# MS-DP420T00: DESIGNING AND IMPLEMENTING CLOUD-NATIVE APPLICATIONS USING MICROSOFT AZURE COSMOS DB

| DURATION | LEVEL | TECHNOLOGY | DELIVERY METHOD | TRAINING CREDITS |
|---|---|---|---|---|
| 4 Days | Intermediate | Azure Data & AI | Instructor-led | NA |

## INTRODUCTION

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.

## AUDIENCE PROFILE

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.

## PREREQUISITES

Before attending this course, students must have:

- Familiarity with Azure and the Azure portal.
- Experience programming with C#. If you have no previous programming experience, we recommend you complete the Take your first steps with C# learning path before starting this one.

## COURSE OBJECTIVES

After completing this course, students will be able to:

- Create and configure Azure Cosmos DB SQL API account, database, and container
- Use the .NET SDK to manage resources and perform operations
- Perform queries of varying complexity
- Design a data modelling and partitioning strategy
- Optimize queries and indexes based on characteristics of an application
- Use the Azure Resource Manager to manage accounts and resources with CLI or JSON and Bicep templates

## COURSE CONTENT

**Module 1: 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.

Lessons

- Introduction
- What is Azure Cosmos DB for NoSQL
- How does Azure Cosmos DB for NoSQL work
- When should you use Azure Cosmos DB for NoSQL
- Knowledge check
- Summary

After completing this module, students will 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.

**Module 2: 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.

Lessons

- Introduction
- Explore resources
- Review basic operations
- Exercise: Create an Azure Cosmos DB for NoSQL account
- Knowledge check
- Summary

After completing this module, students will 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

**Module 3: Plan Resource Requirements**

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

Lessons

- Introduction
- Understand throughput
- Evaluate throughput requirements
- Evaluate data storage requirements
- Time-to-live (TTL)

- Plan for data retention with time-to-live (TTL)
- Knowledge check
- Summary

After completing this module, students will be able to:

- Evaluate various requirements of your application

**Module 4: Configure Azure Cosmos DB for NoSQL**

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

Lessons

- Introduction
- Serverless
- Compare serverless vs. provisioned throughput
- Autoscale throughput
- Compare autoscale vs. standard (manual) throughput
- Migrate between standard (manual) and autoscale throughput
- Exercise: Configure throughput for Azure Cosmos DB for NoSQL with the Azure portal
- Knowledge check
- Summary

After completing this module, students will be able to:

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

**Module 5: Move data into and out of Azure Cosmos DB for NoSQL**

Migrate data into and out of Azure Cosmos DB for NoSQL using Azure services and open-source solutions.

Lessons

- Introduction
- Move data by using Azure Data Factory
- Move data by using a Kafka connector
- Move data by using Stream Analytics
- Move data by using the Azure Cosmos DB Spark connector
- Exercise: Migrate existing data using Azure Data Factory
- Knowledge check
- Summary

After completing this module, students will be able to:

- Create and execute a SQL query
- Project query results
- Use built-in functions in a query
- Implement a corelated subquery
- Create a cross-product query

**Module 6: Use the Azure Cosmos DB for NoSQL SDK**

Learn about the Azure Cosmos DB libraries, and then download the libraries to use in a .NET, Python, or JavaScript (Node.JS) application.

Lessons

- Introduction
- Understand the SDK
- Import from package manager
- Connect to an online account
- Implement client singleton
- Configure connectivity mode
- Exercise: Connect to Azure Cosmos DB for NoSQL with the SDK
- Knowledge check
- Summary

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

- Integrate the Microsoft .NET, Python, and JavaScript Azure Cosmos DB SDK libraries using package managers
- Connect to an Azure Cosmos DB for NoSQL account using an SDK and .NET, Python, or JavaScript

**Module 7: 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.

Lessons

- Introduction
- Enable offline development
- Handle connection errors
- Implement threading and parallelism
- Configure logging
- Exercise: Configure the Azure Cosmos DB for NoSQL SDK for offline development
- Knowledge check
- Summary

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

**Module 8: Implement Azure Cosmos DB for NoSQL point operations**

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

Lessons

- Introduction
- Understand point operations
- Create documents
- Read a document
- Update documents
- Configure time-to-live (TTL) value for a specific document
- Delete documents
- Exercise: Create and update documents with the Azure Cosmos DB for NoSQL SDK
- Knowledge check
- Summary

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

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

**Module 9: Perform cross-document transactional operations with the Azure Cosmos DB for NoSQL**

Perform operations on multiple items in single logical units of work.

