**WEEK – 6 Assignments**

**React**

**Mandatory hands-on :-**

**4. ReactJS-HOL :**

**Objective :**

In this hands-on lab, you will learn how to:

· Implement componentDidMount() hook

· Implementing componentDidCatch() life cycle hook.

**Code :**

[**App.js**](http://app.js) **:**

| **import React from "react"; import Posts from "./Posts";  function App() {  return (  <div className="App">  <Posts />  </div>  ); }  export default App;** |
| --- |

[**Post.js**](http://post.js) **:**

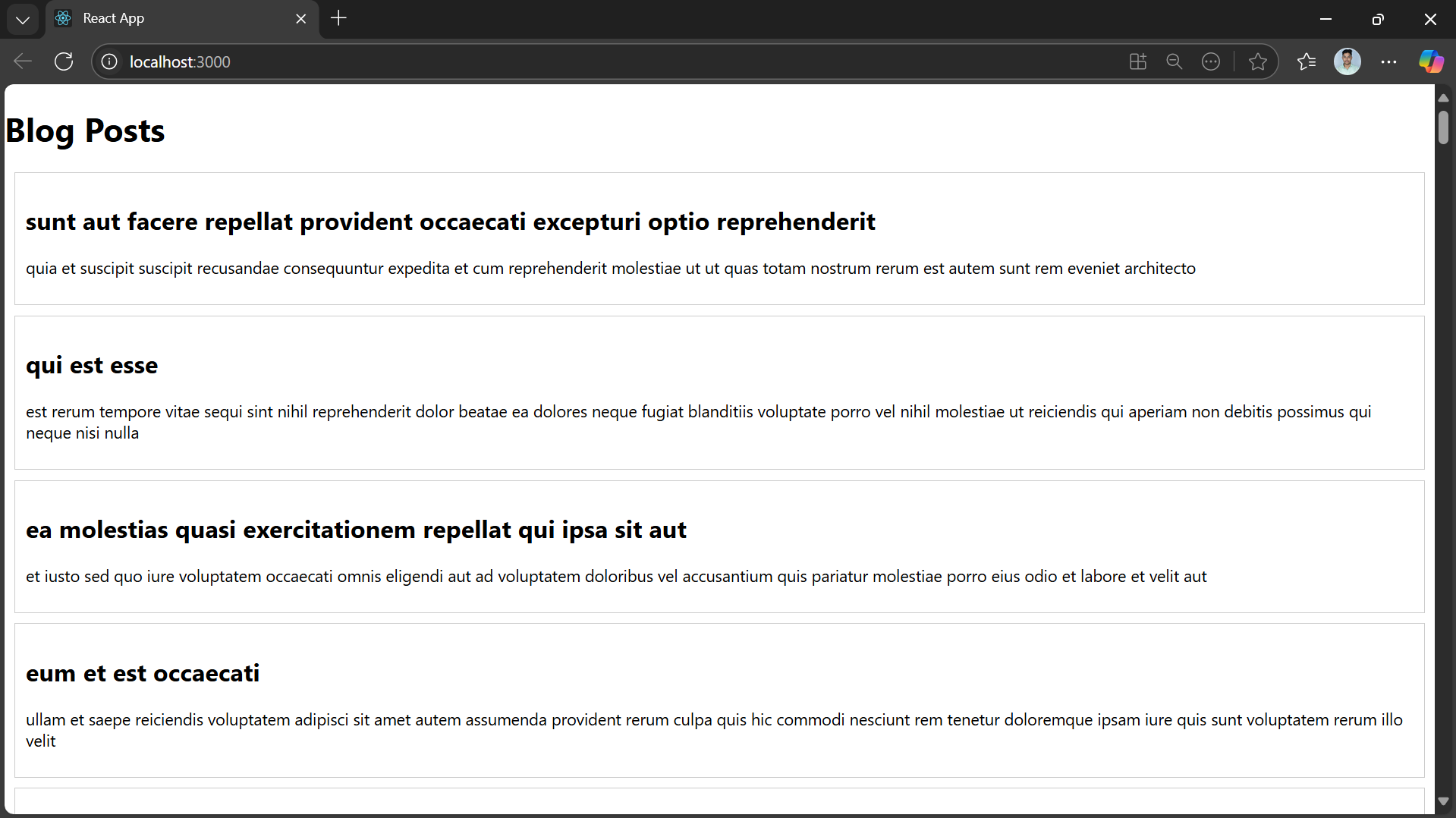
| **class Post {  constructor(id, title, body) {  this.id = id;  this.title = title;  this.body = body;  } }  export default Post;** |
| --- |

