Task

User Management with Conditional Rendering and State Management

Consider we have a list of user information stored in an array of objects. Each object contains details such as the user's name, email, phone number, and address.

AllUsersComponent: Create an AllUsersComponent that receives the list of user information as props and renders a table displaying each user's details. Pass each user's information as props to a SingleUserComponent.

SingleUserComponent: Define a SingleUserComponent that receives the user's information as props and displays it in a table row. Additionally, include a button in each row that, when clicked, removes the corresponding user's information from the table.

Example Data-Store this data in users.js file and import the data into AllUsers component as useState:

```
const users = [
{ id: 1, name: "Alice Johnson", email: "alice@example.com", phone: "555-1234", address: "123 Main St" },
{ id: 2, name: "Bob Smith", email: "bob@example.com", phone: "555-5678", address: "456 Elm St" },
{ id: 3, name: "Charlie Brown", email: "charlie@example.com", phone: "555-9101", address: "789 Oak St" }
];
```

With reference to the project created in one of the assignment in Component, Props and State.

Add the following features to that Project:

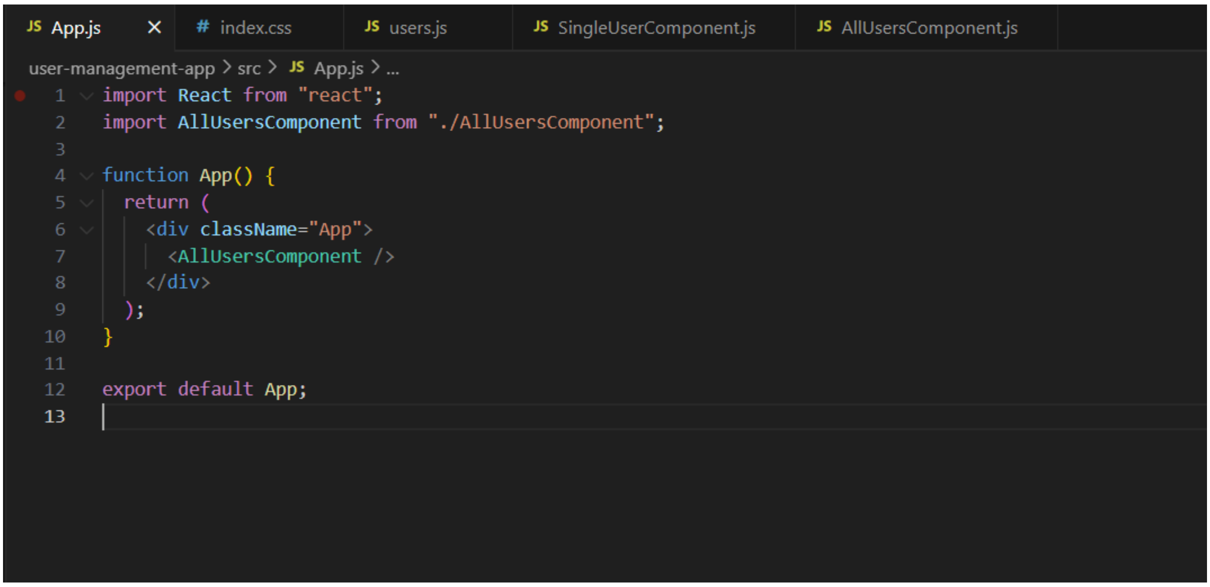
1. Create an Add All Users button that needs to be displayed on the screen initially
2. All the user information with the delete button should be displayed only after add all users button is clicked.
3. Once the users are listed on the screen, the Add All Users button should toggle to Delete All Users button
4. Once the Delete All Users button is clicked, all the users needs to be removed

from the screen Or when we click on delete button of each user and once all the users are deleted - It should show Add All Users button automatically.

Solution –

App.js

```
import React from "react";
import AllUsersComponent from "./AllUsersComponent";

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

export default App;
```

