**CODE**

**public/index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<meta name="theme-color" content="#000000" />

<meta

name="description"

content="Web site created using create-react-app"

/>

<link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />

<!--

manifest.json provides metadata used when your web app is installed on a

user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/

-->

<link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />

<!--

Notice the use of %PUBLIC\_URL% in the tags above.

It will be replaced with the URL of the `public` folder during the build.

Only files inside the `public` folder can be referenced from the HTML.

Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC\_URL%/favicon.ico" will

work correctly both with client-side routing and a non-root public URL.

Learn how to configure a non-root public URL by running `npm run build`.

-->

<title>React App</title>

</head>

<body>

<noscript>You need to enable JavaScript to run this app.</noscript>

<div id="root"></div>

<!--

This HTML file is a template.

If you open it directly in the browser, you will see an empty page.

You can add webfonts, meta tags, or analytics to this file.

The build step will place the bundled scripts into the <body> tag.

To begin the development, run `npm start` or `yarn start`.

To create a production bundle, use `npm run build` or `yarn build`.

-->

</body>

</html>

**src/index.js**

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import App from './App';

import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(

<React.StrictMode>

<App />

</React.StrictMode>

);

// If you want to start measuring performance in your app, pass a function

// to log results (for example: reportWebVitals(console.log))

// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals

reportWebVitals();

**src/App.js**

import React from 'react';

import CohortDetails from './components/CohortDetails';

function App() {

// Data based on the final result image [cite: 38]

const cohorts = [

{

name: 'INTADMF10 - .NET FSD',

startedOn: '22-Feb-2022',

status: 'Scheduled',

coach: 'Aathma',

trainer: 'Jojo Jose',

},

{

name: 'ADM21JF014 - Java FSD',

startedOn: '10-Sep-2021',

status: 'Ongoing',

coach: 'Apoorv',

trainer: 'Elisa Smith',

},

{

name: 'CDBJF21025 - Java FSD',

startedOn: '24-Dec-2021',

status: 'Ongoing',

coach: 'Aathma',

trainer: 'John Doe',

},

];

return (

<div style={{ padding: '20px' }}>

<h1>Cohorts Details</h1>

<hr />

<div>

{cohorts.map((cohort) => (

<CohortDetails key={cohort.name} cohort={cohort} />

))}

</div>

</div>

);

}

export default App;

**src/components/CohortDetails.module.css**

/\*

CSS class with the name "box" [cite: 25]

- Width = 300px [cite: 26]

- Display = inline block [cite: 27]

- Overall 10px margin [cite: 28]

- Top and bottom padding as 10px [cite: 29]

- Left and right padding as 20px [cite: 30]

- 1 px border in black color [cite: 31]

- A border radius of 10px [cite: 32]

\*/

.box {

width: 300px;

display: inline-block;

margin: 10px;

padding: 10px 20px;

border: 1px solid black;

border-radius: 10px;

font-family: sans-serif;

}

/\* CSS style for html <dt> element [cite: 33]

- Set the font weight to 500 [cite: 33]

\*/

.dt {

font-weight: 500;

}

**src/components/CohortDetails.js**

import React from 'react';

import styles from './CohortDetails.module.css'; // Import the CSS Module [cite: 34]

function CohortDetails({ cohort }) {

// Define the style for <h3> based on cohort status [cite: 36]

const headerStyle = {

color: cohort.status === 'Ongoing' ? 'green' : 'blue',

};

return (

// Apply the box class to the container div [cite: 35]

<div className={styles.box}>

<h3 style={headerStyle}>

{cohort.name}

</h3>

<dl>

<dt>Started On</dt>

<dd>{cohort.startedOn}</dd>

<dt>Current Status</dt>

<dd>{cohort.status}</dd>

<dt>Coach</dt>

<dd>{cohort.coach}</dd>

<dt>Trainer</dt>

<dd>{cohort.trainer}</dd>

</dl>

</div>

);

}

export default CohortDetails;

**OUTPUT**

