**Digital Nurture 4.0 Week 6**

**REACT**

ADDITIONAL EXERCISES

**File name: 6. ReactJS-HOL**

1. **Create React App**

npx create-react-app TrainersApp

cd TrainersApp

**2. Create Trainer.js model**

Inside src/Trainer.js:

class Trainer {

constructor(trainerId, name, email, phone, technology, skills) {

this.trainerId = trainerId;

this.name = name;

this.email = email;

this.phone = phone;

this.technology = technology;

this.skills = skills;

}

}

export default Trainer;

**3. Create mock data in TrainersMock.js**

Inside src/TrainersMock.js:

const trainers = [

{

trainerId: "T101",

name: "John Doe",

email: "john@example.com",

phone: "1234567890",

technology: "Java",

skills: ["Spring", "Hibernate", "Microservices"]

},

{

trainerId: "T102",

name: "Jane Smith",

email: "jane@example.com",

phone: "9876543210",

technology: "React",

skills: ["Hooks", "Redux", "Router"]

}

];

export default trainers;

**4. Install React Router**

npm install react-router-dom

**5. Create TrainersList.js component**

Inside src/TrainersList.js:

import React from 'react';

import { Link } from 'react-router-dom';

const TrainersList = ({ trainers }) => {

return (

<div>

<h2>Trainers List</h2>

<ul>

{trainers.map((trainer) => (

<li key={trainer.trainerId}>

<Link to={`/trainer/${trainer.trainerId}`}>{trainer.name}</Link>

</li>

))}

</ul>

</div>

);

};

export default TrainersList;

**6. Create TrainerDetails.js component**

Inside src/TrainerDetails.js:

import React from 'react';

import { useParams } from 'react-router-dom';

import trainers from './TrainersMock';

const TrainerDetails = () => {

const { id } = useParams();

const trainer = trainers.find(t => t.trainerId === id);

return (

<div>

<h2>{trainer.name}</h2>

<p><strong>Email:</strong> {trainer.email}</p>

<p><strong>Phone:</strong> {trainer.phone}</p>

<p><strong>Technology:</strong> {trainer.technology}</p>

<p><strong>Skills:</strong> {trainer.skills.join(", ")}</p>

</div>

);

};

export default TrainerDetails;

**7. Create Home.js component**

Inside src/Home.js

import React from 'react';

const Home = () => {

return (

<div>

<h2>Welcome to Cognizant Trainers App</h2>

<p>This app shows trainer details and their skills.</p>

</div>

);

};

export default Home;

8. **Update App.js with Router**

import React from 'react';

import { BrowserRouter, Routes, Route, Link } from 'react-router-dom';

import Home from './Home';

import TrainersList from './TrainersList';

import TrainerDetails from './TrainerDetails';

import trainers from './TrainersMock';

const App = () => {

return (

<BrowserRouter>

<div>

<h1>Trainer Management</h1>

<nav>

<Link to="/">Home</Link> |{" "}

<Link to="/trainers">Trainers List</Link>

</nav>

<hr />

<Routes>

<Route path="/" element={<Home />} />

<Route path="/trainers" element={<TrainersList trainers={trainers} />} />

<Route path="/trainer/:id" element={<TrainerDetails />} />

</Routes>

</div>

</BrowserRouter>

);

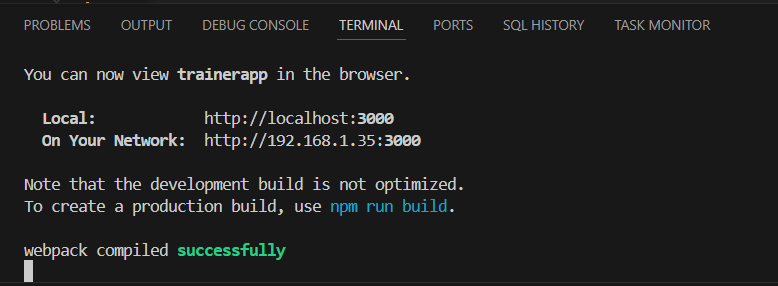
};

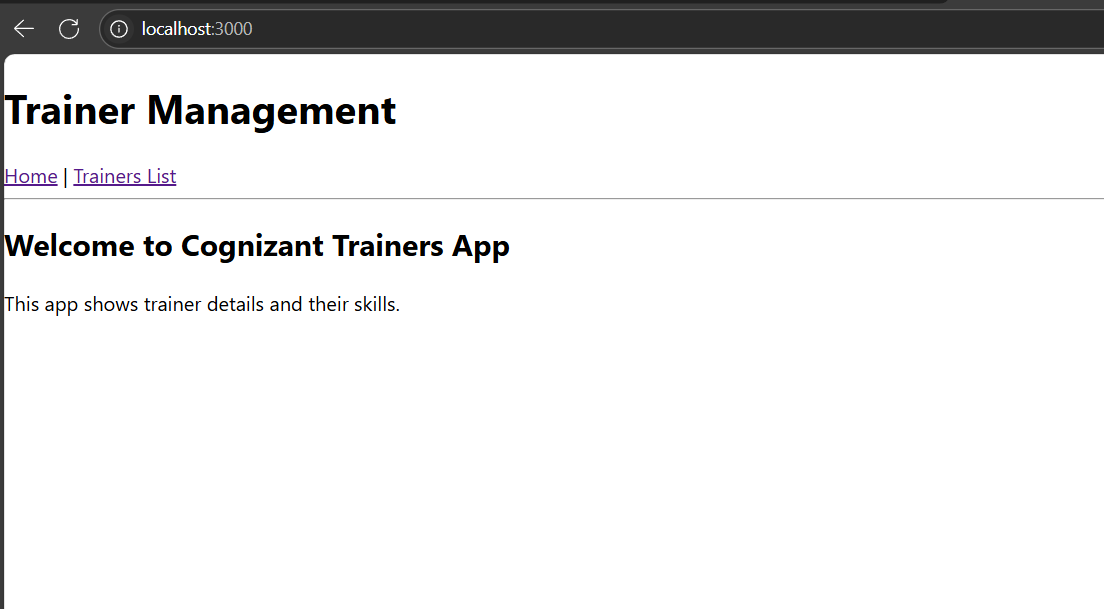
export default App;

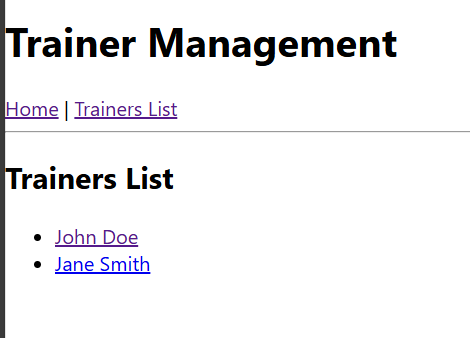
9. **Run the App**

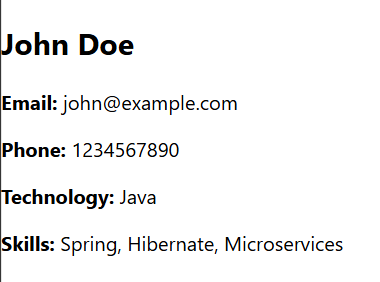
npm start

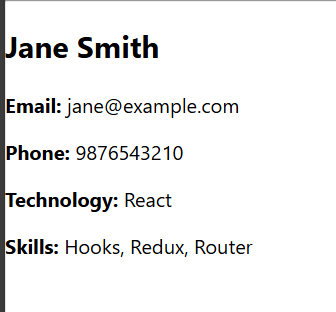
OUTPUT:











**File name: 7. ReactJS-HOL**

**1. Create React App**

Open your terminal and run:

npx create-react-app shoppingapp

cd shoppingapp

**2. Create Cart.js – Class Component for Cart**

import React from 'react';

class Cart extends React.Component {

render() {

return (

<div style={{ border: "1px solid gray", padding: "10px", margin: "10px" }}>

<h3>Item: {this.props.itemname}</h3>

<p>Price: ₹{this.props.price}</p>

</div>

);

}

}

Cart.defaultProps = {

itemname: "Unknown Item",

price: 0

};

export default Cart;

3. Create OnlineShopping.js – Class Component for Main App

import React from 'react';

import Cart from './Cart';

class OnlineShopping extends React.Component {

render() {

const items = [

{ itemname: 'Laptop', price: 45000 },

{ itemname: 'Headphones', price: 2000 },

{ itemname: 'Smartphone', price: 15000 },

{ itemname: 'Keyboard', price: 800 },

{ itemname: 'Mouse', price: 500 }

];

return (

<div>

<h1>Welcome to Online Shopping</h1>

{items.map((item, index) => (

<Cart key={index} itemname={item.itemname} price={item.price} />

))}

</div>

);

}

}

export default OnlineShopping;

4. Use ReactDOM.render() in index.js

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import OnlineShopping from './OnlineShopping';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(

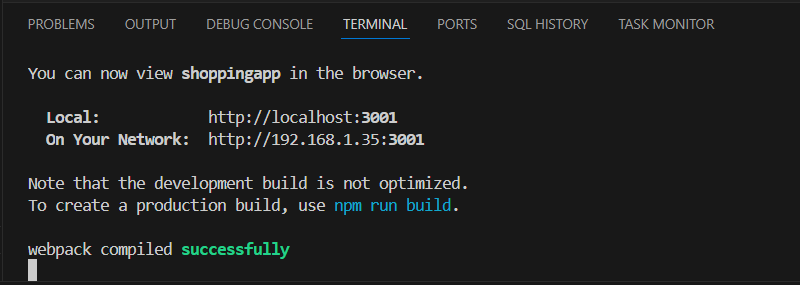
<React.StrictMode>

<OnlineShopping />

</React.StrictMode>

);

Output:





**File name: 8. ReactJS-HOL**

**1. Create React App**

Open terminal or command prompt and run:

npx create-react-app counterapp

cd counterapp

1. Create CountPeople.js in src folder

import React from 'react';

class CountPeople extends React.Component {

constructor(props) {

super(props);

this.state = {

entryCount: 0,

exitCount: 0

};

}

updateEntry = () => {

this.setState(prevState => ({

entryCount: prevState.entryCount + 1

}));

};

updateExit = () => {

this.setState(prevState => ({

exitCount: prevState.exitCount + 1

}));

};

render() {

return (

<div style={{ margin: '50px', padding: '20px', textAlign: 'center', border: '2px solid gray', borderRadius: '10px', width: '400px', marginLeft: 'auto', marginRight: 'auto' }}>

<h1>Mall Entry Counter</h1>

<p><strong>Number of People Entered:</strong> {this.state.entryCount}</p>

<p><strong>Number of People Exited:</strong> {this.state.exitCount}</p>

<button onClick={this.updateEntry} style={{ marginRight: '10px', padding: '10px 20px' }}>Login</button>

<button onClick={this.updateExit} style={{ padding: '10px 20px' }}>Exit</button>

</div>

);

}

}

export default CountPeople;

1. Modify App.js to use CountPeople component

import React from 'react';

import './App.css';

import CountPeople from './CountPeople';

function App() {

return (

<div className="App">

<CountPeople />

</div>

);

}

export default App;

**4. Run the Application**

In terminal:

npm start

OUTPUT:

