**Problem Statement:**

Implement componentDidMount() hook. And Implement componentDidCatch() life cycle hook.

**Solution:**

[**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**

import React, { Component } from 'react';

import Post from './post';

class Posts extends Component {

constructor(props) {

super(props);

this.state = {

posts: [],

};

}

loadPosts = async () => {

try {

const response = await fetch('https://jsonplaceholder.typicode.com/posts');

const data = await response.json();

const postList = data.map(

(post) => new Post(post.id, post.title, post.body)

);

this.setState({ posts: postList });

} catch (error) {

console.error('Error fetching posts:', error);

}

};

componentDidMount() {

this.loadPosts();

}

componentDidCatch(error, info) {

alert("An error occurred in Posts component!");

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

}

render() {

return (

<div>

<h2>Posts</h2>

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

<div key={post.id} style={{ marginBottom: '20px' }}>

<h3>{post.title}</h3>

<p>{post.body}</p>

</div>

))}

</div>

);

}

}

export default Posts;

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

import React from 'react';

import Posts from './Posts';

function App() {

return (

<div className="App">

<h1>Blog Application</h1>

<Posts />

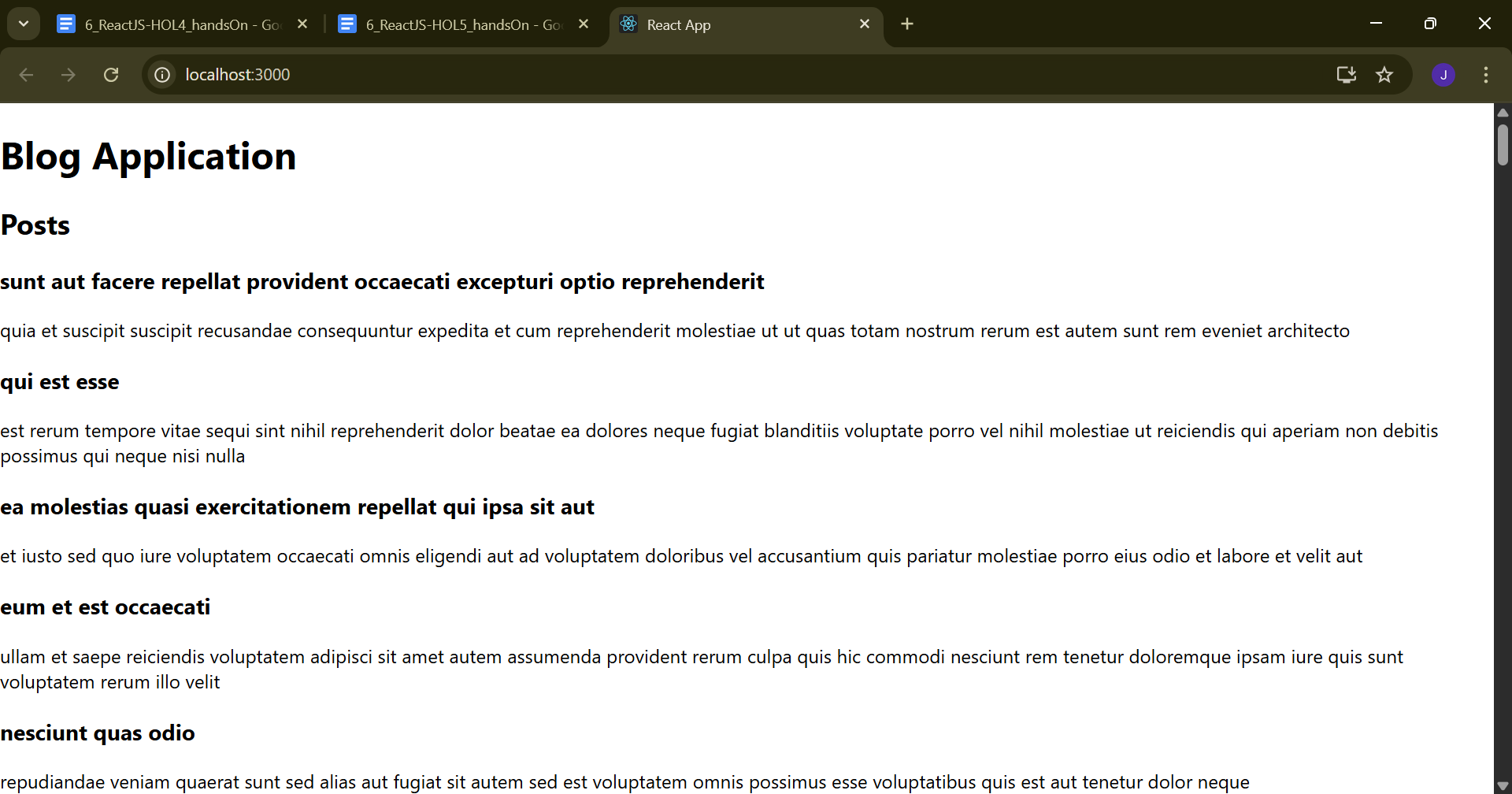
</div>

);

}

export default App;

**Output:**

****