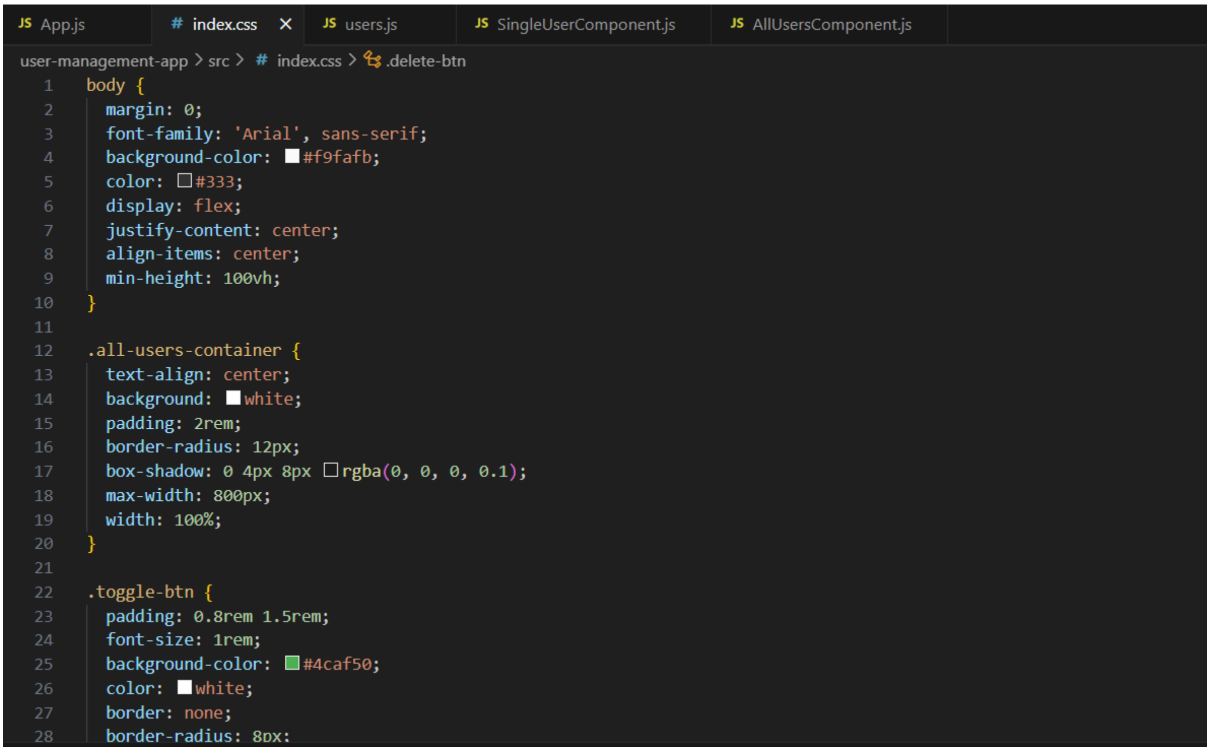


JS App.js ×
# index.css
JS users.js
JS SingleUserComponent.js
JS AllUsersComponent.js
user-management-app > src > JS App.js > ...

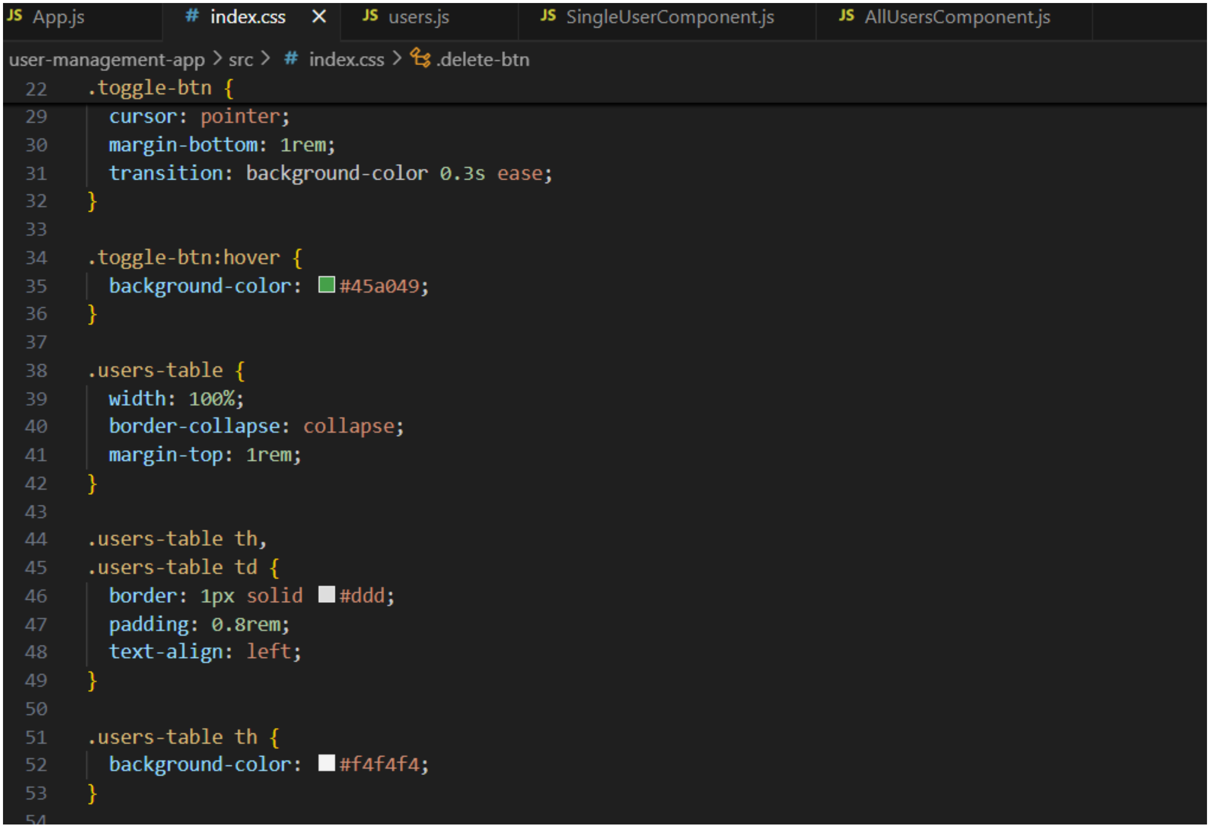
```
import React from "react";
import AllUsersComponent from "./AllUsersComponent";

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

export default App;

