NAME	Aditya Raj
UID	23BCS11031
CLASS	622-B

➤ REACT PRACTISE 2

Title

Library Management UI with Search, Add, and Remove Book Functionality **Objective**

Build an interactive library management interface using React that allows users to view a list of books and perform actions like searching, adding, and removing books. This task strengthens your understanding of state management, event handling, and dynamic rendering in React.

Task Description

Create a React-based UI that displays a list of books (each with a title and author). Add a search input box to filter books by title or author as the user types. Provide an input form to add new books to the list, and include a remove button next to each book to delete it from the list. All actions (searching, adding, removing) should work dynamically without reloading the page, demonstrating React's ability to update the UI based on state changes.

CODE

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

function App() {
  const [books, setBooks] = useState([
      { title: "1984", author: "George Orwell" },
      { title: "The Great Gatsby", author: "F. Scott Fitzgerald" },
      { title: "To Kill a Mockingbird", author: "Harper Lee" },
    ]);

  const [search, setSearch] = useState("");
  const [newTitle, setNewTitle] = useState("");
  const [newAuthor, setNewAuthor] = useState("");

  const addBook = () => {
  if (newTitle.trim() && newAuthor.trim()) {
    setBooks([...books, { title: newTitle, author: newAuthor }]);
      setNewTitle("");
}
```

```
setNewAuthor("");
  }
};
const removeBook = (index) => {
  setBooks(books.filter((_, i) => i !== index));
};
const filteredBooks = books.filter(
  (book) \Rightarrow
    book.title.toLowerCase().includes(search.toLowerCase()) ||
    book.author.toLowerCase().includes(search.toLowerCase())
);
return (
  <div style={styles.container}>
    <h2 style={styles.heading}>Library Management</h2>
    {/* Search */}
    <input</pre>
      type="text"
      placeholder="Search by title or author"
      value={search}
      onChange={(e) => setSearch(e.target.value)}
      style={styles.input}
    />
    {/* Add Book */}
    <div style={{ marginTop: "10px" }}>
      <input
        type="text"
        placeholder="New book title"
        value={newTitle}
        onChange={(e) => setNewTitle(e.target.value)}
        style={styles.input}
      />
      <input</pre>
        type="text"
        placeholder="New book author"
        value={newAuthor}
        onChange={(e) => setNewAuthor(e.target.value)}
        style={styles.input}
      />
      <button onClick={addBook} style={styles.button}>
        Add Book
```

```
</button>
       </div>
       {/* Book List */}
       <div style={{ marginTop: "20px" }}>
         {filteredBooks.map((book, index) => (
           <div key={index} style={styles.bookItem}>
              <span>
                <strong>{book.title}</strong> by {book.author}
              </span>
              <button onClick={() => removeBook(index)}
 style={styles.removeButton}>
               Remove
              </button>
           </div>
         ))}
       </div>
     </div>
   );
 }
 const styles = {
   container: {
     backgroundColor: "#ffffff",
     border: "1px solid #ddd",
     borderRadius: "8px",
     padding: "20px",
     margin: "20px auto",
     width: "600px",
     boxShadow: "0px 4px 10px rgba(0,0,0,0.05)",
   },
heading: {
  marginBottom: "10px",
     color: "#333",
   },
   input: {
  padding: "8px",
     marginRight: "8px",
     border: "1px solid #ccc",
     borderRadius: "4px",
   },
   button: {
     padding: "8px 12px",
     backgroundColor: "#007bff",
```

```
color: "Black",
    border: "none",
    borderRadius: "4px",
    cursor: "pointer",
  },
  bookItem: {
    display: "flex",
    justifyContent: "space-between",
    alignItems: "center",
    border: "1px solid #eee",
    padding: "10px",
    marginBottom: "8px",
    borderRadius: "5px",
    backgroundColor: "#fdfdfd",
    boxShadow: "0px 2px 4px rgba(0,0,0,0.05)",
  },
  removeButton: {
    padding: "5px 10px",
    backgroundColor: "#dc3545",
    color: "Black",
    border: "none",
    borderRadius: "4px",
    cursor: "pointer",
 },
};
export default App;
```

OUTPUT DEPLOY LINK - https://trstr6s.bytexl.live/

