Week 6

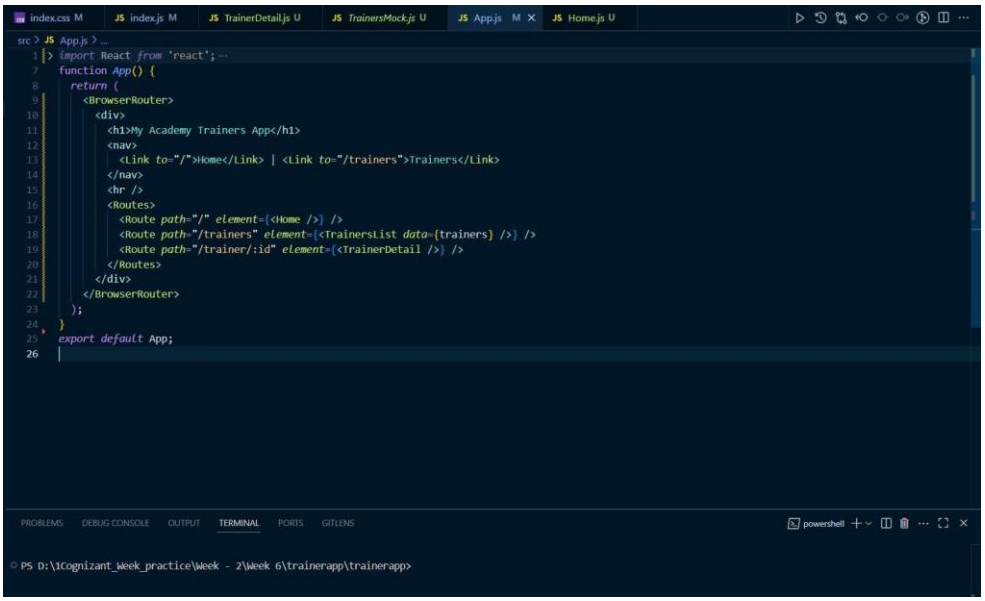
6. ReactJS-HOL

Implement a Simple Navigation Menu

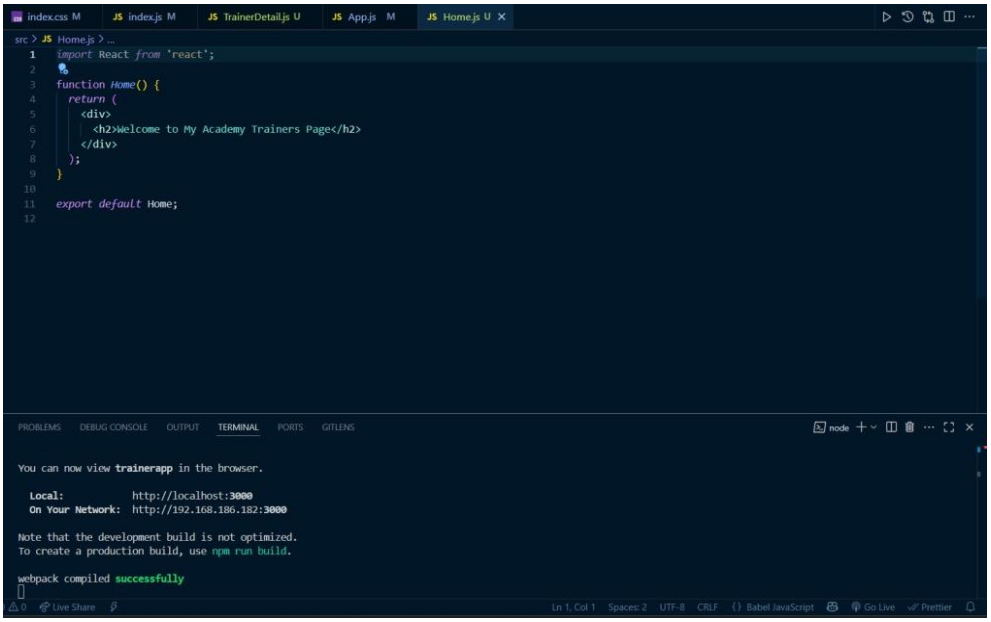
Add Basic Routes (install, configure)

