

1. ¿Cómo manejarías un flujo de errores en una API con Express usando middleware centralizado en Node.js?

En Express, la forma más fácil es poner un middleware de errores al final de todo. Es una función especial que siempre tiene cuatro argumentos: (err, req, res, next).

2. ¿Qué ventajas ofrece un ORM como Sequelize o TypeORM contra un modelo de consultas con SQL plano y cuáles son sus limitaciones?

Un ORM (como Sequelize) es una herramienta que permite usar código JavaScript para comunicarse con la base de datos, en lugar de escribir SQL.

3. ¿Cómo implementarías lazy loading en AngularJS o cómo modularizarías una aplicación grande?

Usando ui-router para definir qué partes de la aplicación se cargan solo cuando alguien visita una ruta específica.

4. ¿Qué diferencias hay entre \$scope, \$rootScope y servicios en AngularJS?

\$scope conecta la vista con el controlador (local), \$rootScope es global, y los servicios son la forma correcta y reusable de compartir datos y lógica en la app.

5. ¿Cómo integrarías una librería de machine learning en un microservicio Python?

Se podría crear una API con Flask o FastAPI. La API tendría una ruta, por ejemplo /predecir, que recibe un JSON. Dentro de esa ruta, cargo mi modelo de machine learning que ya está entrenado.

6. ¿En Bases de Datos relacionales, qué son índices compuestos y cuándo los usarías?

Un índice compuesto es como un índice de un libro, pero para dos o más columnas. Lo usarías cuando las consultas deben realizar filtros por varias columnas al mismo tiempo.

7. ¿Cómo manejarías una migración de datos entre ambientes productivos para no afectar la integridad?

Primero generaría un backup y scripts transaccionales que se aseguren un revert si algo falla dentro del proceso.

8. Diferencias entre blockchain pública y privada.

Pública: Cualquiera puede unirse, ver las transacciones, y ser parte de la red. Son lentas pero súper seguras. Piénsalo como una Wikipedia gigante.

Privada: La de un grupo cerrado. Solo los que tienen permiso pueden entrar.

9. ¿Qué es un hash y en qué proceso tecnológico lo aplicarías?

Un hash es el resultado de una función matemática que convierte cualquier cosa (texto, archivo, lo que sea) en una cadena de letras y números de tamaño fijo. Es único, y si cambias un solo carácter, el hash cambia por completo.

10. Si tuvieras que integrar la API de OpenAI para enriquecer datos, ¿qué técnicas usarías para controlar costos y latencia?

Caché: Si un usuario me pregunta algo que ya había preguntado antes, en lugar de llamar a la API de nuevo, le doy la respuesta que ya guardé.

Agrupar peticiones: Si tengo que hacer 100 llamadas pequeñas, las agrupo en una sola si es posible.