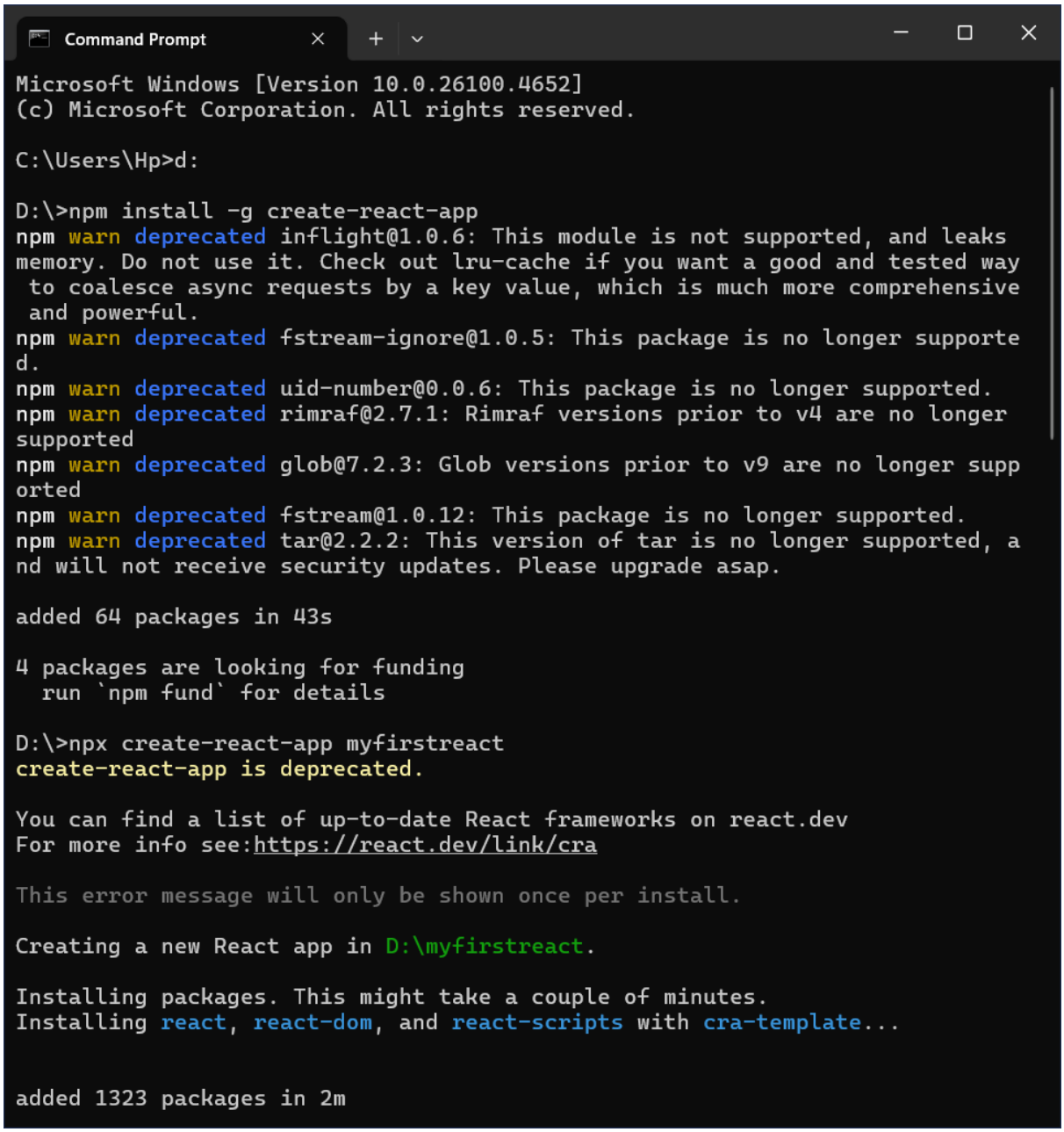
1. ReactJS-Handson

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.



```
Success! Created myfirstreact at D:\myfirstreact
Inside that directory, you can run several commands:

  npm start
    Starts the development server.

  npm run build
    Bundles the app into static files for production.

  npm test
    Starts the test runner.

  npm run eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd myfirstreact
  npm start

Happy hacking!

D:\>cd myfirstreact
```

App.js

src > App.js > ...

```
1  import logo from './logo.svg';
2  import './App.css';
3
4  function App() {
5    return (
6      <h1>Welcome the first session of React</h1>
7    );
8  }
9
10 export default App;
11
```

```
D:\myfirstreact>npm start

> myfirstreact@0.1.0 start
> react-scripts start

(node:13112) [DEP_WEBPACK_DEV_SERVER_ON_AFTER_SETUP_MIDDLEWARE] DeprecationW
arning: 'onAfterSetupMiddleware' option is deprecated. Please use the 'setup
Middlewares' option.
(Use `node --trace-deprecation ...` to show where the warning was created)
(node:13112) [DEP_WEBPACK_DEV_SERVER_ON_BEFORE_SETUP_MIDDLEWARE] Deprecation
Warning: 'onBeforeSetupMiddleware' option is deprecated. Please use the 'set
upMiddlewares' option.
Starting the development server...
Compiled with warnings.

[eslint]
src\App.js
  Line 1:8:  'logo' is defined but never used  no-unused-vars

Search for the keywords to learn more about each warning.
To ignore, add // eslint-disable-next-line to the line before.

WARNING in [eslint]
src\App.js
  Line 1:8:  'logo' is defined but never used  no-unused-vars

webpack compiled with 1 warning
```

OUTPUT:

2. ReactJS_Hands_On

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.