Lessons

- Introduction
- Create a transactional batch with the SDK
- Review batch operation results with the SDK
- Exercise: Batch multiple point operations together with the Azure Cosmos DB for NoSQL SDK
- Implement optimistic concurrency control
- Knowledge check
- Summary

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

- Create a transactional batch and review results
- Implement optimistic concurrency control for an operation

**Module 10: Process bulk data in Azure Cosmos DB for NoSQL**

Perform bulk operations on Azure Cosmos DB in bulk from code using the SDK.

Lessons

- Introduction
- Create bulk operations with the SDK
- Review bulk operation caveats
- Implement bulk best practices
- Exercise: Move multiple documents in bulk with the Azure Cosmos DB for NoSQL SDK
- Knowledge check
- Summary

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

- Use C# task asynchronous programming model and "bulk" support in the Azure Cosmos DB for NoSQL .NET SDK

**Module 11: Query the Azure Cosmos DB for NoSQL**

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

Lessons

- Introduction
- Understand NoSQL query language
- Create queries with NoSQL
- Project query results
- Implement type-checking in queries
- Use built-in functions
- Execute queries in the SDK
- Exercise: Execute a query with the Azure Cosmos DB for NoSQL SDK
- Knowledge check
- Summary

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

**Module 12: 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.

Lessons

- Introduction
- Create cross-product queries
- Implement correlated subqueries
- Implement variables in queries
- Paginate query results
- Exercise: Paginate cross-product query results with the Azure Cosmos DB for NoSQL SDK
- Knowledge check
- Summary

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

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

**Module 13: Build Generative AI applications with Azure Cosmos DB**

Build advanced Generative AI application with Python and Azure Cosmos DB for NoSQL.

Lessons

- Introduction
- Configure the Vector Search and storage feature of Azure Cosmos DB NoSQL
- Exercise - Enable the Azure Cosmos DB for NoSQL Vector Search feature
- Generate embeddings using Azure OpenAI Service
- Exercise - Generate vector embeddings with Azure OpenAI and store them in Azure Cosmos DB for NoSQL
- Build Generative AI applications with Azure Cosmos DB NoSQL and Python
- Perform vector searches using Azure Cosmos DB for NoSQL from a Generative AI application
- Exercise - Build a Generative AI application with Python and Azure Cosmos DB for NoSQL
- Integrate LangChain orchestration to improve efficiency and code maintainability in a Python Generative AI application
- Exercise - Implement RAG with LangChain and Azure Cosmos DB for NoSQL Vector Search
- Knowledge check
- Summary

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

- Build a Generative AI application capable of interacting with private data using Python and Azure Cosmos DB for NoSQL.
- Efficiently retrieve and store vectors in Azure Cosmos DB for NoSQL.
- Perform similarity searches using Azure Cosmos DB for NoSQL.
- Integrate data and AI models using LangChain orchestration to create intelligent and adaptable Generative AI application.

**Module 14: Define indexes in Azure Cosmos DB for NoSQL**

Discover indexes and indexing policies in Azure Cosmos DB for NoSQL.

- Introduction
- Understand indexes
- Understand indexing policies
- Review indexing policy strategies
- Exercise: Review the default index policy for an Azure Cosmos DB for NoSQL container with the portal
- Knowledge check
- Summary

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

**Module 15: Customize indexes in Azure Cosmos DB for NoSQL**

Customize indexing policies for a container in Azure Cosmos DB for NoSQL.

Lessons

- Introduction
- Customize the indexing policy
- Evaluate composite indexes
- Exercise: Configure an Azure Cosmos DB for NoSQL container's index policy with the portal
- Knowledge check
- Summary

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

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

**Module 16: Consume an Azure Cosmos DB for NoSQL change feed using the SDK**

Process change feed events using the change feed processor in the Azure Cosmos DB for NoSQL .NET SDK.

Lessons

- Introduction
- Understand change feed features in the SDK
- Implement a delegate for the change feed processor
- Implement the change feed processor
- Implement the change feed estimator
- Exercise: Process change feed events using the Azure Cosmos DB for NoSQL SDK
- Knowledge check
- Summary

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

- Create a change feed processor in the .NET SDK
- Author a delegate to handle a batch of changes in a client-side application

**Module 17: Handle events with Azure Functions and Azure Cosmos DB for NoSQL change feed**

Use Azure Functions bindings to integrate a function with Azure Cosmos DB for NoSQL.

Lessons

- Introduction
- Understand Azure Function bindings for Azure Cosmos DB for NoSQL
- Configure function bindings
- Develop function
- Exercise: Process Azure Cosmos DB for NoSQL data using Azure Functions
- Knowledge check
- Summary

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

- Create an Azure Function using the Azure Cosmos DB trigger

**Module 18: Search Azure Cosmos DB for NoSQL data with Azure Cognitive Search**

Index Azure Cosmos DB for NoSQL data with Azure Cognitive Search.