```

Index.css

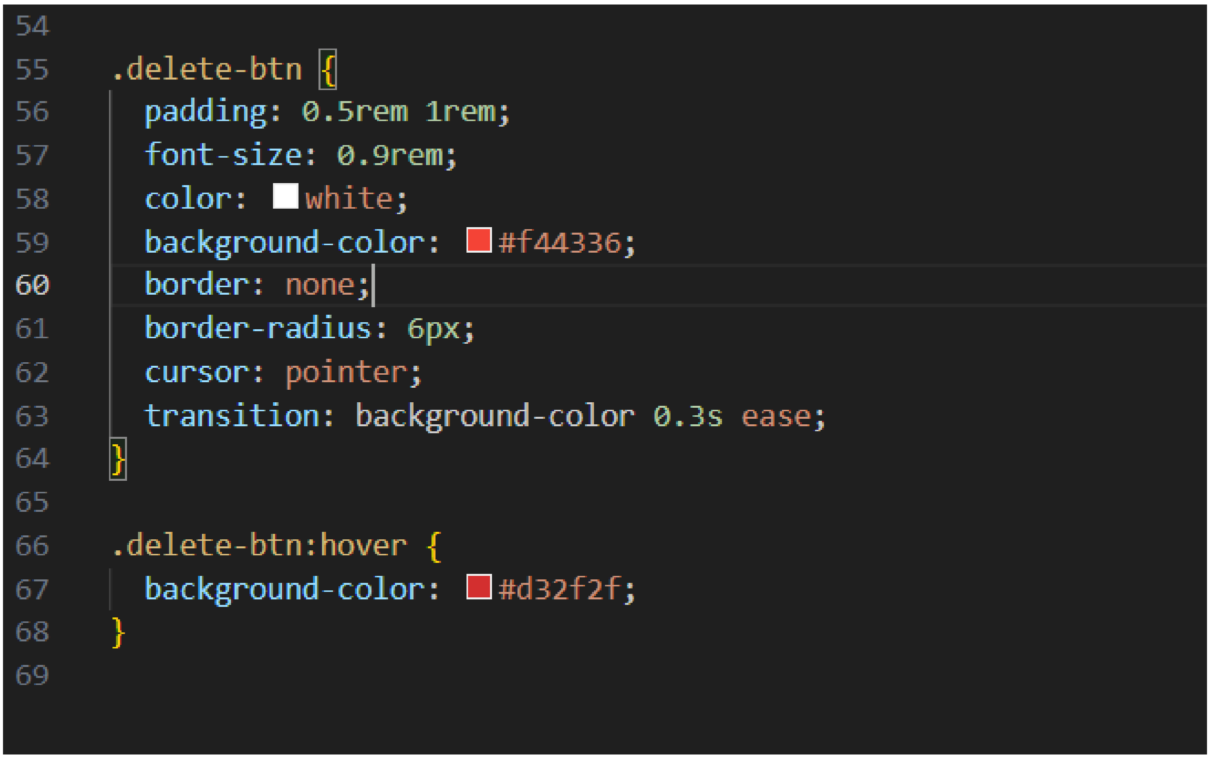
```
body {
  margin: 0;
  font-family: 'Arial', sans-serif;
  background-color: #f9fafb;
  color: #333;
  display: flex;
  justify-content: center;
  align-items: center;
  min-height: 100vh;
}
.all-users-container {
  text-align: center;
  background: white;
  padding: 2rem;
  border-radius: 12px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
  max-width: 800px;
  width: 100%;
}
.toggle-btn {
  padding: 0.8rem 1.5rem;
  font-size: 1rem;
  background-color: #4caf50;
  color: white;
  border: none;
  border-radius: 8px;
  cursor: pointer;
  margin-bottom: 1rem;
  transition: background-color 0.3s ease;
}
.toggle-btn:hover {
  background-color: #45a049;
}
.users-table {
  width: 100%;
  border-collapse: collapse;
  margin-top: 1rem;
}
.users-table th,
.users-table td {
  border: 1px solid #ddd;
  padding: 0.8rem;
  text-align: left;
}
.users-table th {
  background-color: #f4f4f4;
}
.delete-btn {
  padding: 0.5rem 1rem;
  font-size: 0.9rem;
  color: white;
  background-color: #f44336;
  border: none;
  border-radius: 6px;
  cursor: pointer;
  transition: background-color 0.3s ease;
}
.delete-btn:hover {
  background-color: #d32f2f;
}
```


