



Free Questions for **DP-420** by **braindumpscollection**

Shared by **Rodriguez** on **29-01-2024**

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

You have a database in an Azure Cosmos DB for NoSQL account that is configured for multi-region writes.

You need to use the Azure Cosmos DB SDK to implement the conflict resolution policy for a container. The solution must ensure that any conflict sent to the conflict feed.

Solution: You set ConflictResolutionMode to Custom. You Set ResolutionProcedures to a custom stored procedure. You configure the custom stored procedure to use the conflictingItems parameter to resolve conflict.

Does this meet the goal?

Options:

A- Yes

B- No

Answer:

B

Explanation:

Setting ConflictResolutionMode to Custom and configuring a custom stored procedure with the 'conflictingItems' parameter will allow you to implement a custom conflict resolution policy. This will ensure that any conflicts are sent to the conflict feed for resolution.

Question 2

Question Type: MultipleChoice

You have a database in an Azure Cosmos DB for NoSQL account that is configured for multi-region writes.

You need to use the Azure Cosmos DB SDK to implement the conflict resolution policy for a container. The solution must ensure that any conflict sent to the conflict feed.

Solution: You set ConflictResolutionMode to Custom. You Set ResolutionProcedures to a custom stored procedure. You configure the custom stored procedure to use the isTomstone parameter to resolve conflict.

Does this meet the goal?

Options:

A- Yes

B- No

Answer:

B

Explanation:

The solution is incorrect because there is no 'isTom' parameter in the Azure Cosmos DB SDK. The correct parameter is 'isTombstone'.

Question 3

Question Type: Hotspot

You have an Azure Cosmos DB account named account1 that has a default consistency level of session.

You have an app named App1.

You need to ensure that the read operations of App1 can request either bounded staleness or consistent prefix consistency.

What should you modify for each consistency level? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

The screenshot shows an exam question interface. At the top left, it says "Answer Area". Below that, the question text is partially visible: "Bounded staleness:" and "Consistent prefix:". To the right of the question text is a list of three options: "The account level options", "The database level options", and "The request level options". A mouse cursor is pointing at the top of the options list. Below the question text, there is a red horizontal line. To the left of this line, the text "Question 4" is written in red. Below "Question 4", the text "Question Type: Hotspot" is written in red. Below "Question Type: Hotspot", there is a red horizontal line. Below this line, the question text continues: "You have a multi-region Azure Cosmos DB account named account1 that has a default consistency level of strong."

You have an app named App1 that is configured to request a consistency level of session.

How will the read and write operations of App1 be handled? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Answer Area

Write operations will:

Be applied only to the replica to which App1 is connected
Write and replicate data to every region a
Write and replicate data to every region s

Answer:

Question 5

Read operations will be performed by the replica:

To which App1 is connected
That has the lowest estimated latency to t
That responds first

Question Type: MultipleChoice

You have an Azure Cosmos DB for NoSQL account named account1 that has a single read-write region and one additional read region. Account1 uses the strong default consistency level.

You have an application that uses the eventual consistency level when submitting requests to account1.

How will writes from the application be handled?

Options:

- A- Writes will use the strong consistency level.
- B- Azure Cosmos DB will reject writes from the application.
- C- The write order is not guaranteed during replication.

D- Writes will use the eventual consistency level.

Answer:

A

Explanation:

This is because the write concern is mapped to the default consistency level configured on your Azure Cosmos DB account, which is strong in this case. Strong consistency ensures that every write operation is synchronously committed to every region associated with your Azure Cosmos DB account. The eventual consistency level that the application uses only applies to the read operations. Eventual consistency offers higher availability and better performance, but it does not guarantee the order or latency of the reads.

Question 6

Question Type: Hotspot

You are developing an application that will connect to an Azure Cosmos DB for NoSQL account. The account has a single readme region and one agonal read region. The regions are configured for automatic failover.