1. Create a new folder under Src folder with the name “Components”. Add a new file named “Home.js”

src > Components > Home.js > ...

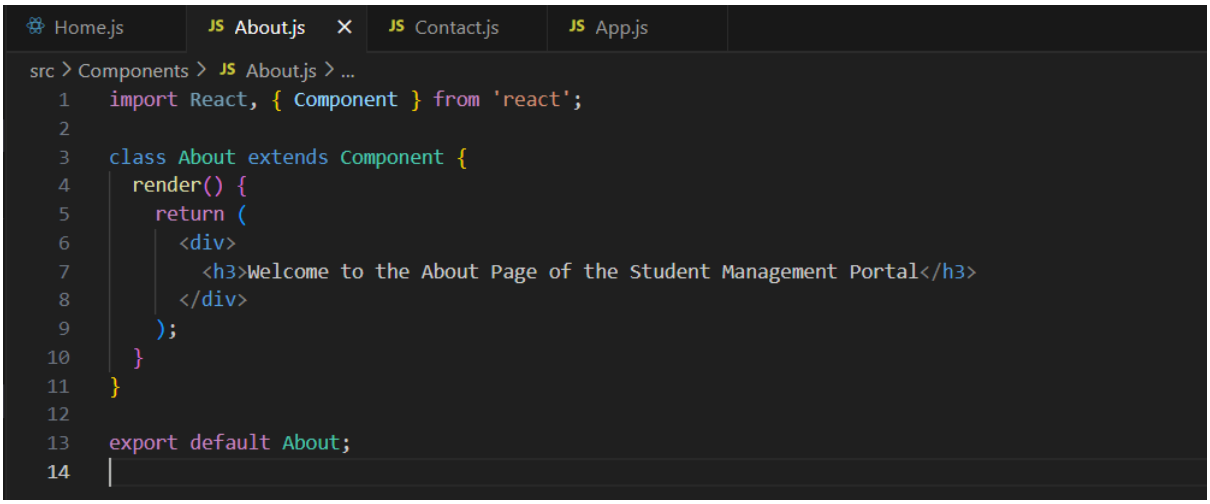
```
import React, { Component } from 'react';

class Home extends Component {
  render() {
    return (
      <div>
        <h3>Welcome to the Home Page of the Student Management Portal</h3>
      </div>
    );
  }
}

export default Home;
```

2. Add a new file named “Contact.js” in Components folder

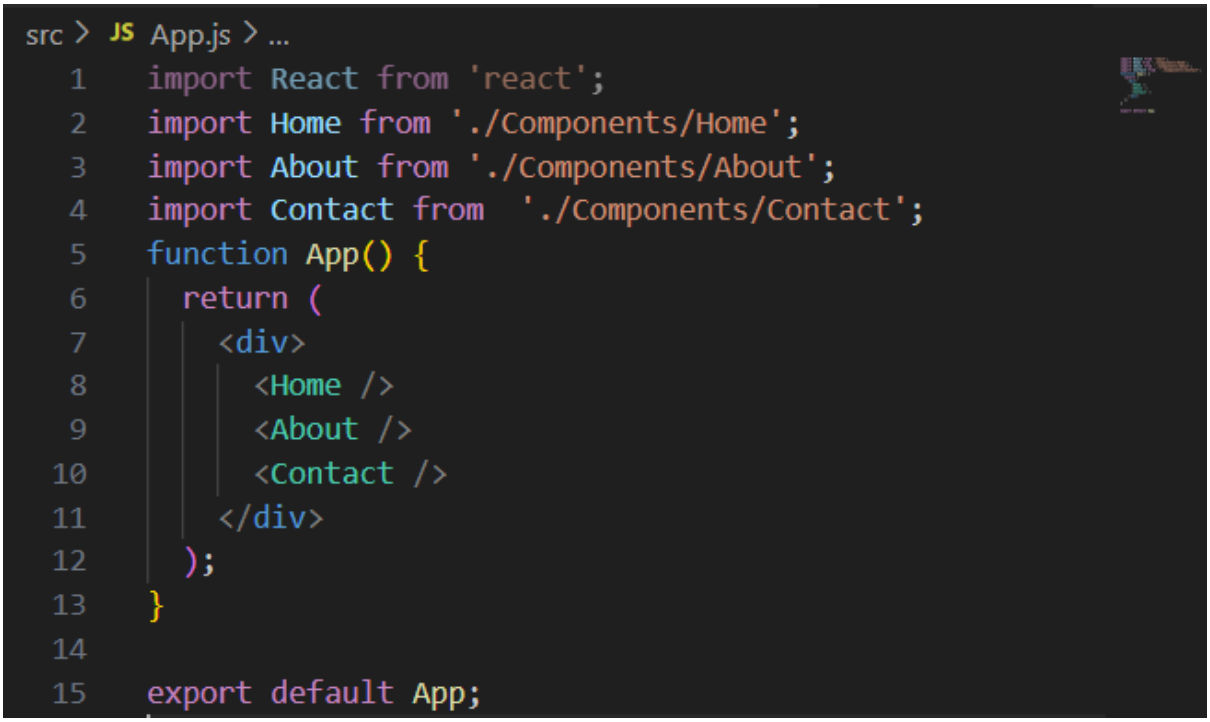
src > Components > Contact.js > ...

```
import React, { Component } from 'react';

class About extends Component {
  render() {
    return (
      <div>
        <h3>Welcome to the Contact Page of the Student Management Portal</h3>
      </div>
    );
  }
}

export default About;
```

