WEEK-7

React-Mandatory And Additional Hands-On

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

1. **ListofPlayers**
2. **Declare an array with 11 players and store details of their names and scores using the map feature of ES56**
3. **Filter the players with scores below 70 using arrow functions of ES6.**
4. **IndianPlayers**
   1. **Display the Odd Team Player and Even Team players using the Destructuring features of ES6**
   2. **Declare two arrays T20players and RanjiTrophy players and merge the two arrays and display them using the Merge feature of ES6**

**Display these two components in the same home page using a simple if else in the flag variable**.

**ListofPlayers.js:**

import React from 'react';

const ListofPlayers = () => {

const players = [

{ name: "Kohli", score: 80 },

{ name: "Rahul", score: 50 },

{ name: "Jadeja", score: 45 },

{ name: "Rohit", score: 90 },

{ name: "Gill", score: 65 },

{ name: "Dhoni", score: 95 },

{ name: "Surya", score: 72 },

{ name: "Ishan", score: 55 },

{ name: "Axar", score: 35 },

{ name: "Pujara", score: 60 },

{ name: "Karun", score: 75 }

];

const lowScorers = players.filter(player => player.score < 70);

return (

<div>

<h2>All Players</h2>

<ul>

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

<li key={index}>{player.name} - {player.score}</li>

))}

</ul>

<h3>Players with Score Below 70</h3>

<ul>

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

<li key={index}>{player.name} - {player.score}</li>

))}

</ul>

</div>

);

};

export default ListofPlayers;

**IndianPlayers.js:**

import React from 'react';

const IndianPlayers = () => {

const team = {

oddTeam: ["Rohit", "Gill", "Dhoni"],

evenTeam: ["Kohli", "Rahul", "Jadeja"]

};

// Destructuring

const { oddTeam, evenTeam } = team;

const T20players = ["Surya", "Ishan", "Axar"];

const RanjiTrophyPlayers = ["Pujara", "Rahane", "Karun Nair"];

const allPlayers = [...T20players, ...RanjiTrophyPlayers];

return (

<div>

<h2>Odd Team Players</h2>

<ul>

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

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

))}

</ul>

<h2>Even Team Players</h2>

<ul>

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

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

))}

</ul>

<h2>All Players (T20 + Ranji)</h2>

<ul>

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

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

))}

</ul>

</div>

);

};

export default IndianPlayers;

**App.js:**

import React from 'react';

import ListofPlayers from './ListofPlayers';

import IndianPlayers from './IndianPlayers';

function App() {

const flag = true; // Change this to false to switch component

return (

<div className="App">

<h1>Cricket App</h1>

{flag ? <ListofPlayers /> : <IndianPlayers />}

</div>

);

}

export default App;

**OUTPUT:**

A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

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

**App.js:**

import React from 'react';

function App() {

// Office object array

const officeSpaces = [

{

name: 'DBS',

rent: 50000,

address: 'Chennai',

image: 'https://i.imgur.com/QC4Hh7e.png'

},

{

name: 'WeWork',

rent: 65000,

address: 'Bangalore',

image: 'https://i.imgur.com/AbC4hD9.jpeg'

}

];

return (

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

<h1 style={{ fontWeight: 'bold' }}>Office Space , at Affordable Range</h1>

{officeSpaces.map((office, index) => {

const rentStyle = {

color: office.rent < 60000 ? 'red' : 'green',

fontWeight: 'bold'

};

return (

<div key={index} style={{ marginBottom: '40px' }}>

<img

src={office.image}

alt={`Office - ${office.name}`}

width="300"

height="300"

/>

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

<p style={rentStyle}>Rent: Rs. {office.rent}</p>

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

</div>

);

})}

</div>

);

}

export default App;

**OUTPUT:**

A screenshot of a computer

AI-generated content may be incorrect.

**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.**
2. **Create a button “Say Welcome” which invokes the function which takes “welcome” as an argument.**

**App.js:**

import React, { Component } from 'react';

import CurrencyConverter from './CurrencyConverter';

class App extends Component {

constructor(props) {

super(props);

this.state = {

count: 5

};

// Binding `this` to methods

this.increment = this.increment.bind(this);

this.sayHello = this.sayHello.bind(this);

this.handleIncrement = this.handleIncrement.bind(this);

}

// Method to increment counter

increment() {

this.setState({ count: this.state.count + 1 });

}

// Method to display a hello message

sayHello() {

alert("Hello! Member1");

}

// Combine methods for Increment button

handleIncrement() {

this.increment();

this.sayHello();

}

// Decrement method

decrement = () => {

this.setState({ count: this.state.count - 1 });