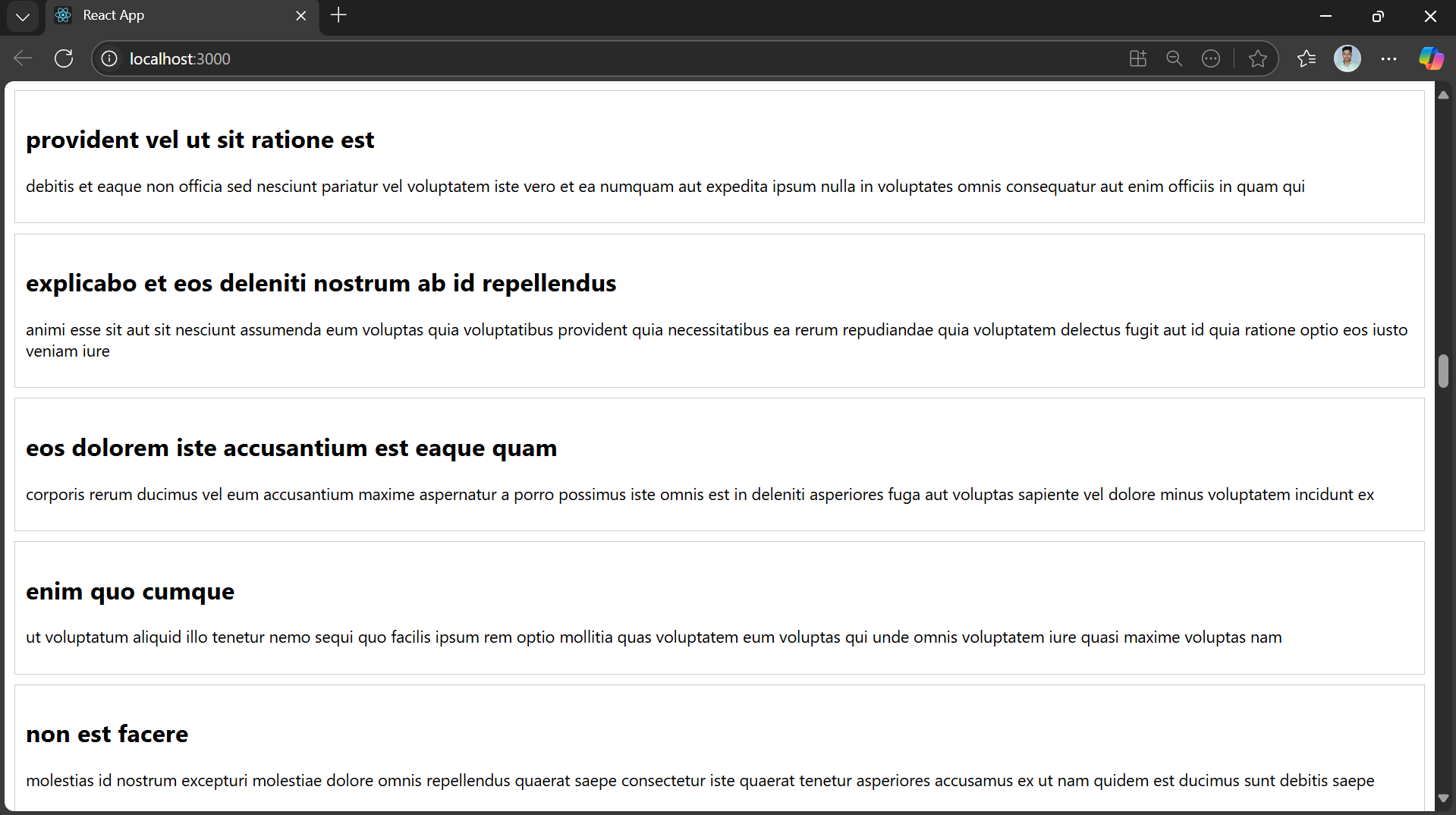
[**Posts.js**](http://posts.js) **:**

| **import React from "react"; import Post from "./Post";  class Posts extends React.Component {  constructor(props) {  super(props);  this.state = {  posts: [],  error: null,  };  }   loadPosts() {  fetch("https://jsonplaceholder.typicode.com/posts")  .then((response) => response.json())  .then((data) => {  const posts = data.map(  (item) => new Post(item.id, item.title, item.body)  );  this.setState({ posts });  })  .catch((error) => {  throw error;  });  }   componentDidMount() {  this.loadPosts();  }   componentDidCatch(error, info) {  this.setState({ error });  alert("Something went wrong: " + error.message);  }   render() {  if (this.state.error) {  return <h2>Failed to load posts due to an error.</h2>;  }   return (  <div>  <h1>Blog Posts</h1>  {this.state.posts.length === 0 && <p>Loading posts...</p>}  {this.state.posts.map((post) => (  <div  key={post.id}  style={{  border: "1px solid #ccc",  margin: "10px",  padding: "10px",  }}  >  <h2>{post.title}</h2>  <p>{post.body}</p>  </div>  ))}  </div>  );  } }  export default Posts;** |
| --- |

**OUTPUT :**

**Localhost:3000 :**

****

****