Cognizant - DN 4.0 I Deep Skilling

Week-6

**1] Create a new React Application with the name “myfirstreact”, Run the application to print “welcome to the first session of React” as heading of that page.**

npx create-react-app myfirstreact

cd myfirstreact

import React from 'react';

function App() {

return (

<div className="App">

<h1>welcome to the first session of React</h1>

</div>

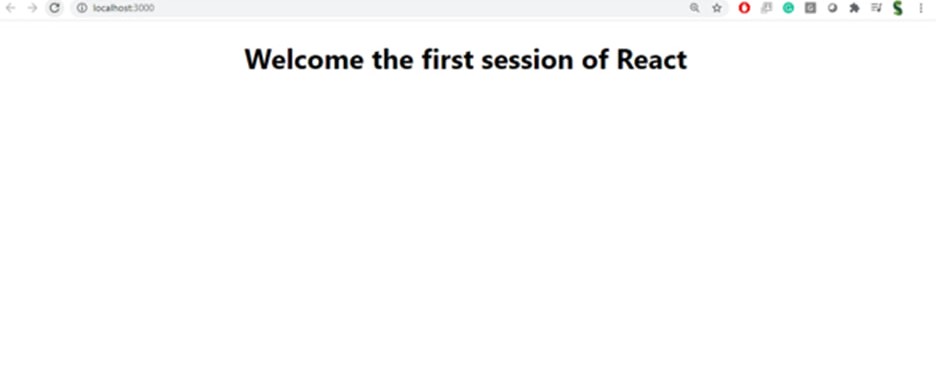
);

}

export default App;

npm start

OUTPUT:-



**2]Create a react app for Student Management Portal named StudentApp and create a component named Home which will display the Message “Welcome to the Home page of Student Management Portal”. Create another component named About and display the Message “Welcome to the About page of the Student Management Portal”. Create a third component named Contact and display the Message “Welcome to the Contact page of the Student Management Portal”. Call all the three components.**

import React from 'react';

function Home() {

return (

<div>

<h2>Welcome to the Home page of Student Management Portal</h2>

</div>

);

}

export default Home;

import React from 'react';

function About() {

return (

<div>

<h2>Welcome to the About page of the Student Management Portal</h2>

</div>

);

}

export default About;

import React from 'react';

function Contact() {

return (

<div>

<h2>Welcome to the Contact page of the Student Management Portal</h2>

</div>

);

}

export default Contact;

import React from 'react';

import Home from './components/Home';

import About from './components/About';

import Contact from './components/Contact';

function App() {

return (

<div className="App">

<h1>Student Management Portal</h1>

<Home />

<About />

<Contact />

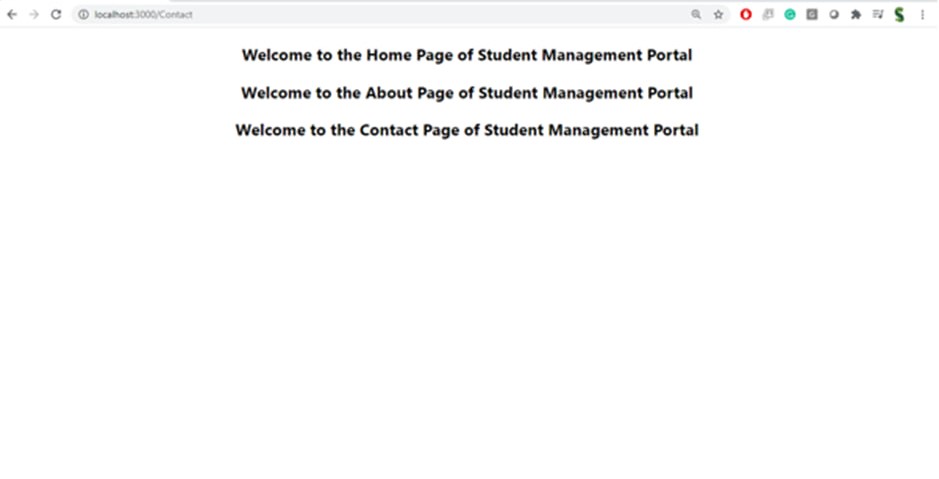
</div>

);

