



Custom Hooks exercises

1. Counter Custom Hook

Create a custom hook called `useCounter` that manages the state and logic for a counter component.

The hook should return the current count, an increment function, and a decrement function.

Implement a simple counter component using the `useCounter` hook

2. Mouse Position Custom Hook

Create a custom hook called `useMousePosition` that tracks the user's mouse position on the screen.

The hook should return the current x and y coordinates of the mouse pointer.

Implement a component that displays the mouse position on the screen using the `useMousePosition` hook.

3. Fetch Data Custom Hook

Create a custom hook called `useFetch` to fetch data from an API.

The hook should accept a URL as a parameter and return the fetched data, loading state, and error state.

Implement a component that fetches data from a sample API using the `useFetch` hook and displays the results.

4. Local Storage Custom Hook

Create a custom hook called `useLocalStorage` to store and retrieve data from the browser's local storage.

The hook should accept a key and an initial value, and return the current value and a function to update the value.

Implement a component that uses the `useLocalStorage` hook to store and retrieve user input.