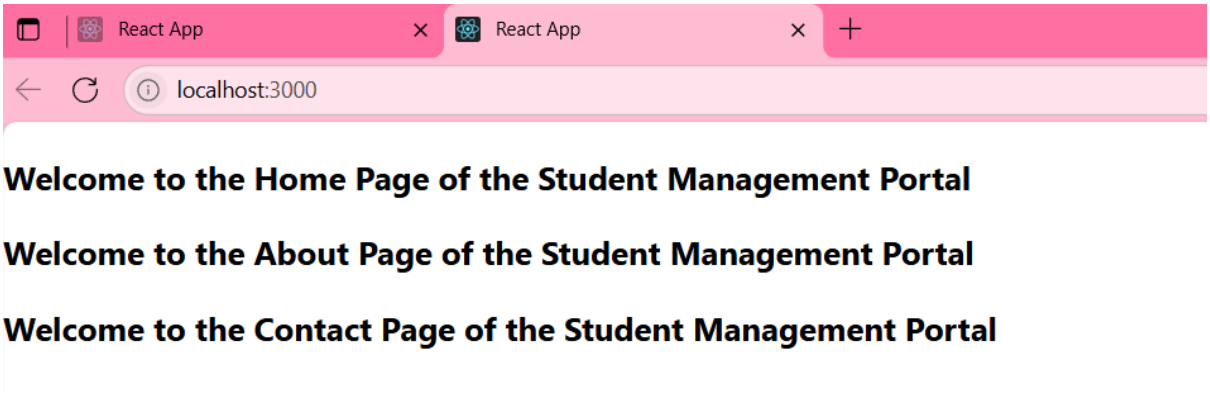
3. Add a new file named “About.js” in Components folder



4. Add a new file named “App.js” in src folder



OUTPUT:


React App
React App
localhost:3000
Welcome to the Home Page of the Student Management Portal
Welcome to the About Page of the Student Management Portal
Welcome to the Contact Page of the Student Management Portal

3. ReactJS-Handson

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.

src > JS App.js > ...

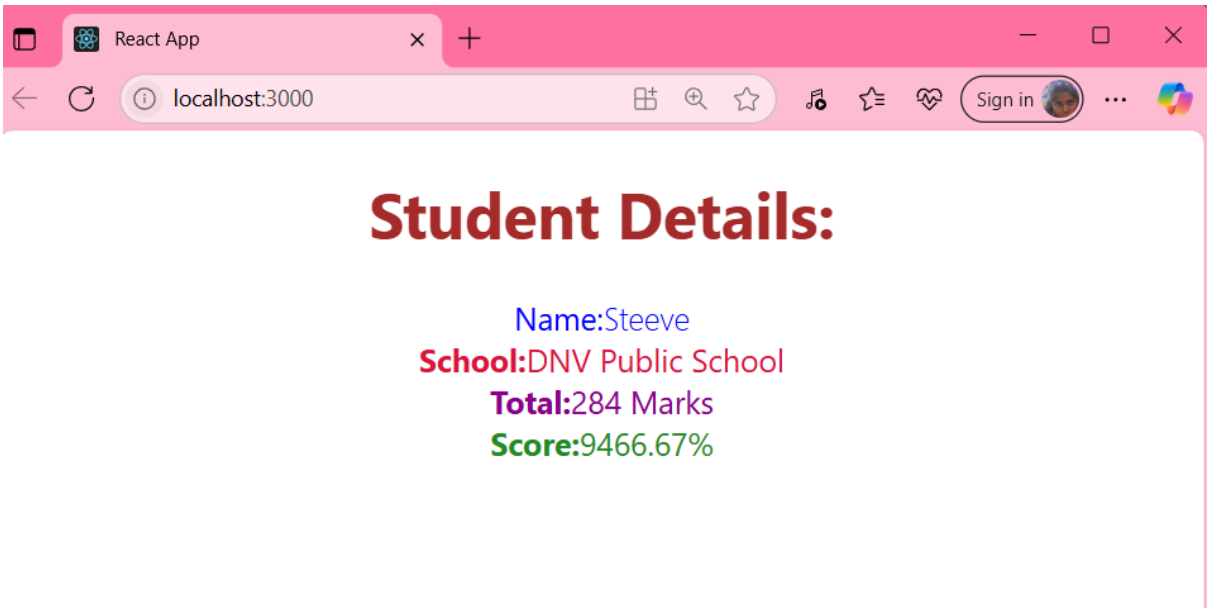
```
1   import logo from './logo.svg';
2   import './App.css';
3   import { CalculateScore } from './CalculateScore';
4   function App() {
5     return (
6       <div>
7         <CalculateScore Name={"Steeve"}
8         School={"DNV Public School"}
9         total={284}
10        goal={3}
11        />
12      </div>
13    );
14  }
15
16  export default App;
17
```

