DP-420: Designing and Implementing Cloud-Native Applications Using Microsoft Azure Cosmos DB

Microsoft - Data & AI

- **Localidade:** Lisboa
- **Data:** 16 Dec 2024
- **Preço:** 1590 € (Os valores apresentados não incluem IVA. Oferta de IVA a particulares e estudantes.)
- **Horário:** Laboral das 9h00 às 17h00
- **Nível:** Intermédio
- **Duração:** 28h

Sobre o curso

This course teaches developers how to create application using the SQL API and SDK for Azure Cosmos DB.

Students will learn how to write efficient queries, create indexing policies, manage and provisioned resources, and perform common operations with the SDK.

Destinatários

- Software engineers tasked with authoring cloud-native solutions that leverage Azure Cosmos DB SQL API and its various SDKs. They are familiar with C#, Python, Java, or JavaScript. They also have experience writing code that interacts with a SQL or NoSQL database platform.

Objetivos

You will learn to:

- Prepare for Exam DP-420: Designing and Implementing Cloud-Native Applications Using Microsoft Azure Cosmos DB
- Use the .NET SDK for Azure Cosmos DB for NoSQL to perform common operations on databases, containers, and item
- Create SQL queries for the Azure Cosmos DB for NoSQL using the Data Explorer and the .NET SDK

- Create custom indexing policies for Azure Cosmos DB for NoSQL containers
- Integrate Azure Cosmos DB for NoSQL with Azure Cognitive Search, Azure Functions, and your own solutions
- Optimize the performance of your queries and operations using Azure Cosmos DB for NoSQL
- Plan and implement techniques to replicate data across the globe in Azure Cosmos DB for NoSQL
- Optimize the performance of your queries and operations using Azure Cosmos DB for NoSQL
- Review the common Cosmos DB administrative tasks of monitor, performance metrics, backup and security used in Azure
- Use the command line and Azure Resource Manager to automate common management tasks for Azure Cosmos DB for NoSQL
- Use JavaScript to author server-side stored procedures, user-defined functions, and triggers

---

Pré-requisitos

Before attending this course, students must have:

- Knowledge of Microsoft Azure and ability to navigate the Azure portal (AZ-900 equivalent)
- Experience writing in an Azure-supported language at the intermediate level. (C#, JavaScript, Python, or Java)
- Ability to write code to connect and perform operations on a SQL or NoSQL database product. (SQL Server, Oracle, MongoDB, Cassandra or similar)

Programa

- Get started with Azure Cosmos DB SQL API
- Plan and implement Azure Cosmos DB SQL API
- Connect to Azure Cosmos DB SQL API with the SDK
- Access and manage data with the Azure Cosmos DB SQL API SDKs
- Execute queries in Azure Cosmos DB SQL API
- Define and implement an indexing strategy for Azure Cosmos DB SQL API
- Integrate Azure Cosmos DB SQL API with Azure services
- Implement a data modeling and partitioning strategy for Azure Cosmos DB SQL API
- Design and implement a replication strategy for Azure Cosmos DB SQL API
- Optimize query performance in Azure Cosmos DB SQL API
- Monitor and troubleshoot an Azure Cosmos DB for NoSQL solution
- Manage an Azure Cosmos DB for NoSQL solution using DevOps practices
- Create server-side programming constructs in Azure Cosmos DB for NoSQL

Get started with Azure Cosmos DB SQL API

- Introduction to Azure Cosmos DB for NoSQL
- Try Azure Cosmos DB for NoSQL

Plan and implement Azure Cosmos DB SQL API

- Plan Resource Requirements
- Configure Azure Cosmos DB for NoSQL database and containers
- Move data into and out of Azure Cosmos DB for NoSQL

Connect to Azure Cosmos DB SQL API with the SDK

- Use the Azure Cosmos DB for NoSQL SDK
- Configure the Azure Cosmos DB for NoSQL SDK

Access and manage data with the Azure Cosmos DB SQL API SDKs

- Implement Azure Cosmos DB for NoSQL point operations
- Perform cross-document transactional operations with the Azure Cosmos DB for NoSQL
- Process bulk data in Azure Cosmos DB for NoSQL

Execute queries in Azure Cosmos DB SQL API

- Query the Azure Cosmos DB for NoSQL
- Author complex queries with the Azure Cosmos DB for NoSQL

Define and implement an indexing strategy for Azure Cosmos DB SQL API

- Define indexes in Azure Cosmos DB for NoSQL
- Customize indexes in Azure Cosmos DB for NoSQL

Integrate Azure Cosmos DB SQL API with Azure services

- Consume an Azure Cosmos DB for NoSQL change feed using the SDK
- Handle events with Azure Functions and Azure Cosmos DB for NoSQL change feed
- Search Azure Cosmos DB for NoSQL data with Azure Cognitive Search

Implement a data modeling and partitioning strategy for Azure Cosmos DB SQL API

- Implement a non-relational data model
- Design a data partitioning strategy

Design and implement a replication strategy for Azure Cosmos DB SQL API

- Configure replication and manage failovers in Azure Cosmos DB
- Use consistency models in Azure Cosmos DB for NoSQL
- Configure multi-region write in Azure Cosmos DB for NoSQL

Optimize query performance in Azure Cosmos DB SQL API

- Customize an indexing policy in Azure Cosmos DB for NoSQL
- Measure index performance in Azure Cosmos DB for NoSQL
- Implement integrated cache in Azure Cosmos DB for NoSQL

Monitor and troubleshoot an Azure Cosmos DB for NoSQL solution

- Measure performance in Azure Cosmos DB for NoSQL
- Monitor responses and events in Azure Cosmos DB for NoSQL
- Implement backup and restore for Azure Cosmos DB for NoSQL
- Implement security in Azure Cosmos DB for NoSQL

Manage an Azure Cosmos DB for NoSQL solution using DevOps practices

- Write management scripts for Azure Cosmos DB for NoSQL
- Create resource template for Azure Cosmos DB for NoSQL

Create server-side programming constructs in Azure Cosmos DB for NoSQL

- Build multi-item transactions with the Azure Cosmos DB for NoSQL
- Expand query and transaction functionality in Azure Cosmos DB for NoSQL