

WEEK-7

1. ReactJS-HOL(9)

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

- List of players
- Indian players

[ListofPlayer.js](#)

```
import React from 'react';

function ListofPlayers() {
  const players = [
    { name: 'Virat', score: 80 },
    { name: 'Rohit', score: 45 },
    { name: 'Rahul', score: 90 },
    { name: 'Dhoni', score: 100 },
    { name: 'Jadeja', score: 30 },
    { name: 'Ashwin', score: 75 },
    { name: 'Pant', score: 60 },
    { name: 'Bumrah', score: 50 },
    { name: 'Shami', score: 20 },
    { name: 'Siraj', score: 85 },
    { name: 'Gill', score: 40 },
  ];

  const below70 = 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>

      <h2>Players with score ≤ 70</h2>
      <ul>
        {below70.map((player, index) => (
          <li key={index}>{player.name} - {player.score}</li>
        ))}
      </ul>
    </div>
  );
}
```

```
export default ListofPlayers;
```

[IndianPlayers.js](#)

```
import React from 'react';

export function OddPlayers([first,, third,, fifth]) {
  return (
    <div>
      <h3>Odd Team Players</h3>
      <ul>
        <li>First: {first}</li>
        <li>Third: {third}</li>
        <li>Fifth: {fifth}</li>
      </ul>
    </div>
  );
}

export function EvenPlayers([, second,, fourth,, sixth]) {
  return (
    <div>
      <h3>Even Team Players</h3>
      <ul>
        <li>Second: {second}</li>
        <li>Fourth: {fourth}</li>
        <li>Sixth: {sixth}</li>
      </ul>
    </div>
  );
}

export function MergedPlayers() {
  const T20Players = ['First Player', 'Second Player', 'Third Player'];
  const RanjiPlayers = ['Fourth Player', 'Fifth Player', 'Sixth Player'];

  const AllPlayers = [...T20Players, ...RanjiPlayers];

  return (
    <div>
      <h3>Merged Player List</h3>
      <ul>
        {AllPlayers.map((player, index) => (
          <li key={index}>{player}</li>
        ))}
      </ul>
    </div>
  );
}
```

```

    </ul>
  </div>
);
}

```

App.js

```

import React from 'react';
import './App.css';
import ListofPlayers from './Components/ListofPlayers';
import { OddPlayers, EvenPlayers, MergedPlayers } from
'./Components/IndianPlayers';

function App() {
  const showList = true;

  const players = ['Virat', 'Rohit', 'Rahul', 'Dhoni', 'Jadeja', 'Ashwin'];

  return (
    <div className="App">
      <h1>Cricket App</h1>
      {showList ? (
        <ListofPlayers />
      ) : (
        <>
          <OddPlayers {...[players]} />
          <EvenPlayers {...[players]} />
          <MergedPlayers />
        </>
      )}
    </div>
  );
}

export default App;

```

Output:

Cricket App	
All Players	
•	Virat - 80
•	Rohit - 45
•	Rahul - 90
•	Dhoni - 100
•	Jadeja - 30
•	Ashwin - 75
•	Pant - 60
•	Bumrah - 50
•	Shami - 20
•	Siraj - 85
•	Gill - 40
Players with score ≤ 70	
•	Rohit - 45
•	Jadeja - 30
•	Pant - 60
•	Bumrah - 50
•	Shami - 20
•	Gill - 40