Use Routes in React Applications

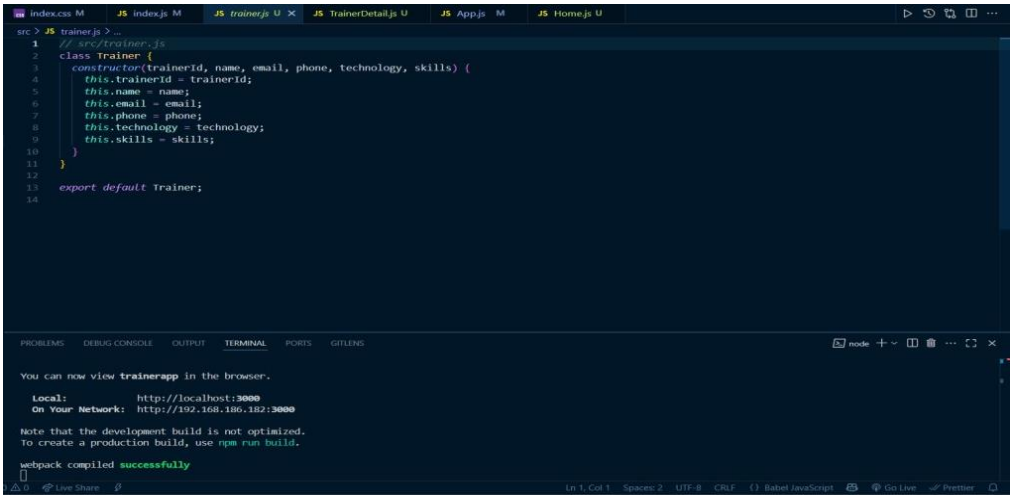
App.js



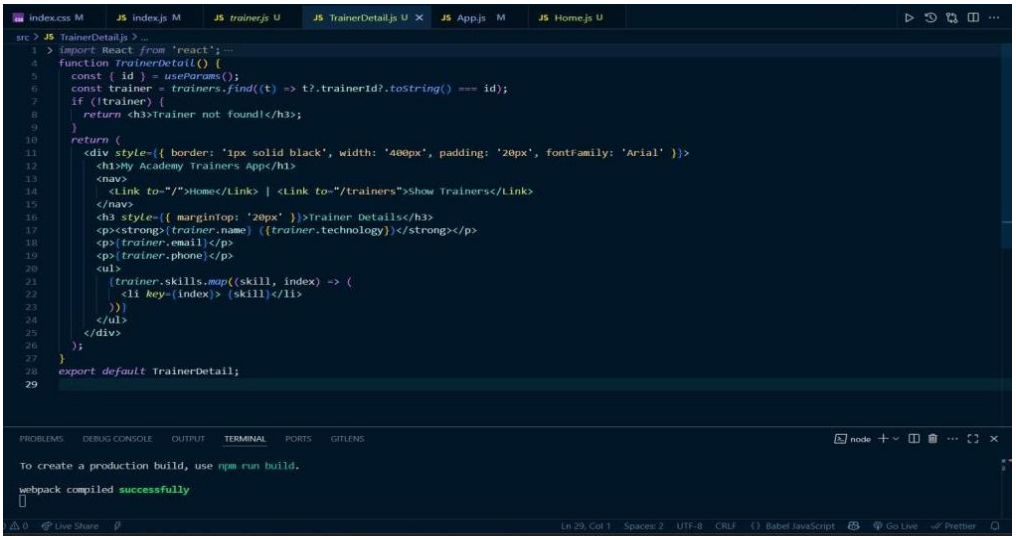
Home.js



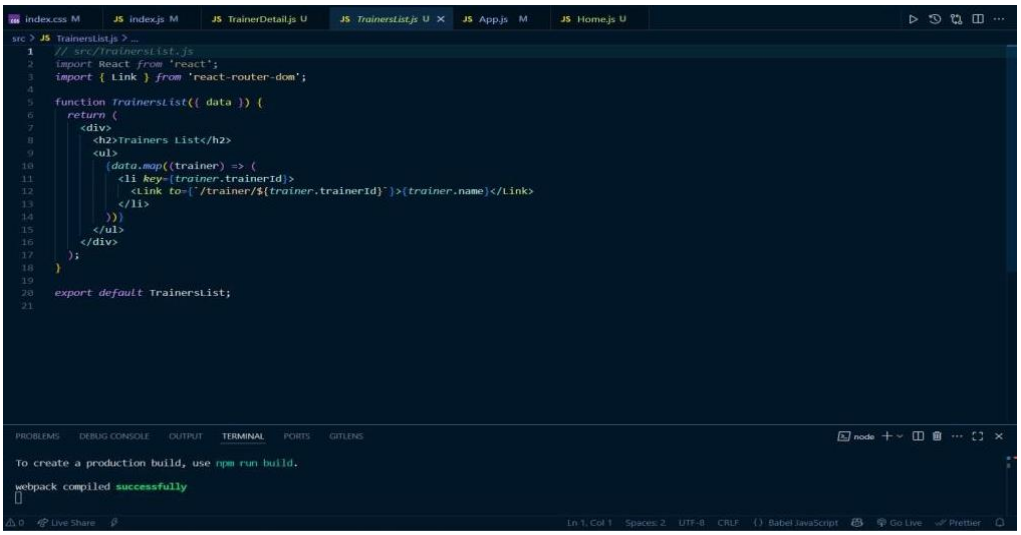
Trainer.js



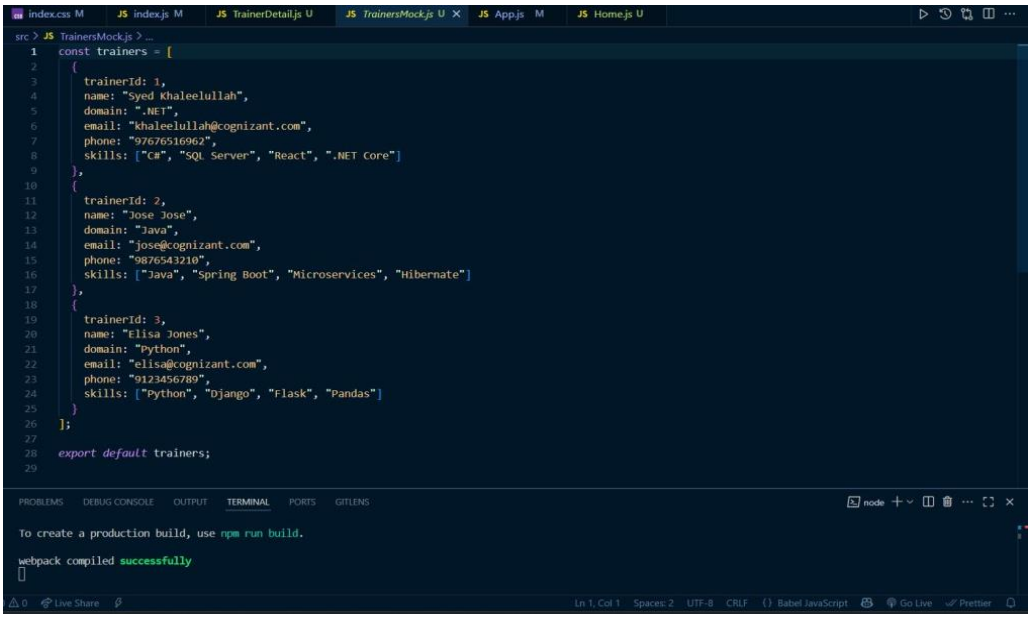
TrainerDetails.js



