1. Prompt ban đầu: “Giải thích cấu trúc dữ liệu Stack.”

**Câu trả lời từ AI**:  
“Stack là một cấu trúc dữ liệu tuyến tính, tuân theo nguyên tắc LIFO (Last In, First Out – vào sau ra trước). Các thao tác chính trên Stack gồm: push (thêm phần tử), pop (loại bỏ phần tử), peek/top (xem phần tử trên cùng). Stack thường được dùng trong việc quản lý lời gọi hàm, xử lý biểu thức, hoặc undo/redo.”

**Nhận xét**: chính xác, nhưng hơi khô khan, khó tiếp cận với người mới.

1. Prompt sau đó: “Tôi chưa biết gì về cấu trúc dữ liệu Stack. Hãy giải thích đơn giản kèm ví dụ đời thực (ví dụ: chồng đĩa).”

**Câu trả lời từ AI**:  
“Stack giống như một **chồng đĩa** trong nhà bếp. Bạn chỉ có thể thêm một chiếc đĩa mới lên trên cùng (push), và cũng chỉ có thể lấy chiếc đĩa trên cùng ra trước (pop). Nguyên tắc là **vào sau ra trước** (LIFO).  
Ví dụ: Nếu bạn xếp lần lượt 3 chiếc đĩa A, B, C (theo thứ tự), thì chiếc C ở trên cùng. Khi lấy ra, bạn sẽ lấy C trước, rồi đến B, cuối cùng mới tới A.  
Trong lập trình, Stack được dùng khi bạn muốn lưu trữ dữ liệu và xử lý theo đúng nguyên tắc này, ví dụ như quản lý quá trình gọi hàm trong chương trình.”

**Nhận xét**: dễ hiểu, gần gũi, người mới học dễ hình dung hơn.

1. Bảng so sánh

|  |  |  |
| --- | --- | --- |
| Tiêu chí | Prompt 1: “Giải thích cấu trúc dữ liệu Stack.” | Prompt 2: “Tôi chưa biết gì... kèm ví dụ chồng đĩa” |
| Cách diễn đạt | Thuật ngữ chuyên môn (LIFO, push, pop, peek) | Dùng ví dụ đời thực (chồng đĩa) trước, rồi mới liên hệ kỹ thuật |
| Mức độ dễ hiểu | Dành cho người đã biết khái niệm cơ bản | Dành cho người mới bắt đầu, không cần nền tảng trước |
| Hình ảnh minh họa | Không có | Có ví dụ trực quan (chồng đĩa) |
| Ứng dụng | Đề cập đến ứng dụng trong lập trình | Vừa có ứng dụng thực tế, vừa có ứng dụng trong lập trình |
| Trải nghiệm người học | Khó tiếp cận nếu chưa quen thuật ngữ | Thân thiện, dễ hình dung và ghi nhớ |