

Hooks

What are Hooks?

- With hooks we can add special feature to react components
- Managing data (state) with **useState**.
- Doing something when the component updates (side effects) with **useEffect**.

States in the lifecycle of an app

1. UI State

Controls what the user sees on the screen (e.g., a loading spinner, showing/hiding a modal).

Example:

- Is the button disabled?
- Is a dropdown open?

2. Data State

Stores the actual data your app works with (e.g., a list of tasks, user input).

Example:

- What's typed in the search bar?
- The list of items fetched from an API.

3. Communication State

Tracks interactions with the outside world (e.g., API request status, online/offline status).

Example:

- Is the API call in progress?
- Did the data load successfully?

4. Session State

Tracks the user's current session (e.g., logged-in user details).

Example:

- Who is the current user?
- What's in the shopping cart?

useState Hook

- Managing component state with `useState`.
- Examples: Counter, Input fields.

useEffect Hook

- Adding side effects with `useEffect`.
- Dependency array and lifecycle behavior.

USESTATE HOOK

```
const [state, setState] = useState(initialState);
```

`state`: The current state value.

`setState`: Function to update the state.

`initialState`: The initial value of the state.

Example: Counter component

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

```
function Counter() {
```

```
  const [count, setCount] = useState(0);
```

```
  return (
```

```
    <div>
```

```
    <h2>Counter: {count}</h2>
```

```
    <button onClick={() => setCount(count + 1)}>Increment</button>
```

```
    <button onClick={() => setCount(count - 1)}>Decrement</button>
```

```
    <button onClick={() => setCount(0)}>Reset</button>
```

```
    </div>
```

```
  );
```

```
}
```

```
export default Counter;
```

Here we can have a default value of count, and update it with `setCount`, then display the new, updated value of count.

useEffect

Runs some code when the component renders or updates (e.g., fetching data or logging).

```
import React, { useState, useEffect } from "react";

function Logger() {
  const [value, setValue] = useState("");

  useEffect(() => {
    console.log("Value changed to:", value);
  }, [value]); // Runs only when 'value' changes

  return (
    <input
      type="text"
      value={value}
      onChange={(e) => setValue(e.target.value)}
    />
  );
}
```