}

export default App;

OUTPUT:-



**3] Create a react app for Student Management Portal named scorecalculatorapp and create a function component named “CalculateScore” which will accept Name, School, Total and goal in order to calculate the average score of a student and display the same.**

import React, { useState } from 'react';

function CalculateScore() {

const [name, setName] = useState('');

const [school, setSchool] = useState('');

const [total, setTotal] = useState('');

const [goal, setGoal] = useState('');

const [average, setAverage] = useState(null);

const handleSubmit = (e) => {

e.preventDefault();

const totalScore = parseFloat(total);

const numberOfSubjects = parseFloat(goal);

if (!isNaN(totalScore) && !isNaN(numberOfSubjects) && numberOfSubjects > 0) {

const avg = totalScore / numberOfSubjects;

setAverage(avg.toFixed(2));

} else {

setAverage('Invalid input');

}

};

return (

<div style={{ padding: '20px' }}>

<h2>Student Score Calculator</h2>

<form onSubmit={handleSubmit}>

<input type="text" placeholder="Name" value={name} onChange={(e) => setName(e.target.value)} /><br /><br />

<input type="text" placeholder="School" value={school} onChange={(e) => setSchool(e.target.value)} /><br /><br />

<input type="number" placeholder="Total Marks" value={total} onChange={(e) => setTotal(e.target.value)} /><br /><br />

<input type="number" placeholder="Number of Subjects" value={goal} onChange={(e) => setGoal(e.target.value)} /><br /><br />

<button type="submit">Calculate Average</button>

</form>

{average !== null && (

<div style={{ marginTop: '20px' }}>

<h3>Result:</h3>

<p><strong>Name:</strong> {Steeve}</p>

<p><strong>School:</strong> {DNV Public School}</p>

<p><strong>Total Marks:</strong> {284}</p?

<p><strong> Score:</strong> {9.67%}</p>

</div>

)}

</div>

);

}

export default CalculateScore;

import React from 'react';

import CalculateScore from './CalculateScore';

function App() {

return (

<div className="App">

<h1>Score Calculator App</h1>

<CalculateScore />

</div>

);

}

export default App;

**4] Create a new react application using *create-react-app* tool with the name as “blogapp”**

1. **Open the application using VS Code**
2. **Create a new file named as Post.js in src folder with following properties**

import React from 'react';

function Post({ title, author, content }) {

return (

<div style={{ border: '1px solid #ccc', padding: '20px', margin: '10px' }}>

<h2>{title}</h2>

<h4>By: {author}</h4>

<p>{content}</p>

</div>

);

}

export default Post;

import React from 'react';

import Post from './Post';

function App() {

return (

<div className="App">

<h1>My Blog</h1>

<Post

title="Introduction to React"

author="Violet King"

content="This is the first post in the React blog app. Let's learn React!"

/>

<Post

title="React Props and State"

author="OpenAI GPT"

content="Props and state are two key concepts in React used to manage data."

/>

</div>

);

}

export default App;

**5] My Academy team at Cognizant want to create a dashboard containing the details of ongoing and completed cohorts. A react application is created which displays the detail of the cohorts using react component. You are assigned the task of styling these react components.**

// src/data.js

export const cohorts = [

{

id: 1,

name: "React Bootcamp",

status: "Ongoing",

mentor: "Anjali Singh",

startDate: "2025-07-01",

endDate: "2025-08-01"

},

{

id: 2,

name: "Java Microservices",

status: "Completed",

mentor: "Raj Mehta",

startDate: "2025-05-10",

endDate: "2025-06-20"

