

## What is Azure Cosmos DB?

- Azure Cosmos DB is a globally distributed, multi-model database service designed for massive scale, low-latency, and high-performance applications. It supports multiple data models, including document, graph, key-value, table, and column-family.

## Explain the key features of Azure Cosmos DB.

- Key features include global distribution, automatic and instant scalability, multiple consistency models, multiple data models, and comprehensive SLAs for performance, availability, and latency.

## What are the different consistency models supported by Azure Cosmos DB?

- Azure Cosmos DB offers five consistency models: Strong, Bounded staleness, Session, Consistent prefix, and Eventual. Each model provides a different trade-off between consistency, availability, and latency.

## How does global distribution work in Azure Cosmos DB?

- Global distribution allows you to replicate data across multiple Azure regions. It enables low-latency access for users in different geographic locations and provides high availability and disaster recovery.

## Explain the concept of partitioning in Azure Cosmos DB.

- Partitioning involves dividing data into logical partitions to distribute it across multiple physical partitions for scalable and efficient operations. It's a crucial concept for achieving high performance in Azure Cosmos DB.

## What is the role of the Azure Cosmos DB API in supporting multiple data models?

- Azure Cosmos DB supports multiple data models (document, graph, key-value, table, and column-family) through the use of APIs. Each API provides a specific data model, such as SQL API for document data, Gremlin API for graph data, etc.

## How does Azure Cosmos DB handle schema changes in a document database?

- Azure Cosmos DB is schema-agnostic, meaning it can handle dynamic schema changes seamlessly. It allows you to insert documents with different structures within the same collection.

## What are the security features available in Azure Cosmos DB?

- Azure Cosmos DB provides security features such as Virtual Network Service Endpoints, Firewall rules, Role-Based Access Control (RBAC), Encryption at Rest, and Azure AD-based authentication.

Explain the concept of indexing in Azure Cosmos DB.

- Indexing in Azure Cosmos DB is critical for enabling efficient query execution. It allows you to define and customize indexes based on your query patterns and performance requirements.

How do you migrate data to Azure Cosmos DB from other databases?

- Data migration to Azure Cosmos DB can be performed using various methods, including the Azure Cosmos DB Data Migration tool, Azure Data Factory, and custom scripts. The choice depends on the source database and migration requirements.

What are the best practices for designing and optimizing queries in Azure Cosmos DB?

- Best practices include proper indexing, understanding and selecting the appropriate consistency level, using partitioning effectively, and optimizing queries based on the chosen data model.