

DP-3015: Getting Started with Cosmos DB NoSQL Development

Course Details

Course Code: DP-3015

Duration: 1 day

Notes:

- This course syllabus should be used to determine whether the course is appropriate for the students, based on their current skills and technical training needs.
- Course content, prices, and availability are subject to change without notice.
- Terms and Conditions apply

Elements of this syllabus are subject to change.

About this course

This course teaches developers to utilize Azure Cosmos DB for NoSQL API and SDK. Students will learn query execution, resource configuration, SDK operations, and design strategies for non-relational data modelling and data partitioning.

Audience Profile

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

Academy IT Pty Ltd

Level 4, 45 Grenfell Street
ADELAIDE 5000

Email: sales@academyit.com.au

Web: www.academyit.com.au

Phone: 08 7324 9800

Brian: 0400 112 083

Introduction to Azure Cosmos DB for NoSQL

Learn about the Azure Cosmos DB for NoSQL and determine if it is a good fit for your application.

Learning objectives

After completing this module, you'll be able to:

- Evaluate whether Azure Cosmos DB for NoSQL is the right database for your application.
- Describe how the features of the Azure Cosmos DB for NoSQL are appropriate for modern applications.

Try Azure Cosmos DB for NoSQL

Learn how to use the Azure Cosmos DB for NoSQL to create an account, and then use the account to create Cosmos DB resources.

Learning objectives

After completing this module, you'll be able to:

- Create a new Azure Cosmos DB for NoSQL account
- Create database, container, and item resources for an Azure Cosmos DB for NoSQL account

Plan Resource Requirements

Familiarize yourself with the various configuration options for a new Azure Cosmos DB for NoSQL account.

Learning objectives

After completing this module, you'll be able to:

- Evaluate various requirements of your application

Configure Azure Cosmos DB for NoSQL database and containers

Select between the various throughput offerings in Azure Cosmos DB for NoSQL.

Learning objectives

After completing this module, you'll be able to:

- Compare the various service and throughput offerings for Azure Cosmos DB
- Migrate between standard and autoscale throughput

Use the Azure Cosmos DB for NoSQL SDK

Learn about the Microsoft.Azure.Cosmos library, and then download the library to use in a .NET application.

Learning objectives

After completing this module, you'll be able to:

- Integrate the Microsoft.Azure.Cosmos SDK library from NuGet
- Connect to an Azure Cosmos DB for NoSQL account using the SDK and .NET

Configure the Azure Cosmos DB for NoSQL SDK

Learn how to configure the Azure Cosmos DB for NoSQL SDK in various ways including how to integrate with the emulator, implement parallelism, and create a custom logger.

Learning objectives

After completing this module, you'll be able to:

- Configure the SDK for offline development
- Troubleshoot common connection errors
- Implement parallelism in the SDK
- Configure logging using the SDK

Implement Azure Cosmos DB for NoSQL point operations

Write code to create, read, update, and delete items in Azure Cosmos DB for NoSQL.

Learning objectives

After completing this module, you'll be able to:

- Perform CRUD operations using the SDK
- Configure TTL for a specific item

Query the Azure Cosmos DB for NoSQL

Author queries for Azure Cosmos DB for NoSQL using the SQL query language.

Learning objectives

After completing this module, you'll be able to:

- Create and execute a SQL query
- Project query results
- Use built-in functions in a query

Author complex queries with the Azure Cosmos DB for NoSQL

Create SQL queries for Azure Cosmos DB for NoSQL that uses subqueries or cross-products.

Learning objectives

After completing this module, you'll be able to:

- Implement a correlated subquery
- Create a cross-product query

Implement a non-relational data model

Identify an application's key access patterns.

Define the entities' data model, along with containers to store the data with a partition key that will result in an efficient and scalable data store for the application.

Learning objectives

In this module, you will:

- Determine access patterns for data.
- Apply data model and partitioning strategies to support an efficient and scalable NoSQL database.

Design a data partitioning strategy

Identify strategies for managing relationships between data entities, such as customers and sales data. Improve data-model performance and scaling by pre-aggregating and denormalizing your data. Use change-feed to maintain your data's referential integrity.

Learning objectives

At the completing this module we will learn:

- Manage relationships between data entities by using advanced modeling and partitioning strategies.
- Maintain the referential integrity of your data by using change feed.
- Implement pre-aggregating and denormalizing data strategies to improve data-model performance and scaling.
- Optimizing storage and compute by mixing entity types in a single container