src > JS CalculateScore.js > ...

```jsx
import '../src/mystyle.css';

const percentToDecimal = (decimal) => {
  return (decimal * 100).toFixed(2) + '%';
};

const calcScore = (total, goal) => {
  return percentToDecimal(total / goal);
};

export const CalculateScore = ({ Name, School, total, goal }) => (
  <div className="formatstyle">
    <h1><font color="Brown">Student Details:</font></h1>

    <div className="Name">
      <b><span>Name:</span></b>
      <span>{Name}</span>
    </div>

    <div className="School">
      <b><span>School:</span></b>
      <span>{School}</span>
    </div>

    <div className="Total">
      <b><span>Total:</span></b>
      <span>{total}</span>
      <span> Marks</span>
    </div>

    <div className="Score">
      <b>Score:</b>
      <span>{calcScore(total, goal)}</span>
    </div>
  </div>
);
```

```
src > # mystyle.css > .Score
1   .Name{
2       font-weight:300;
3       color: blue;
4   }
5   .School{
6       color: crimson
7   }
8   .Total{
9       color: darkmagenta;
10  }
11  .formatstyle{
12      text-align:center;
13      font-size: large;
14  }
15  .Score{
16      color: forestgreen;
17  }
```

OUTPUT:

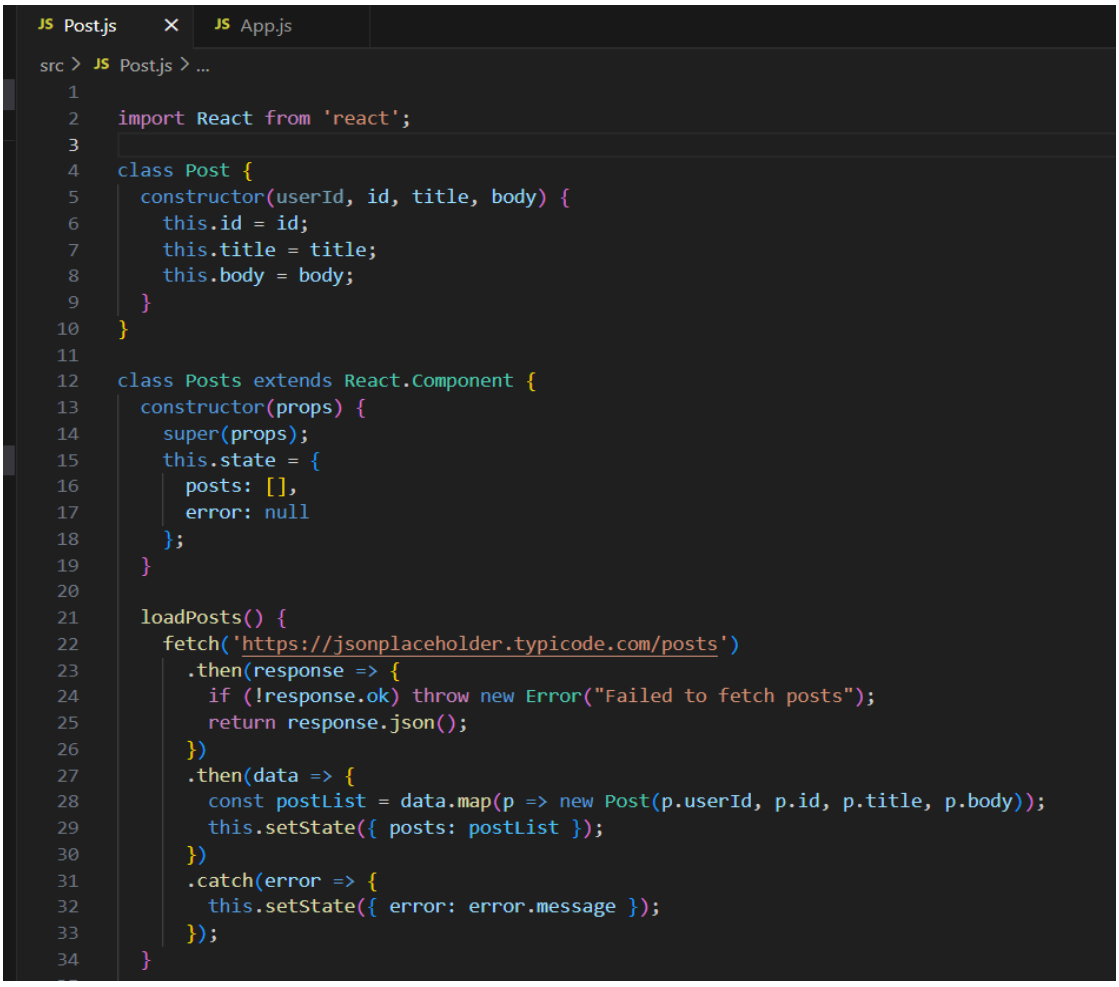


4. ReactJS-Handson

1. Create a new react application using *create-react-app* tool with the name as “blogapp”
2. Open the application using VS Code
3. Create a new file named as **Post.js** in **src folder** with following properties
