1. NodeJs là gì?

NodeJs là một platform (không phải framework hay ngôn ngữ lập trình) được xây dựng trên Javascript runtime của Chrome. Phần core của NodeJs được viết bằng C++. Không có đa luồng nhưng có cơ chế bất đồng bộ

2. Cookie, session là gì? Hoạt động như thế nào?

+ Cookie được tạo bởi server và lưu trữ ở client giúp lưu trữ và truyền đạt thông tin. Mỗi lần gửi request sẽ gửi kèm cookie

+ Session là một phiên làm việc giữa các request của client và response của server. Thường khi để xác thực phiên làm việc sẽ xử lý qua cookie

+ Cách hoạt động: Điển hình nhất là khi có request đăng nhập từ client lên server. Server sẽ validate và trả về một cookie chứa các thông tin cần thiết để xác định (thường là để author) về client.

3. Phân quyền, jwt là gì, lưu ở đâu?

+ Phân quyền là một cách ngăn chặn các request không được phép hoặc không hợp lệ từ client lên server, để nhận được response thì request đấy phải có quyền “lấy” response

+ Jwt là viết tắt của jsonwebtoken, là một token dưới dạng json. Được server tạo ra và gửi về client những data cần thiết cho việc authentication, authorization. Thường lưu ở header của client

4. Index là gì? Có đánh được cho nhiều bảng không? Có bao nhiêu loại index, cách sử dụng của chúng?

+ Index là chỉ mục giúp việc truy vấn dữ liệu trong database hiệu quả hơn. Ví dụ như khi vào trong thư viện mượn sách, các cô quản thư tìm quyển sách mà mình muốn mượn dựa vào các index

+ Các loại index:

+ Đơn mức: Primary, clustering, secondary

+ Đa mức: B tree, B+ tree.

B tree –cây cân bằng AVL, insert, update, delete đều có độ phức tạp O(log n).

B+ tree – các dữ liệu chỉ lưu ở nút lá

5. Khi chèn 5 bảng mà mới bị 3 thì mất kết nối, muốn chèn lại thì như nào?

Dùng transaction, rollback lại vì transaction đảm bảo tính ACID

6. Promise, async await, callback, closure

+ Callback: Hàm gọi lại. Được thực thi sau khi 1 hàm khác thực thi xong. Tại sao có thể có callback trong JS? Vì ở JS thì function cũng có thể coi là một object. Việc trả về một object trong các ngôn ngữ là đúng chứ không sai

+ Promise: là hứa. Mang đủ tính chất của hứa, nếu thực hiện được thì resolve còn không sẽ bị thất hứa và reject. Mà đã hứa thì có thể hứa 1 hay nhiều lời hứa, việc làm được hay không đã có resolve và reject lo. Promise sinh ra giúp tránh việc callback hell khiến code dirty

+ Async await: Async biết một function bình thường thành một promise. Await là dừng lại việc async. Await chỉ sử dụng với async và promise

+ Closure: Hiểu đơn giản là các hàm bên trong 1 hàm truy cập được biến của hàm bên ngoài

<https://viblo.asia/p/ban-nen-biet-closure-hoat-dong-nhu-the-nao-aWj53VQQl6m>

+ Higher order function: Hàm có thể trả về 1 hàm, 1 hàm có thể truyền như 1 tham số