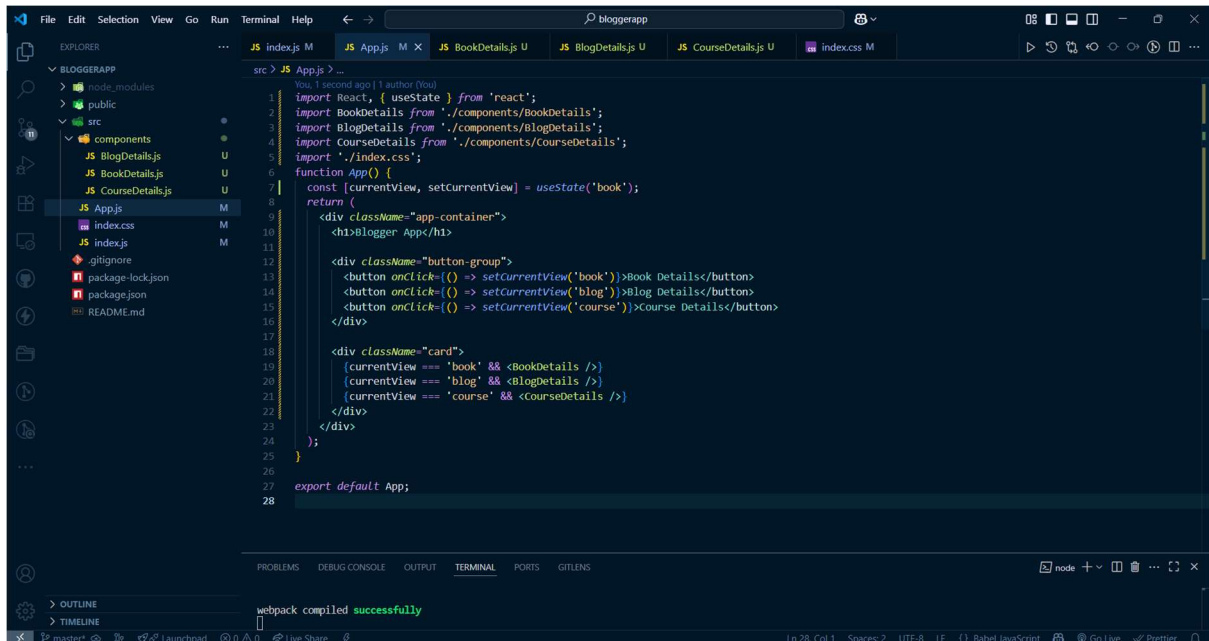


## Week - 7

### 13. ReactJS-HOL

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

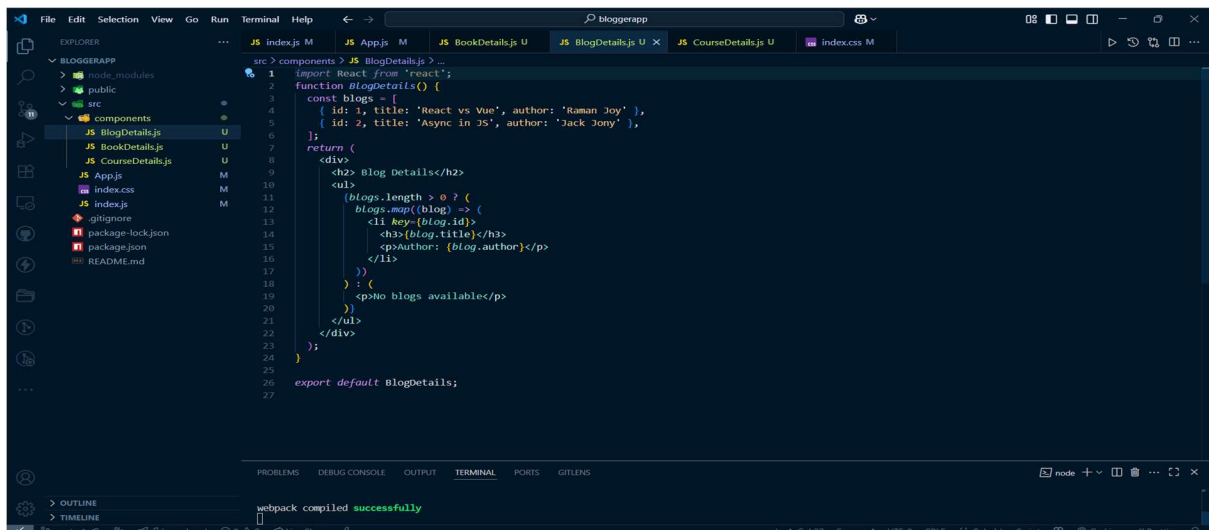
#### App.js



The screenshot shows the VS Code editor with the 'bloggerapp' project open. The Explorer panel on the left shows the file structure: `node_modules`, `public`, `src` (containing `components`, `index.css`, `index.js`), and `package-lock.json`. The `App.js` file is selected in the Explorer and its content is displayed in the editor. The code defines a functional component `App` using `useState` to manage the current view. It includes buttons to switch between 'book', 'blog', and 'course' details. The `render` method uses conditional rendering to display the selected component. The `index.js` file in the `src` directory is also visible in the Explorer, showing the import of `App` and the rendering of `<App />`.

```
src > JS App.js > ...
1  import React, { useState } from 'react';
2  import BookDetails from './components/BookDetails';
3  import BlogDetails from './components/BlogDetails';
4  import CourseDetails from './components/courseDetails';
5  import './index.css';
6
7  function App() {
8    const [currentView, setCurrentView] = useState('book');
9    return (
10     <div className="app-container">
11       <h1>Blogger App</h1>
12
13       <div className="button-group">
14         <button onClick={() => setCurrentView('book')}>Book Details</button>
15         <button onClick={() => setCurrentView('blog')}>Blog Details</button>
16         <button onClick={() => setCurrentView('course')}>Course Details</button>
17       </div>
18
19       <div className="card">
20         {currentView === 'book' && <BookDetails />}
21         {currentView === 'blog' && <BlogDetails />}
22         {currentView === 'course' && <CourseDetails />}
23       </div>
24     </div>
25   );
26 }
27
28 export default App;
```

#### BlogDetails.js



The screenshot shows the VS Code editor with the 'bloggerapp' project open. The Explorer panel on the left shows the file structure: `node_modules`, `public`, `src` (containing `components`, `index.css`, `index.js`), and `package-lock.json`. The `BlogDetails.js` file is selected in the Explorer and its content is displayed in the editor. The code defines a functional component `BlogDetails` that uses `useState` to manage a list of blogs. It renders a list of blogs with their titles and authors, or a message if no blogs are available. The `index.js` file in the `src` directory is also visible in the Explorer, showing the import of `BlogDetails` and the rendering of `<BlogDetails />`.

```
src > components > JS BlogDetails.js > ...
1  import React from 'react';
2  function BlogDetails() {
3    const blogs = [
4      { id: 1, title: 'React vs Vue', author: 'Raman Joy' },
5      { id: 2, title: 'Async in JS', author: 'Jack Jony' },
6    ];
7    return (
8     <div>
9       <h2>Blog Details</h2>
10       <ul>
11         {blogs.length > 0 ? (