JS App.js
# index.css ×
JS users.js
JS SingleUserComponent.js
JS AllUsersComponent.js
user-management-app > src > # index.css > .delete-btn

```css
body {
  margin: 0;
  font-family: 'Arial', sans-serif;
  background-color: #f9fafb;
  color: #333;
  display: flex;
  justify-content: center;
  align-items: center;
  min-height: 100vh;
}

.all-users-container {
  text-align: center;
  background: white;
  padding: 2rem;
  border-radius: 12px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
  max-width: 800px;
  width: 100%;
}

.toggle-btn {
  padding: 0.8rem 1.5rem;
  font-size: 1rem;
  background-color: #4caf50;
  color: white;
  border: none;
  border-radius: 8px;
```


JS App.js
# index.css ×
JS users.js
JS SingleUserComponent.js
JS AllUsersComponent.js
user-management-app > src > # index.css > .delete-btn

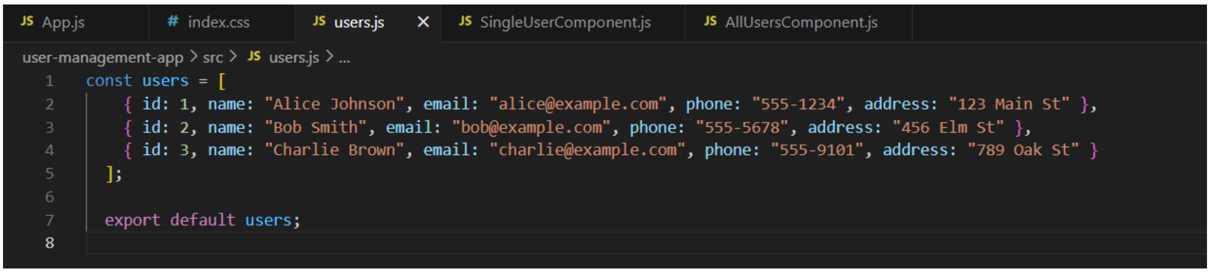
```css
.toggle-btn {
  cursor: pointer;
  margin-bottom: 1rem;
  transition: background-color 0.3s ease;
}

.toggle-btn:hover {
  background-color: #45a049;
}

.users-table {
  width: 100%;
  border-collapse: collapse;
  margin-top: 1rem;
}

.users-table th,
.users-table td {
  border: 1px solid #ddd;
  padding: 0.8rem;
  text-align: left;
}

.users-table th {
  background-color: #f4f4f4;
}
```



```css

.delete-btn {
  padding: 0.5rem 1rem;
  font-size: 0.9rem;
  color: white;
  background-color: #f44336;
  border: none;
  border-radius: 6px;
  cursor: pointer;
  transition: background-color 0.3s ease;
}

.delete-btn:hover {
  background-color: #d32f2f;
}

