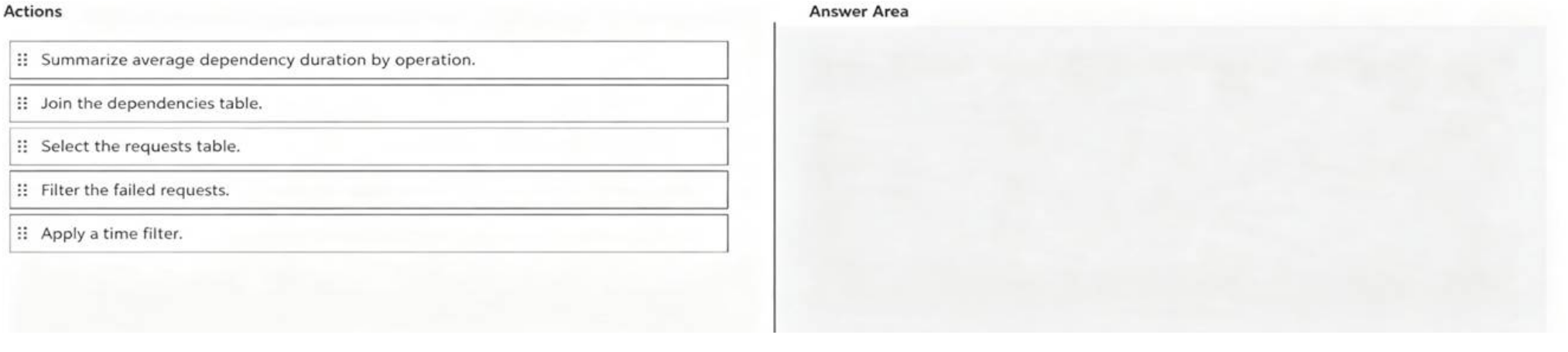
Microsoft

Exam Questions AI-200

Developing AI Cloud Solutions on Azure

NEW QUESTION 1
You are investigating high latency in an AI search application that processes millions of requests daily. Telemetry is stored in Azure Monitor Logs.
You must create a KQL query that correlates information from the AppRequests table and the AppDependencies table. The query must meet the following requirements:
Include only data from the last 24 hours.
Filter for failed requests only.
Calculate the average duration of dependencies, grouped by operation.
The query must follow best practice to optimize the performance by minimizing the initial data scan.
You need to create the query.
In which order should perform the actions? To answer, move all actions from the list of actions to the answer area and arrange them in the correct order.

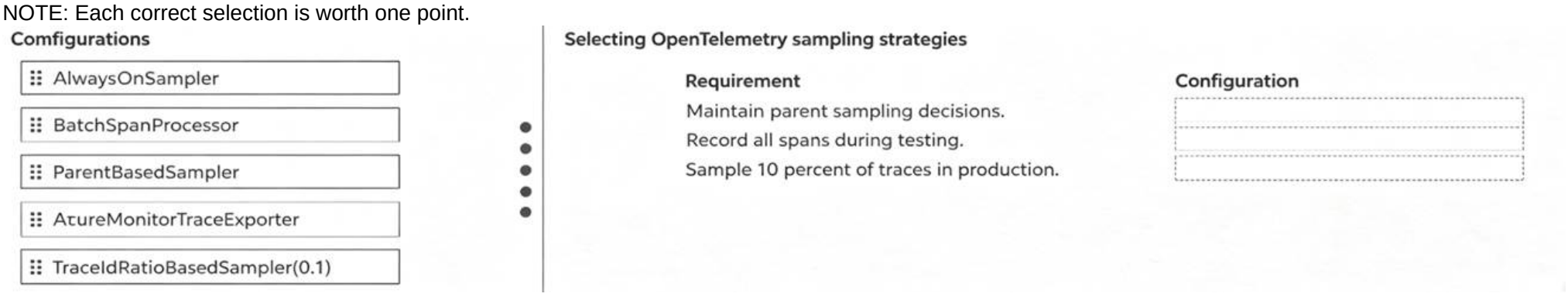


A. Mastered
B. Not Mastered

Answer: A

Explanation:
Verified Answer:1) Select the requests table; 2) apply the 24-hour time filter; 3) filter failed requests; 4) join the dependencies table; 5) summarize average dependency duration by operation.
Detailed Explanation:KQL performs best when high-selectivity filters are applied as early as possible, especially the time predicate that limits the amount of data scanned. Starting from AppRequests, restricting to the last 24 hours, and then filtering failures reduces the left-side dataset before the join. The dependency table is then correlated with those requests, and aggregation is performed last to calculate average dependency duration by operation. Joining or summarizing before the time and failure filters would process more data than necessary.
Study Guide Alignment:Security and operations: Key Vault, App Configuration, managed identity, OpenTelemetry, Azure Monitor, and KQL-based troubleshooting.
Official Microsoft Learn References:AI-200 Study Guide | Optimize log queries in Azure Monitor

NEW QUESTION 2
You are configuring sampling for a distributed application that sends traces to Azure Monitor. The solution must:
• Preserve upstream sampling decisions across distributed traces
• Capture all spans during local testing.
• Sample 10 percent of traces in production.
You need to apply the appropriate sampling configuration for each requirement.
What should you do? To answer, move the appropriate configurations to the correct requirements. You may use each configuration once, more than once or not at all. You may need to move the split bar Between panes or scroll to view content.
NOTE: Each correct selection is worth one point.



