## **Objectives**

* Explain various ways of conditional rendering
* Explain how to render multiple components
* Define list component
* Explain about keys in React applications
* Explain how to extract components with keys
* Explain React Map, map() function

In this hands-on lab, you will learn how to:

* Implement conditional rendering in React applications

## **Prerequisites**

The following is required to complete this hands-on lab:

* Node.js
* NPM
* Visual Studio Code

## **Notes**

Estimated time to complete this lab: **60 minutes.**

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.



**Hint:**







Sample Code and Sample OUTPUT:

App.js

import React, { useState } from 'react';

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

import BookDetails from './components/BookDetails';

import BlogDetails from './components/BlogDetails';

import CourseDetails from './components/CourseDetails';

const App = () => {

const [showBlog, setShowBlog] = useState(true);

const [courseType, setCourseType] = useState('react');

return (

<div>

<h1>Blogger App</h1>

<h2>Book Details (if-else):</h2>

<BookDetails books={books} />

<h2>Blog Details (ternary and &&):</h2>

<BlogDetails showBlog={showBlog} />

<button onClick={() => setShowBlog((prev) => !prev)}>

Toggle Blog

</button>

<h2>Course Details (switch-case):</h2>

<select onChange={(e) => setCourseType(e.target.value)} value={courseType}>

<option value="react">React</option>

<option value="angular">Angular</option>

<option value="vue">Vue</option>

</select>

<CourseDetails courseType={courseType} />

</div>

);

};

export default App;  
CourseDetails.js:

import React from 'react';

const CourseDetails = ({ courseType }) => {

  let courseContent;

  switch (courseType) {

    case 'react':

      courseContent = <p>React is a JavaScript library for building UIs.</p>;

      break;

    case 'angular':

      courseContent = <p>Angular is a TypeScript-based framework for building apps.</p>;

      break;

    case 'vue':

      courseContent = <p>Vue is a progressive JavaScript framework.</p>;

      break;

    default:

      courseContent = <p>Please select a course.</p>;

  }

  return (

    <div>

      <h2>Course Info</h2>

      {courseContent}

    </div>

  );

};

export default CourseDetails;

BookDetails.js:

import React from 'react';

const BookDetails = ({ books }) => {

  if (!books || books.length === 0) {

    return <p>No books available.</p>;

  }

  return (

    <ul>

      {books.map((book) => (

        <div key={book.id}>

          <h3>{book.bname}</h3>

          <h4>{book.price}</h4>

        </div>

      ))}

    </ul>

  );

};

export default BookDetails;

BlogDetails.js:

import React from 'react';

const BlogDetails = ({ showBlog }) => {

  const blog = {

    title: 'Understanding React Lifecycle',

    content: 'React lifecycle methods help manage component behavior...'

  };

  return (

    <div>

      {showBlog ? (

        <>

          <h3>{blog.title}</h3>

          <p>{blog.content}</p>

        </>

      ) : (

        <p>Blog is hidden</p>

      )}

      {showBlog && <button>Read More</button>}

    </div>

  );

};

export default BlogDetails;

Books.js:

export const books = [

  { id: 101, bname: 'Master React', price: 670 },

  { id: 102, bname: 'Deep Dive into Angular 11', price: 800 },

  { id: 103, bname: 'Mongo Essentials', price: 450 },

];