```

Users.js

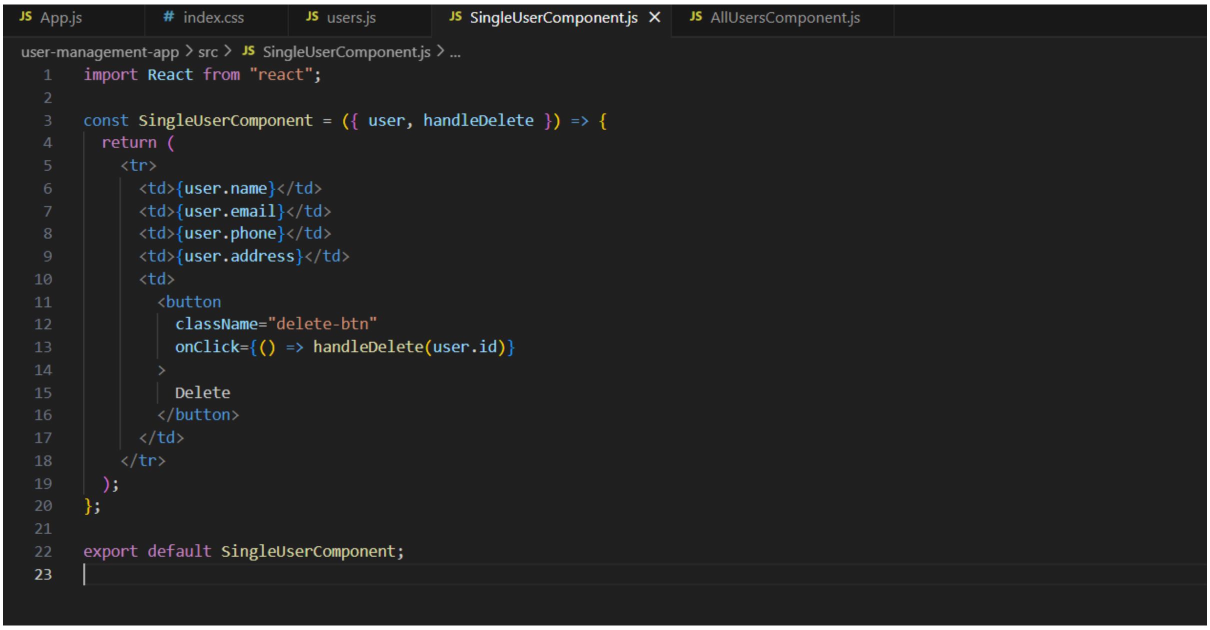
```
const users = [
   { id: 1, name: "Alice Johnson", email: "alice@example.com", phone: "555-1234", address: "123 Main St" },
   { id: 2, name: "Bob Smith", email: "bob@example.com", phone: "555-5678", address: "456 Elm St" },
   { id: 3, name: "Charlie Brown", email: "charlie@example.com", phone: "555-9101", address: "789 Oak St" }
 ];


 export default users;
```


JS App.js
# index.css
JS users.js ×
JS SingleUserComponent.js
JS AllUsersComponent.js
user-management-app > src > JS users.js > ...

```javascript
const users = [
    { id: 1, name: "Alice Johnson", email: "alice@example.com", phone: "555-1234", address: "123 Main St" },
    { id: 2, name: "Bob Smith", email: "bob@example.com", phone: "555-5678", address: "456 Elm St" },
    { id: 3, name: "Charlie Brown", email: "charlie@example.com", phone: "555-9101", address: "789 Oak St" }
  ];

  export default users;

```

SingleUserComponent.js

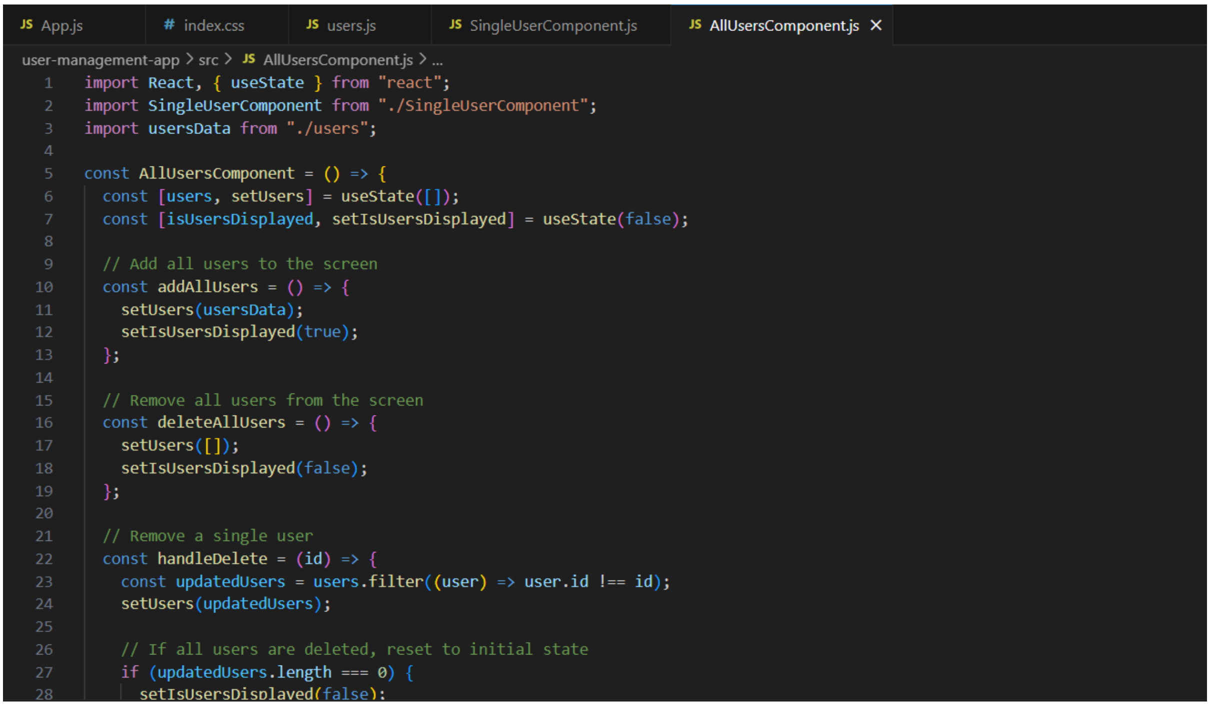
```
import React from "react";

const SingleUserComponent = ({ user, handleDelete }) => {
  return (
    <tr>
      <td>{user.name}</td>
      <td>{user.email}</td>
      <td>{user.phone}</td>
      <td>{user.address}</td>
      <td>
        <button
          className="delete-btn"
          onClick={() => handleDelete(user.id)}
        >
          Delete
        </button>
      </td>
    </tr>
  );
};

export default SingleUserComponent;
```