Lessons

- Introduction
- Create an indexer for data in Azure Cosmos DB for NoSQL
- Implement a change detection policy
- Manage a data deletion detection policy
- Exercise - Search data using Azure Cognitive Search and Azure Cosmos DB for NoSQL
- Knowledge check
- Summary

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

- Create an indexer to migrate data from Azure Cosmos DB for NoSQL to an Azure Cognitive Search index

**Module 19: 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.

Lessons

- Introduction
- What's the difference between NoSQL and relational databases?
- Identify access patterns for your app
- When to embed or reference data
- Exercise: Measure performance for customer entities
- Choose a partition key
- Model small lookup entities
- Knowledge check
- Summary

In this module, you will:

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

**Module 20: 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.

Lessons

- Introduction
- Denormalize data in your model
- Manage referential integrity by using change feed
- Combine multiple entities in the same container
- Denormalize aggregates in the same container
- Finalize the data model
- Exercise advanced modeling patterns
- Knowledge check
- Summary

After completing this module, we'll 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

**Module 21: Configure replication and manage failovers in Azure Cosmos DB**

Replicate data across the globe and manage automatic or manual failovers in Azure Cosmos DB.

Lessons

- Introduction
- Understand replication
- Distribute data across regions
- Evaluate the cost of distributing data globally
- Define automatic failover policies
- Perform manual failovers
- Configure SDK region
- Exercise: Connect different regions with the Azure Cosmos DB for NoSQL SDK
- Knowledge check
- Summary

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

- Distribute data across global data centers
- Configure automatic failover and perform a manual failover
- Configure the Azure Cosmos DB for NoSQL .NET SDK to use a specific region

**Module 22: Use consistency models in Azure Cosmos DB for NoSQL**

Configure consistency on a sliding scale for Azure Cosmos DB for NoSQL.

Lessons

- Introduction
- Understand consistency models
- Configure default consistency model in the portal
- Change consistency model with the SDK
- Use session tokens
- Exercise: Configure consistency models in the portal and the Azure Cosmos DB for NoSQL SDK
- Knowledge check
- Summary

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

- Configure default consistency level for an Azure Cosmos DB account
- Change the consistency level on a per-request basis

**Module 23: Configure multi-region write in Azure Cosmos DB for NoSQL**

Write data to multiple regions using Azure Cosmos DB for NoSQL.

Lessons

- Introduction
- Understand multi-region write
- Configure multi-region support in the SDK
- Understand conflict resolution policies
- Create custom conflict resolution policy
- Exercise: Connect multi-region write account with the Azure Cosmos DB for NoSQL SDK
- Knowledge check
- Summary

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

- Configure Azure Cosmos DB for multi-region write
- Use the .NET SDK to select a write region for operations.

**Module 24: Customize an indexing policy in Azure Cosmos DB for NoSQL**

Tune the indexing policy based on your SQL queries in Azure Cosmos DB for NoSQL.

Lessons

- Introduction
- Index usage
- Review read-heavy index patterns
- Review write-heavy index patterns
- Exercise - Optimize an Azure Cosmos DB for NoSQL container's index policy for common operations
- Module assessment
- Summary

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

- Customize an indexing policy for read-heavy workloads
- Customize an indexing policy for write-heavy workloads

**Module 25: Measure index performance in Azure Cosmos DB for NoSQL**

Measure the performance of an indexing policy in Azure Cosmos DB for NoSQL.

Lessons

- Introduction

- Enable indexing metrics
- Analyze indexing metrics results
- Measure query cost
- Measure point operation cost
- Exercise - Optimize an Azure Cosmos DB for NoSQL container's index policy for a specific query
- Knowledge check
- Summary

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

- Tune an indexing policy for specific queries
- Measure the cost for a query or operation

**Module 26: Implement integrated cache in Azure Cosmos DB for NoSQL**

Implement, configure, and monitor integrated cache in Azure Cosmos DB for NoSQL.

Lessons

- Introduction
- Review workloads that benefit from the cache
- Enable integrated cache
- Configure cache staleness
- Knowledge check
- Summary

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

- Implement the integrated cache
- Configure integrated cache options

**Module 27: Measure performance in Azure Cosmos DB for NoSQL**

Measure performance in Azure Cosmos DB for NoSQL

Lessons

- Introduction
- Understand Azure Monitor
- Measure throughput
- Observe rate-limiting events
- Query logs
- Exercise: Use Azure Monitor to analyze an Azure Cosmos DB for NoSQL account
- Knowledge check
- Summary

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

- Understand how Azure Cosmos DB uses Azure Monitor to monitor server-side metrics.
- Measure Cosmos DB's throughput
- Observe rate-limiting events.
- Query telemetry logs
- Measure cross-partition storage distribution throughput

**Module 28: Monitor responses and events in Azure Cosmos DB for NoSQL**

We will learn to use a rich set of REST response codes returned by Azure Cosmos DB request to help you analyze potential issues.