4. Create a new class based component named as **Posts** inside **Posts.js** file
5. Initialize the component with a list of Post in state of the component using the constructor
6. Create a new method in component with the name as **loadPosts()** which will be responsible for using Fetch API and assign it to the component state created earlier. To get the posts use the url (https://jsonplaceholder.typicode.com/posts)
7. Implement the **componentDidMount()** hook to make calls to **loadPosts()** which will fetch the posts
8. Implement the **render()** which will display the title and post of posts in html page using heading and paragraphs respectively.

9. Define a **componentDidCatch()** method which will be responsible for displaying any error happing in the component as alert messages.

Posts.js



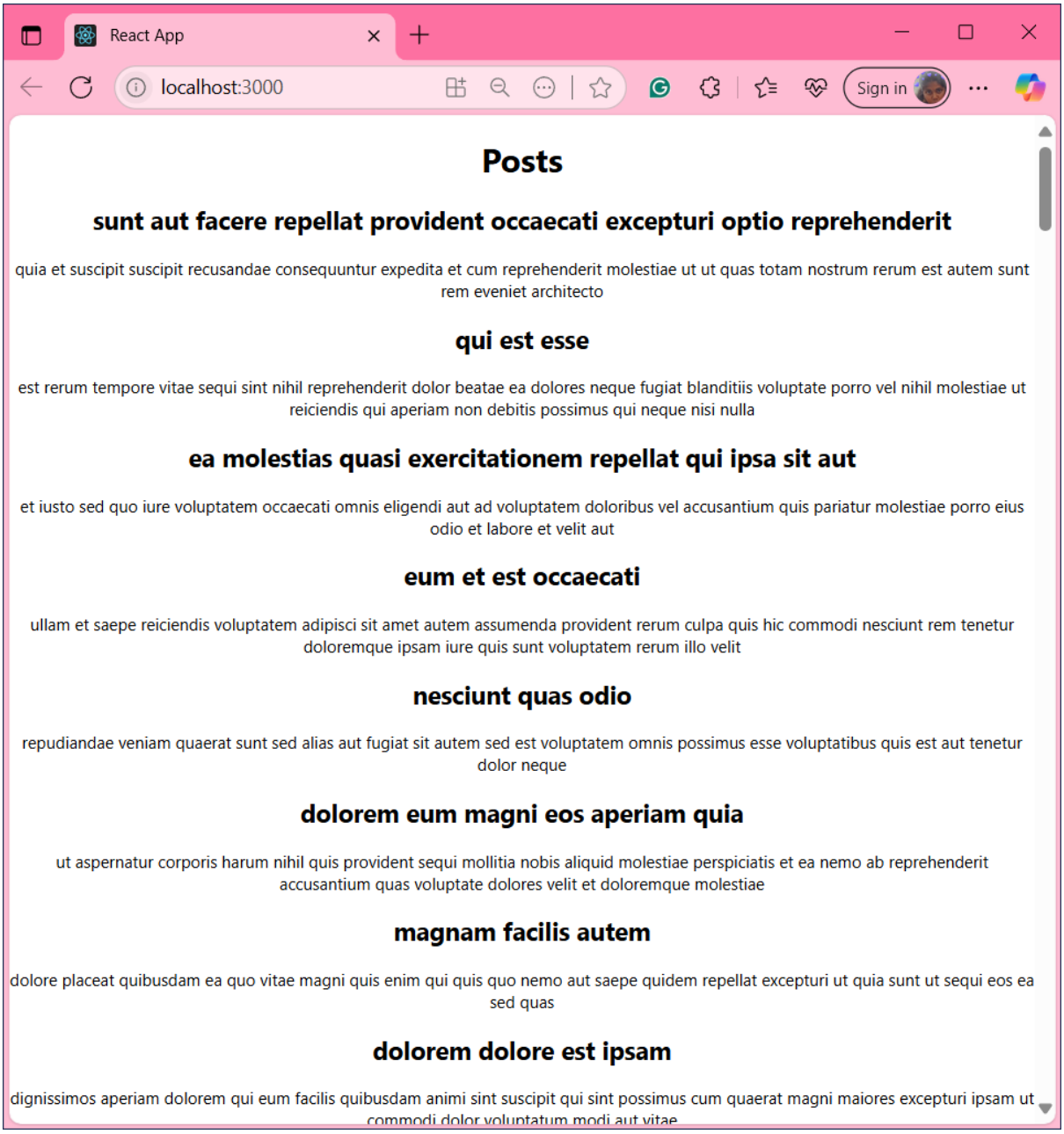
```
  componentDidMount() {
    this.loadPosts();
  }

  componentDidCatch(error, info) {
    alert("An error occurred: " + error);
    console.error("Error details:", info);
  }

  render() {
    const { posts, error } = this.state;

    if (error) {
      return <h2 style={{ color: "red" }}>Error: {error}</h2>;
    }

    return (
      <div>
        <h1>Posts</h1>
        {posts.map(post => (
          <div key={post.id} style={{ marginBottom: '1rem' }}>
            <h2>{post.title}</h2>
            <p>{post.body}</p>
          </div>
        ))}
      </div>
    );
  }
}

export default Posts;
```

