Follow up question:

An important aspect of the role is to be able to meet with non-technical team members and gather requirements for the database. Using the requirements, you would be asked to research different database options and propose a solution that meets the team’s needs. Please share a list of no more than 10 questions you would ask the team to develop a list of requirements you could use to evaluate different database options.

1. **What are the primary types of data you need to store and manage?**

I would want to understand the nature of the data (e.g., structured, unstructured). If the team needs the database for structured/unstructured data such as energy utilization or costs for example, we could consider using Azure SQL Database which can handle structured data, while Azure Cosmos DB is designed for semi-structured data, providing support for multiple data models, including key-value type data.

1. **What are the expected volumes of data currently and in the future?**

If the team needs up to multiple terabytes of storage, Azure blobs for example can be used for large-scale, cost-effective storage of historical data.

1. **What are the performance and speed requirements for the database?**

This will help us chalk out plans to build indexes (clustered/non-clustered) if required, and gauge how many DB servers, VMs and/or CPU cores we need to increase throughput and reduce latency.

1. **What are the security and compliance requirements for your data?**

Understand if there are specific regulations or standards (e.g., GDPR, HIPAA) that the database must comply with. If encryption and access controls are necessary, we need to choose a database that ensures robust data security and regulatory compliance.

1. **How do you plan to access and query the data? Do you plan to visualize on a dashboard?**

This will help us figure out if we need to build a reporting layer over the semantic layer where the data is initially stored. The data from the reporting layer can be used to generate a live feed to either a Power BI dashboard or Tableau dashboard based on costs and requirements.

1. **Do you need real-time data updates?**

If the team needs real-time updates to the data, we can figure out how to design and schedule data pipelines to run periodically. For example, we could set up an Azure Data Factory using batch jobs to trigger the real-time data feed.

1. **What is your budget for the database solution, including initial setup and ongoing maintenance?**

Understand financial constraints to narrow down viable options. Knowing the initial budget and the monthly budget for maintenance, licensing, and scaling will help us choose a database that offers various pricing tiers, including serverless options that allow you to pay only for the compute resources you use, fitting within budget constraints.

1. **Do you have existing systems or technologies that the new database must integrate with?**

Identify dependencies and integration requirements with current systems. This will help us move from a traditional DBMS such as Teradata/Excel to a cloud platform such as Snowflake/Azure for increased flexibility.

1. **What are your backup and recovery requirements?**

I would want to know how fast they need to restore operations in case of a disaster, and how risky data loss can be. This will help us choose a database that provides automated backups and point-in-time restore to meet backup and disaster recovery requirements.

1. **Who will be responsible for managing and maintaining the database, and what is their level of expertise?**

Understand the in-house technical capabilities to choose a solution that fits the team's skill level. If the IT team will be managing the database, we might consider a fully managed service such as Azure to minimize maintenance efforts.