Lessons & Learning Objectives

- Introduction
- Review common response codes
- Understand transient errors
- Review rate-limiting errors
- Configure Alerts
- Audit security
- Exercise: Troubleshoot an application using the Azure Cosmos DB for NoSQL SDK
- Knowledge check
- Summary

**Module 29: Implement backup and restore for Azure Cosmos DB for NoSQL**

We'll learn to use the two backup models Azure Cosmos DB provides.

Lessons

- Introduction
- Evaluate periodic backup
- Configure continuous backup and recovery
- Perform a point-in-time recovery
- Exercise - Recover a database or container from a recovery point
- Module assessment
- Summary

Learning Objectives:

- Understand the different backup and restore options Azure Cosmos DB provides.
- Evaluate periodic backups.
- Configure continuous backups.
- Do point in time recovery.

**Module 30: Implement security in Azure Cosmos DB for NoSQL**

We will learn the different security models that Azure Cosmos DB uses.

Lessons & Objectives

- Introduction
- Implement network-level access control
- Review data encryption options
- Use role-based access control (RBAC)
- Access account resources using Microsoft Entra ID
- Understand Always Encrypted
- Exercise: Store Azure Cosmos DB for NoSQL account keys in Azure Key Vault
- Knowledge check
- Summary

**Module 31: Write management scripts for Azure Cosmos DB for NoSQL**

Learn how to manage Azure Cosmos DB for NoSQL accounts, databases, and containers using the Azure CLI.

Lessons

- Introduction
- Create resources
- Manage index policies
- Configure database or container-provisioned throughput
- Migrate between standard and autoscale throughput
- Change region failover priority
- Initiate failovers
- Exercise: Adjust provisioned throughput using an Azure CLI script
- Module assessment
- Summary

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

- View arguments, groups, and subgroups for a specific CLI command
- Create Azure Cosmos DB accounts, databases, and containers using the CLI
- Manage an indexing policy using the CLI
- Configure database or container throughput using the CLI
- Initiate failovers and manage failover regions using the CLI

**Module 32: Create resource template for Azure Cosmos DB for NoSQL**

Learn about automated Azure Cosmos DB for NoSQL resource deployments using the Azure Resource Manager with JSON and Bicep templates.

Lessons

- Introduction
- Understand Azure Resource Manager resources
- Author Azure Resource Manager templates
- Configure database or container resources
- Configure throughput with an Azure Resource Manager template
- Manage index policies through Azure Resource Manager templates
- Exercise: Create an Azure Cosmos DB for NoSQL container using Azure Resource Manager templates
- Knowledge check
- Summary

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

- Identify the three most common resource types for Azure Cosmos DB for NoSQL accounts
- Create and deploy a JSON Azure Resource Manager template for an Azure Cosmos DB for NoSQL account, database, or container
- Create and deploy a Bicep template for an Azure Cosmos

DB for NoSQL account, database, or container
- Manage throughput and index policies using JSON or Bicep templates

**Module 33: Build multi-item transactions with the Azure Cosmos DB for NoSQL**
Author stored procedures using JavaScript in Azure Cosmos DB for NoSQL.
Lessons
- Introduction
- Understand transactions in the context of JavaScript SDK
- Author Stored procedures
- Rollback transactions
- Create stored procedures with the JavaScript SDK
- Exercise: Create a stored procedure with the Azure portal
- Knowledge check
- Summary

After completing this module, you'll be able to:
- Author stored procedure
- Rollback stored procedure transactions

**Module 34: Expand query and transaction functionality in Azure Cosmos DB for NoSQL**
Author user-defined functions and triggers using JavaScript in Azure Cosmos DB for NoSQL.
Lessons
- Introduction
- Create user-defined functions
- Create user-defined functions with the SDK
- Add triggers to an operation
- Create and use triggers with the SDK
- Exercise: Implement and use user defined functions with the Azure Cosmos DB SDK
- Knowledge check
- Summary

After completing this module, you'll be able to:
- Create user-defined functions
- Create triggers

## ASSOCIATED CERTIFICATIONS & EXAM

This course will prepare delegates to write the Microsoft DP-420: Designing and Implementing Cloud-Native Applications Using Microsoft Azure Cosmos DB exam.