

DP-420 Dumps

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

<https://www.certleader.com/DP-420-dumps.html>



NEW QUESTION 1

- (Exam Topic 2)

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 an Azure Cosmos DB Core (SQL) API account named account1 that uses autoscale throughput. You need to run an Azure function when the normalized request units per second for a container in account1 exceeds a specific value.

Solution: You configure the function to have an Azure CosmosDB trigger. Does this meet the goal?

A. Yes

B. No

Answer: B

Explanation:

Instead configure an Azure Monitor alert to trigger the function.

You can set up alerts from the Azure Cosmos DB pane or the Azure Monitor service in the Azure portal. Reference: <https://docs.microsoft.com/en-us/azure/cosmos-db/create-alerts>

NEW QUESTION 2

- (Exam Topic 2)

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:

For container1 you must resolve conflicts by using the highest value for policyYear.

For container2 you must resolve conflicts by accepting the distance closest to latitude: 40.730610 and longitude: -73.935242.

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.

| Configurations | Answer Area |
|---|-------------|
| Last Write Wins (default) mode | Container1: |
| Merge Procedures (custom) mode | Container2: |
| An application that reads from the conflicts feed | |

A. Mastered

B. Not Mastered

Answer: A

Explanation:

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>

NEW QUESTION 3

- (Exam Topic 2)

You need to configure an Apache Kafka instance to ingest data from an Azure Cosmos DB Core (SQL) API account. The data from a container named telemetry must be added to a Kafka topic named iot. The solution must store the data in a compact binary format.

Which three configuration items should you include in the solution? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

A. "connector.class": "com.azure.cosmos.kafka.connect.source.CosmosDBSourceConnector"

B. "key.converter": "org.apache.kafka.connect.json.JsonConverter"
C. "key.converter": "io.confluent.connect.avro.AvroConverter"
D. "connect.cosmos.containers.topicmap": "iot#telemetry"
E. "connect.cosmos.containers.topicmap": "iot"
F. "connector.class": "com.azure.cosmos.kafka.connect.source.CosmosDBSinkConnector"

Answer: CDF

Explanation:

C: Avro is binary format, while JSON is text.

F: Kafka Connect for Azure Cosmos DB is a connector to read from and write data to Azure Cosmos DB. The Azure Cosmos DB sink connector allows you to export data from Apache Kafka topics to an Azure Cosmos DB database. The connector polls data from Kafka to write to containers in the database based on the topics subscription.

D: Create the Azure Cosmos DB sink connector in Kafka Connect. The following JSON body defines config for the sink connector.

Extract:

"connector.class": "com.azure.cosmos.kafka.connect.sink.CosmosDBSinkConnector", "key.converter": "org.apache.kafka.connect.json.AvroConverter"

"connect.cosmos.containers.topicmap": "hotels#kafka"

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/sql/kafka-connector-sink> <https://www.confluent.io/blog/kafka-connect-deep-dive-converters-serialization-explained/>

NEW QUESTION 4

- (Exam Topic 2)

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

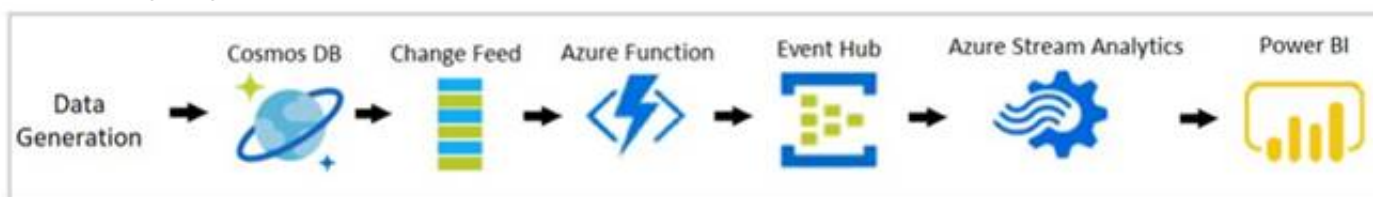
B. No

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:



C:\Users\Admin\Desktop\Data\Odt

data\Untitled.jpg

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

NEW QUESTION 5

- (Exam Topic 2)

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 an Azure Cosmos DB Core (SQL) API account named account 1 that uses autoscale throughput. You need to run an Azure function when the normalized request units per second for a container in account1 exceeds a specific value.

Solution: You configure an application to use the change feed processor to read the change feed and you configure the application to trigger the function.

Does this meet the goal?

A. Yes

B. No

Answer: B

Explanation:

Instead configure an Azure Monitor alert to trigger the function.

You can set up alerts from the Azure Cosmos DB pane or the Azure Monitor service in the Azure portal. Reference: <https://docs.microsoft.com/en-us/azure/cosmos-db/create-alerts>

NEW QUESTION 6

- (Exam Topic 2)

You have an app that stores data in an Azure Cosmos DB Core (SQL) API account The app performs queries that return large result sets.

You need to return a complete result set to the app by using pagination. Each page of results must return 80 items.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

| Actions | Answer Area |
|---|-------------|
| Configure MaxItemCount in QueryRequestOptions | |
| Run the query and provide a continuation token | |
| Configure MaxBufferedItemCount in QueryRequestOptions | |
| Append the results to a variable | |
| Run the query and increment MaxItemCount | |

- A. Mastered
B. Not Mastered

Answer: A

Explanation:

Step 1: Configure the MaxItemCount in QueryRequestOptions

You can specify the maximum number of items returned by a query by setting the MaxItemCount. The MaxItemCount is specified per request and tells the query engine to return that number of items or fewer.

Box 2: Run the query and provide a continuation token

In the .NET SDK and Java SDK you can optionally use continuation tokens as a bookmark for your query's progress. Azure Cosmos DB query executions are stateless at the server side and can be resumed at any time using the continuation token.

