**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.

A screenshot of a computer

AI-generated content may be incorrect.

**Hint:**

A computer screen with text

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

A black background with text

AI-generated content may be incorrect.

**Solution**

Step 1 : create react project

npx create-react-app bloggerapp

cd bloggerapp

npm start

Step 2 : data.js in src

// src/data.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 },

];

export const courses = [

  { id: 201, name: 'Angular', date: '4/5/2021' },

  { id: 202, name: 'React', date: '6/3/2021' },

];

export const blogs = [

  { id: 301, title: 'React Learning', author: 'Stephen Biz', excerpt: 'Welcome to learning React!' },

  { id: 302, title: 'Installation', author: 'Schewzdiener', excerpt: 'You can install React from npm.' },

];

Step 3 : BookDetails.js in src

// src/BookDetails.js

import React from 'react';

function BookDetails(props) {

  // element variable approach:

  const bookdet = (

    <ul style={{ listStyle: 'none', padding: 0 }}>

      {props.books.map(book => (

        <div key={book.id} style={{ marginBottom: 18 }}>

          <h3>{book.bname}</h3>

          {/\* Logical && to show price only when prop true \*/}

          {props.showPrice && <h4>{book.price}</h4>}

        </div>

      ))}

    </ul>

  );

  return (

    <div>

      <div className="st2">

        <h1>Book Details</h1>

        {bookdet}

      </div>

    </div>

  );

}

export default BookDetails;

Step 4 : App.css in src

/\* src/App.css \*/

body {

  font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;

  margin: 0;

}

.container {

  display: grid;

  grid-template-columns: 1fr 1fr 1fr;

  gap: 10px;

  padding: 60px 40px;

  align-items: start;

}

.col {

  padding: 0 30px;

}

.col.center {

  border-left: 6px solid #0b7d39;

  border-right: 6px solid #0b7d39;

  padding-left: 40px;

  padding-right: 40px;

}

h1 {

  font-size: 30px;

  margin-bottom: 18px;

  font-weight: 700;

}

h2 { font-size: 22px; margin: 12px 0 6px; }

h3 { font-size: 18px; margin: 8px 0; }

h4 { font-size: 16px; margin: 6px 0; color: #333; }

.controls {

  text-align: center;

  padding: 10px 20px 40px;

}

.controls button, .controls select, .controls label {

  margin: 6px;

  font-size: 14px;

}

Step 5 : BlogDetails.js in src

// src/BlogDetails.js

import React from 'react';

function BlogDetails({ blogs }) {

  return (

    <div>

      <div className="v1">

        <h1>Blog Details</h1>

        {

          // inline ternary operator: if blogs array empty -> show message

          blogs && blogs.length > 0 ? (

            blogs.map(blog => (

              <div key={blog.id} style={{ marginBottom: 18 }}>

                <h2>{blog.title}</h2>

                <h4>{blog.author}</h4>

                <p>{blog.excerpt}</p>

              </div>

            ))

          ) : (

            <p>No blogs available.</p>

          )

        }

      </div>

    </div>

  );

}

export default BlogDetails;

Step 6 : CourseDetails.js in src

// src/CourseDetails.js

import React from 'react';

function CourseDetails({ courses }) {

  // early return - a common conditional rendering pattern

  if (!courses || courses.length === 0) {

    return (

      <div className="mystyle1">

        <h1>Course Details</h1>

        <p>No courses found.</p>

      </div>

    );

  }

  return (

    <div className="mystyle1">

      <h1>Course Details</h1>

      {courses.map(c => (

        <div key={c.id} style={{ marginBottom: 18 }}>

          <h2>{c.name}</h2>

          {/\* short info always shown, you could also use && for date \*/}

          <p>{c.date}</p>

        </div>

      ))}

    </div>

  );

}

export default CourseDetails;

Step 7 : PanelSwitcher.js in src

// src/PanelSwitcher.js

import React from 'react';

import BookDetails from './BookDetails';

import BlogDetails from './BlogDetails';

import CourseDetails from './CourseDetails';

function PanelSwitcher({ panel, books, blogs, courses, showPrice }) {

  switch (panel) {

    case 'books':

      return <BookDetails books={books} showPrice={showPrice} />;

    case 'blogs':

      return <BlogDetails blogs={blogs} />;

    case 'courses':

      return <CourseDetails courses={courses} />;

    default:

      return <div>Select a panel from the dropdown above.</div>;

  }

}

export default PanelSwitcher;

Step 8 : App.js in src

// src/App.js

import React, { useState } from 'react';

import './App.css';

import { books, courses, blogs } from './data';

import BookDetails from './BookDetails';

import BlogDetails from './BlogDetails';

import CourseDetails from './CourseDetails';

import PanelSwitcher from './PanelSwitcher';

function App() {

  const [showPrice, setShowPrice] = useState(true);

  const [panel, setPanel] = useState('books'); // for PanelSwitcher demo

  return (

    <>

      <div className="container">

        <div className="col">

          <CourseDetails courses={courses} />

        </div>

        <div className="col center">

          <BookDetails books={books} showPrice={showPrice} />

        </div>

        <div className="col">

          <BlogDetails blogs={blogs} />

        </div>

      </div>

      <div className="controls">

        <label>

          <input

            type="checkbox"

            checked={showPrice}

            onChange={() => setShowPrice(p => !p)}

          />

          {' '}Show book prices (logical && demo)

        </label>

        <div style={{ marginTop: 10 }}>

          <label>Switch panel (switch-case demo): </label>

          <select value={panel} onChange={(e) => setPanel(e.target.value)}>

            <option value="books">Books</option>

            <option value="blogs">Blogs</option>

            <option value="courses">Courses</option>

          </select>

        </div>

        <div style={{ marginTop: 12 }}>

          <strong>PanelSwitcher output:</strong>

          <div style={{ marginTop: 8, border: '1px solid #ddd', padding: 14 }}>

            <PanelSwitcher

              panel={panel}

              books={books}

              blogs={blogs}

              courses={courses}

              showPrice={showPrice}

            />

          </div>

        </div>

      </div>

    </>

  );

}

export default App;

Step 9 : Run

npm start

**Output**

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.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.