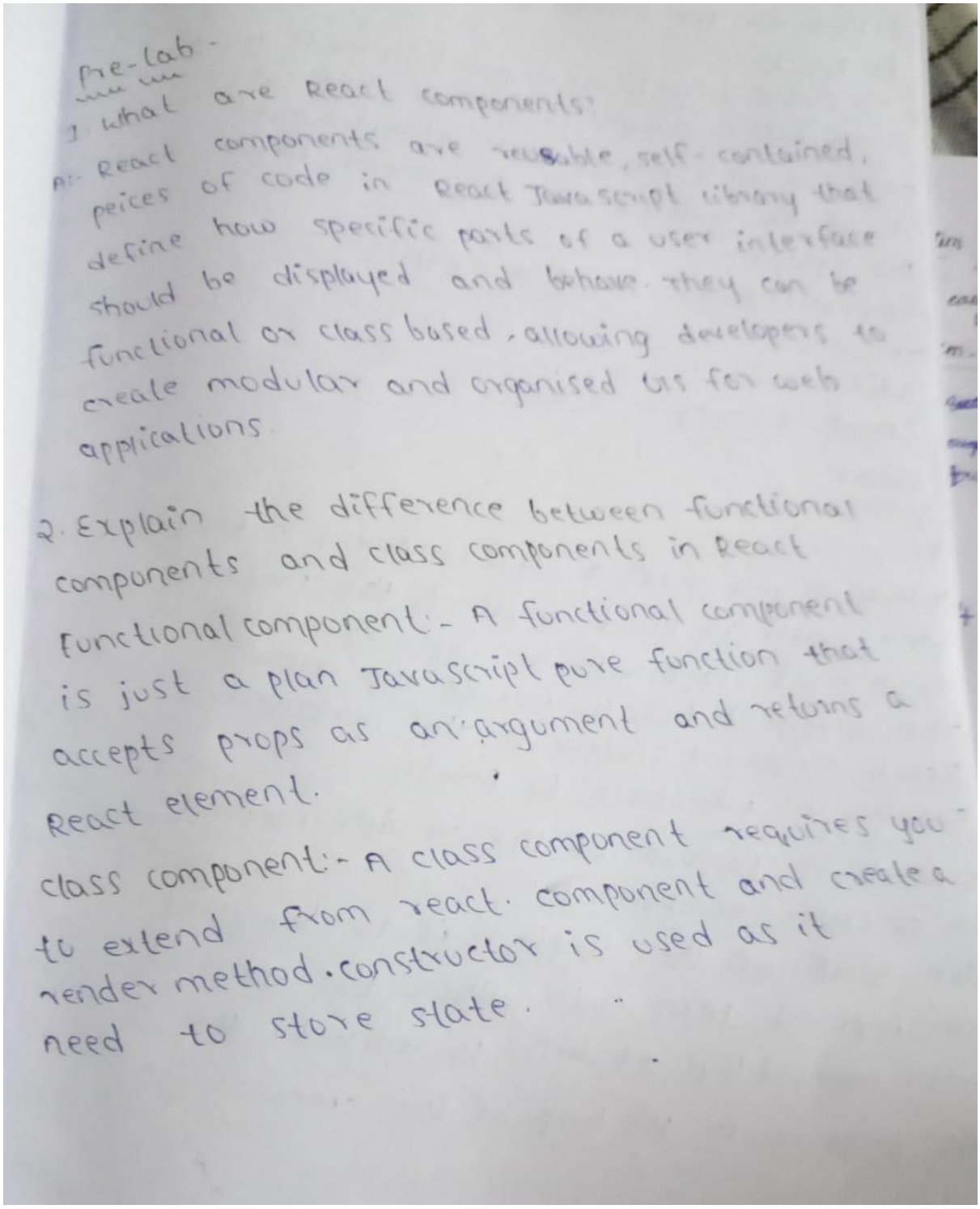
EXPERIMENT 1



Pre-lab:-

1. What are React components?

A:- React components are reusable, self-contained, peices of code in React Javascript library that define how specific parts of a user interface should be displayed and behave. they can be functional or class based, allowing developers to create modular and organised UIs for web applications.

2. Explain the difference between functional components and class components in React

Functional component:- A functional component is just a plan Javascript pure function that accepts props as an argument and returns a React element.

class component:- A class component requires you to extend from react. component and create a render method. constructor is used as it need to store state.

3 How do you create a functional component in React?

To create a functional component in React, you simply define a JavaScript function that takes props as an argument and returns what should be rendered on the screen. Here's the syntax:

```
function MyFunctionalComponent(props) {
return (
<div>
{props.message}
</div>
);
}
```

