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 plan to create an Azure Cosmos DB account that will use the NoSQL API.

You need to create a grouping strategy for items that will be stored in the account. The solution must ensure that write and read operations on the items can be performed within the same transact!

What should you use to group the items?

A. logical partitions

B. physical partitions

C. databases

D. containers

Correct Answer: A

QUESTION 2

HOTSPOT

You have an Azure Cosmos DB Core (SQL) account that has a single write region in West Europe.

You run the following Azure CLI script.

```
az cosmosdb update -n $accountName -g $resourceGroupName \
   --locations regionName='West Europe' failoverPriority=0 isZoneRedundant-False \
   --locations regionName='North Europe' failoverPriority=1 isZoneRedundant-False

az cosmosdb failover-priority-change -n $accountName -g $resourceGroupName \
   --failover-policies 'North Europe=0' 'West Europe=1'
```

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
After running the script, there will be an instance of Azure Cosmos DB in North Europe that is writable	○	○
After running the script, the Azure Cosmos DB instance in West Europe will be writable	○	○
The cost of the Azure Cosmos DB account is unaffected by running the script	○	○

Correct Answer:

Answer Area

Statements	Yes	No
After running the script, there will be an instance of Azure Cosmos DB in North Europe that is writable	○	○
After running the script, the Azure Cosmos DB instance in West Europe will be writable	○	○
The cost of the Azure Cosmos DB account is unaffected by running the script	○	○

Box 1: Yes

The Automatic failover option allows Azure Cosmos DB to failover to the region with the highest failover priority with no user action should a region become unavailable.

Box 2: No

West Europe is used for failover. Only North Europe is writable.

To Configure multi-region set UseMultipleWriteLocations to true.

Box 3: Yes

Provisioned throughput with single write region costs $0.008/hour per 100 RU/s and provisioned throughput with multiple writable regions costs $0.016/per hour per 100 RU/s.

Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/sql/how-to-multi-master https://docs.microsoft.com/en-us/azure/cosmos-db/optimize-cost-regions

QUESTION 3

HOTSPOT

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

The following is a sample of a document in container1.

{

"studentId": "631282",

"firstName": "James",

"lastName": "Smith",

"enrollmentYear": 1990,

"isActivelyEnrolled": true,

"address": {

"street": "",

"city": "",

"stateProvince": "",

"postal": "",

}

}

The container1 container has the following indexing policy.

{

"indexingMode": "consistent",

"includePaths": [

{

"path": "/*"

},

{

"path": "/address/city/?"

}

],

"excludePaths": [

{

"path": "/address/*"

},

{

"path": "/firstName/?"

}

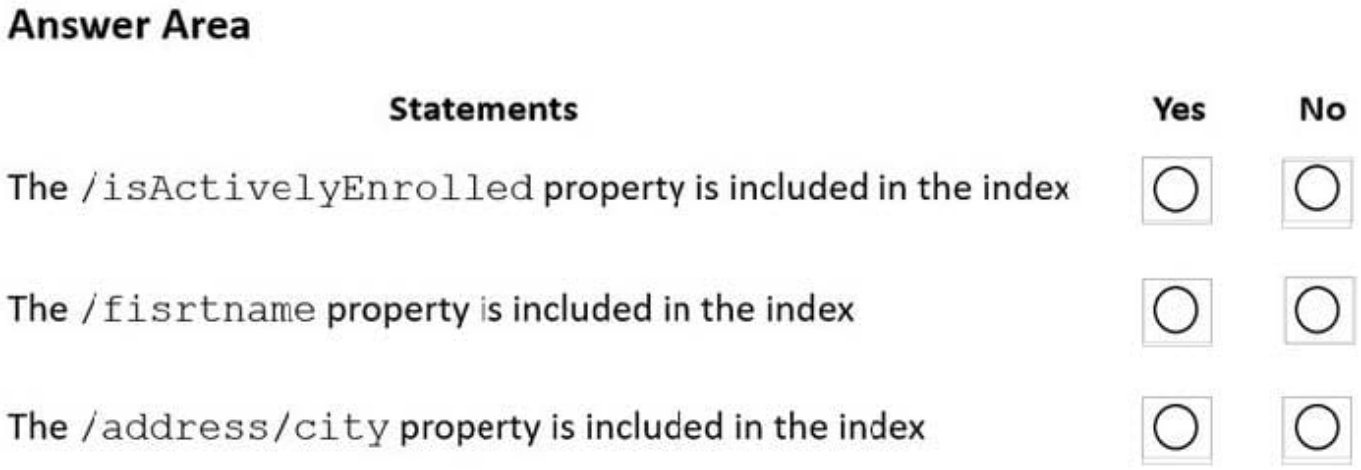
]