The account has the following connect strings. (Line numbers are included for reference only.)

```
01 {
02   "connectionStrings": [
03     {
04       "connectionString":
05       "AccountEndpoint=https://contosodbaccount.documents.azure.com:443/;
06       AccountKey=MwUgRnGti4vErT2rfPPFdTFFyI9KyI9Kbe1RPGv70QdHo6VZ2i45TcJzrd4J80zYxrEATzyZ
07       "description": "Primary SQL Connection String"
08     },
09     {
10       "connectionString"
11       "AccountEndpoint=https://contosodbaccount.documents.azure.com:443/;
12       AccountKey=gfThRnGti4vErT2rfPPFdTFFyI43529Kbe1RPGv70QdHo6VZ2i45TcJzrd4J80zYxrfatzyZh
13       "description": "Secondary SQL Connection String"
14     },
15     {
16       "connectionString":
17       "AccountEndpoint=https://contosodbaccount.documents.azure.com:443/;
18       AccountKey=WGykBc1PHJoos6MdErT2rfPPFx9yI9Kbe1RPGv70QlIQwQNxq6QdOXjxgyLLebXBp8uJu7Fy
19       "description": "Primary Read-Only SQL Connection String"
20     },
21     {
22       "connectionString":
23       "AccountEndpoint=https://contosodbaccount.documents.azure.com:443/;
24       AccountKey=k2DZI0oY4Jc7QeUJqVGH3csda6EyI9Kbe1RPGv70QErT2rfPPFtbwTPfKAg19zVxC0MDNn8x
25       "description": "Secondary Read-Only SQL Connection String"
```


For each of the following statements, select Yes if the statement is true. otherwise, select No.

NOTE: Each correct selection is worth one point.

Answer Area

Statements

Answer:

If the primary write region fails, applications that write to the database must use a different connection string to continue to use the service.

The Primary Read-Only SQL Connection String and the Secondary Read-Only SQL Connection String will connect to different regions from an application running in the East US Azure region.

Question Type: MultipleChoice

Applications can choose from which region they read by setting the PreferredLocations property within their connection properties.

You need to create a data store for a directory of small and medium-sized businesses (SMBs). The data store must meet the following requirements:

- * Store companies and the users employed by them. Each company will have less than 1,000 users.
- * Some users have data that is greater than 2 KB.
- * Associate each user to only one company.
- * Provide the ability to browse by company.
- * Provide the ability to browse the users by company.

* Whenever a company or user profile is selected, show a details page for the company and all the related users.

* Be optimized for reading data.

Which design should you implement to optimize the data store for reading data?

Options:

A- In a directory container, create a document for each company and a document for each user. Use company ID as the partition key.

B- In a company container, create a document for each company. Embed the users into company documents. Use the company ID as the partition key.

C- Create a user container that uses the user ID as the partition key and a company container that container that uses the company ID as the partition key. Add the company ID to each user documents.

D- In a user container, create a document for each user. Embed the company into each user document. Use the user ID as the partition key.

Answer:

B

Explanation:

Azure Cosmos DB is a multi-model database that supports various data models, such as documents, key-value, graph, and column-family³. The core content-model of Cosmos DB's database engine is based on atom-record-sequence (ARS), which allows it to store and query different types of data in a flexible and efficient way³.

To optimize the data store for reading data, you should consider the following factors:

The size and shape of your data

The frequency and complexity of your queries

The latency and throughput requirements of your application

The trade-offs between storage efficiency and query performance

Based on these factors, one possible design that you could implement is B. In a company container, create a document for each company. Embed the users into company documents. Use the company ID as the partition key.

This design has the following advantages:

It stores companies and users as self-contained documents that can be easily retrieved by company ID¹.

It avoids storing redundant data or creating additional containers for users¹.

It allows you to browse by company and browse the users by company with simple queries¹.

It shows a details page for the company and all the related users by fetching a single document¹.

It leverages the benefits of embedding data, such as reducing the number of requests, improving query performance, and simplifying data consistency².

This design also has some limitations, such as:

It may not be suitable for some users who have data that is greater than 2 KB, as it could exceed the maximum document size limit of 2 MB.

It may not be optimal for scenarios where users need to be associated with more than one company or queried independently from companies.

It may not be scalable for companies that have more than 1,000 users, as it could result in hot partitions or throttling.

Depending on your specific use case and requirements, you may need to adjust this design or choose a different one. For example, you could use a hybrid data model that combines embedding and referencing data, or you could use a graph data model that expresses entities and relationships as vertices and edges.

To Get Premium Files for DP-420 Visit

<https://www.p2pexams.com/products/dp-420>

For More Free Questions Visit

<https://www.p2pexams.com/microsoft/pdf/dp-420>

