**Exercise 1: Button Click Counter**

**Task:**

* Create a component with a button.
* Increment a counter each time the button is clicked.

**Exercise 2: Input Field Event**

**Task:**

* Create a component with an input field.
* Display the input text below the field dynamically as the user types.

**Exercise 3: Mouse Hover Effect**

**Task:**

* Create a component with a box.
* Change the color of the box when the mouse hovers over it.

**Exercise 4: Toggle Theme**

**Task:**

* Create a component with a button.
* Toggle between light and dark themes when the button is clicked.

**Exercise 5: Click Tracker**

**Task:**

* Create a component that tracks the number of times a specific area is clicked.
* Reset the count when a reset button is clicked.

**Exercise 6: Todo List Manager**

**Task:**

* Create a Todo app with the following components:
  + **TodoInput**: Input field and button for adding a todo.
  + **TodoItem**: Displays a single todo item with a delete button.
  + **TodoList**: Displays all todo items.

**Exercise 7: User Form with Preview**

**Task:**

* Create a form to collect user data with the following components:
  + **UserForm**: Collects user data (name, email, age).
  + **UserPreview**: Displays live preview of the entered data.
  + **SubmitButton**: Allows submission of data.

**Exercise 8: Form Validation Application**

**Task**:

* Create a form with fields: Name, Email, and Password.
* Include three components: Form, Field, and ValidationMessage.
* Use state to track inputs and show validation messages for required fields.

**Exercise 9: Multi-Step Form Application**

**Task**:  
Create a multi-step form with three steps:

1. **Step 1**: Collect personal information (First Name, Last Name, Age).
2. **Step 2**: Collect account details (Email, Password).
3. **Step 3**: Display confirmation of the entered details.

**Navigation**:

* Use Next and Back buttons to navigate between steps.
* Include validation for required fields before moving to the next step.

**Exercise 10: To-Do List with Filters**

**Task**:  
Create a to-do list application with the following features:

* Add new tasks.
* Mark tasks as completed.
* Filter tasks by:
  + All Tasks.
  + Completed Tasks.
  + Pending Tasks.

**Components**:

1. **TodoInput**: For adding new tasks.
2. **TodoList**: For displaying the list of tasks.
3. **TodoFilters**: For filtering tasks.

**Exercise 11: Recipe Manager**

**Task**:  
Build a Recipe Manager app that allows users to:

* Add new recipes with a title and ingredients.
* Display a list of recipes.
* View detailed information about a selected recipe.

**Components**:

1. **RecipeInput**: For adding new recipes.
2. **RecipeList**: Displays a list of recipes.
3. **RecipeDetails**: Shows the details of the selected recipe.

**Exercise 12: Library Management System**

**Task**:  
Create a Library Management System that allows users to:

* View a list of available books.
* Borrow a book (mark it as borrowed).
* Add a new book to the library.

**Components**:

1. **BookList**: Displays the list of books with a "Borrow" button for each.
2. **AddBookForm**: Allows adding a new book to the list.
3. **BorrowedBooks**: Displays a list of borrowed books.

**Exercise 13: Task Management Application**

**Task**:  
Create a Task Management Application with the following features:

* Add tasks with a description and priority level.
* Mark tasks as completed.
* Filter tasks based on their priority.

**Components**:

1. **TaskList**: Displays the list of tasks with options to mark them as completed.
2. **AddTaskForm**: Allows adding a new task with a priority level.
3. **TaskFilter**: Allows filtering tasks based on priority level.

**Exercise 14: Expense Tracker Application**

**Task**:  
Create an Expense Tracker Application with the following features:

* Add new expenses with a category and amount.
* Display a summary of expenses by category.
* Allow filtering of expenses by category.

**Components**:

1. **ExpenseList**: Displays the list of expenses and their categories.
2. **AddExpenseForm**: Allows adding a new expense with its category and amount.
3. **ExpenseSummary**: Displays the total expenses for each category.

**Exercise 15: Task Management Application**

**Task**:  
Create a Task Management Application with the following features:

* Users can add tasks with a title, description, and due date.
* Tasks can be marked as completed.
* Tasks can be edited or deleted.
* Tasks are categorized (e.g., "Work", "Personal").
* Filter tasks based on category and completion status.
* Display a summary of total tasks and completed tasks.

**Components**:

1. **TaskList**: Displays the list of tasks.
2. **TaskItem**: Represents each task in the list.
3. **AddTaskForm**: Form for adding new tasks.
4. **TaskSummary**: Displays the count of total tasks and completed tasks.
5. **TaskFilter**: Filters tasks based on category and completion status.
6. **CategoryFilter**: Allows filtering tasks by category.
7. **CompletionFilter**: Allows filtering tasks by completion status.
8. **TaskEditForm**: Form to edit an existing task.
9. **DeleteTaskButton**: Button to delete a task.
10. **CompleteTaskButton**: Button to mark a task as completed.