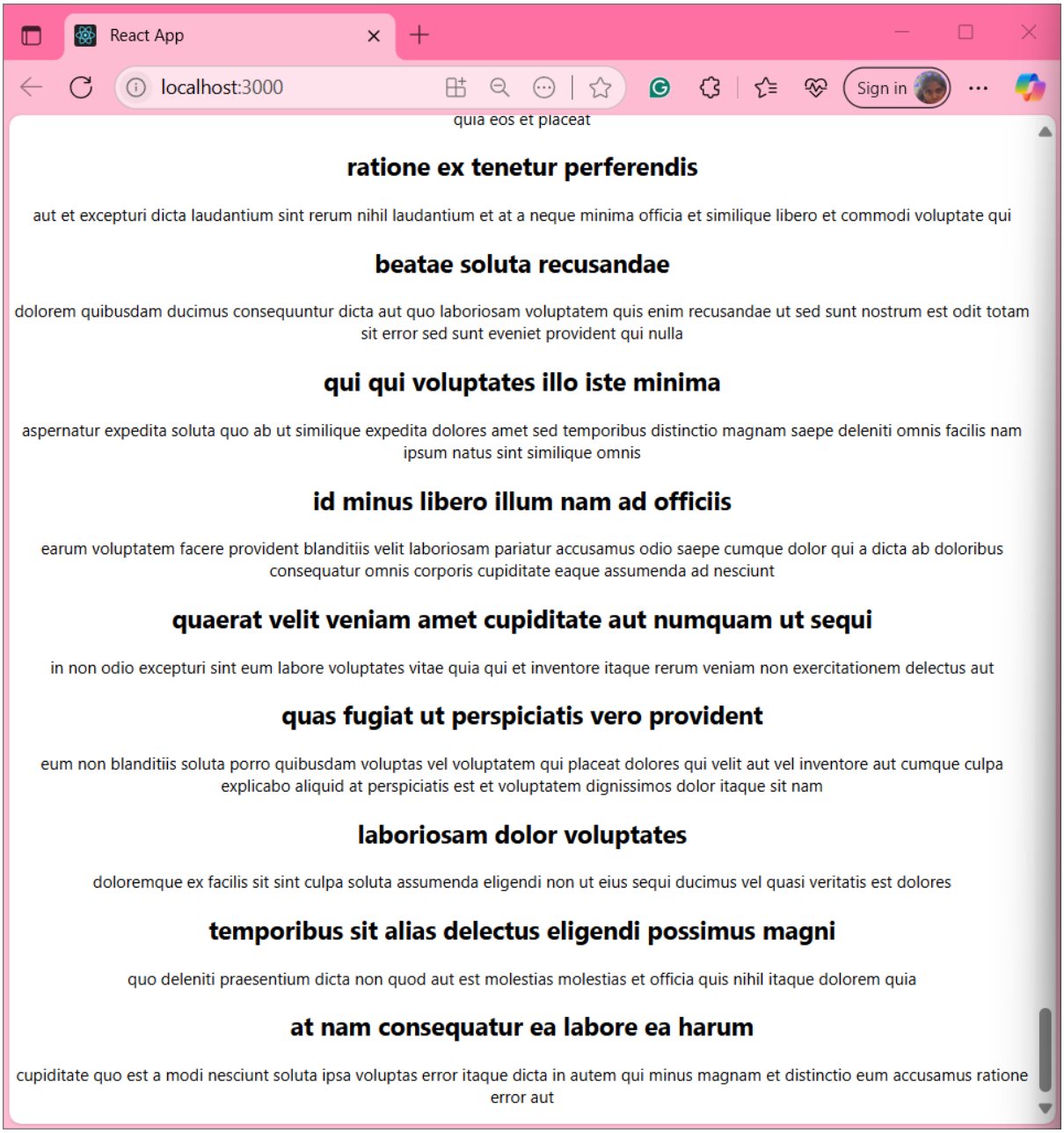
App.js

src > JS App.js > ...

```
import React from 'react';
import './App.css';
import Posts from './Post';

function App() {
  return (
    <div className="App">
      <Posts />
    </div>
  );
}

export default App;
```

OUTPUT:


