

DP-420^{Q&As}

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

Pass Microsoft DP-420 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass2lead.com/dp-420.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Microsoft Official Exam Center

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers



QUESTION 1

You need to provide a solution for the Azure Functions notifications following updates to con-product. The solution must meet the business requirements and the product catalog requirements.

Which two actions should you perform? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Configure the trigger for each function to use a different leaseCollectionPrefix
- B. Configure the trigger for each function to use the same leaseCollectionNaire
- C. Configure the trigger for each function to use a different leaseCollectionName
- D. Configure the trigger for each function to use the same leaseCollectionPrefix

Correct Answer: AC

leaseCollectionPrefix: when set, the value is added as a prefix to the leases created in the Lease collection for this Function. Using a prefix allows two separate Azure Functions to share the same Lease collection by using different prefixes.

Scenario: Use Azure Functions to send notifications about product updates to different recipients.

Trigger the execution of two Azure functions following every update to any document in the con-product container.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-cosmosdb-v2-trigger>

QUESTION 2

DRAG DROP

You have an Azure Cosmos DB Core (SQL) API account that is configured for multi-region writes. The account contains a database that has two containers named container1 and container2.

The following is a sample of a document in container1:

```
{  
  "customerId": 1234,  
  "firstName": "John",  
  "lastName": "Smith",  
  "policyYear": 2021  
}
```

The following is a sample of a document in container2:

```
{  
  "gpsId": 1234,  
  "latitude": 38.8951,  
  "longitude": -77.0364  
}
```

You need to configure conflict resolution to meet the following requirements:

1.
For container1 you must resolve conflicts by using the highest value for policyYear.
2.
For container2 you must resolve conflicts by accepting the distance closest to latitude: 40.730610and longitude: -73.935242.
3.
Administrative effort must be minimized to implement the solution.

What should you configure for each container? To answer, drag the appropriate configurations to the correct containers. Each configuration may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Select and Place:

Configurations

- Last Write Wins (default) mode
- Merge Procedures (custom) mode
- An application that reads from the conflicts feed

Answer Area

Container1:

Container2:



Correct Answer:

Configurations

An application that reads from the conflicts feed

Answer Area

Container1:

Last Write Wins (default) mode

Container2:

Merge Procedures (custom) mode

Box 1: Last Write Wins (LWW) (default) mode

Last Write Wins (LWW): This resolution policy, by default, uses a system-defined timestamp property. It's based on the time-synchronization clock protocol.

Box 2: Merge Procedures (custom) mode

Custom: This resolution policy is designed for application-defined semantics for reconciliation of conflicts. When you set this policy on your Azure Cosmos container, you also need to register a merge stored procedure. This procedure is

automatically invoked when conflicts are detected under a database transaction on the server. The system provides exactly once guarantee for the execution of a merge procedure as part of the commitment protocol.

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/conflict-resolution-policies>

<https://docs.microsoft.com/en-us/azure/cosmos-db/sql/how-to-manage-conflicts>

QUESTION 3

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:

1.

You set ConflictResolutionMode to Custom.

2.

You Set ResolutionProcedures to a custom stored procedure.

3.

You configure the custom stored procedure to use the isTomstone parameter to resolve conflict.

Does this meet the goal?

A. Yes

B. No

Correct Answer: A

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

QUESTION 4

You maintain a relational database for a book publisher. The database contains the following tables.

Name	Column
Author	authorId (primary key)
	fullname
	address
	contactinfo
Book	bookId (primary key)
	isbn
	title
	genre
BookauthorInk	authorId (foreign key)
	bookId (foreign key)

The most common query lists the books for a given authorId.

You need to develop a non-relational data model for Azure Cosmos DB Core (SQL) API that will replace the relational database. The solution must minimize latency and read operation costs.

What should you include in the solution?

A. Create a container for Author and a container for Book. In each Author document, embed bookId for each book by

the author. In each Book document embed authorId of each author.

B. Create Author, Book, and Bookauthorlnk documents in the same container.

C. Create a container that contains a document for each Author and a document for each Book. In each Book document, embed authorId.

D. Create a container for Author and a container for Book. In each Author document and Book document embed the data from Bookauthorlnk.

Correct Answer: A

Store multiple entity types in the same container.

QUESTION 5

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

Correct Answer: B

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.

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

[Latest DP-420 Dumps](#)

[DP-420 PDF Dumps](#)

[DP-420 Study Guide](#)