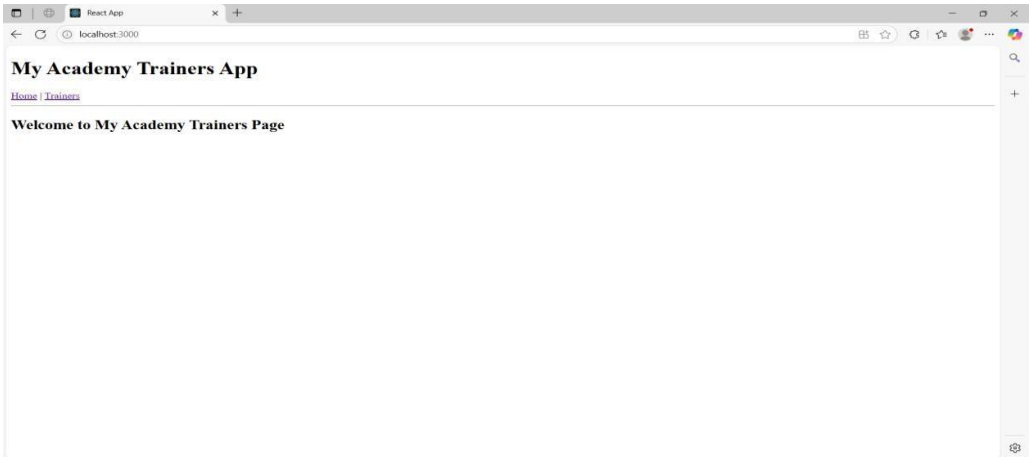
TrainersList.js

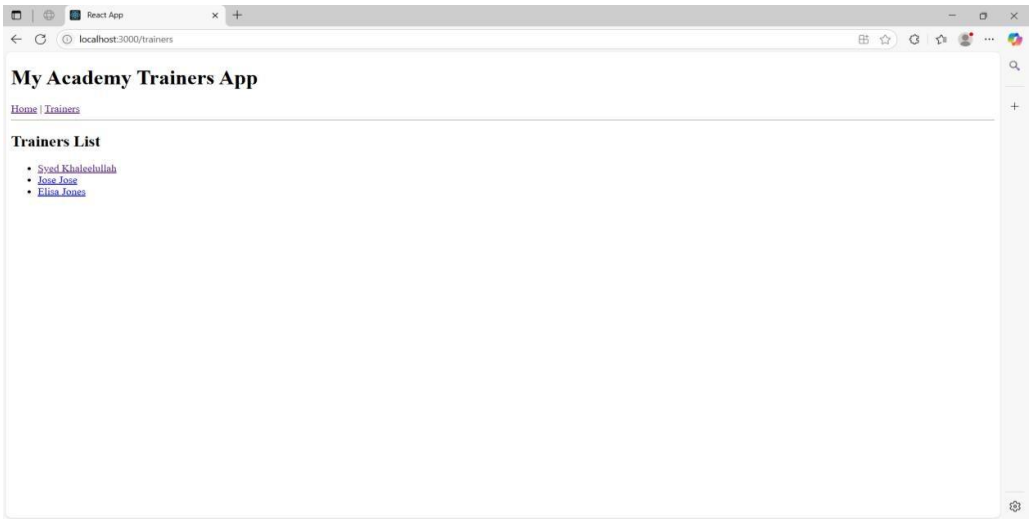


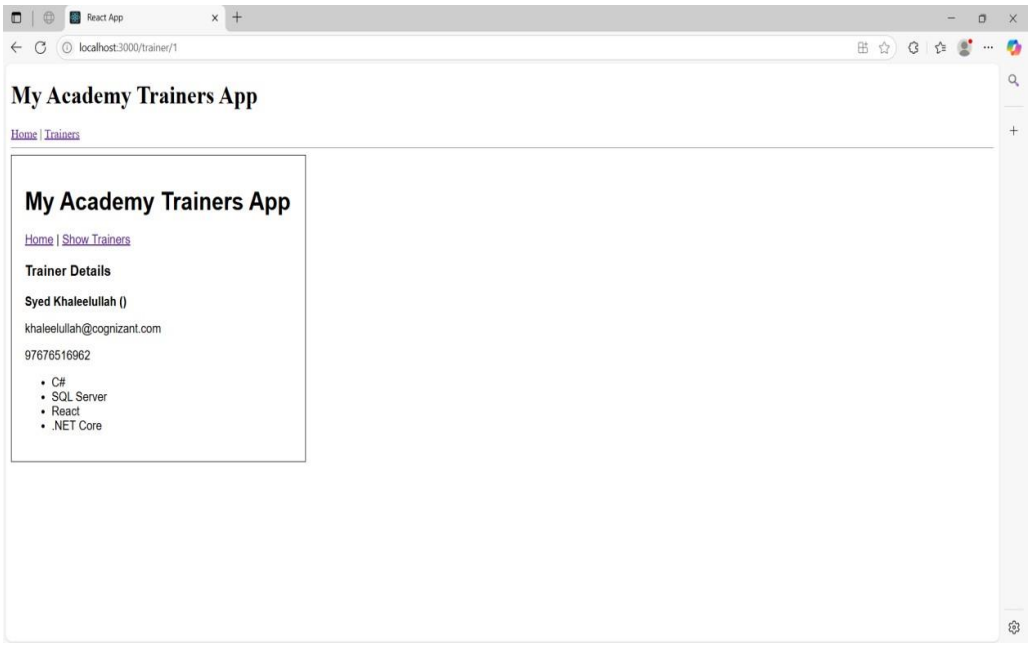
TrainersMock.js

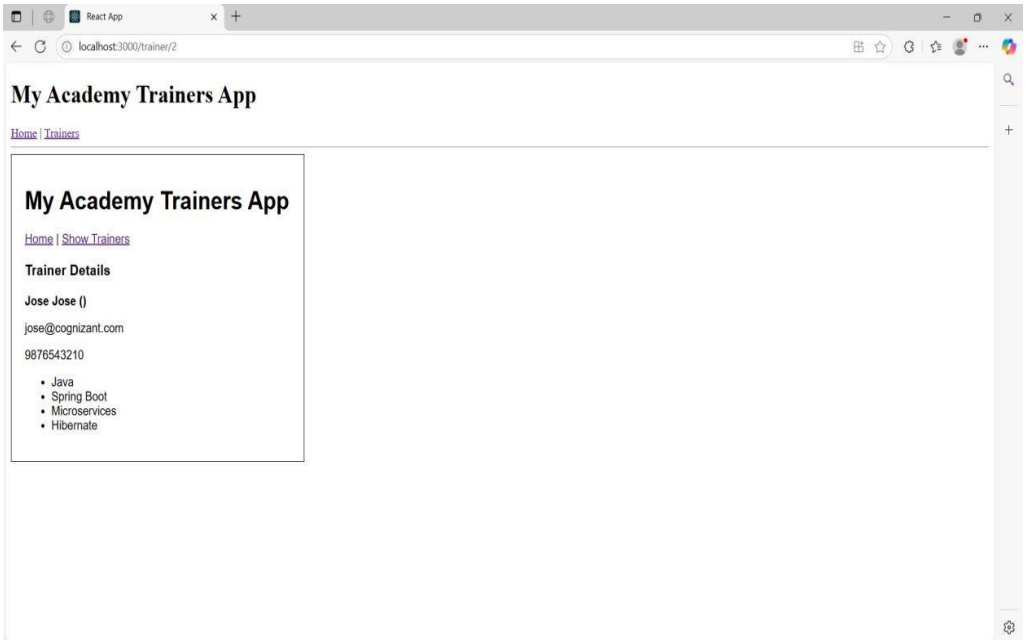


Output






React App
localhost:3000/trainer/1
My Academy Trainers App
Home | Trainers