React App
localhost:3000
Sign in
Posts
sunt aut facere repellat provident occaecati excepturi optio reprehenderit
quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto
qui est esse
est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non debitis possimus qui neque nisi nulla
ea molestias quasi exercitationem repellat qui ipsa sit aut
et iusto sed quo iure voluptatem occaecati omnis eligendi aut ad voluptatem doloribus vel accusantium quis pariatur molestiae porro eius odio et labore et velit aut
eum et est occaecati
ullam et saepe reiciendis voluptatem adipisci sit amet autem assumenda provident rerum culpa quis hic commodi nesciunt rem tenetur doloremque ipsam iure quis sunt voluptatem rerum illo velit
nesciunt quas odio
repudiandae veniam quaerat sunt sed alias aut fugiat sit autem sed est voluptatem omnis possimus esse voluptatibus quis est aut tenetur dolor neque
dolorem eum magni eos aperiam quia
ut aspernatur corporis harum nihil quis provident sequi mollitia nobis aliquid molestiae perspiciatis et ea nemo ab reprehenderit accusantium quas voluptate dolores velit et doloremque molestiae
magnam facilis autem
dolore placeat quibusdam ea quo vitae magni quis enim qui quis quo nemo aut saepe quidem repellat excepturi ut quia sunt ut sequi eos ea sed quas
dolorem dolore est ipsam
dignissimos aperiam dolorem qui eum facilis quibusdam animi sint suscipit qui sint possimus cum quaerat magni maiores excepturi ipsam ut commodi dolor voluptatum modi aut vitae


React App
localhost:3000
Sign in
quia eos et placeat
ratione ex tenetur perferendis
aut et excepturi dicta laudantium sint rerum nihil laudantium et at a neque minima officia et similique libero et commodi voluptate qui
beatae soluta recusandae
dolorem quibusdam ducimus consequuntur dicta aut quo laboriosam voluptatem quis enim recusandae ut sed sunt nostrum est odit totam sit error sed sunt eveniet provident qui nulla
qui qui voluptates illo iste minima
aspernatur expedita soluta quo ab ut similique expedita dolores amet sed temporibus distinctio magnam saepe deleniti omnis facilis nam ipsum natus sint similique omnis
id minus libero illum nam ad officiis
earum voluptatem facere provident blanditiis velit laboriosam pariatur accusamus odio saepe cumque dolor qui a dicta ab doloribus consequatur omnis corporis cupiditate eaque assumenda ad nesciunt
quaerat velit veniam amet cupiditate aut numquam ut sequi
in non odio excepturi sint eum labore voluptates vitae quia qui et inventore itaque rerum veniam non exercitationem delectus aut
quas fugiat ut perspiciatis vero provident
eum non blanditiis soluta porro quibusdam voluptas vel voluptatem qui placeat dolores qui velit aut vel inventore aut cumque culpa explicabo aliquid at perspiciatis est et voluptatem dignissimos dolor itaque sit nam
laboriosam dolor voluptates
doloremque ex facilis sit sint culpa soluta assumenda eligendi non ut eius sequi ducimus vel quasi veritatis est dolores
temporibus sit alias delectus eligendi possimus magni
quo deleniti praesentium dicta non quod aut est molestias molestias et officia quis nihil itaque dolorem quia
at nam consequatur ea labore ea harum
cupiditate quo est a modi nesciunt soluta ipsa voluptas error itaque dicta in autem qui minus magnam et distinctio eum accusamus ratione error aut

5. ReactJS-Handson

TASK: 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.

CohortDetails.js

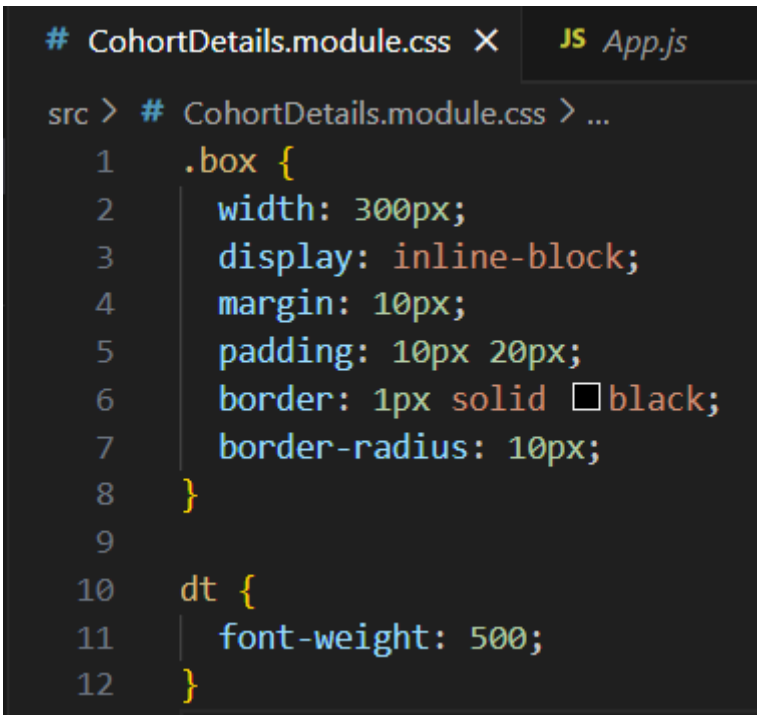
src > JS CohortDetails.js > ...