4. How do you create a class component in React?

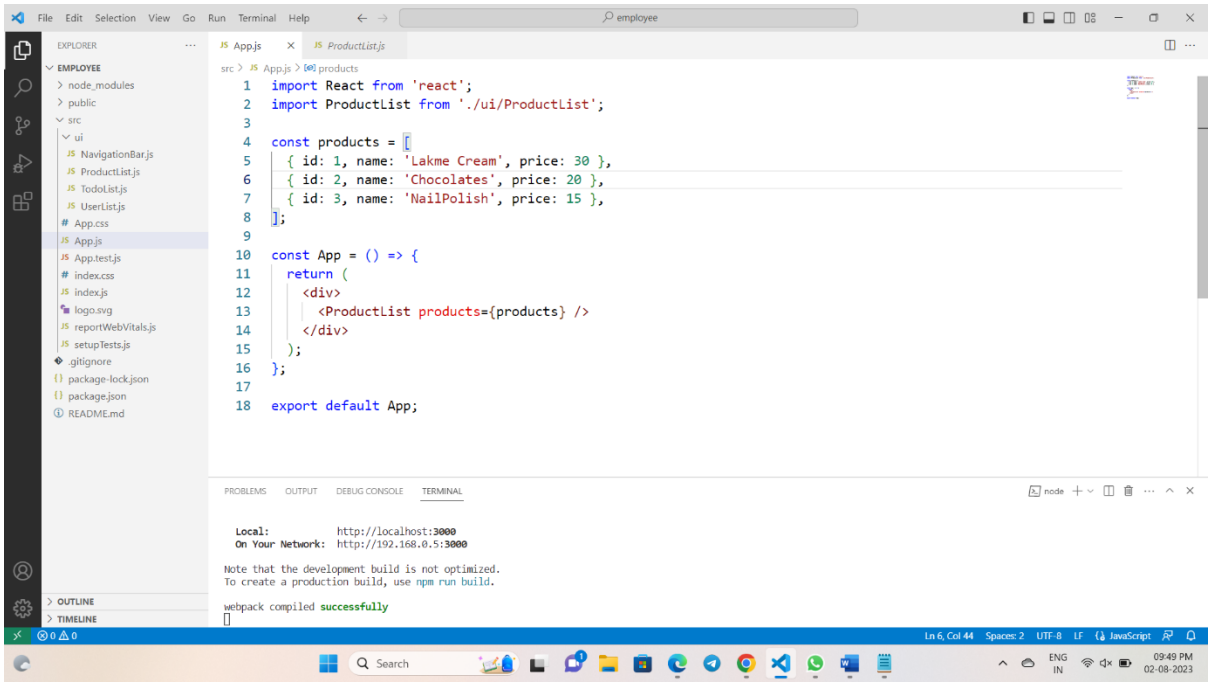
When creating a React component, the component's name must start with an upper case letter. The component has to include the extends React.Component statements, this statement creates an inheritance to React.

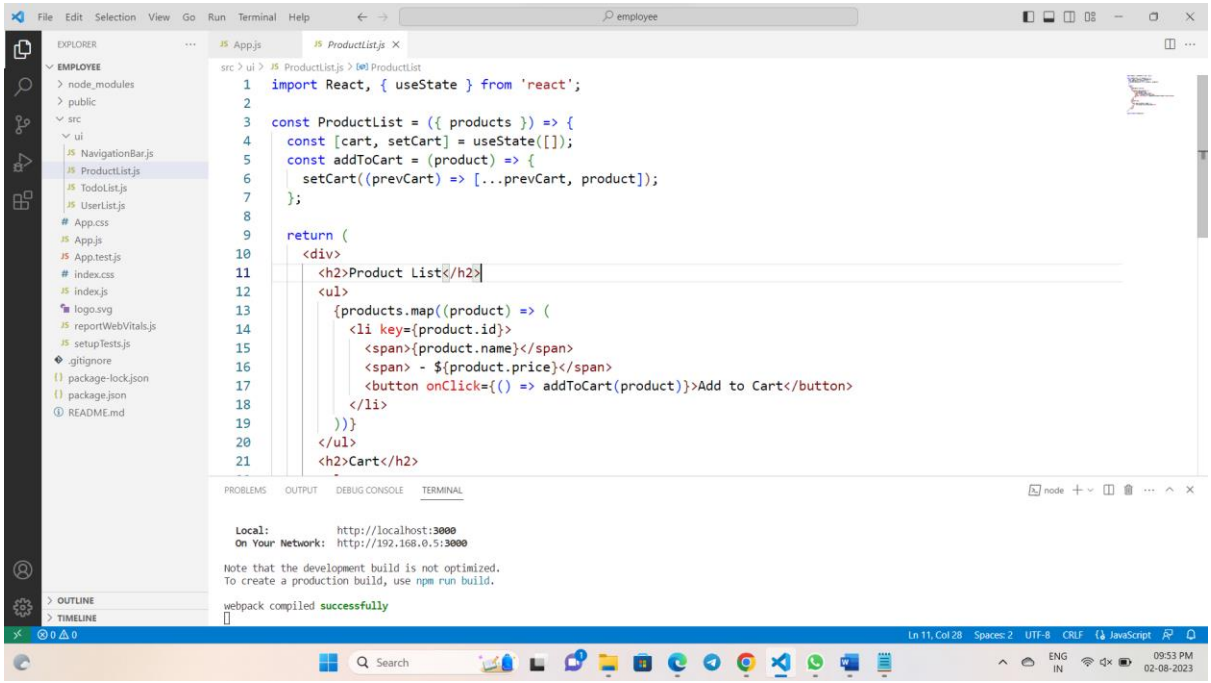
5. What is purpose of the render method in a class component?