My Academy Trainers App
Home | Show Trainers
Trainer Details
Syed Khaleelullah ()
khaleelullah@cognizant.com
97676516962
C#
SQL Server
React
.NET Core


React App
localhost:3000/trainer/2
My Academy Trainers App
Home | Trainers
My Academy Trainers App
Home | Show Trainers
Trainer Details
Jose Jose ()
jose@cognizant.com
9876543210
Java
Spring Boot
Microservices
Hibernate

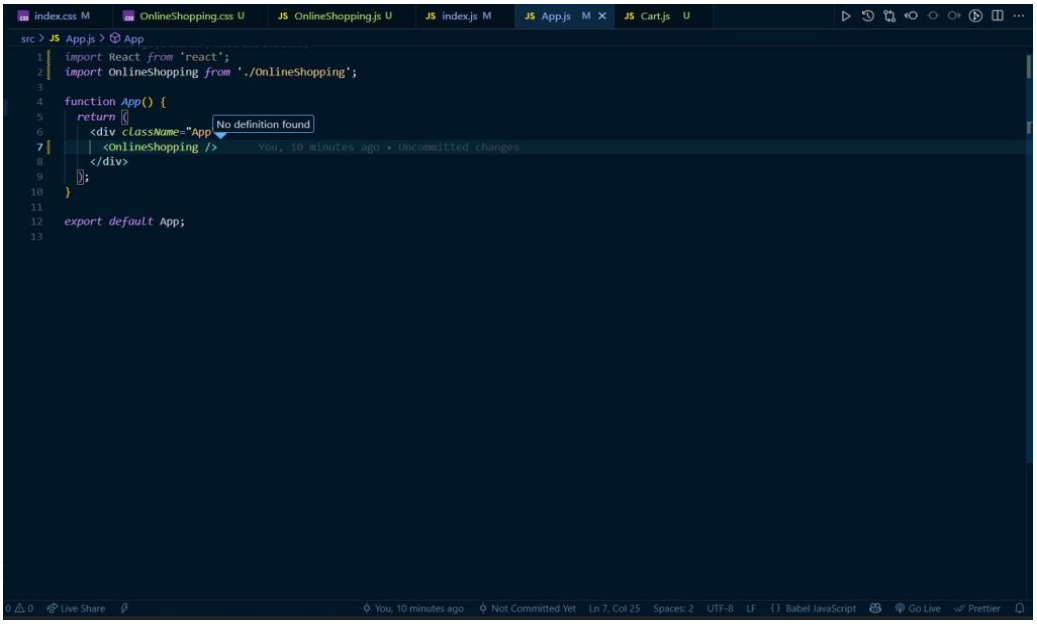
7. ReactJS-HOL

Create a React Application named “shoppingapp” with a class component named “OnlineShopping” and “Cart”.