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

function CohortDetails(props) {
    const cohort = props.cohort;

    return (
        <div className={styles.box}>
            <h3 style={{ color: cohort.currentStatus === 'ongoing' ? 'green' : 'blue' }}>
                {cohort.cohortCode} -
                <span> {cohort.technology}</span>
            </h3>
            <dl>
                <dt>Started On</dt>
                <dd>{cohort.startDate}</dd>
                <dt>Current Status</dt>
                <dd>{cohort.currentStatus}</dd>
                <dt>Coach</dt>
                <dd>{cohort.coachName}</dd>
                <dt>Trainer</dt>
                <dd>{cohort.trainerName}</dd>
            </dl>
        </div>
    );
}

export default CohortDetails;
```

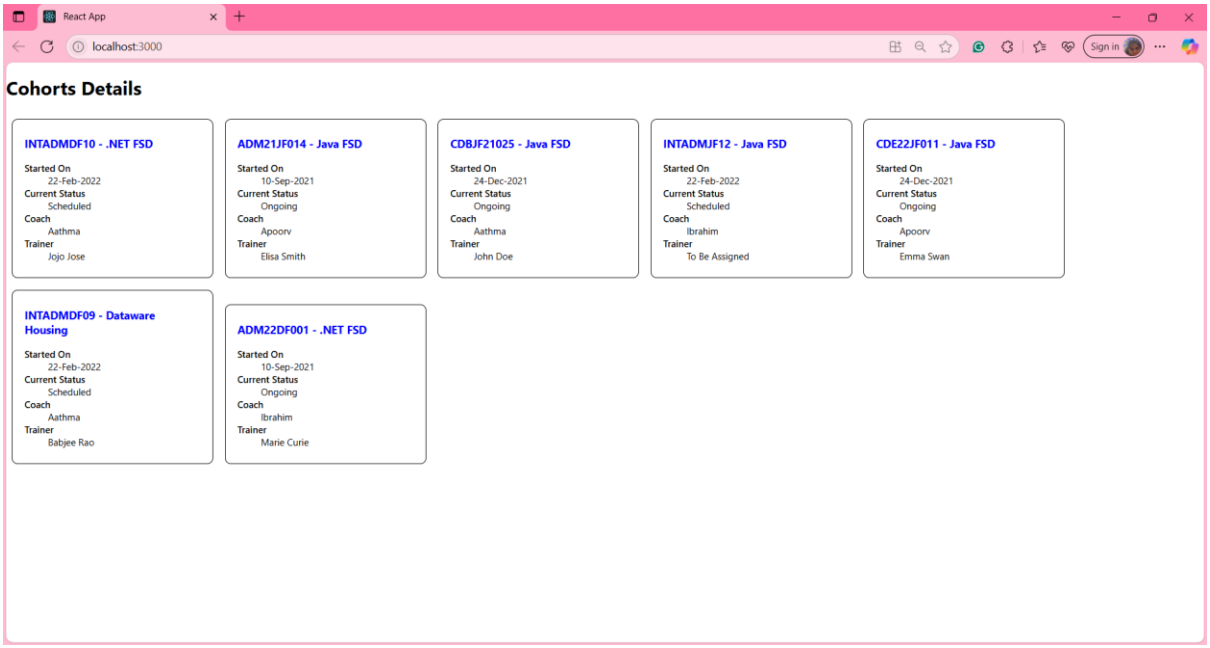
App.js

```
src > JS App.js > ...
1   import logo from './logo.svg';
2   import './App.css';
3   import { CohortsData} from './Cohort'
4   import CohortDetails from './CohortDetails';
5
6   function App() {
7
8     return (
9     <div>
10      <h1>Cohorts Details</h1>
11      {CohortsData.map(cohort => <CohortDetails cohort={cohort}/>)}
12    </div>
13    );
14  }
15
16  export default App;
17
```

CohortDetails.module.css



OUTPUT:


React App
localhost:3000
Sign in
Cohorts Details
INTADMDF10 - .NET FSD
Started On
22-Feb-2022
Current Status
Scheduled
Coach
Aathma
Trainer
Jojo Jose
ADM21JF014 - Java FSD
Started On
10-Sep-2021
Current Status
Ongoing
Coach
Apoorv
Trainer
Elisa Smith
CDBJF21025 - Java FSD
Started On
24-Dec-2021
Current Status
Ongoing
Coach
Aathma
Trainer
John Doe
INTADMJF12 - Java FSD
Started On
22-Feb-2022
Current Status
Scheduled
Coach
Ibrahim
Trainer
To Be Assigned
CDE22JF011 - Java FSD
Started On
24-Dec-2021
Current Status
Ongoing
Coach
Apoorv
Trainer
Emma Swan
INTADMDF09 - Dataware Housing
Started On
22-Feb-2022
Current Status
Scheduled
Coach
Aathma
Trainer
Babjee Rao
ADM22DF001 - .NET FSD
Started On
10-Sep-2021
Current Status
Ongoing
Coach
Ibrahim
Trainer
Marie Curie