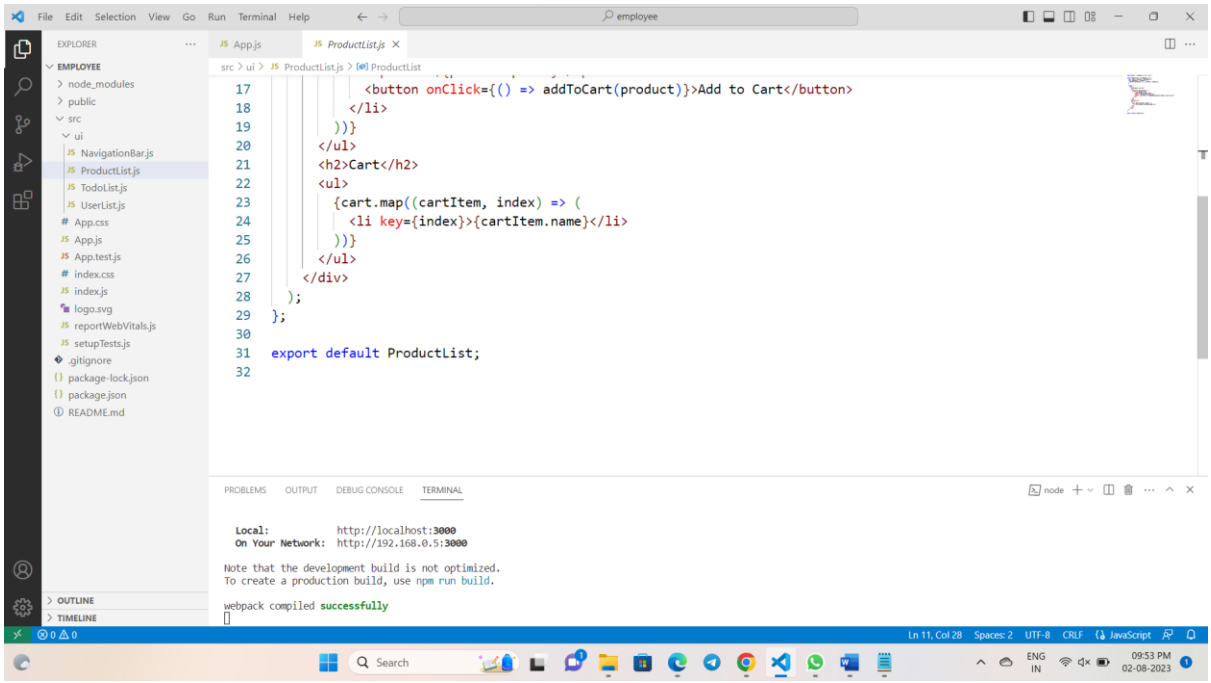
The goal of this function is to represent the imposed HTML code within the specified HTML element tags. It helps to redirect the HTML page with the help of the render() function.

INLAB QUESTIONS:

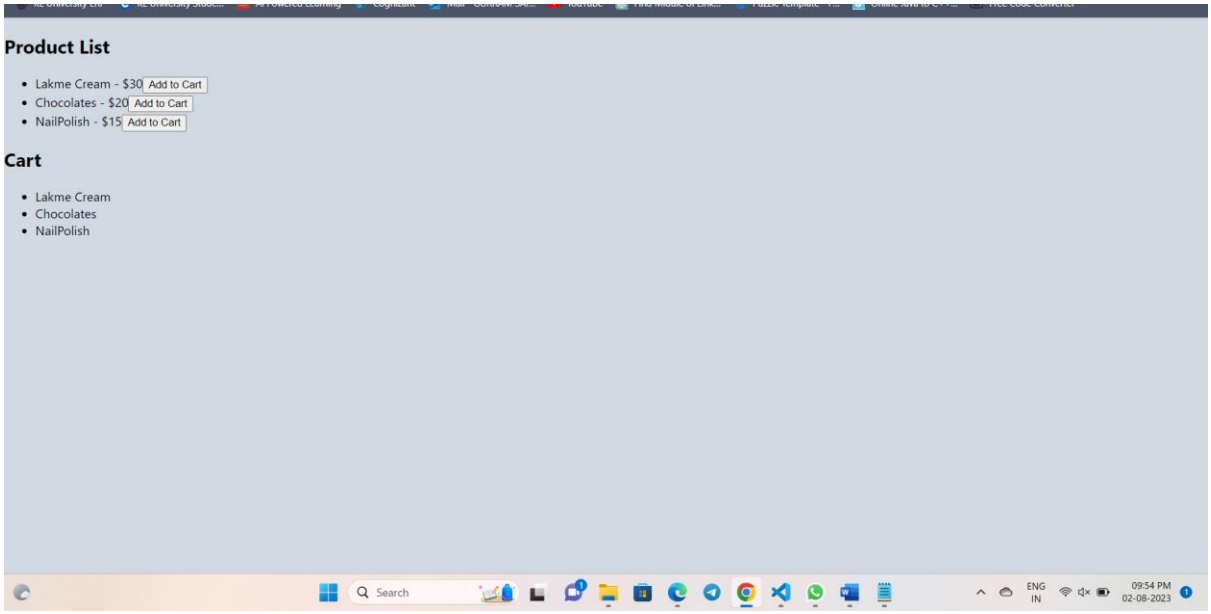
Exercise 1: Create a React component called ProductList that receives an array of products as props and renders a list of product items. Each product item should display the product name, price, and a button to add the product to the cart. Implement the functionality to update the cart when a product is added.