};

// Function that takes argument

sayWelcome = (msg) => {

alert(msg);

};

// Synthetic event handler

handleClick = (event) => {

alert("I was clicked");

};

render() {

return (

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

<h3>{this.state.count}</h3>

<button onClick={this.handleIncrement}>Increment</button>

<br /><br />

<button onClick={this.decrement}>Decrement</button>

<br /><br />

<button onClick={() => this.sayWelcome("welcome")}>Say welcome</button>

<br /><br />

<button onClick={this.handleClick}>Click on me</button>

<hr />

<CurrencyConverter />

</div>

);

}

}

export default App;

**CurrencyConverter.js:**

import React, { useState } from 'react';

function CurrencyConverter() {

const [amount, setAmount] = useState('');

const [currency, setCurrency] = useState('');

const handleSubmit = (e) => {

e.preventDefault();

const inr = parseFloat(amount);

const euroRate = 0.011; // Example conversion rate

const result = inr \* euroRate;

alert(`₹${inr} = €${result.toFixed(2)}`);

};

return (

<div>

<h1 style={{ color: "green" }}>Currency Convertor!!!</h1>

<form onSubmit={handleSubmit}>

<div>

<label>Amount: </label>

<input

type="number"

value={amount}

onChange={(e) => setAmount(e.target.value)}

/>

</div>

<br />

<div>

<label>Currency: </label>

<textarea

value={currency}

onChange={(e) => setCurrency(e.target.value)}

/>

</div>

<br />

<button type="submit">Submit</button>

</form>

</div>

);

}

export default CurrencyConverter;

**OUTPUT:**

A screenshot of a computer

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**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.**

**GuestPage.js:**

import React from 'react';

function GuestPage() {

return (

<div>

<h1>Please sign up.</h1>

</div>

);

}

export default GuestPage;

**UserPage.js:**

import React from 'react';

function UserPage() {

return (

<div>

<h1>Welcome back</h1>

</div>

);

}

export default UserPage;

**LoginControl.js:**

import React, { useState } from 'react';

import GuestPage from './GuestPage';

import UserPage from './UserPage';

function LoginControl() {

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

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

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

let button;

let page;

if (isLoggedIn) {

button = <button onClick={handleLogoutClick}>Logout</button>;

page = <UserPage />;

} else {

button = <button onClick={handleLoginClick}>Login</button>;

page = <GuestPage />;

}

return (

<div>

{page}

{button}

</div>

);

}

export default LoginControl;

**App.js:**

import React from 'react';

import LoginControl from './components/LoginControl';

function App() {

return (

<div className="App">

<LoginControl />

</div>

);

}

export default App;

**OUTPUT:**

A screen shot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

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

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

**Implement this with as many ways possible of Conditional Rendering.**

**App.js:**

import React, { useState } from 'react';

import CourseDetails from './components/CourseDetails';

import BookDetails from './components/BookDetails';

import BlogDetails from './components/BlogDetails';

function App() {

const [showSection, setShowSection] = useState('all');

return (

<div style={{ display: 'flex', justifyContent: 'space-evenly', padding: 20 }}>

{(showSection === 'all' || showSection === 'courses') && <CourseDetails />}

{(showSection === 'all' || showSection === 'books') && <BookDetails />}

{(showSection === 'all' || showSection === 'blogs') && <BlogDetails />}

{/\* Buttons to change condition \*/}

<div style={{ position: 'fixed', bottom: 20, left: 20 }}>

<button onClick={() => setShowSection('all')}>Show All</button>

<button onClick={() => setShowSection('courses')}>Show Courses</button>

<button onClick={() => setShowSection('books')}>Show Books</button>

<button onClick={() => setShowSection('blogs')}>Show Blogs</button>

</div>

</div>

);

}

export default App;

**CourseDetails.js:**

import React from 'react';

const courses = [

{ name: 'Angular', date: '4/5/2021' },

{ name: 'React', date: '6/3/20201' }

];

function CourseDetails() {

return (

<div style={{ borderRight: '4px solid green', paddingRight: 20 }}>

<h2>Course Details</h2>

{courses.map((course, index) => (

<div key={index}>

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

<p>{course.date}</p>

</div>

))}

</div>

);

}

export default CourseDetails;

BookDetails.js:

import React from 'react';

const books = [

{ title: 'Master React', price: 670 },

{ title: 'Deep Dive into Angular 11', price: 800 },

{ title: 'Mongo Essentials', price: 450 }

];

function BookDetails() {

return (

<div style={{ borderRight: '4px solid green', paddingRight: 20 }}>

<h2>Book Details</h2>

{books.map((book, index) => (

<div key={index}>

<h3>{book.title}</h3>

<p>{book.price}</p>

</div>

))}

</div>

);

}

export default BookDetails;

**BlogDetails.js:**

import React from 'react';

const blogs = [

{

title: 'React Learning',

author: 'Stephen Biz',

content: 'Welcome to learning React!'