}

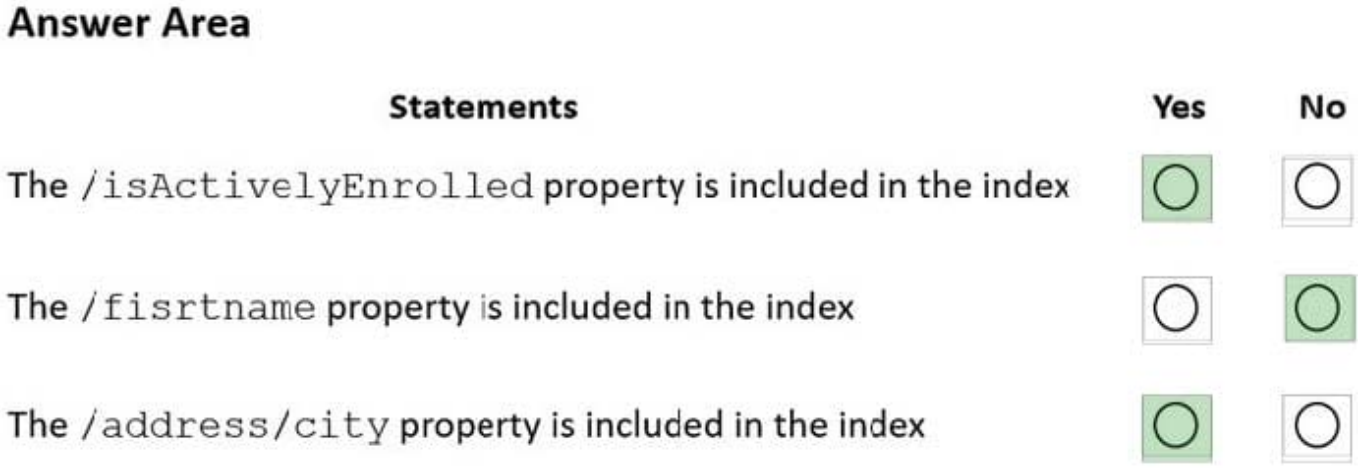
For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

Hot Area:



Correct Answer:



Box 1: Yes

"path": "/*" is in includePaths.

Include the root path to selectively exclude paths that don\\'t need to be indexed. This is the recommended approach as it lets Azure Cosmos DB proactively index any new property that may be added to your model.

Box 2: No

"path": "/firstName/?" is in excludePaths.

Box 3: Yes

"path": "/address/city/?" is in includePaths

Reference:

https://docs.microsoft.com/en-us/azure/cosmos-db/index-policy

QUESTION 4

You configure multi-region writes for account1.

You need to ensure that App1 supports the new configuration for account1. The solution must meet the business requirements and the product catalog requirements.

What should you do?

A. Set the default consistency level of account1 to bounded staleness.

B. Create a private endpoint connection.

C. Modify the connection policy of App1.

D. Increase the number of request units per second (RU/s) allocated to the con-product and con-productVendor containers.

Correct Answer: D

App1 queries the con-product and con-productVendor containers.

Note: Request unit is a performance currency abstracting the system resources such as CPU, IOPS, and memory that are required to perform the database operations supported by Azure Cosmos DB.

Scenario:

Develop an app named App1 that will run from all locations and query the data in account1.

Once multi-region writes are configured, maximize the performance of App1 queries against the data in account1.

Whenever there are multiple solutions for a requirement, select the solution that provides the best performance, as long as there are no additional costs associated.

Incorrect Answers:

A:

Bounded staleness relates to writes. App1 only do reads.

Note: Bounded staleness is frequently chosen by globally distributed applications that expect low write latencies but require total global order guarantee. Bounded staleness is great for applications featuring group collaboration and sharing, stock ticker, publish-subscribe/queueing etc.

Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/consistency-levels

QUESTION 5

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 result1 these questions will not appear in the review screen.

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 OB SDK to implement the conflict resolution policy for a container. The solution must ensure that any conflicts are sent to the conflicts feed.

Solution: You set ConflictResolutionMode to Laswriterwins and you use the default settings for the policy.

Does this meet the goal?

A. Yes

B. No

Correct Answer: A