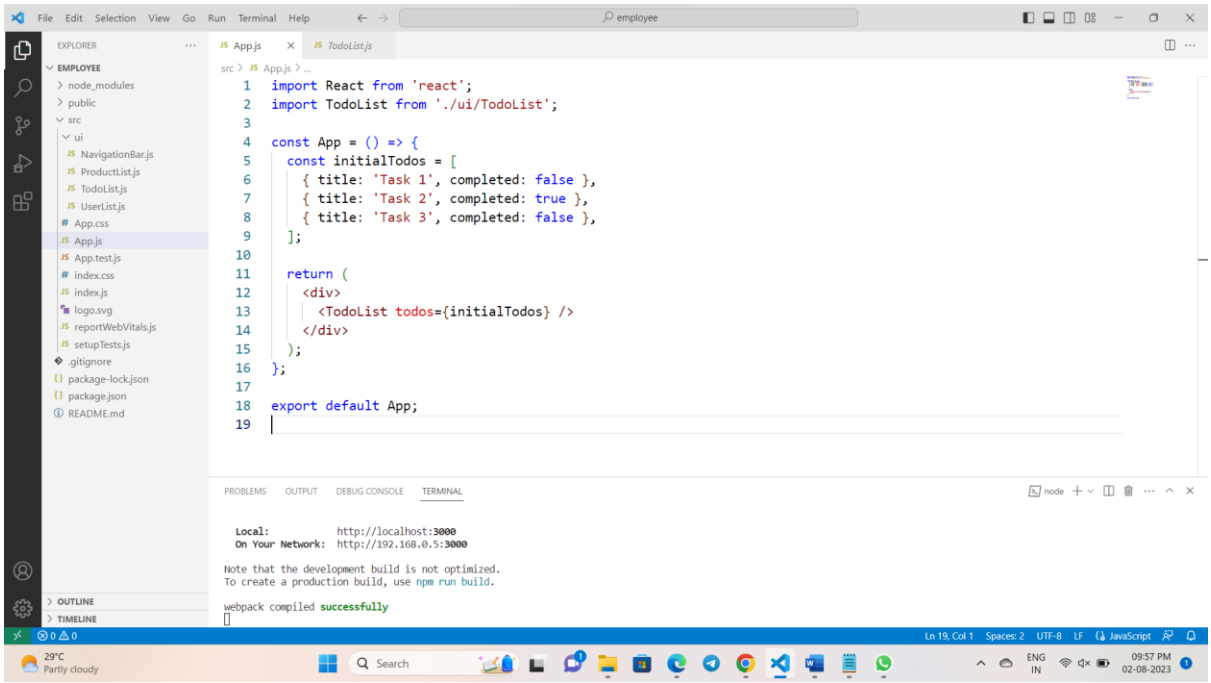


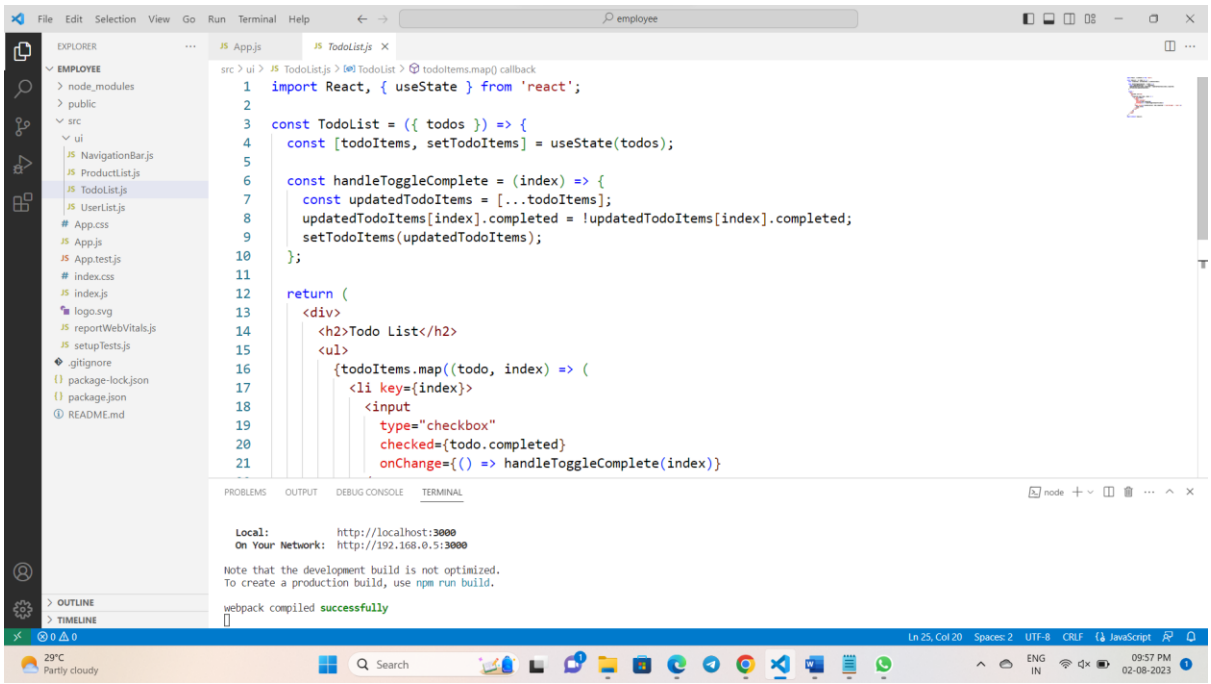


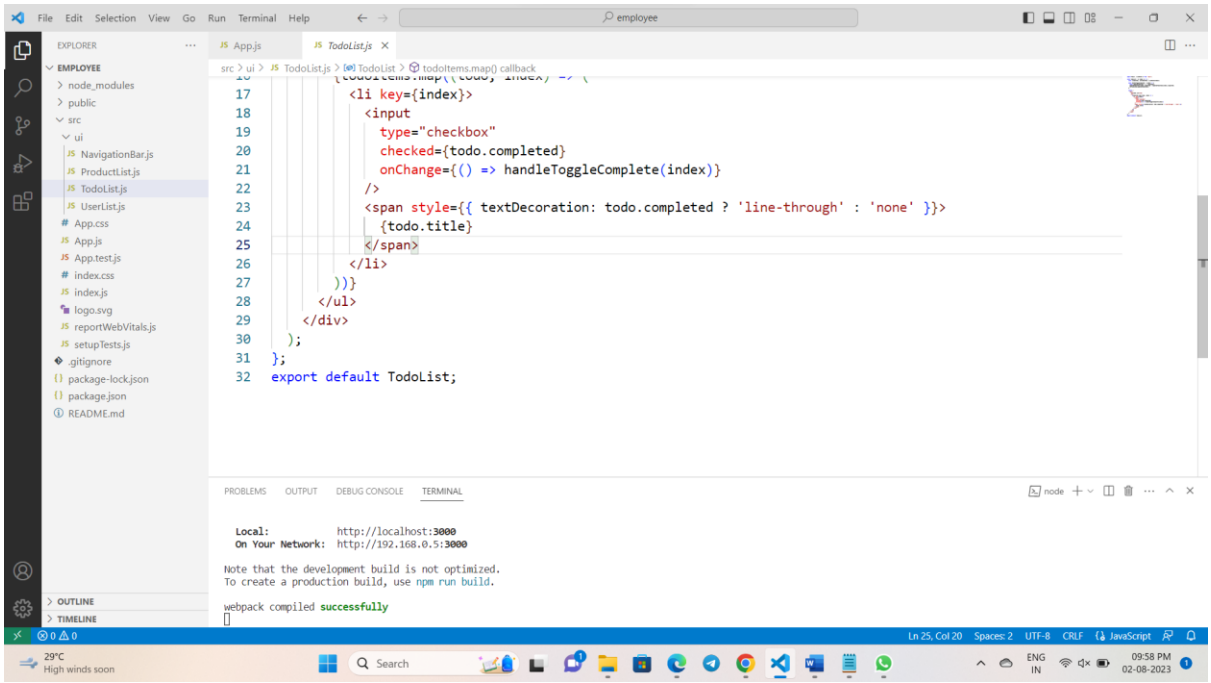
OUTPUT:



Exercise 2: Create a React component called TodoList that receives an array of todo items as props. Each todo item should have a title and a completed status. Render the todo items as a list with checkboxes indicating the completion status. Implement the functionality to toggle the completion status when a checkbox is clicked.


File Edit Selection View Go Run Terminal Help
employee
EXPLORER
App.js
TodoList.js
src > App.js > ...
1 import React from 'react';
2 import TodoList from './ui/TodoList';
3
4 const App = () => {
5 const initialTodos = [
6 { title: 'Task 1', completed: false },
7 { title: 'Task 2', completed: true },
8 { title: 'Task 3', completed: false },
9];
10
11 return (
12
13 <TodoList todos={initialTodos} />
14
15);
16 };
17
18 export default App;
19
EMPLOYEE
node_modules
public
src
ui
NavigationBar.js
ProductList.js
TodoList.js
UserList.js
App.css
App.js
App.test.js
index.css
index.js
logo.svg
reportWebVitals.js
setupTests.js
.gitignore
package-lock.json
package.json
README.md
OUTLINE
TIMELINE
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
node
Local: http://localhost:3000
On Your Network: http://192.168.0.5:3000
Note that the development build is not optimized.
To create a production build, use npm run build.
webpack compiled successfully
Ln 19, Col 1
Spaces: 2
UTF-8
LF
JavaScript
29°C
Partly cloudy
Search
ENG
IN
09:57 PM
02-08-2023

