* Implement componentDidMount() hook
* Implementing componentDidCatch() life cycle hook.

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.
2. Initialize the component with a list of Post in state of the component using the constructor
3. 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>)
4. Implement the **componentDidMount()** hook to make calls to **loadPosts()** which will fetch the posts
5. Implement the **render()** which will display the title and post of posts in html page using heading and paragraphs respectively.
6. Define a **componentDidCatch()** method which will be responsible for displaying any error happing in the component as alert messages.

Posts.js

import React from 'react';

class Posts extends React.Component{

    constructor(props){

        super(props);

        this.state={

          posts:[],

          hasError: false

    };

}

loadPosts =()=>

{

    fetch('https://jsonplaceholder.typicode.com/posts')

    .then(response =>{

        if(!response.ok){

            throw new Error("Network response was not ok");

        }

        return response.json();

    })

    .then(data =>{

        this.setState({posts :data});

    })

    .catch(error =>{

        console.error("Fetch error", error);

        this.setState({hasError:true});

    });

}

componentDidMount(){

    this.loadPosts();

}

componentDidCatch(error,info)

{

    alert("AN ERROR ACCURED:"+ error.message);

    console.error("Error caught in Posts component:", error, info);

    this.setState({hasError: true});

}

render(){

    const{posts,hasError}= this.state;

    if(hasError)

    {

        return <p>Error Loading Posts Processing.</p>;

    }

    return(

        <div>

            <h1>

                ALL BLOG POSTS

            </h1>

            {

                posts.map(post =>(

                    <div key={post.id}>

                        <h2>{post.title}</h2>

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

                    </div>

                ))

            }

        </div>

    );

}

}

export default Posts;

1. Add the Posts component to App component.

import React from 'react';

import Posts from './Posts';

function App(){

  return(

    <div className="App">

      <Posts />

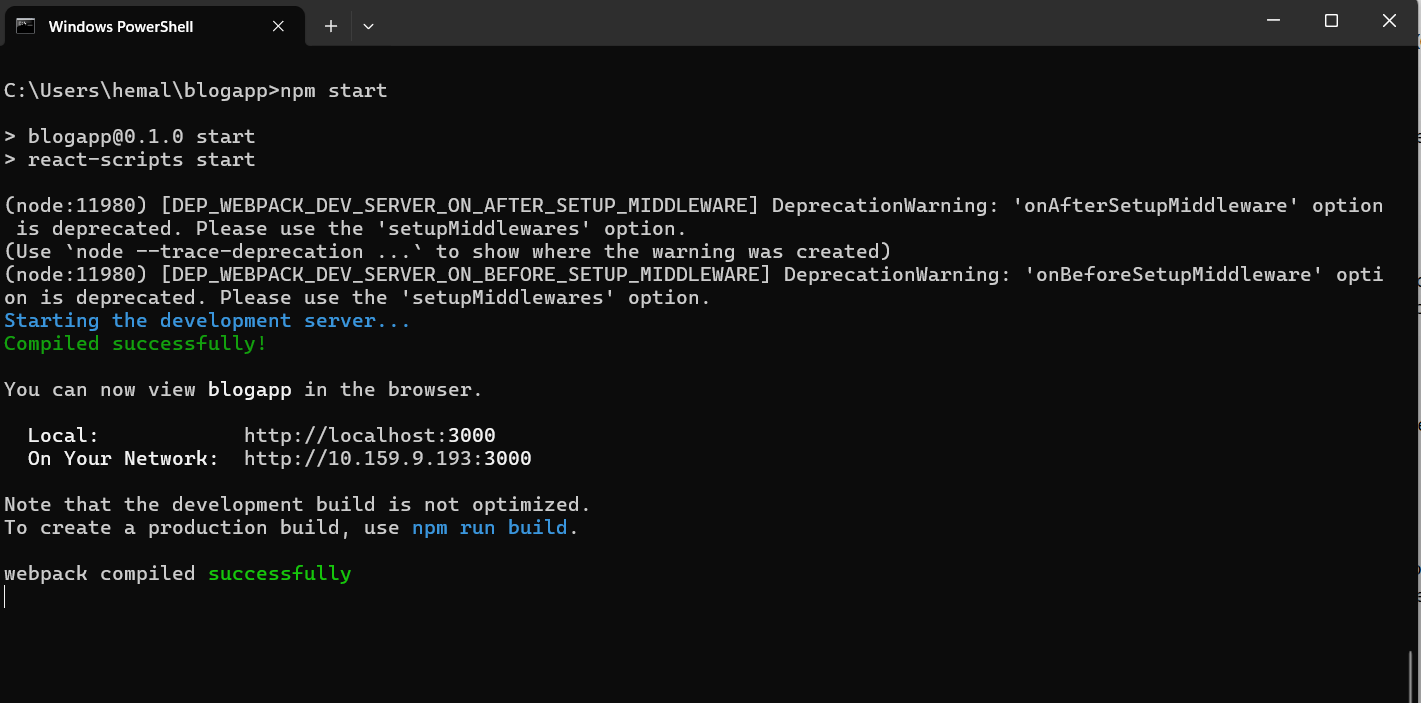
    </div>

  );

}

export default App;

1. Build and Run the application using *npm start* command.



1. Execution of the code

