**- Trình bày về state trong ReactJS?**

+ State trong ReactJS là một đối tượng đậc biệt giúp quản lý dữ liệu động của một component. Khi state thay đổi, component sẽ tự động render lại để phản ánh sự thay đổi đó trên giao diện.

**- setState trong React hoạt động đồng bộ hay bất đồng bộ? Tại sao?**

+ setState trong React hoạt động bất đồng bộ

Lý do:

1. Tối ưu hiệu suất: React gom nhóm nhiều thay đổi state lại và chỉ cập nhật một lần để giảm số lần re-render không cần thiết.
2. Batching (Gộp các cập nhật): Nếu nhiều setState được gọi trong cùng một sự kiện, React sẽ gộp chúng lại và cập nhật cùng lúc.
3. Cập nhật theo trạng thái mới nhất: Vì setState là bất đồng bộ, bạn nên sử dụng hàm callback để đảm bảo nhận giá trị state mới nhất.

**- Giải thích chi tiết các lifecycle methods của Component React**

Lifecycle methods chỉ có trong Class Component, được chia thành 3 giai đoạn:

**Giai đoạn 1: Mounting (Khi component được tạo)**

* constructor(): Khởi tạo state, bind event.
* render(): Trả về JSX để hiển thị giao diện.
* componentDidMount(): Chạy sau khi component hiển thị lên UI, dùng để gọi API hoặc thao tác DOM.

**Giai đoạn 2: Updating (Khi component được cập nhật)**

* shouldComponentUpdate(nextProps, nextState): Quyết định có re-render hay không.
* render(): Hiển thị lại UI khi state/props thay đổi.
* componentDidUpdate(prevProps, prevState): Chạy sau khi component cập nhật, thường dùng để gọi API dựa vào state mới.

**Giai đoạn 3: Unmounting (Khi component bị xóa khỏi UI)**

* componentWillUnmount(): Dọn dẹp dữ liệu, clear timeout, remove event listener.