

# DP-420 Dump Collection | DP-420 Valid Test Simulator



What's more, part of that ExamTorrent DP-420 dumps now are free: <https://drive.google.com/open?id=1Ks6GUxLajM8ogkfOweZjSRb09XbA1o2>

A lot of people are dreaming to pass the DP-420 exam. Also, you can start your own business after you totally master the skills of the DP-420 preparation exam expertly. Quickly, you will become the millionaire. Then it is time for others to envy your luxury life. All in all, our DP-420 Exam Prep is worth for investing. After all, you are the main beneficiary. Please follow your hearts and begin your new challenges bravely.

Microsoft DP-420 Exam is designed to test the candidate's knowledge and skills in designing and implementing cloud-native applications using Microsoft Azure Cosmos DB. DP-420 exam is intended for developers and architects who have an understanding of cloud computing and have experience in designing and developing cloud-native applications.

>> DP-420 Dump Collection <<

## DP-420 Valid Test Simulator, Reliable DP-420 Study Plan

Although our DP-420 exam braindumps have been recognised as a famous and popular brand in this field, but we still can be better by our efforts. In the future, our DP-420 study materials will become the top selling products. Although we come across some technical questions of our DP-420 learning guide during development process, we still never give up to developing our DP-420 practice engine to be the best in every detail.

## Microsoft Designing and Implementing Cloud-Native Applications Using Microsoft Azure Cosmos DB Sample Questions (Q96-Q101):

### NEW QUESTION # 96

You have an Azure subscription that contains a resource group named RG1. RG1 contains an Azure Cosmos DB for NoSQL account named account1.

You plan to audit changes to the control plane of account1.

You need to ensure that audit events include the details of the security principal that performs each control plane change.

How should you complete the az command? To answer select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

#### Answer Area

```
az cosmosdb  --name Account1 --resource-group RG1
```

|         |
|---------|
| sql     |
| service |
| sql     |
| update  |

|   |
|---|
| <input type="checkbox"/> --analytical-storage-schema-type FullFidelity            |
| <input checked="" type="checkbox"/> --analytical-storage-schema-type FullFidelity |
| <input type="checkbox"/> --disable-key-based-metadata-write-access true           |
| <input type="checkbox"/> --network-acl-bypass None                                |
| <input type="checkbox"/> --server-version 4.2                                     |

Answer:

Explanation:



Explanation:



**NEW QUESTION # 97**

You are designing a data model for an Azure Cosmos DB for NoSQL account.

What are the partition limits for request units per second (RU/s) and storage? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.



**Answer:**

Explanation:



**NEW QUESTION # 98**

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have a container named container1 in an Azure Cosmos DB Core (SQL) API account.

You need to make the contents of container1 available as reference data for an Azure Stream Analytics job.

Solution: You create an Azure function that uses Azure Cosmos DB Core (SQL) API change feed as a trigger and Azure event hub

as the output.

Does this meet the goal?

- A. Yes
- B. No

**Answer: A**

Explanation:

The Azure Cosmos DB change feed is a mechanism to get a continuous and incremental feed of records from an Azure Cosmos container as those records are being created or modified. Change feed support works by listening to container for any changes. It then outputs the sorted list of documents that were changed in the order in which they were modified.

The following diagram represents the data flow and components involved in the solution:



Reference: <https://docs.microsoft.com/en-us/azure/cosmos-db/sql/changefeed-ecommerce-solution>

### NEW QUESTION # 99

You have a database named db1 in an Azure Cosmos DB. You have a third-party application that is exposed through an OData endpoint. You need to migrate data from the application to a container in Azure Cosmos DB. What should you use?

- A. Database Migration Assistant
- B. Azure Data Factory
- C. Azure Migrate

**Answer: B**

Explanation:

You can migrate data from various data sources to Azure Cosmos DB using different tools and methods. The choice of the migration tool depends on factors such as the data source, the Azure Cosmos DB API, the size of data, and the expected migration duration. Some of the common migration tools are:

**Azure Cosmos DB Data Migration tool:** This is an open source tool that can import data to Azure Cosmos DB from sources such as JSON files, MongoDB, SQL Server, CSV files, and Azure Cosmos DB collections. This tool supports the SQL API and the Table API of Azure Cosmos DB.

**Azure Data Factory:** This is a cloud-based data integration service that can copy data from various sources to Azure Cosmos DB using connectors. This tool supports the SQL API, MongoDB API, Cassandra API, Gremlin API, and Table API of Azure Cosmos DB.

**Azure Cosmos DB live data migrator:** This is a command-line tool that can migrate data from one Azure Cosmos DB container to another container within the same or different account. This tool supports live migration with minimal downtime and works with any Azure Cosmos DB API.

For your scenario, if you want to migrate data from a third-party application that is exposed through an OData endpoint to a container in Azure Cosmos DB for NoSQL, you should use Azure Data Factory. Azure Data Factory has an OData connector that can read data from an OData source and write it to an Azure Cosmos DB sink using the SQL API. You can create a copy activity in Azure Data Factory that specifies the OData source and the Azure Cosmos DB sink, and run it on demand or on a schedule.

### NEW QUESTION # 100

You have a database in an Azure Cosmos DB Core (SQL) API account.

You plan to create a container that will store employee data for 5,000 small businesses. Each business will have up to 25 employees. Each employee item will have an emailAddress value.

You need to ensure that the emailAddress value for each employee within the same company is unique.

To what should you set the partition key and the unique key? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Partition key

|                        |   |
|------------------------|---|
| Microsoft              | ▼ |
| companyId              |   |
| companyId+emailAddress |   |
| emailAddress           |   |
| employeeId             |   |

Unique key

|              |   |
|--------------|---|
|              | ▼ |
| companyId    |   |
| emailAddress |   |
| employeeId   |   |

Answer:

Explanation:

Partition key

|                        |   |
|------------------------|---|
| Microsoft              | ▼ |
| companyId              |   |
| companyId+emailAddress |   |
| emailAddress           |   |
| employeeId             |   |

Unique key

|              |   |
|--------------|---|
|              | ▼ |
| companyId    |   |
| emailAddress |   |
| employeeId   |   |

Explanation

Partition key

|                        |   |
|------------------------|---|
|                        | ▼ |
| companyId              |   |
| companyId+emailAddress |   |
| emailAddress           |   |
| employeeId             |   |

Unique key

|              |   |
|--------------|---|
|              | ▼ |
| companyId    |   |
| emailAddress |   |
| employeeId   |   |

Box 1: CompanyID

After you create a container with a unique key policy, the creation of a new or an update of an existing item resulting in a duplicate within a logical partition is prevented, as specified by the unique key constraint. The partition key combined with the unique key guarantees the uniqueness of an item within the scope of the container.