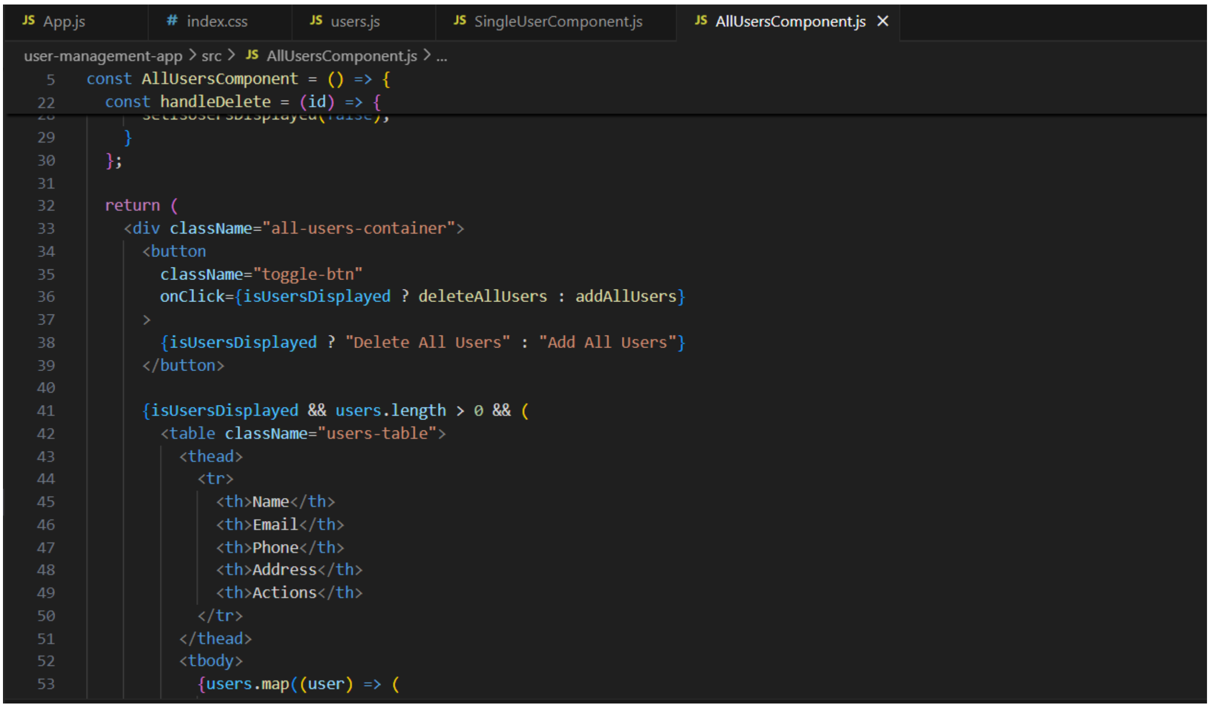
```
import React from "react";

const SingleUserComponent = ({ user, handleDelete }) => {
  return (
    <tr>
      <td>{user.name}</td>
      <td>{user.email}</td>
      <td>{user.phone}</td>
      <td>{user.address}</td>
      <td>
        <button
          className="delete-btn"
          onClick={() => handleDelete(user.id)}
        >
          Delete
        </button>
      </td>
    </tr>
  );
};

export default SingleUserComponent;