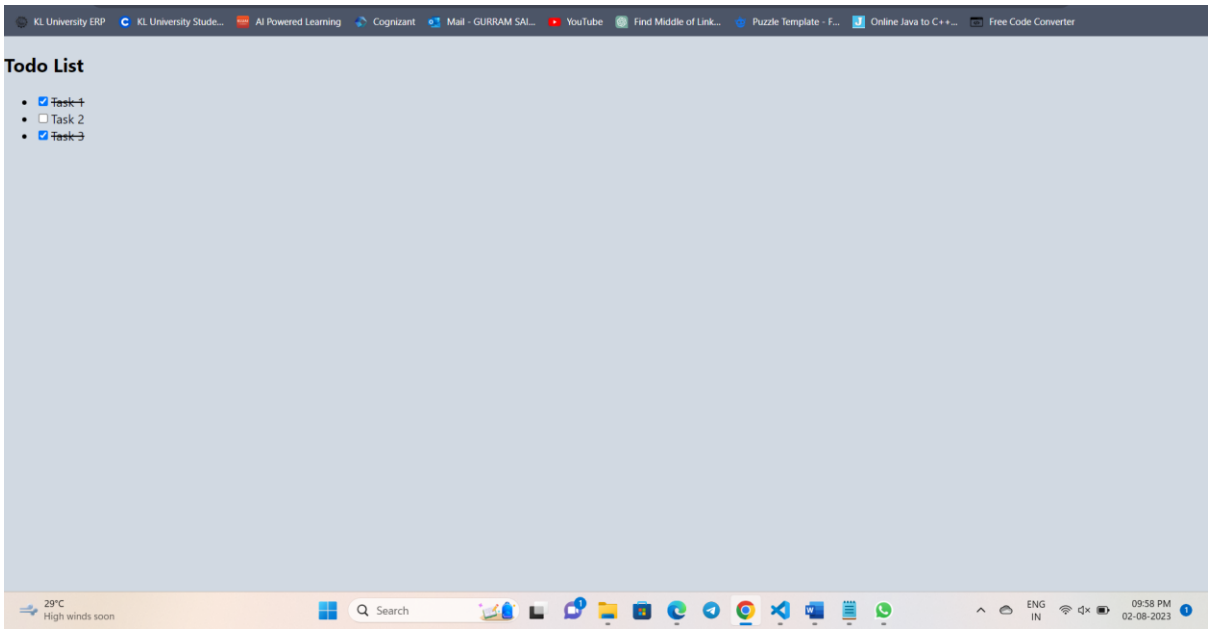

File Edit Selection View Go Run Terminal Help
employee
EXPLORER
App.js
TodoList.js
src > ui > TodoList.js > TodoList > todoItems.map() callback
1 import React, { useState } from 'react';
2
3 const TodoList = ({ todos }) => {
4 const [todoItems, setTodoItems] = useState(todos);
5
6 const handleToggleComplete = (index) => {
7 const updatedTodoItems = [...todoItems];
8 updatedTodoItems[index].completed = !updatedTodoItems[index].completed;
9 setTodoItems(updatedTodoItems);
10 };
11
12 return (
13
14 <h2>Todo List</h2>
15
16 {todoItems.map((todo, index) => (
17 <li key={index}>
18 <input
19 type="checkbox"
20 checked={todo.completed}
21 onChange={() => handleToggleComplete(index)}
EMPLOYEE
node_modules
public
src
ui
NavigationBar.js
ProductList.js
TodoList.js
UserList.js
App.css
App.js
App.test.js
index.css
index.js
logo.svg
reportWebVitals.js
setupTests.js
.gitignore
package-lock.json
package.json
README.md
OUTLINE
TIMELINE
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
node
Local: http://localhost:3000
On Your Network: http://192.168.0.5:3000
Note that the development build is not optimized.
To create a production build, use npm run build.
webpack compiled successfully
Ln 25, Col 20
Spaces: 2
UTF-8
CRLF
JavaScript
29°C
Partly cloudy
Search
ENG
IN
09:57 PM
02-08-2023


File Edit Selection View Go Run Terminal Help
employee
EXPLORER
EMPLOYEE
node_modules
public
src
ui
NavigationBar.js
ProductList.js
TodoList.js
UserList.js
App.css
App.js
App.test.js
index.css
index.js
logo.svg
reportWebVitals.js
setupTests.js
.gitignore
package-lock.json
package.json
README.md
App.js
TodoList.js
src > ui > TodoList.js > TodoList > todoItems.map() callback
<li key={index}>
<input
type="checkbox"
checked={todo.completed}
onChange={() => handleToggleComplete(index)}
/>
<span style={{ textDecoration: todo.completed ? 'line-through' : 'none' }}>
{todo.title}

))}

);
};
export default TodoList;
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Local: http://localhost:3000
On Your Network: http://192.168.0.5:3000
Note that the development build is not optimized.
To create a production build, use npm run build.
webpack compiled successfully
OUTLINE
TIMELINE
Ln 25, Col 20
Spaces: 2
UTF-8
CRLF
JavaScript
29°C
High winds soon
Search
ENG
IN
09:58 PM
02-08-2023

OUTPUT:


KL University ERP
KL University Stude...
AI Powered Learning
Cognizant
Mail - GURRAM SAI...
YouTube
Find Middle of Link...
Puzzle Template - F...
Online Java to C++...
Free Code Converter
Todo List
Task 1
Task 2
Task 3
29°C
High winds soon
Search
ENG
IN
09:58 PM
02-08-2023

Viva questions(In-lab):-

1. What are the features of React:
JSX, virtual DOM, one-way data binding, performance, extensions, conditional statements, components, simplicity.

2. What is JSX?
JSX stands for JavaScript XML. JSX allows us to write HTML in React. JSX makes it easier to write and add HTML in React.

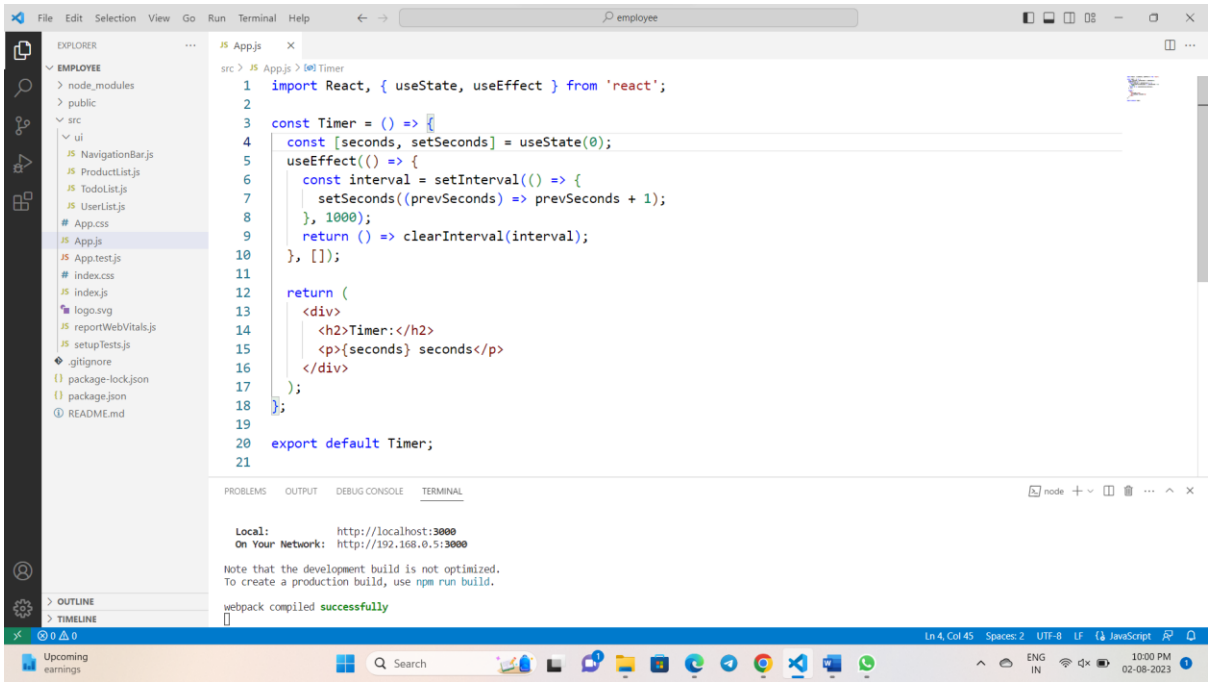
3. Can web browsers read JSX directly?
Browsers can't read JSX because there is no inherent implementation for the browser engines to read and understand them.