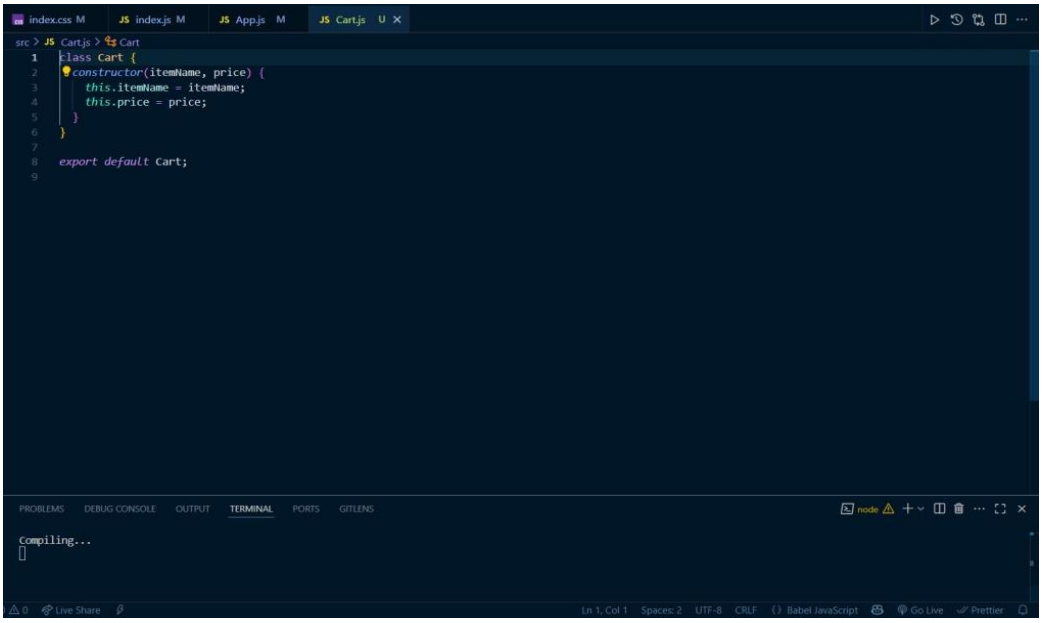
In Cart class, create 2 properties as mentioned below:

Itemname Price

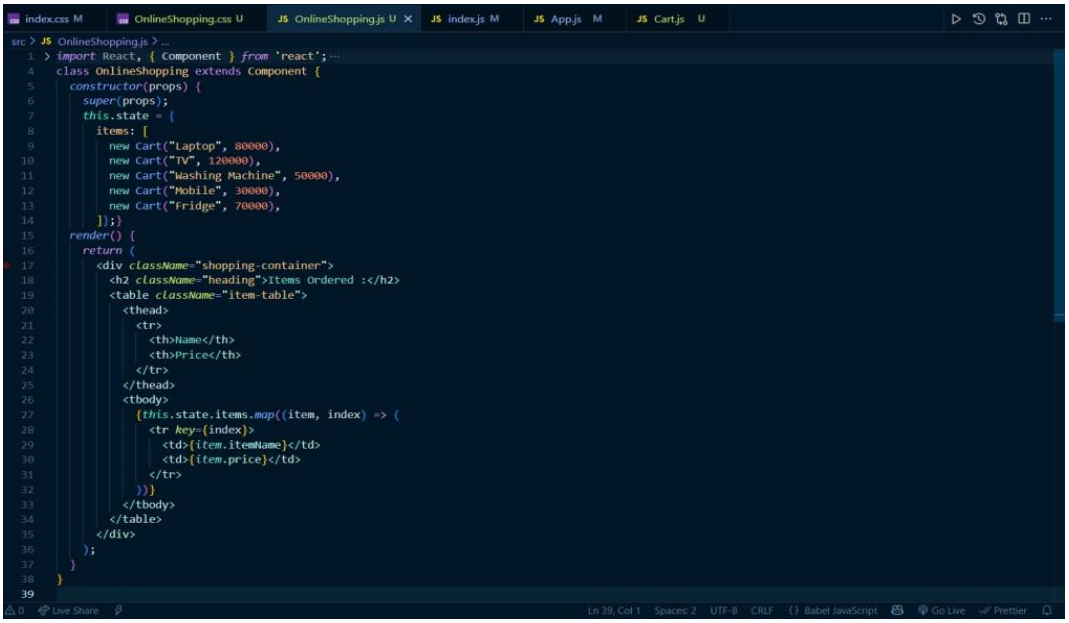
App.js



Cart.js

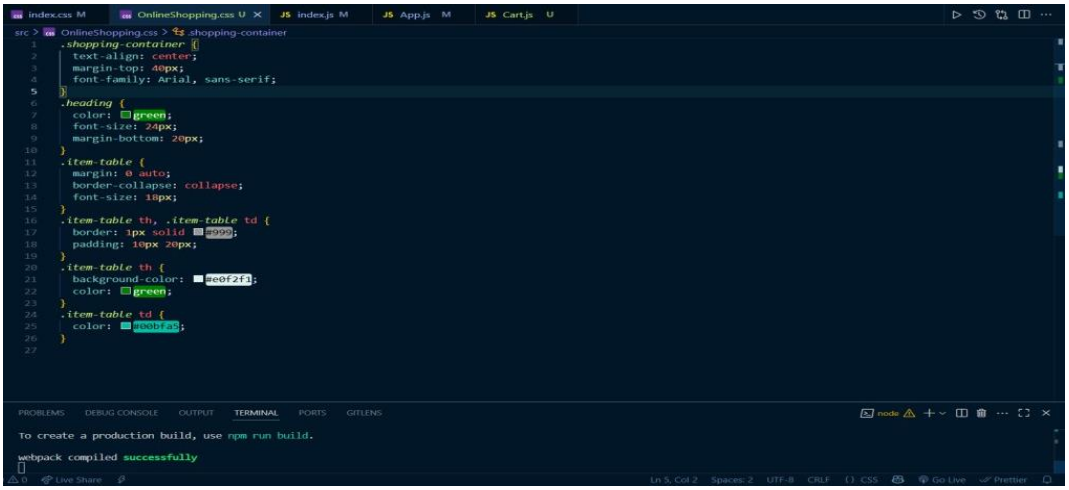


OnlineShopping.js



```
import React, { Component } from 'react';
class OnlineShopping extends Component {
  constructor(props) {
    super(props);
    this.state = {
      items: [
        new Cart("Laptop", 80000),
        new Cart("TV", 120000),
        new Cart("Washing Machine", 50000),
        new Cart("Mobile", 30000),
        new Cart("Fridge", 70000),
      ]};}
  render() {
    return (
      <div className="shopping-container">
        <h2 className="heading">Items Ordered :</h2>
        <table className="item-table">
          <thead>
            <tr>
              <th>Name</th>
              <th>Price</th>
            </tr>
          </thead>
          <tbody>
            {this.state.items.map((item, index) => (
              <tr key={index}>
                <td>{item.itemName}</td>
                <td>{item.price}</td>
              </tr>
            ))}
          </tbody>
        </table>
      </div>
    );
  }
}
```

OnlineShopping.css



```
.shopping-container {
  text-align: center;
  margin-top: 40px;
  font-family: Arial, sans-serif;
}
.heading {
  color: green;
  font-size: 24px;
  margin-bottom: 20px;
}
.item-table {
  margin: 0 auto;
  border-collapse: collapse;
  font-size: 18px;
}
.item-table th, .item-table td {
  border: 1px solid #999;
  padding: 10px 20px;
}
.item-table th {
  background-color: #e0f2f1;
  color: green;
}
.item-table td {
  color: #00bfa5;
}
```

```
To create a production build, use npm run build.

webpack compiled successfully
```