```

AllUsersComponent.js

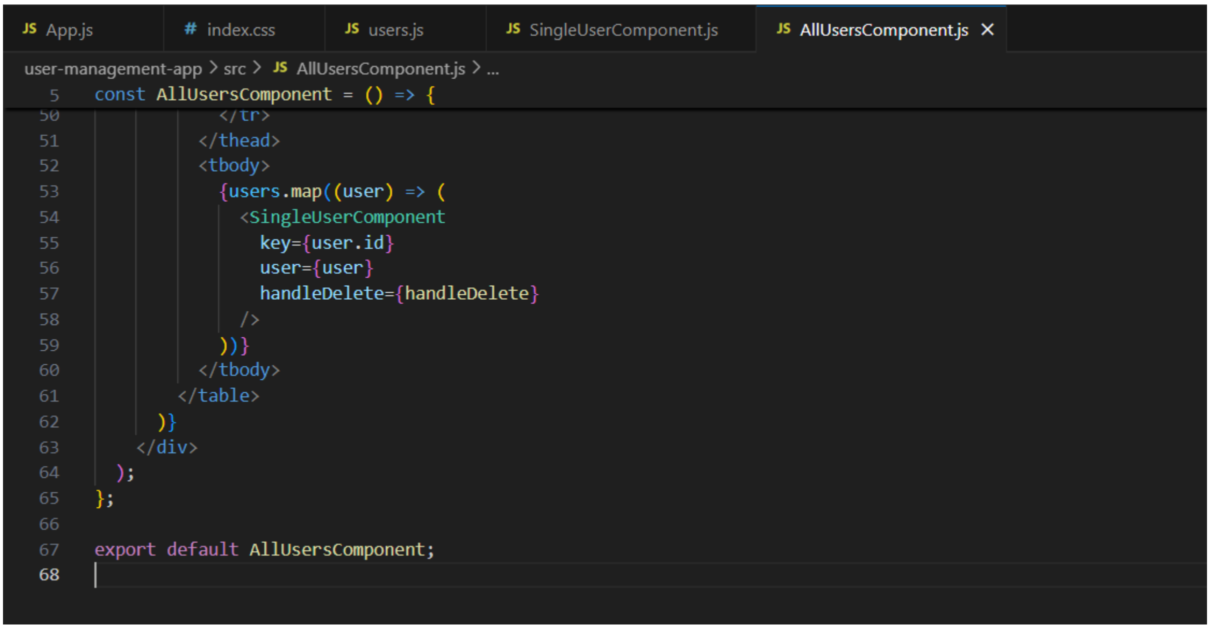
```
import React, { useState } from "react";
import SingleUserComponent from "./SingleUserComponent";
import usersData from "./users";

const AllUsersComponent = () => {
  const [users, setUsers] = useState([]);
  const [isUsersDisplayed, setIsUsersDisplayed] = useState(false);
  // Add all users to the screen
  const addAllUsers = () => {
    setUsers(usersData);
    setIsUsersDisplayed(true);
  };

  // Remove all users from the screen
  const deleteAllUsers = () => {
    setUsers([]);
    setIsUsersDisplayed(false);
  };
  // Remove a single user
  const handleDelete = (id) => {
    const updatedUsers = users.filter((user) => user.id !== id);
    setUsers(updatedUsers);
    // If all users are deleted, reset to initial state
    if (updatedUsers.length === 0) {
      setIsUsersDisplayed(false);
    }
  };
  return (
    <div className="all-users-container">
      <button
        className="toggle-btn"
        onClick={isUsersDisplayed ? deleteAllUsers : addAllUsers}>
        {isUsersDisplayed ? "Delete All Users" : "Add All Users"}
      </button>
      {isUsersDisplayed && users.length > 0 && (
        <table className="users-table">
          <thead>
            <tr>
              <th>Name</th>
              <th>Email</th>
              <th>Phone</th>
              <th>Address</th>
              <th>Actions</th>
              </tr>
            </thead>
            <tbody>
              {users.map((user) => (
                <SingleUserComponent
                  key={user.id}
                  user={user}
                  handleDelete={handleDelete}
                />
              ))}
            </tbody>
          </table>
        )}
      </div>
    );
  };
  export default AllUsersComponent;
```



```
import React, { useState } from "react";
import SingleUserComponent from "./SingleUserComponent";
import usersData from "./users";

const AllUsersComponent = () => {
  const [users, setUsers] = useState([]);
  const [isUsersDisplayed, setIsUsersDisplayed] = useState(false);

  // Add all users to the screen
  const addAllUsers = () => {
    setUsers(usersData);
    setIsUsersDisplayed(true);
  };

  // Remove all users from the screen
  const deleteAllUsers = () => {
    setUsers([]);
    setIsUsersDisplayed(false);
  };

  // Remove a single user
  const handleDelete = (id) => {
    const updatedUsers = users.filter((user) => user.id !== id);
    setUsers(updatedUsers);

    // If all users are deleted, reset to initial state
    if (updatedUsers.length === 0) {
      setIsUsersDisplayed(false);
```