2. ReactJS-HOL(10)

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

[OfficeList.js](#)

```
import React from 'react';
import img1 from './images/AA.jpeg';
import img2 from './images/AB.jpeg';
import img3 from './images/AC.jpeg';
import img4 from './images/AD.jpeg';

function OfficeList() {

  const featuredOffice = {
    name: "Tech Hub Workspace",
    rent: 55000,
    address: "123, IT Park Road, Bangalore",
    image: img1
  };

  const officeSpaces = [
    {
      name: "Innovate Co-working",
      rent: 45000,
      address: "22, MG Road, Pune",
      image: img2
    },
    {
      name: "Elite Business Center",
      rent: 70000,
      address: "101, Ring Road, Hyderabad",
      image: img3
    },
    {
      name: "Startup Den",
      rent: 60000,
      address: "50, Koramangala, Bangalore",
      image: img4
    }
  ];

  return (
    <div>
```

```

<h1>Office Space Rental App</h1>

<h2>Featured Office</h2>
<img src={featuredOffice.image} alt={featuredOffice.name} style={{ width:
"400px" }} />
<h3>{featuredOffice.name}</h3>
<p style={{ color: featuredOffice.rent > 60000 ? 'green' : 'red' }}>
  Rent: ₹{featuredOffice.rent}
</p>
<p>{featuredOffice.address}</p>

<h2>Available Offices</h2>
{officeSpaces.map((office, index) => (
  <div key={index} style={{ border: '1px solid #ccc', padding: 10,
marginBottom: 20 }}>
    <img src={office.image} alt={office.name} style={{ width: "400px" }} />
    <h3>{office.name}</h3>
    <p style={{ color: office.rent > 60000 ? 'green' : 'red' }}>
      Rent: ₹{office.rent}
    </p>
    <p>{office.address}</p>
  </div>
))}
</div>
);
}

export default OfficeList;

```

Output:

Office Space Rental App

Featured Office



Tech Hub Workspace

Rent: ₹55000

123, IT Park Road, Bangalore

Available Offices



Elite Business Center

Rent: ₹70000

101, Ring Road, Hyderabad

3. ReactJS-HOL(11)

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

EventExample.js

```
import React, { useState } from 'react';

function EventExamples() {
  const [count, setCount] = useState(0);
  const [message, setMessage] = useState("");
  const [rupees, setRupees] = useState("");
  const [euro, setEuro] = useState("");

  const sayHello = () => {
    console.log("Hello! This is a static message.");
    setMessage("Hello! This is a static message.");
  };

  const increment = () => {
    setCount(count + 1);
    sayHello();
  };

  const decrement = () => {
    setCount(count - 1);
  };

  const sayWelcome = (msg) => {
    alert(`Welcome Message: ${msg}`);
  };

  const handleClick = () => {
    alert("I was clicked");
  };

  const handleRupeesChange = (e) => {
    setRupees(e.target.value);
  };

  const handleSubmit = (e) => {
    e.preventDefault();
    const converted = (parseFloat(rupees) / 89.5).toFixed(2);
    setEuro(converted);
  };

  return (
```

```

<div style={{ padding: "20px" }}>
  <h2>Event Handling Examples</h2>

  <h3>Counter: {count}</h3>
  <button onClick={increment}>Increment</button>
  <button onClick={decrement}>Decrement</button>

  <p>{message}</p>

  <button onClick={() => sayWelcome("Welcome to React!")}>Say Welcome</button>
  <button onClick={handleClick}>OnPress</button>

  <hr />
  <h3>Currency Converter</h3>
  <form onSubmit={handleSubmit}>
    <label>Enter amount in Rupees: </label>
    <input type="number" value={rupees} onChange={handleRupeesChange} />
    <button type="submit">Convert</button>
  </form>
  {euro && <p>Equivalent Euro: €{euro}</p>}
</div>
);
}

export default EventExamples;

```

App.js

```

import './App.css';
import EventExamples from './Components/EventExamples';

function App() {
  return (
    <div className="App">
      <EventExamples />
    </div>
  );
}

export default App;

```

Output:

Event Handling Examples

Counter: 3

Increment Decrement

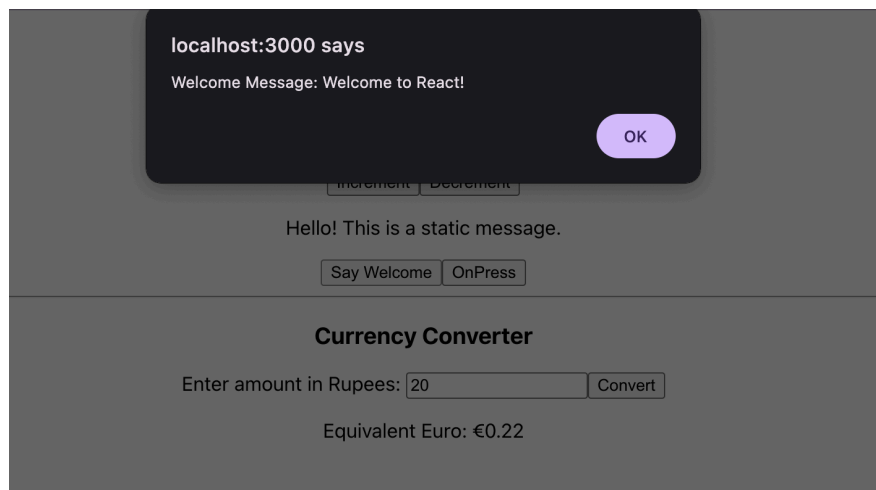
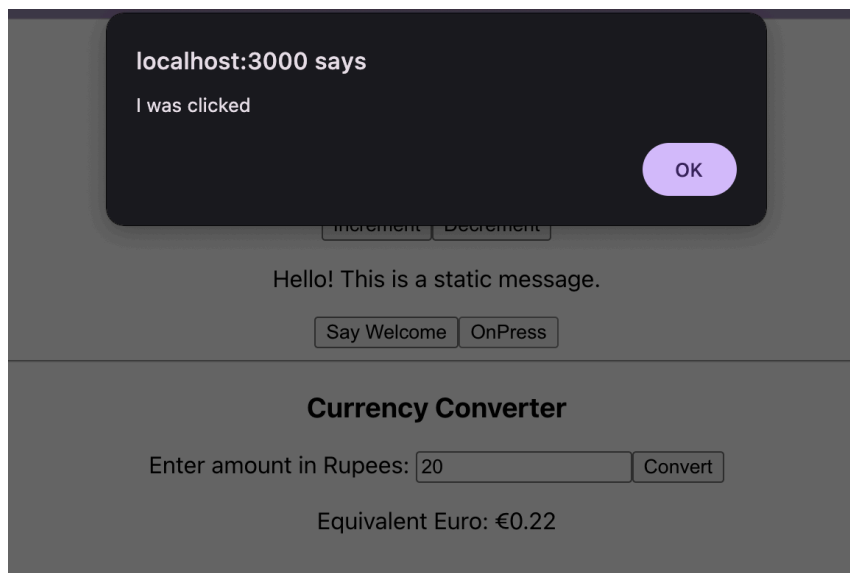
Hello! This is a static message.

Say Welcome OnPress

Currency Converter

Enter amount in Rupees: Convert

Equivalent Euro: €0.22



4. ReactJS-HOL(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.

GuestPage.js

```
import React from 'react';

function GuestPage() {
  return (
    <div>
      <h2>Welcome, Guest!</h2>
      <p>Browse available flight details below:</p>
      <ul>
        <li>Flight A123 - Hyderabad to Delhi - ₹4500</li>
        <li>Flight B456 - Bangalore to Mumbai - ₹5200</li>
      </ul>
      <p><i>Login to book your tickets.</i></p>
    </div>
  );
}

export default GuestPage;
```

UserPage.js

```
import React from 'react';

function UserPage() {
  return (
    <div>
      <h2>Welcome, User!</h2>
      <p>Book your flights below:</p>
      <ul>
        <li>Flight A123 - Hyderabad to Delhi - ₹4500 <button>Book</button></li>
        <li>Flight B456 - Bangalore to Mumbai - ₹5200 <button>Book</button></li>
      </ul>
    </div>
  );
}

export default UserPage;
```

[LoginButton.js](#)

```
import React from 'react';

function LoginButton({ onLogin }) {
  return <button onClick={onLogin}>Login</button>;
}

export default LoginButton;
```

[LogoutButton.js](#)

```
import React from 'react';

function LoginButton({ onLogin }) {
  return <button onClick={onLogin}>Login</button>;
}

export default LoginButton;
```

[App.js](#)

```
import React, { useState } from 'react';
import GuestPage from './Components/GuestPage';
import UserPage from './Components/UserPage';
import LoginButton from './Components/LoginButton';
import LogoutButton from './Components/LogoutButton';

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

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

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

  return (
    <div className="App" style={{ padding: "20px" }}>
      <h1>Ticket Booking App</h1>
      {isLoggedIn ? <LogoutButton onLogout={handleLogout} /> : <LoginButton
onLogin={handleLogin} />}
      <hr />
    </div>
  );
}
```

```
{isLoggedIn ? <LoginPage /> : <GuestPage />}  
  </div>  
);  
}  
  
export default App;
```

Output:

Ticket Booking App

Login

Welcome, Guest!

