React Hooks Assignment Questions

1.UseState Hook:

- Create a React component with a button and a counter.
- Implement the `useState` hook to update the counter when the button is clicked.
- Display the updated counter value on the screen.

2.UseEffect Hook:

- Build a React component that fetches data from an API on component mount.
- Use the `useEffect` hook to perform the data fetching operation.
- Display the fetched data on the screen.

3.UseReducer Hook:

- Develop an application for managing a shopping cart.
- Implement the `useReducer` hook to manage the cart's state.
- Enable users to add or remove items from the cart.

4.UseContext Hook:

- Create a multi-level nested component structure.
- Use the `useContext` hook to pass data from a top-level component to a deeply nested component.
- Demonstrate the avoidance of prop drilling.

5.UseRef Hook:

- Construct a form with an input field and a button.
- Implement the `useRef` hook to focus the input field on component mount.
- Allow clearing the input field when the button is clicked.

6.UseMemo Hook:

- Develop a computationally intensive component with complex calculations or operations.

- Implement the `useMemo` hook to optimize performance by memoizing the computation result.

- Show prevention of unnecessary re-renders on state changes.

7.UseCallback Hook:

- Create a parent component and a child component with a callback function from the parent.

- Implement the `useCallback` hook to prevent unnecessary re-renders of the child component on parent state changes.

- Show optimization of rendering performance.

8.Custom Hook:

- Design a custom hook for handling complex logic or repetitive tasks.

- Implement the custom hook in multiple components to showcase reusability.

- Demonstrate how the custom hook simplifies component logic and promotes code reuse.