

# Experiment-8

## Library Management UI with Search, Add, and Remove Book Functionality

### Code Implementation

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

// Main component for the Library Management UI
export default function App() {
  // State for the list of books, initialized with some default data
  const [books, setBooks] = useState([
    { id: 1, title: '1984', author: 'George Orwell' },
    { id: 2, title: 'The Great Gatsby', author: 'F. Scott Fitzgerald' },
    { id: 3, title: 'To Kill a Mockingbird', author: 'Harper Lee' },
  ]);

  // State for the search term input
  const [searchTerm, setSearchTerm] = useState('');
  // State for the new book's title input
  const [newBookTitle, setNewBookTitle] = useState('');
  // State for the new book's author input
  const [newBookAuthor, setNewBookAuthor] = useState('');

  // Function to handle adding a new book
  const handleAddBook = (e) => {
    e.preventDefault(); // Prevent form from reloading the page
    if (newBookTitle.trim() === '' || newBookAuthor.trim() === '') {
      return; // Do not add empty books
    }
    const newBook = {
      id: Date.now(), // Use a timestamp for a unique ID
      title: newBookTitle,
      author: newBookAuthor,
    };
    setBooks([...books, newBook]);
    // Clear the input fields after adding
    setNewBookTitle('');
    setNewBookAuthor('');
  };
}
```

```
};
```

```
// Function to handle removing a book by its ID
const handleRemoveBook = (id) => {
  setBooks(books.filter(book => book.id !== id));
};
```

```
// Filter the books based on the search term (case-insensitive)
const filteredBooks = books.filter(book =>
  book.title.toLowerCase().includes(searchTerm.toLowerCase()) ||
  book.author.toLowerCase().includes(searchTerm.toLowerCase())
);
```

```
return (
  <div className="bg-gray-100 min-h-screen font-sans">
    <div className="container mx-auto p-8 max-w-3xl">
      <div className="bg-white p-8 rounded-lg shadow-lg border border-gray-200">
        <h1 className="text-3xl font-bold mb-6 text-gray-800">Library Management</h1>
```

```
        {/* Search Input */}
        <div className="mb-6">
          <input
            type="text"
            placeholder="Search by title or author"
            className="w-full px-4 py-2 border border-gray-300 rounded-md
focus:outline-none focus:ring-2 focus:ring-indigo-500"
            value={searchTerm}
            onChange={(e) => setSearchTerm(e.target.value)}
          />
        </div>

        {/* Add New Book Form */}
        <form onSubmit={handleAddBook} className="mb-8 flex flex-col sm:flex-row gap-4">
          <input
            type="text"
            placeholder="New book title"
            className="flex-grow px-4 py-2 border border-gray-300 rounded-md
focus:outline-none focus:ring-2 focus:ring-indigo-500"
            value={newBookTitle}
            onChange={(e) => setNewBookTitle(e.target.value)}
          />
          <input
            type="text"
```

```

        placeholder="New book author"
        className="flex-grow px-4 py-2 border border-gray-300 rounded-md
focus:outline-none focus:ring-2 focus:ring-indigo-500"
        value={newBookAuthor}
        onChange={(e) => setNewBookAuthor(e.target.value)}
    />
    <button
        type="submit"
        className="bg-indigo-600 text-white px-6 py-2 rounded-md hover:bg-indigo-700
transition-colors focus:outline-none focus:ring-2 focus:ring-indigo-500 focus:ring-offset-2"
        >
        Add Book
    </button>
</form>

{/* Book List */}
<div>
    {filteredBooks.map(book => (
        <div
            key={book.id}
            className="flex items-center justify-between p-4 mb-3 border border-gray-200
rounded-lg shadow-sm hover:bg-gray-50 transition-colors"
            >
            <div>
                <span className="font-bold text-gray-900">{book.title}</span>
                <span className="text-gray-600"> by {book.author}</span>
            </div>
            <button
                onClick={() => handleRemoveBook(book.id)}
                className="bg-red-500 text-white px-4 py-1 rounded-md hover:bg-red-600
transition-colors focus:outline-none focus:ring-2 focus:ring-red-500 focus:ring-offset-2
text-sm"
                >
                Remove
            </button>
        </div>
    ))}
</div>
</div>
</div>
</div>
);
}

```