Browse available flight details below:

- Flight A123 – Hyderabad to Delhi – ₹4500
- Flight B456 – Bangalore to Mumbai – ₹5200

Login to book your tickets.

Ticket Booking App

Logout

Welcome, User!

Book your flights below:

- Flight A123 – Hyderabad to Delhi – ₹4500

Book
- Flight B456 – Bangalore to Mumbai – ₹5200

Book

5. ReactJS-HOL(13)

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.

[BookDetails.js](#)

```
import React from 'react';

function BookDetails() {
  return (
    <div>
      <h2>Book Details</h2>
      <p>Title: Book1</p>
      <p>Author: Book Author</p>
    </div>
  );
}

export default BookDetails;
```

[BlogDetails.js](#)

```
import React from 'react';

function BlogDetails() {
  return (
    <div>
      <h2>Blog Details</h2>
      <p>Title: Blog1</p>
      <p>Author: Blog Author</p>
    </div>
  );
}

export default BlogDetails;
```

[CourseDetails.js](#)

```
import React from 'react';
```

```
function CourseDetails() {
  return (
    <div>
      <h2>Course Details</h2>
      <p>Course: Course 1</p>
      <p>Instructor: Course Author</p>
    </div>
  );
}

export default CourseDetails;
```

[App.js](#)

```
import React, { useState } from 'react';
import BookDetails from './Components/BookDetails';
import BlogDetails from './Components/BlogDetails';
import CourseDetails from './Components/CourseDetails';

function App() {
  const [selected, setSelected] = useState('book');

  const renderComponentIfElse = () => {
    if (selected === 'book') {
      return <BookDetails />;
    } else if (selected === 'blog') {
      return <BlogDetails />;
    } else {
      return <CourseDetails />;
    }
  };

  const renderComponentSwitch = () => {
    switch (selected) {
      case 'book':
        return <BookDetails />;
      case 'blog':
        return <BlogDetails />;
      case 'course':
        return <CourseDetails />;
      default:
        return null;
    }
  };
};
```

```

const renderComponentTernary =
  selected === 'book' ? (
    <BookDetails />
  ) : selected === 'blog' ? (
    <BlogDetails />
  ) : (
    <CourseDetails />
  );

const componentMap = {
  book: <BookDetails />,
  blog: <BlogDetails />,
  course: <CourseDetails />
};

return (
  <div style={{ padding: '20px' }}>
    <h1> Blogger App</h1>
    <div>
      <button onClick={() => setSelected('book')}>Book</button>
      <button onClick={() => setSelected('blog')}>Blog</button>
      <button onClick={() => setSelected('course')}>Course</button>
    </div>

    <hr />

    <h3> Conditional Rendering using if-else</h3>
    {renderComponentIfElse()}

    <h3> Conditional Rendering using switch-case</h3>
    {renderComponentSwitch()}

    <h3> Conditional Rendering using ternary operator</h3>
    {renderComponentTernary}

    <h3> Conditional Rendering using object mapping</h3>
    {componentMap[selected]}
  </div>
);
}

export default App;

```

Output:

Blogger App

Book Blog Course

Conditional Rendering using if-else

Blog Details

Title: Blog1

Author: Blog Author

Conditional Rendering using switch-case

Blog Details

Title: Blog1

Author: Blog Author

Conditional Rendering using ternary operator

Blog Details

Title: Blog1

Author: Blog Author

Conditional Rendering using object mapping

Blog Details

Title: Blog1

Author: Blog Author

Blogger App

Book Blog Course

Conditional Rendering using if-else

Course Details

Course: Course 1

Instructor: Course Author

Conditional Rendering using switch-case

Course Details

Course: Course 1

Instructor: Course Author

Conditional Rendering using ternary operator

Course Details

Course: Course 1

Instructor: Course Author

Conditional Rendering using object mapping

Course Details

Course: Course 1

Instructor: Course Author

Blogger App

Book Blog Course

Conditional Rendering using if-else

Book Details

Title: Book1

Author: Book Author

Conditional Rendering using switch-case

Book Details

Title: Book1

Author: Book Author

Conditional Rendering using ternary operator

Book Details

Title: Book1

Author: Book Author

Conditional Rendering using object mapping

Book Details

Title: Book1

Author: Book Author