},

{

title: 'Installation',

author: 'Schewzdenier',

content: 'You can install React from npm.'

}

];

function BlogDetails() {

return (

<div>

<h2>Blog Details</h2>

{blogs.map((blog, index) => (

<div key={index}>

<h3>{blog.title}</h3>

<strong>{blog.author}</strong>

<p>{blog.content}</p>

</div>

))}

</div>

);

}

export default BlogDetails;

**OUTPUT:**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Developers of Apps Centric Solutions have created an employee management application which supports light and dark themes for the buttons. The current solution uses the react state and props to provide the theme name to be used from App component to Employee List component and from there to Employee Card component. Quality assurance team analyzed the solutions and found the technique being used to be a substandard one. React architect suggested to use the react context API to share the theme name with nested child components instead of passing them down using props from the parent component.**

**ThemeContext.js:**

import { createContext } from 'react';

const ThemeContext = createContext('light');

export default ThemeContext;

**App.js:**

import React, { useState } from 'react';

import EmployeesList from './EmployeesList';

import ThemeContext from './ThemeContext';

function App() {

const [theme, setTheme] = useState('light');

const toggleTheme = () => {

setTheme(prev => (prev === 'light' ? 'dark' : 'light'));

};

return (

<ThemeContext.Provider value={theme}>

<div className={`App ${theme}`}>

<button onClick={toggleTheme}>

Toggle Theme (Current: {theme})

</button>

<EmployeesList />

</div>

</ThemeContext.Provider>

);

}

export default App;

**EmployeesList.js:**

import React from 'react';

import EmployeeCard from './EmployeeCard';

function EmployeesList() {

const employees = [

{ id: 1, name: 'Alice' },

{ id: 2, name: 'Bob' },

{ id: 3, name: 'Charlie' }

];

return (

<div>

{employees.map(emp => (

<EmployeeCard key={emp.id} employee={emp} />

))}

</div>

);

}

export default EmployeesList;

**EmployeeCard.js:**

import React, { useContext } from 'react';

import ThemeContext from './ThemeContext';

function EmployeeCard({ employee }) {

const theme = useContext(ThemeContext); // Access theme from context

return (

<div className={`employee-card ${theme}`}>

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

<button className={`btn ${theme}`}>View Profile</button>

</div>

);

}

export default EmployeeCard;

**App.css:**

.App {

font-family: sans-serif;

padding: 20px;

}

.employee-card {

border: 1px solid #ccc;

margin: 10px;

padding: 10px;

border-radius: 8px;

}

.employee-card.light {

background-color: #f9f9f9;

color: #000;

}

.employee-card.dark {

background-color: #333;

color: #fff;

}

.btn {

padding: 8px 12px;

border: none;

border-radius: 4px;

cursor: pointer;

}

.btn.light {

background-color: #fff;

color: #000;

border: 1px solid #ccc;

}

.btn.dark {

background-color: #000;

color: #fff;

border: 1px solid #555;

}

**OUTPUT:**

A screen shot of a computer

AI-generated content may be incorrect.

**Create a React App named “ticketraisingapp” which will help to raise a complaint and get it resolved.**

**Create a component named “ComplaintRegister” with a form containing a textbox to enter the employee name and a textarea to enter the complaint. Use “handleSubmit” event of the button to submit the complaint and generate a Reference number for further follow ups in the alert box.**

**ComplaintRegister.js:**

import React, { useState } from 'react';

function ComplaintRegister() {

const [employeeName, setEmployeeName] = useState('');

const [complaint, setComplaint] = useState('');

const handleSubmit = (e) => {

e.preventDefault();

if (!employeeName || !complaint) {

alert('Please fill out both fields.');

return;

}

// Generate a simple reference number

const referenceNumber = 'REF' + Math.floor(Math.random() \* 1000000);

alert(`Complaint submitted successfully!\nReference No: ${referenceNumber}`);

// Clear form

setEmployeeName('');

setComplaint('');

};

return (

<div style={styles.container}>

<h2>Raise a Complaint</h2>

<form onSubmit={handleSubmit} style={styles.form}>

<label>

Employee Name:

<input

type="text"

value={employeeName}

onChange={(e) => setEmployeeName(e.target.value)}

style={styles.input}

/>

</label>

<label>

Complaint:

<textarea

value={complaint}

onChange={(e) => setComplaint(e.target.value)}

style={styles.textarea}

/>

</label>

<button type="submit" style={styles.button}>

Submit Complaint

</button>

</form>

</div>

);

}

const styles = {

container: {

maxWidth: '500px',

margin: '40px auto',

padding: '20px',

border: '1px solid #ccc',

borderRadius: '8px',

fontFamily: 'sans-serif'

},

form: {

display: 'flex',

flexDirection: 'column',

gap: '15px'

},

input: {

width: '100%',

padding: '8px'

},

textarea: {

width: '100%',

padding: '8px',

height: '100px'

},

button: {

padding: '10px',

backgroundColor: '#1976d2',

color: 'white',

border: 'none',

borderRadius: '4px',

cursor: 'pointer'

}

};

export default ComplaintRegister;

**App.js:**

import React from 'react';

import ComplaintRegister from './ComplaintRegister';

function App() {

return (

<div className="App">

<ComplaintRegister />

</div>

);

}

export default App;

**OUTPUT:**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Create a React App named “mailregisterapp” which will have a component named “register.js”. Create a form which accepts the name, email and password and validate the fields as per the following:**

1. **Name should have atleast 5 characters**
2. **Email should have @ and .**
3. **Password should have atleast 8 characters.**

**Ensure that validations are implemented through eventhandle and eventsubmit of a form.**

**Register.js:**

import React, { useState } from 'react';

function Register() {

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

const [email, setEmail] = useState('');

const [password, setPassword] = useState('');

const validateForm = (e) => {

e.preventDefault();

if (name.length < 5) {

alert('Full Name must be 5 characters long!');

return;

}

if (!email.includes('@') || !email.includes('.')) {

alert('Email is not valid!');

return;

}

if (password.length < 8) {

alert('Password must be 8 characters long!');

return;

}

alert('Registration Successful!');

// Clear form (optional)

setName('');

setEmail('');

setPassword('');

};

return (

<div style={styles.container}>

<h1 style={styles.header}>Register Here!!!</h1>

<form onSubmit={validateForm}>

<div>

<label>Name:&nbsp;</label>

<input

type="text"

value={name}

onChange={(e) => setName(e.target.value)}

/>

</div><br />

<div>

<label>Email:&nbsp;</label>

<input

type="email"

value={email}

onChange={(e) => setEmail(e.target.value)}

/>

</div><br />

<div>

<label>Password:&nbsp;</label>

<input

type="password"

value={password}

onChange={(e) => setPassword(e.target.value)}

/>

</div><br />

<button type="submit">Submit</button>

</form>

</div>

);

}

const styles = {

container: {

marginTop: '50px',

textAlign: 'center',

fontFamily: 'Arial',

},

header: {

color: 'red',

fontWeight: 'bold'

}

};

export default Register;

**App.js:**

import React from 'react';

import Register from './Register';

function App() {

return (

<div className="App">

<Register />

</div>

);

}

export default App;

**OUTPUT:**  
A screenshot of a computer

AI-generated content may be incorrect.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Create a React Application “fetchuserapp” which will retrieve the user details from** [**https://api.randomuser.me/**](https://api.randomuser.me/) **and display the title, firstname and image of a user.**

**Create a component named “Getuser” and in the asynchronous method “ComponentDidMount ()” invoke the URL using fetch method and the response can be displayed in the render method of the component.**

**Getuser.js:**

import React from 'react';

class Getuser extends React.Component {

constructor(props) {

super(props);

this.state = {

title: '',

firstName: '',

picture: '',

};

}

async componentDidMount() {

try {

const response = await fetch('https://api.randomuser.me/');

const data = await response.json();

const user = data.results[0];

this.setState({

title: user.name.title,

firstName: user.name.first,

picture: user.picture.large,

});

} catch (error) {

console.error('Failed to fetch user:', error);

}

}

render() {

const { title, firstName, picture } = this.state;

return (

<div style={styles.container}>

<h1>

{title} {firstName}

</h1>

{picture && <img src={picture} alt="User" style={styles.image} />}

</div>

);

}

}

const styles = {

container: {

textAlign: 'center',

marginTop: '50px',

},

image: {

borderRadius: '8px',

marginTop: '10px',

},

};

export default Getuser;

**App.js:**

import React from 'react';

import Getuser from './Getuser';

function App() {

return (

<div className="App">

<Getuser />

</div>

);

}

export default App**;**

**OUTPUT:**

**A screenshot of a computer

AI-generated content may be incorrect.**