

DEVOPS AND FULLSTACK

LAB-09(2/2/26)

HTNO:2303A51184

BATCH-03

LAB PRACTICAL 9

Scenario: Counter Application (Understanding State)

Scenario Description: You are developing a simple **Counter Application** to understand **React state**. The application should allow users to **increment, decrement, and reset** a counter value.

Task Instructions

Setup

- Create a React project using `create-react-app`.
- Start the development server.

Implementation Tasks

- Create a Counter component.
- Initialize state using `useState()`.
- Add buttons for:
 - Increment
 - Decrement
 - Reset
- Update the counter value using state setters.

Bonus Challenge

- Prevent the counter from going below zero.
- Display a message when a certain value is reached.

Expected Output

- Counter value updates dynamically.
- Proper usage of React state.
- Re-rendering on state change.

CODE:

<!DOCTYPE html>

<html lang="en">

```
<head>

  <meta charset="UTF-8">
  <title>Counter App</title>

<script
src="https://unpkg.com/react@18/umd/react.development.js"></script>

<script src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></script>

<script
src="https://unpkg.com/@babel/standalone/babel.min.js"
"></script>

<style>

  body {
    text-align: center;
    margin-top: 50px;
    font-family: Arial;
  }

  button {
    margin: 5px;
  }

</style>
```

```
padding: 10px 20px;  
font-size: 16px;  
}  
</style>  
</head>  
<body>  
  
<div id="root"></div>  
  
<script type="text/babel">  
function Counter() {  
  const [count, setCount] = React.useState(0);  
  
  const increment = () => {  
    setCount(count + 1);  
  };  
  
  const decrement = () => {  
    if (count > 0) {  
      setCount(count - 1);  
    }  
  };  
}</script>
```

```
    }  
};
```

```
const reset = () => {  
  setCount(0);  
};
```

```
return (  
  <div>
```

```
    <button  
      onClick={increment}>Increment</button>  
    <button  
      onClick={decrement}>Decrement</button>  
    <button onClick={reset}>Reset</button>
```

```
  /* Message when certain value is reached */
```

```
  {count === 5 && <p>🎉 You reached 5!</p>}
```

```
</div>
```

```
    );  
}  
  
ReactDOM.createRoot(document.getElementById("root"))  
.render(<Counter />);  
  
</script>  
  
</body>  
</html>
```

OUTPUT:

