Follow-Up on Interview & Next Steps

Part 1: SQL - Creating a Table

Task:

Write an SQL script to create a table named employees with the following columns:

1. id (INTEGER, Primary Key, Auto Increment)
2. first_name (VARCHAR, 50)
3. last_name (VARCHAR, 50)
4. email (VARCHAR, 100, Unique)
5. phone_number (VARCHAR, 20, Nullable)
6. hire_date (DATE, Not Null)
7. job_title (VARCHAR, 50)
8. salary (DECIMAL(10,2))
9. department (VARCHAR, 50)
10. manager_id (INTEGER, Nullable, Foreign Key referencing id in the same table)

Requirements:

- Ensure that id is the primary key and auto-increments.
- The email field should be unique.
- The hire_date field cannot be null.
- manager_id should reference another id in the same table.

Part 2: Python - Loading Data into the Table

Task:

Write a Python program that performs the following:

1. Connects to a Microsoft SQL Server database (company_db), creating it if it doesn't exist.
2. Creates the employees table (if it does not already exist).
3. Inserts the following sample employee data into the table:

first_name	last_name	email	phone_number	hire_date	job_title	salary	department	manager_id

first_name	last_name	email	phone_number	hire_date	job_title	salary	department	manager_id
Alice	Smith	alice.smith@email.com	0987654321	2023-06-22	Manager	90000.00	HR	NULL
Bob	Johnson	bob.johnson@email.com	1231231234	2022-09-14	Analyst	65000.00	Finance	2

Requirements:

- Use pyodbc for database operations.
- Ensure the script checks if the table already exists before creating it.
- Use parameterized queries to prevent SQL injection.
- Print confirmation messages after inserting the data.

Bonus:

- Write a function to fetch and display all records from the employees table after insertion.