If the query returns a continuation token, then there are additional query results.

Step 3: Append the results to a variable

Reference: <https://docs.microsoft.com/en-us/azure/cosmos-db/sql/sql-query-pagination>

NEW QUESTION 7

- (Exam Topic 2)

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

Answer: C

Explanation:

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.

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

NEW QUESTION 8

- (Exam Topic 2)

HOTSPOT

You configure Azure Cognitive Search to index a container in an Azure Cosmos DB Core (SQL) API account as shown in the following exhibit.

| + Add field + Add subfield Delete | | | | | | | | |
|-----------------------------------|-----------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|-------------------------------------|-------------------|-----------|
| Field Name | Type | Retrievable | Filterable | Sortable | Facetable | Searchable | Analyzer | Suggester |
| id | Edm.String | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | ... |
| name | Edm.String | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Standard - Lucene | ... |
| ▼ headquarters | Edm.ComplexType | | | | | | | ... |
| country | Edm.String | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | ... |
| iso | Edm.String | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | ... |
| employees | Edm.Int32 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | | ... |
| ... | Edm.String | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | ... |

Use the drop-down menus to select the answer choice that completes each statement based on the information presented in the graphic.

NOTE: Each correct selection is worth one point.

Answer Area

The [answer choice] field is limited to exact match comparisons

| | |
|---------|---|
| | ▼ |
| country | |
| id | |
| name | |

The [answer choice] field is hidden form the search results

| | |
|---------|---|
| | ▼ |
| country | |
| id | |
| name | |

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1: country

The country field is filterable.

Note: filterable: Indicates whether to enable the field to be referenced in \$filter queries. Filterable differs from searchable in how strings are handled. Fields of type Edm.String or Collection(Edm.String) that are filterable do not undergo lexical analysis, so comparisons are for exact matches only.

Box 2: name

The name field is not Retrievable.

Retrievable: Indicates whether the field can be returned in a search result. Set this attribute to false if you want to use a field (for example, margin) as a filter, sorting, or scoring mechanism but do not want the field to be visible to the end user.

Note: searchable: Indicates whether the field is full-text searchable and can be referenced in search queries. Reference: <https://docs.microsoft.com/en-us/rest/api/searchservice/create-index>

NEW QUESTION 9

- (Exam Topic 2)

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.

id
deviceid
timestamp
ownerid
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.

Answer Area

| Statements | Yes | No |
|---|-----------------------|-----------------------|
| To return for all devices owned by a specific emailaddress, multiple queries must be performed | <input type="radio"/> | <input type="radio"/> |
| To return deviceid, ownerid, timestamp, and value for a specific metricid, a join must be performed | <input type="radio"/> | <input type="radio"/> |
| To return deviceid, ownerid, emailaddress, and model, a join must be performed | <input type="radio"/> | <input type="radio"/> |

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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.

NEW QUESTION 10

- (Exam Topic 2)

You are creating a database in an Azure Cosmos DB Core (SQL) API account. The database will be used by an application that will provide users with the ability to share online posts. Users will also be able to submit comments on other users' posts.

You need to store the data shown in the following table.

| Type | Description |
|-----------|--|
| Users | Information about a user who will use the application |
| Posts | Text of up to 1,000 characters that a user will share with other users |
| Comments | Text of up to 280 characters that users will submit as a comment on a post |
| Interests | Information about a user's interests |

The application has the following characteristics: Users can submit an unlimited number of posts.

The average number of posts submitted by a user will be more than 1,000. Posts can have an unlimited number of comments from different users.

The average number of comments per post will be 100, but many posts will exceed 1,000 comments. Users will be limited to having a maximum of 20 interests.

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 | Yes | No |
|---|-----------------------|-----------------------|
| If you embed the posts data into the users data instead of creating a separate document for each post, you will increase the write operation costs for new posts | <input type="radio"/> | <input type="radio"/> |
| If you embed the comments data into the posts data instead of creating a separate document for each comment you will increase the write operation costs for new comments | <input type="radio"/> | <input type="radio"/> |
| If you embed the interests data into the users data instead of creating a separate document for each interest, you will increase the read operation costs for displaying the users and their associated interests | <input type="radio"/> | <input type="radio"/> |

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1: Yes

Non-relational data increases write costs, but can decrease read costs.

Box 2: Yes

Non-relational data increases write costs, but can decrease read costs.

Box 3: No

Non-relational data increases write costs, but can decrease read costs.

NEW QUESTION 10

- (Exam Topic 2)

You have a container named container1 in an Azure Cosmos DB Core (SQL) API account. Upserts of items in container1 occur every three seconds.

You have an Azure Functions app named function1 that is supposed to run whenever items are inserted or replaced in container1.

You discover that function1 runs, but not on every upsert.

You need to ensure that function1 processes each upsert within one second of the upsert. Which property should you change in the Function.json file of function1?

- A. checkpointInterval
- B. leaseCollectionsThroughput
- C. maxItemsPerInvocation
- D. feedPollDelay

Answer: D

Explanation:

With an upsert operation we can either insert or update an existing record at the same time.

FeedPollDelay: The time (in milliseconds) for the delay between polling a partition for new changes on the feed, after all current changes are drained. Default is 5,000 milliseconds, or 5 seconds.

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

NEW QUESTION 13

.....

Thank You for Trying Our Product

* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

* One year free update

You can enjoy free update one year. 24x7 online support.

* Trusted by Millions

We currently serve more than 30,000,000 customers.

* Shop Securely

All transactions are protected by VeriSign!

100% Pass Your DP-420 Exam with Our Prep Materials Via below:

<https://www.certleader.com/DP-420-dumps.html>