Day 3 — Props & State (Basics)

Goal:

Understand how data flows in React using **Props** and how components manage data using **State**.

---

1. What are Props?

Props (properties) are used to **pass data from one component to another**.

Think of props like **function arguments**.

Example:

```
function Greeting(props) {
  return <h2>Hello {props.name}</h2>;
}

function App() {
  return <Greeting name="Malik" />;
}
```

---

2. Why Props are Important?

- Make components reusable
- Same component, different data
- Clean and dynamic UI

---

3. Destructuring Props (Recommended)

Cleaner way to use props:

```
function Greeting({ name }) {
  return <h2>Hello {name}</h2>;
}
```

---

4. What is State?

State is **component's own data** that can change over time.

For state, we use the `useState` hook.

Example:

```
import { useState } from 'react';

function Counter() {
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>Count: {count}</p>
      <button onClick={() => setCount(count + 1)}>+</button>
    </div>
  );
}
```

---

5. Difference Between Props & State

Props	State
Passed from parent	Managed inside component
Read-only	Can be updated
Used for data	Used for interactivity

---

6. Simple Example (Props + State Together)

```
function User({ name }) {
  const [online, setOnline] = useState(false);

  return (
    <div>
      <h3>{name}</h3>
      <p>Status: {online ? 'Online' : 'Offline'}</p>
      <button onClick={() => setOnline(!online)}>Toggle</button>
    </div>
```

```
  );
}
```

---

7. Practice Tasks (Must Do)

Task 1:

Create a `UserCard` component that receives: - name - email

Task 2:

Add a button to show/hide email using state

Task 3:

Render multiple `UserCard` components with different props

---

8. Common Mistakes

- Trying to change props ❌
- Forgetting to import `useState`
- Updating state directly ❌(always use setter)

---

9. End of Day Result

By end of Day 3, you should: - Understand props vs state - Pass data between components - Create interactive UI

Next Day Preview (Day 4)

- Event handling
- Forms & inputs
- Controlled components