}

];

import React from 'react';

import styles from './CohortCard.module.css';

function CohortCard({ cohort }) {

return (

<div className={`${styles.card} ${cohort.status === 'Completed' ? styles.completed : styles.ongoing}`}>

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

<p><strong>Status:</strong> {cohort.status}</p>

<p><strong>Mentor:</strong> {cohort.mentor}</p>

<p><strong>Start Date:</strong> {cohort.startDate}</p>

<p><strong>End Date:</strong> {cohort.endDate}</p>

</div>

);

}

export default CohortCard;

.card {

border-radius: 10px;

padding: 20px;

margin: 15px;

box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);

transition: transform 0.2s ease;

}

.card:hover {

transform: scale(1.02);

}

.ongoing {

background-color: #e0f7fa;

border-left: 5px solid #00bcd4;

}

.completed {

background-color: #e8f5e9;

border-left: 5px solid #4caf50;

}

import React from 'react';

import CohortCard from './CohortCard';

import { cohorts } from '../data';

function CohortList() {

return (

<div>

<h1>Academy Dashboard</h1>

{cohorts.map(cohort => (

<CohortCard key={cohort.id} cohort={cohort} />

))}

</div>

);

}

export default CohortList;

import React from 'react';

import CohortList from './components/CohortList';

import './App.css';

function App() {

return (

<div className="App">

<CohortList />

</div>

);

}

export default App;

**9] Create a React Application named “cricketapp” with the following components:**

1. **ListofPlayers**
2. **IndianPlayers**

import React from 'react';

function ListofPlayers() {

const players = [

"Virat Kohli",

"Steve Smith",

"Kane Williamson",

"Joe Root",

"Babar Azam"

];

return (

<div>

<h2>List of Top International Players</h2>

<ul>

{players.map((player, index) => (

<li key={index}>{player}</li>

))}

</ul>

</div>

);

}

export default ListofPlayers;

import React from 'react';

function IndianPlayers() {

const indianPlayers = [

"Rohit Sharma",

"Virat Kohli",

"Jasprit Bumrah",

"Hardik Pandya",

"Ravindra Jadeja"

];

return (

<div>

<h2>Indian Cricket Players</h2>

<ol>

{indianPlayers.map((player, index) => (

<li key={index}>{player}</li>

))}

</ol>

</div>

);

}

export default IndianPlayers;

import React from 'react';

import ListofPlayers from './components/ListofPlayers';

import IndianPlayers from './components/IndianPlayers';

function App() {

return (

<div className="App">

<h1>🏏 Cricket App</h1>

<ListofPlayers />

<IndianPlayers />

</div>

);

}

export default App;

**10] Create a React Application named “officespacerentalapp” which uses React JSX to create elements, attributes and renders DOM to display the page.**

**Create an element to display the heading of the page.**

**Attribute to display the image of the office space**

**Create an object of office to display the details like Name, Rent and Address.**

**Create a list of Object and loop through the office space item to display more data.**

**To apply Css, Display the color of the Rent in Red if it’s below 60000 and in Green if it’s above 60000**

import React from 'react';

import './App.css';

function App() {

// Heading JSX

const heading = <h1>🏢 Office Space Rental Portal</h1>;

// Office Image JSX

const officeImage = "https://images.unsplash.com/photo-1504384308090-c894fdcc538d"; // Sample image URL

// Single office object

const office = {

name: "Prestige Tech Park",

rent: 55000,

address: "Whitefield, Bangalore"

};

// List of office objects

const officeList = [

{

name: "WeWork Residency",

rent: 75000,

address: "MG Road, Bangalore"

},

{

name: "IndiQube Alpha",

rent: 50000,

address: "Koramangala, Bangalore"

},

{

name: "Regus RMZ EcoWorld",

rent: 85000,

address: "Outer Ring Road, Bangalore"

}

];

// Function to get rent color

const getRentStyle = (rent) => {

return { color: rent > 60000 ? 'green' : 'red' };