4. What is virtual DOM?
A virtual DOM is a lightweight JavaScript representation of the document object model used in declarative web frameworks such as React, Vue.js, and Elm.

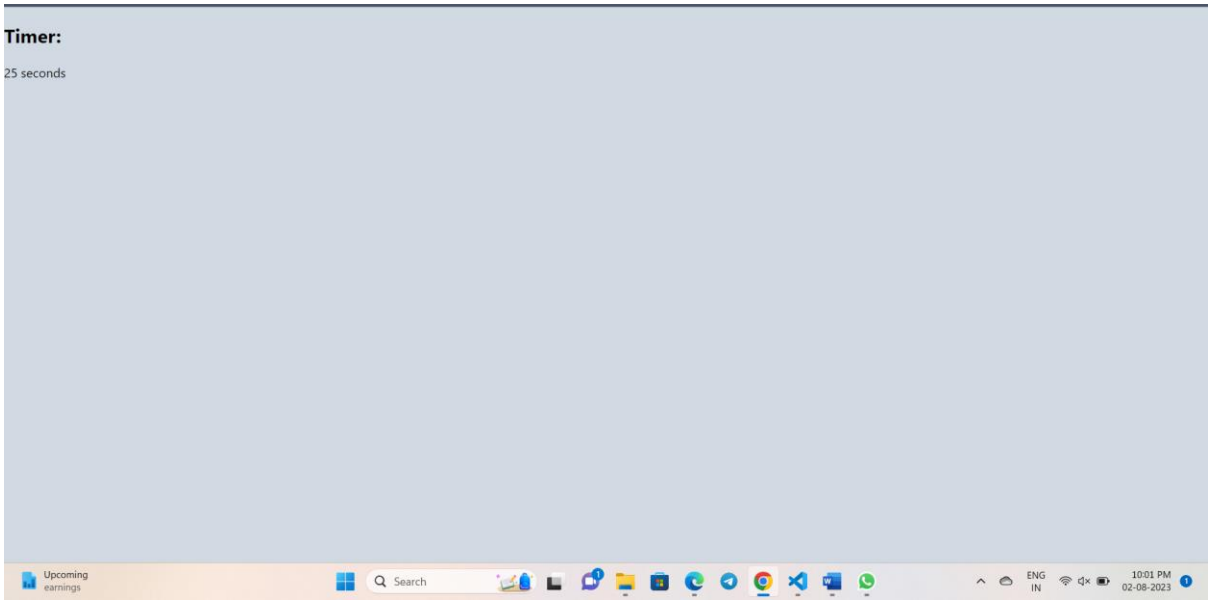
5. Why use react instead of other frameworks, like angular?
In terms of performance, bundle size, and backward compatability, React outperforms angular. The component-driven architecture of React allows developers to reuse code components, which tends to save time and cost.

POSTLAB

question 1: Implement a Timer component in React that displays the number of seconds since it has mounted. The timer should start at 0 and increment by 1 every second. Write the code for the Timer component using functional components and hooks.

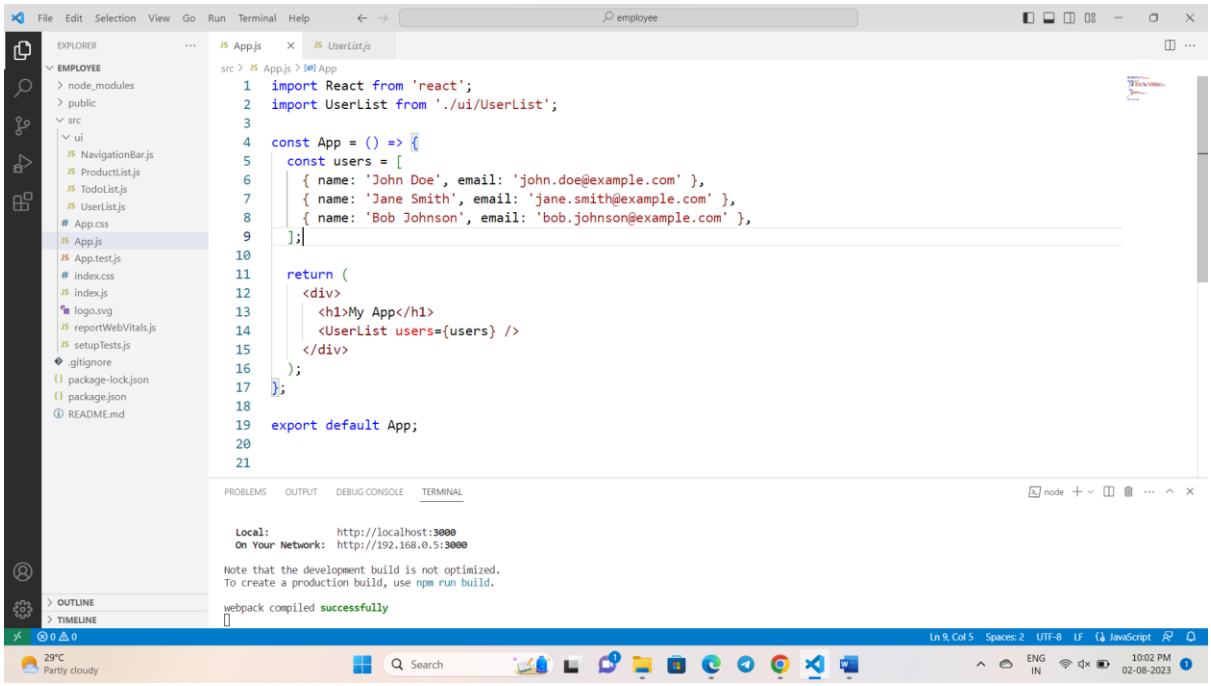


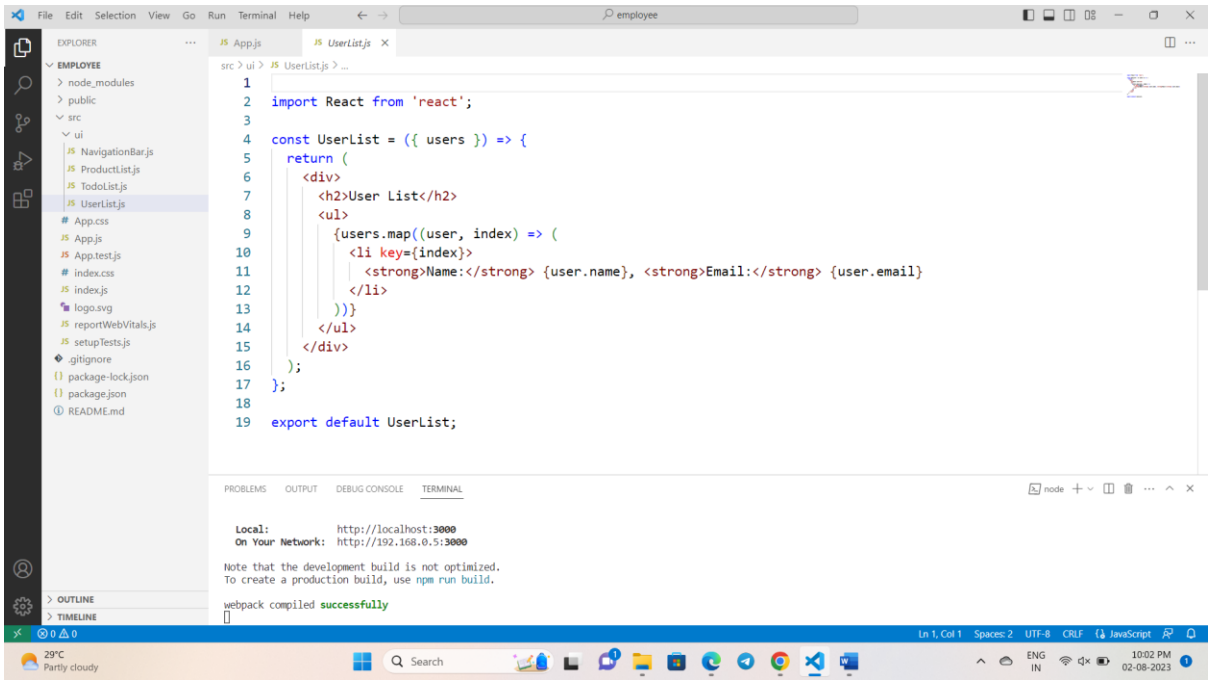
OUTPUT:



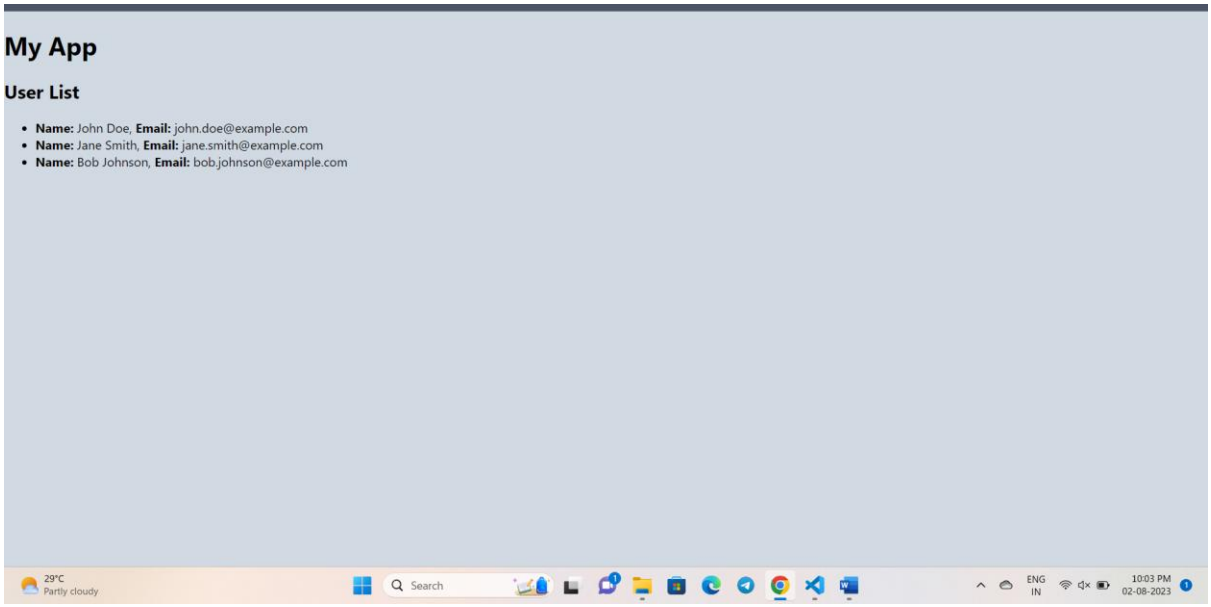
Question 2: Create a User List component in React that receives an array of user objects as props. The component should render a list of users with their

names and email addresses. Each user should be displayed as a list item with their name and email. Write the code for the User List component.





OUTPUT:


My App
User List
Name: John Doe, Email: john.doe@example.com
Name: Jane Smith, Email: jane.smith@example.com
Name: Bob Johnson, Email: bob.johnson@example.com
29°C
Partly cloudy
Search
ENG
IN
10:03 PM
02-08-2023