;

**✅ 5. Modify App.js**

Replace contents of src/App.js with:

import React from 'react';

import Posts from './Posts';

function App() {

return (

<div className="App">

<Posts />

</div>

);

}

export default App;

**✅ 6. Run the Application**

In the terminal (inside the blogapp folder), run:

npm start

**Output:- A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.**

**5 .** My Academy team at Cognizant want to create a dashboard containing the details of ongoing and completed cohorts. A react application is created which displays the detail of the cohorts using react component. You are assigned the task of styling these react components

**A:**- **Step - 1. Create React App**

Open **Command Prompt** and run:

npx create-react-app cohortdashboard

cd cohortdashboard

code .

**Step - 2:- Create CohortDetails Component**

Inside src/, create a new file:  
**CohortDetails.js**

import React from 'react';

import styles from './CohortDetails.module.css';

function CohortDetails({ name, mentor, status }) {

const headingStyle = {

color: status === 'ongoing' ? 'green' : 'blue'

};

return (

<div className={styles.box}>

<h3 style={headingStyle}>Cohort: {name}</h3>

<dl>

<dt>Mentor:</dt>

<dd>{mentor}</dd>

<dt>Status:</dt>

<dd>{status}</dd>

</dl>

</div>

);

}

export default CohortDetails;

**Step - 3. Create CSS Module**

📁 In src/, create a new file:  
**CohortDetails.module.css**

.box {

width: 300px;

display: inline-block;

margin: 10px;

padding: 10px 20px;

border: 1px solid black;

border-radius: 10px;

}

dt {

font-weight: 500;

}

**Step – 4:- Modify App.js to Use the Component**

In src/App.js, replace everything with:

import React from 'react';

import CohortDetails from './CohortDetails';

function App() {

return (

<div>

<CohortDetails name="React Basics" mentor="Alice" status="ongoing" />

<CohortDetails name="Node Advanced" mentor="Bob" status="completed" />

</div>

);

}

export default App;

**Step - 5. Run the Application**

npm start

Output:- A screenshot of a computer

AI-generated content may be incorrect.