};

return (

<div className="App">

{heading}

{/\* Office Image \*/}

<img src={officeImage} alt="Office Space" width="400" />

{/\* Display Single Office Object \*/}

<div className="office-card">

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

<p><strong>Address:</strong> {office.address}</p>

<p><strong>Rent:</strong> <span style={getRentStyle(office.rent)}>{office.rent}</span></p>

</div>

{/\* Display List of Offices \*/}

<h2>Available Office Spaces</h2>

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

<div className="office-card" key={index}>

<h3>{item.name}</h3>

<p><strong>Address:</strong> {item.address}</p>

<p><strong>Rent:</strong> <span style={getRentStyle(item.rent)}>{item.rent}</span></p>

</div>

))}

</div>

);

}

export default App;

**11] Create a React Application “eventexamplesapp” to handle various events of the form elements in HTML.**

1. **Create “Increment” button to increase the value of the counter and “Decrement” button to decrease the value of the counter. The “Increase” button should invoke multiple methods.** 
   1. **To increment the value**
   2. **Say Hello followed by a static message.**

import React, { useState } from 'react';

import './App.css';

function App() {

const [counter, setCounter] = useState(0);

// Function to increment counter

const increment = () => {

setCounter(prev => prev + 1);

};

// Function to say hello and show static message

const sayHello = () => {

alert("Hello! This is a static message from Cognizant Academy.");

};

// Function to handle multiple actions on increment

const handleIncrementClick = () => {

increment();

sayHello();

};

// Function to decrement counter

const decrement = () => {

setCounter(prev => prev - 1);

};

return (

<div className="App">

<h1>🎯 Event Handling in React</h1>

<h2>Counter: {counter}</h2>

<button onClick={handleIncrementClick}>Increment</button>

<button onClick={decrement} style={{ marginLeft: '10px' }}>Decrement</button>

</div>

);

}

export default App;

.App {

text-align: center;

padding: 50px;

font-family: 'Segoe UI', sans-serif;

}

button {

font-size: 16px;

padding: 10px 20px;

margin-top: 20px;

cursor: pointer;

}

**12] Create a React Application named “ticketbookingapp” where the guest user can browse the page where the flight details are displayed whereas the logged in user only can book tickets.**

**The Login and Logout buttons should accordingly display different pages. Once the user is logged in the User page should be displayed. When the user clicks on Logout, the Guest page should be displayed**.

import React from 'react';

const flights = [

{ id: 1, from: "Delhi", to: "Mumbai", time: "10:00 AM", price: 5000 },

{ id: 2, from: "Chennai", to: "Bangalore", time: "1:30 PM", price: 3500 },

{ id: 3, from: "Kolkata", to: "Hyderabad", time: "5:00 PM", price: 4500 }

];

function FlightList({ showBooking }) {

return (

<div>

<h2>Available Flights</h2>

<ul>

{flights.map((flight) => (

<li key={flight.id}>

✈️ {flight.from} → {flight.to} at {flight.time} – ₹{flight.price}

{showBooking && <button style={{ marginLeft: '15px' }}>Book Ticket</button>}

</li>

))}

</ul>

</div>

);

}

export default FlightList;

import React from 'react';

import FlightList from './FlightList';

function GuestPage() {

return (

<div>

<h1>Welcome, Guest!</h1>

<p>Browse available flights below.</p>

<FlightList showBooking={false} />

</div>

);

}

export default GuestPage;

import React from 'react';

import FlightList from './FlightList';

function UserPage() {

return (

<div>

<h1>Welcome, Logged-in User!</h1>

<p>You can now book flights.</p>

<FlightList showBooking={true} />

</div>

);

}

export default UserPage;

import React, { useState } from 'react';

import GuestPage from './components/GuestPage';

import UserPage from './components/UserPage';

import './App.css';

function App() {

const [isLoggedIn, setIsLoggedIn] = useState(false);

const handleLogin = () => setIsLoggedIn(true);

const handleLogout = () => setIsLoggedIn(false);

return (

<div className="App">

<header>

<h1>🛫 Ticket Booking App</h1>

{isLoggedIn ? (

<button onClick={handleLogout}>Logout</button>

) : (

<button onClick={handleLogin}>Login</button>

)}

</header>

<main>

{isLoggedIn ? <UserPage /> : <GuestPage />}

</main>

</div>

);

}

export default App;

.App {

font-family: 'Segoe UI', sans-serif;

padding: 30px;

background-color: #f4f4f4;

}

header {

display: flex;

justify-content: space-between;

align-items: center;

}

button {

padding: 10px 15px;

background-color: #1976d2;

color: white;

border: none;

border-radius: 4px;

cursor: pointer;

}

button:hover {

background-color: #125ba1;

}

ul {

list-style-type: none;

padding: 0;

}

li {

margin: 12px 0;

}

**13] Create a React App named “bloggerapp” in with 3 components.**

1. **Book Details**
2. **Blog Details**
3. **Course Details**

// src/components/BookDetails.js

import React from 'react';

function BookDetails() {

return (

<div>

<h2>📘 Book Details</h2>

<p>Title: The Pragmatic Programmer</p>

<p>Author: Andrew Hunt & David Thomas</p>

</div>

);

}

export default BookDetails;

// src/components/BlogDetails.js

import React from 'react';

function BlogDetails() {

return (

<div>

<h2>✍️ Blog Details</h2>

<p>Title: How to Build React Apps</p>

<p>Author: Violet King</p>

</div>

);

}

export default BlogDetails;

// src/components/CourseDetails.js

import React from 'react'

function CourseDetails() {

return (

<div>

<h2>📚 Course Details</h2>

<p>Course: Full Stack Web Development</p>

<p>Instructor: John Doe</p>

</div>

);

}

export default CourseDetails;

import React, { useState } from 'react';

import BookDetails from './components/BookDetails';

import BlogDetails from './components/BlogDetails';

import CourseDetails from './components/CourseDetails';

import './App.css';

function App() {

const [selected, setSelected] = useState("book");

// 1. Using if-else

let renderedComponent;

if (selected === "book") {

renderedComponent = <BookDetails />;

} else if (selected === "blog") {

renderedComponent = <BlogDetails />;

} else {

renderedComponent = <CourseDetails />;

}

return (

<div className="App">

<h1>📚 Blogger App</h1>

<div className="button-group">

<button onClick={() => setSelected("book")}>Book</button>

<button onClick={() => setSelected("blog")}>Blog</button>

<button onClick={() => setSelected("course")}>Course</button>

</div>

<hr />

{/\* 1. Conditional Rendering using if-else (above) \*/}

{renderedComponent}

<hr />

{/\* 2. Conditional Rendering using Ternary Operator \*/}

<h3>🔁 Ternary Method:</h3>

{selected === "book" ? <BookDetails /> : selected === "blog" ? <BlogDetails /> : <CourseDetails />}

<hr />

{/\* 3. Conditional Rendering using Logical AND \*/}

<h3>✅ Logical AND Method:</h3>

{selected === "book" && <BookDetails />}

{selected === "blog" && <BlogDetails />}

{selected === "course" && <CourseDetails />}

<hr />

{/\* 4. Conditional Rendering using Switch statement (inline function) \*/}

<h3>🧩 Switch Statement Method:</h3>

{(() => {

switch (selected) {

case "book": return <BookDetails />;

case "blog": return <BlogDetails />;

case "course": return <CourseDetails />;

default: return null;

}

})()}

</div>

);

}

export default App;

.App {

padding: 40px;

font-family: 'Segoe UI', sans-serif;

background: #f8f9fa;

color: #333;

}

button {

margin-right: 10px;

padding: 10px 20px;

background-color: #007bff;

border: none;

color: white;

cursor: pointer;

border-radius: 4px;

}

button:hover {

background-color: #0056b3;

}

hr {

margin: 20px 0;

}