Output

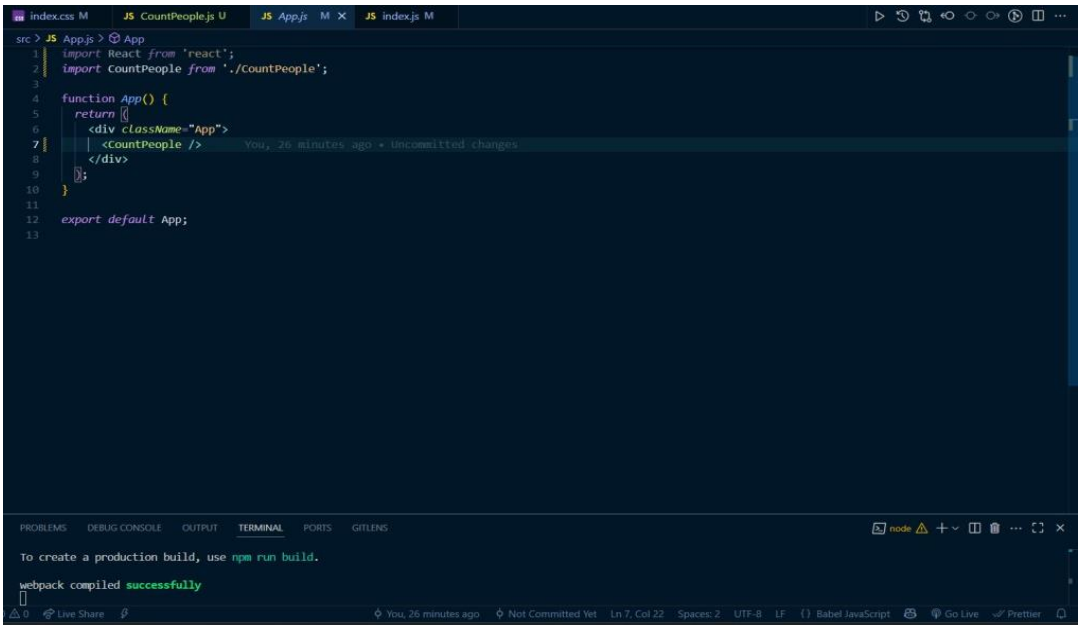
8. ReactJS-HOL

Create a React App “counterapp” which will have a component named “CountPeople” which will have 2 methods.

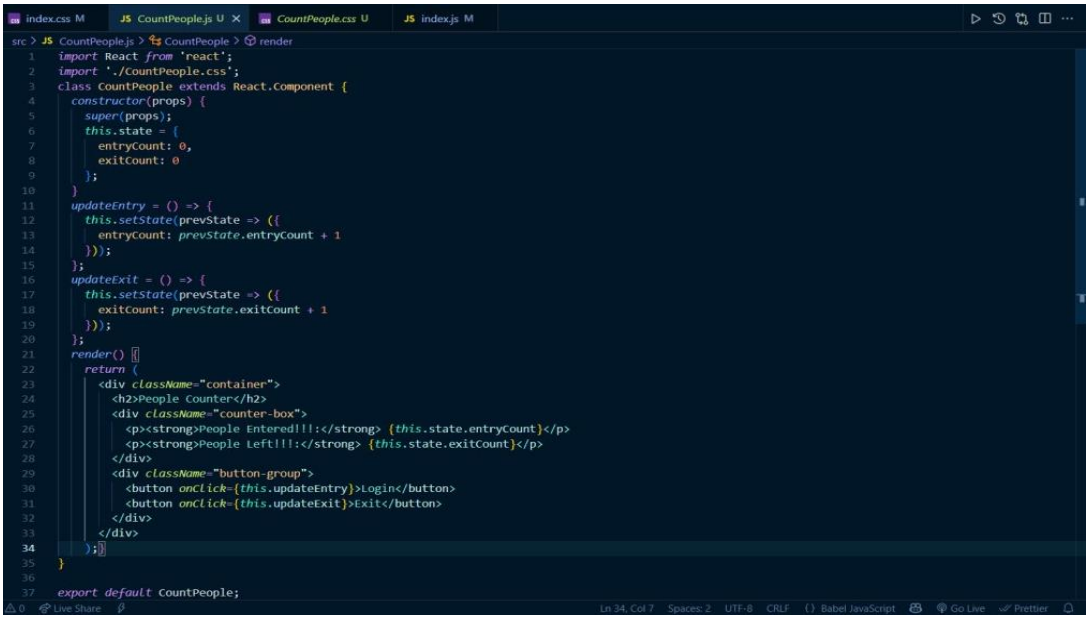
UpdateEntry() → which will display the number of people who entered the mall.

UpdateExit() → which will display the number of people who exited the mall.

App.js



CountPeople.js



CountPeople.css

```css
.container {
  text-align: center;
  margin-top: 50px;
  font-family: Arial, sans-serif;
}

.counter-box {
  border: 2px solid #4CAF50;
  border-radius: 10px;
  display: inline-block;
  padding: 20px;
  margin-bottom: 20px;
  background-color: #f9fff9;
}

.button-group button {
  padding: 10px 20px;
  margin: 10px;
  border: none;
  border-radius: 5px;
  font-weight: bold;
  cursor: pointer;
  color: white;
}

.button-group button:first-child {
  background-color: #4CAF50;
}

.button-group button:last-child {
  background-color: #f44336;
}
```

Output