

React State

১. State কী?

উত্তর:

State হলো **React component**-এর **internal data** যেটা সময়ের সাথে পরিবর্তন হতে পারে। যখন **state** পরিবর্তন হয়, তখন **React UI** আবার **re-render** করে।

২. Props আর State-এর মধ্যে পার্থক্য কী?

<u>বিষয়</u>	<u>Props</u>	<u>State</u>
কে সেট করে	Parent component	Component নিজে
পরিবর্তন করা যায়?	না (read-only)	হ্যাঁ (mutable)
ব্যবহার	Data pass করতে	Data পরিবর্তন ও UI control করতে
ধরন	External data	Internal data

৩. State সরাসরি পরিবর্তন করা যায় কি?

উত্তর: না ❌, কখনো সরাসরি **state** পরিবর্তন করা যায় না।

React-এ **state** পরিবর্তন করতে হলে **setState()** (class) বা **setter function (setCount)** (functional) ব্যবহার করতে হয়।

কারণ **React direct change detect** করতে পারে না, কিন্তু **setter** দিয়ে **change** করলে **React** বুঝতে পারে এবং **UI update** করে।

৪. Class Component-এ State কিভাবে declare ও update করা হয়?

উত্তর:

```
constructor() {  
  super();  
  this.state = { count: 0 };  
}  
this.setState({ count: this.state.count + 1 });
```

- **constructor** এ **state set** হয়
- **this.setState()** দিয়ে **state** আপডেট হয়

৫. **useState()** কী এবং এটা কিভাবে কাজ করে?

উত্তর: **useState()** হলো **React** এর **built-in function** যা **state update** করে এবং **UI automatically re-render** করে।

React asynchronously এই কাজ করে যাতে **performance** ভালো থাকে।

৬. **Functional Component**-এ **useState()** কী কাজ করে?

উত্তর:

useState() হলো **React hook** যা **functional component**-এ **state** যোগ করতে সাহায্য করে।

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

- **count** → **state variable**
- **setCount()** → **state পরিবর্তনের function**

৭. **useState()** কয়টা **value return** করে?

উত্তর: দুইটা **value return** করে:

- 1 **state variable**
- 2 **setter function**

উদাহরণ:

```
const [name, setName] = useState("Sona");
```

৮. **State**-এর মধ্যে কী ধরনের **data** রাখা যায়?

উত্তর:

সব ধরনের **data** রাখা যায় — যেমন:

- **number**
- **string**
- **array**

- object
- boolean
- even another function

উদাহরণ :

```
const [user, setUser] = useState({ name: "Babu", age: 20 });
```

৯. এক **component**-এ একাধিক **useState()** ব্যবহার করা যায় কি?

উত্তর: হ্যাঁ ☒

১০. **useState()** কি **component**-এর বাইরে ব্যবহার করা যায়?

উত্তর: না ☒, **useState()** কেবল **React functional component** বা **custom hook** এর ভিতরেই ব্যবহার করা যায়।

১১. **State mutable** না **immutable**?

উত্তর: **State immutable**, মানে সরাসরি পরিবর্তন করা যায় না।
Setter function দিয়েই নতুন মান দেওয়া যায়।

১২. কেন **useState()** **class component**-এর থেকে ভালো?

উত্তর:

- ☒ কোড ছোট ও সহজ
- ☒ **this** বা **constructor** লাগে না
- ☒ **Multiple state** সহজে **handle** করা যায়
- ☒ **Functional component** দ্রুত চলে
- ☒ **Debug** করা সহজ

১৩. যখন **state change** হয় তখন কী হয়?

উত্তর: **React UI automatically re-render** করে এবং নতুন **data** দেখায়।

🌿 ১৪. State vs Props – ছোট উত্তর

- **Props:** বাইরের থেকে আসে, পরিবর্তন করা যায় না
- **State:** component-এর ভিতরের **data**, পরিবর্তন করা যায়

🌿 ১৫. setState() **synchronous** না **asynchronous**?

উত্তর:

Asynchronous ⚡

মানে, **React** একটু দেরি করে **state update** করে যাতে **performance** ভালো থাকে এবং একাধিক **update** একসাথে **handle** করা যায়।

🌿 ১৬. State **synchronous** না **asynchronous**?

উত্তর: এইটা একটু **clarify** করা দরকার — কারণ **state** নিজেই **synchronous** বা **asynchronous** নয়, কিন্তু **state update** এর **behavior asynchronous**।

🌿 ১৭. React-এ state কেন দরকার?

উত্তর: কারণ **UI dynamic** রাখতে হলে **data** পরিবর্তনের সাথে **React** কে জানাতে হয়।
State ছাড়া **React static** থেকে যেত।