DP-420^{Q&As}

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.passapply.com/dp-420.html

100% Passing Guarantee
100% Money Back Assurance

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

QUESTION 1

You are implementing an Azure Data Factory data flow that will use an Azure Cosmos DB (SQL API) sink to write a dataset. The data flow will use 2,000 Apache Spark partitions. You need to ensure that the ingestion from each Spark partition is balanced to optimize throughput.

Which sink setting should you configure?

A. Throughput

B. Write throughput budget

C. Batch size

D. Collection action

Correct Answer: C

Batch size: An integer that represents how many objects are being written to Cosmos DB collection in each batch. Usually, starting with the default batch size is sufficient. To further tune this value, note:

Cosmos DB limits single request\\'s size to 2MB. The formula is "Request Size = Single Document Size * Batch Size". If you hit error saying "Request size is too large", reduce the batch size value.

The larger the batch size, the better throughput the service can achieve, while make sure you allocate enough RUs to empower your workload.

Incorrect Answers:

A: Throughput: Set an optional value for the number of RUs you\\'d like to apply to your CosmosDB collection for each execution of this data flow. Minimum is 400.

B: Write throughput budget: An integer that represents the RUs you want to allocate for this Data Flow write operation, out of the total throughput allocated to the collection.

D: Collection action: Determines whether to recreate the destination collection prior to writing.

None: No action will be done to the collection. Recreate: The collection will get dropped and recreated

Reference: https://docs.microsoft.com/en-us/azure/data-factory/connector-azure-cosmos-db

---

QUESTION 2

HOTSPOT

You have a database named telemetry in an Azure Cosmos DB Core (SQL) API account that stores IoT data. The database contains two containers named readings and devices.

Documents in readings have the following structure.

1.

id

2.

deviceid

3.

timestamp

4.

ownerid

5.

measures (array)

-type

-value

-metricid

Documents in devices have the following structure.

id deviceid owner

- ownerid

-emailaddress

-name brand model

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

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

Statements	Yes	No
To return for all devices owned by a specific `emailaddress`, multiple queries must be performed	O	O
To return `deviceid`, `ownerid`, `timestamp`, and `value` for a specific `metricid`, a join must be performed	O	O
To return `deviceid`, `ownerid`, `emailaddress`, and `model`, a join must be performed	O	O

Correct Answer:

Answer Area

Statements	Yes	No
To return for all devices owned by a specific `emailaddress`, multiple queries must be performed	○	○
To return `deviceid`, `ownerid`, `timestamp`, and `value` for a specific `metricid`, a join must be performed	○	○
To return `deviceid`, `ownerid`, `emailaddress`, and `model`, a join must be performed	○	○

Box 1: Yes

Need to join readings and devices.

Box 2: No

Only readings is required. All required fields are in readings.

Box 3: No

Only devices is required. All required fields are in devices.

QUESTION 3

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 authorIdof each author.

B. Create Author, Book, and BookauthorInk 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 BookauthorInk.

Correct Answer: A

Store multiple entity types in the same container.

QUESTION 4

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 ConfilictResolutionMode to Custom.

2.

 You Set ResolutionProcedures to a custom stored procedure.

3.

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

Does this meet the goal?

A. Yes

B. No

Correct Answer: A

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 5

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 ConfilictResolutionMode to Custom and you use the default settings for the policy.

Does this meet the goal?

A. Yes

B. No

Correct Answer: B

Setting ConflictResolutionMode to Custom and using the default settings for the policy will not ensure that conflicts are sent to the conflict feed. You need to define a custom stored procedure using the "conflictingItems" parameter to handle conflicts properly.

[DP-420 VCE Dumps](#)   [DP-420 Practice Test](#)   [DP-420 Study Guide](#)