12           blogs.map(blog => (
13             <li key={blog.id}>
14               <h3>{blog.title}</h3>
15               <p>Author: {blog.author}</p>
16             </li>
17           ))
18         ) : (
19           <p>No blogs available</p>
20         )}
21       </ul>
22     </div>
23   );
24 }
25
26 export default BlogDetails;
```

## BookDetails.js

The screenshot shows a VS Code editor with a React application. The Explorer sidebar on the left shows the file structure with 'BookDetails.js' selected. The main editor displays the code for 'BookDetails.js', which includes a function 'BookDetails' that returns a JSX element. The code uses 'import React from 'react';' and 'const books = [...]'. The JSX element uses 'books.map' to render a list of books. The bottom status bar shows 'webpack compiled successfully'.

## CourseDetails.js

The screenshot shows a VS Code editor window with the following components:

- Explorer Panel (Left):** Displays the file structure of the project. The files listed are:
  - `node_modules`
  - `public`
  - `src`
    - `components`
      - `BlogDetails.js` (U)
      - `BookDetails.js` (U)
      - `CourseDetails.js` (U) - Selected
      - `App.js` (M)
      - `index.css` (M)
      - `index.js` (M)
    - `gitignore`
    - `package-lock.json`
    - `package.json`
    - `README.md`

- Main Editor:** Displays the code for `src/components/CourseDetails.js`. The code is as follows:
 

```
1 import React from 'react';
2
3 function CourseDetails() {
4   const showCourse = true;
5
6   return (
7     <div>
8       <h2> Course Details</h2>
9
10      <div>
11        <p>Full Stack Development - 6 Months</p>
12        <p>course available</p>
13      </div>
14    </div>
15  );
16
17 export default CourseDetails;
```
- Terminal Panel (Bottom):** Shows the message `webpack compiled successfully`.

## Index.css

The screenshot shows the VS Code editor interface with the 'index.css' file open. The Explorer sidebar on the left shows the project structure, including 'src' and 'components' folders. The main editor area displays the CSS code for 'index.css', which includes styles for the body, a container, a heading, and buttons. The status bar at the bottom indicates that 'webpack compiled successfully'.

```

1  body {
2      margin: 0;
3      font-family: 'Segoe UI', sans-serif;
4      background-color: #f7f9fc;
5  }
6  .app-container {
7      max-width: 800px;
8      margin: 40px auto;
9      padding: 30px;
10     background: white;
11     border-radius: 12px;
12     box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
13 }
14 h1 {
15     text-align: center;
16     color: #333;
17     margin-bottom: 30px;
18 }
19 .button-group {
20     display: flex;
21     justify-content: center;
22     gap: 15px;
23     margin-bottom: 30px;
24 }
25 .button-group button {
26     padding: 10px 16px;
27     background-color: #4caf50;
28     color: white;
29     border: none;
30     border-radius: 8px;
31     cursor: pointer;

```

# Output