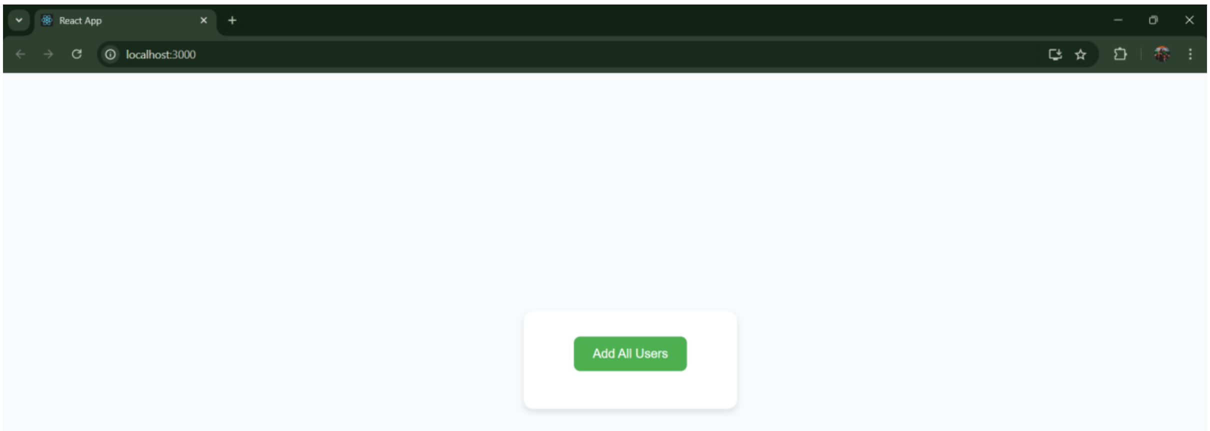
```
const AllUsersComponent = () => {
  const handleDelete = (id) => {
      setIsUsersDisplayed(false);
    }
  };

  return (
    <div className="all-users-container">
      <button
        className="toggle-btn"
        onClick={isUsersDisplayed ? deleteAllUsers : addAllUsers}
      >
        {isUsersDisplayed ? "Delete All Users" : "Add All Users"}
      </button>

      {isUsersDisplayed && users.length > 0 && (
        <table className="users-table">
          <thead>
            <tr>
              <th>Name</th>
              <th>Email</th>
              <th>Phone</th>
              <th>Address</th>
              <th>Actions</th>
            </tr>
          </thead>
          <tbody>
            {users.map((user) => (
```



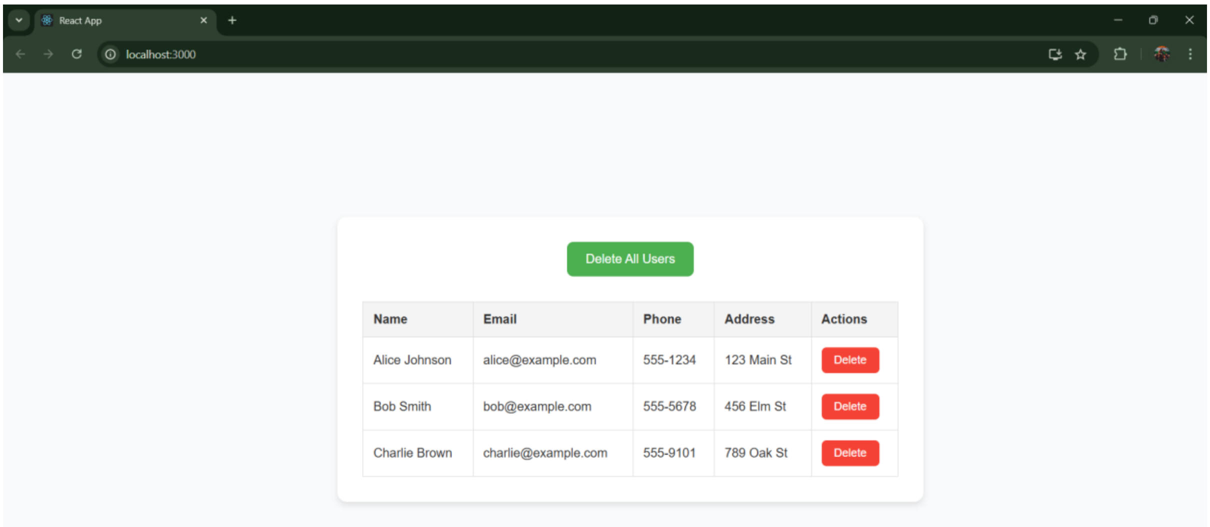
```
const AllUsersComponent = () => {
            </tr>
          </thead>
          <tbody>
            {users.map((user) => (
              <SingleUserComponent
                key={user.id}
                user={user}
                handleDelete={handleDelete}
              />
            ))}
          </tbody>
        </table>
      )}
    </div>
  );
};

export default AllUsersComponent;
```

OUTPUT –

Before Clicking on the Add All Users Button



After Clicking on the Add All Users Button



Name	Email	Phone	Address	Actions
Alice Johnson	alice@example.com	555-1234	123 Main St	Delete
Bob Smith	bob@example.com	555-5678	456 Elm St	Delete
Charlie Brown	charlie@example.com	555-9101	789 Oak St	Delete