**4. ReactJS-HOL**

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

**Post.js**

class Post {

constructor(id, title, body){

this.id = id;

this.title = title;

this.body = body;

}

}

export default Post;

1. Create a new class based component named as **Posts** inside **Posts.js** file

**Posts.js**

class Posts extends React.Component {

constructor(props){

super(props);

}

}

1. Initialize the component with a list of Post in state of the component using the constructor
2. 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>)

class Posts extends React.Component {

constructor(props){

super(props);

//code

}

loadPosts() {

//code

}

}

implement the **componentDidMount()** hook to make calls to **loadPosts()** which will fetch the posts

class Posts extends React.Component {

constructor(props){

super(props);

//code

}

loadPosts() {

//code

}

componentDidMount() {

//code

}

}

Implement the **render()** which will display the title and post of posts in html page using heading and paragraphs respectively.

class Posts extends React.Component {

constructor(props) {

//...

}

loadPosts() {

//...

}

componentDidMount() {

//...

}

render() {

//code

}

}

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

class Posts extends React.Component {

constructor(props) {

//...

}

loadPosts() {

//...

}

componentDidMount() {

//...

}

render() {

//...

}

componentDidCatch(error, info) {

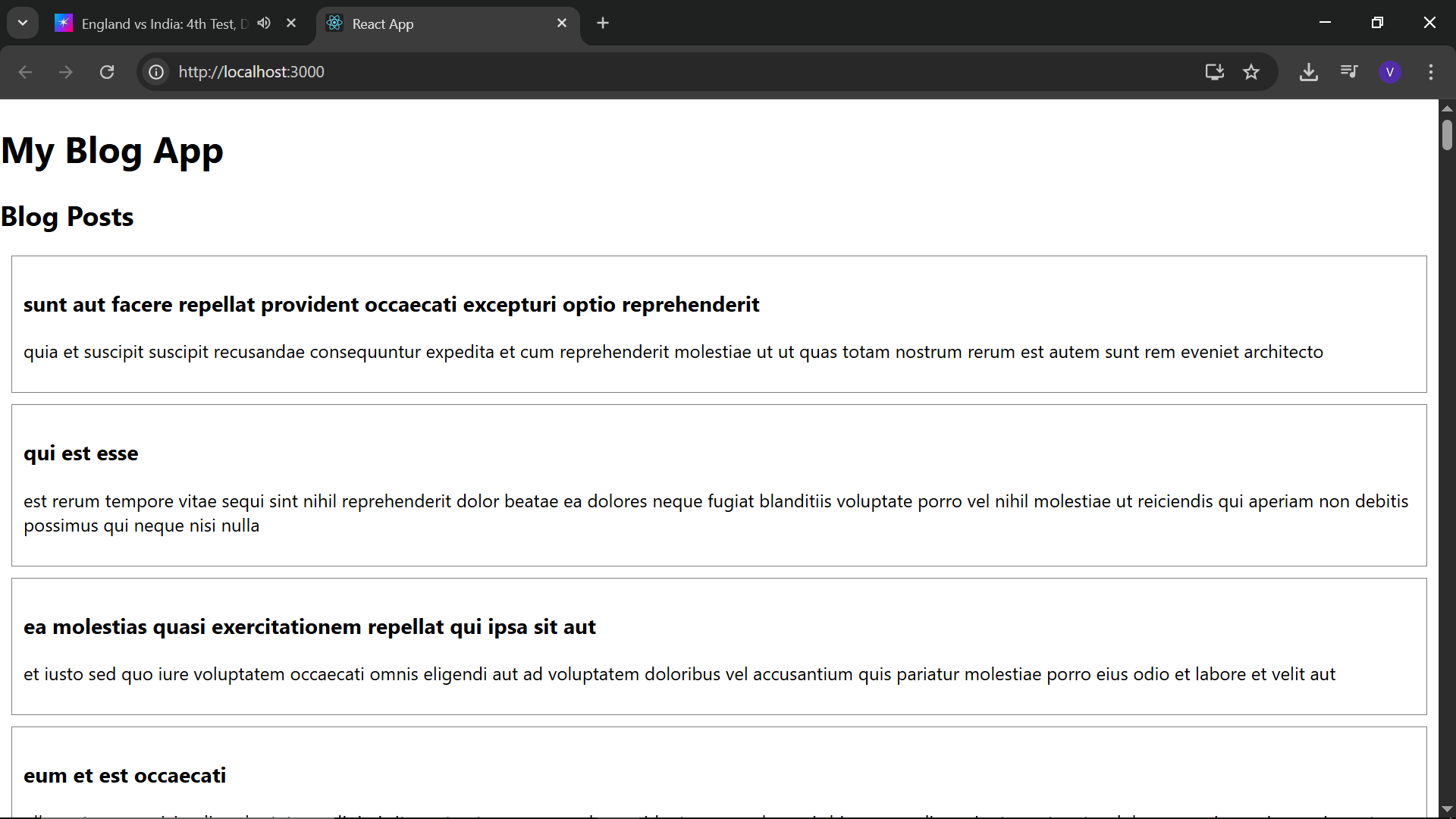
//code

}

}

1. Add the Posts component to App component.
2. Build and Run the application using *npm start* command.

**Output**

****