A. Mastered
B. Not Mastered

Answer: A

Explanation:
Verified Answer:Preserve parent sampling decisions: ParentBasedSampler. Capture all spans locally: AlwaysOnSampler. Sample 10% in production: TraceIdRatioBasedSampler(0.1).
Detailed Explanation:Parent-based sampling propagates the sampling decision from the upstream parent, preventing broken sampling decisions across a distributed trace. AlwaysOnSampler records every span, which is suitable for local test environments where complete trace visibility is more important than telemetry volume. TraceIdRatioBasedSampler with 0.1 selects approximately ten percent of traces based on trace identifiers, providing deterministic fixed-ratio sampling for production. A batch span processor and an exporter control processing/export, not the sampling decision itself.
Study Guide Alignment:Security and operations: Key Vault, App Configuration, managed identity, OpenTelemetry, Azure Monitor, and KQL-based troubleshooting.
Official Microsoft Learn References:AI-200 Study Guide | OpenTelemetry sampling in Azure Monitor | Enable Azure Monitor OpenTelemetry

NEW QUESTION 3
You need to deploy Azure function resources and apps by using an automated, version-controlled CI/CD pipeline that supports declarative infrastructure deployment. What should you use?

A. Local Git deployment
B. GitHub Actions
C. Azure Functions Core Tools
D. Azure CLI

Answer: B

Explanation:
Detailed Explanation:GitHub Actions is the correct orchestration mechanism among the options because the case requires an automated, version-controlled pipeline and Bicep-based declarative deployment. Microsoft documents GitHub Actions as a supported way to deploy Bicep and automate Azure resource deployment. Local Git, Azure Functions Core Tools, and Azure CLI can participate in development or scripted deployment, but they do not by themselves satisfy the stated requirement for a repository-driven, repeatable CI/CD pipeline.
Study Guide Alignment:Security and operations: Key Vault, App Configuration, managed identity, OpenTelemetry, Azure Monitor, and KQL-based troubleshooting.
Official Microsoft Learn References:AI-200 Study Guide | Deploy Bicep with GitHub Actions | Automate Function app resource deployment

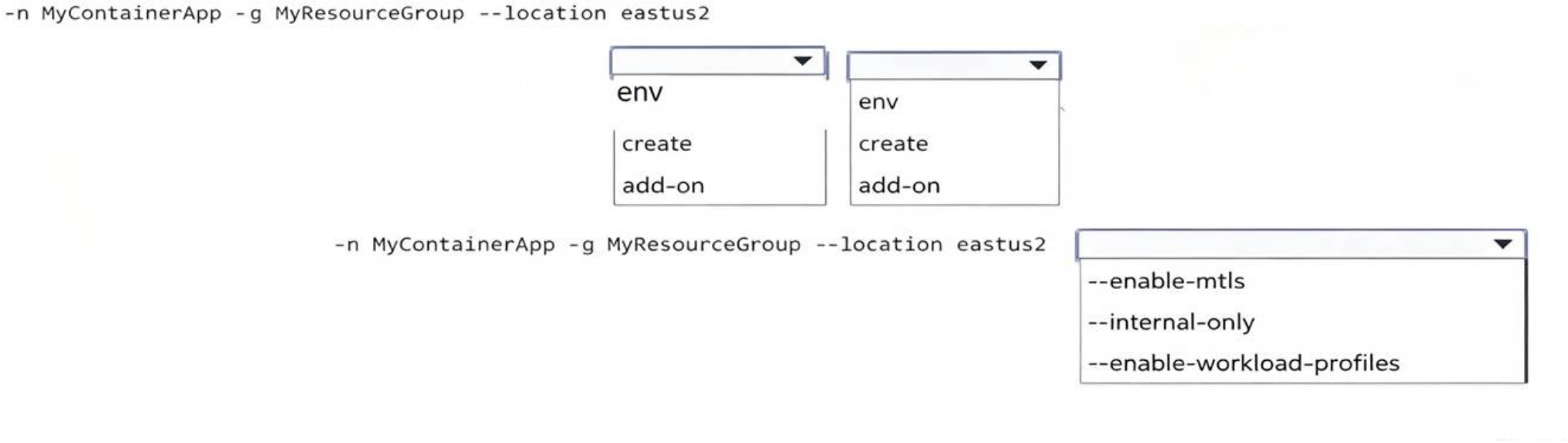
NEW QUESTION 4
You are developing a microservices-based application that uses Azure Container Apps. The application consists of several containerized services that handle tasks, such as processing orders, managing inventory, and generating reports. You deploy a new revision of the processing orders app.
Processing orders must be triggered by a web request and must always be available based on incoming web requests.
You need to validate that the replica is ready to handle incoming requests.
What should you implement?

A. HTTP liveness probe
B. HTTP startup probe
C. TCP readiness probe
D. HTTP readiness probe
E. TCP liveness probe

Answer: D

Explanation:
Detailed Explanation:A readiness probe answers whether a running replica is currently ready to receive traffic. For a web-triggered order-processing service, an HTTP readiness probe can test the application endpoint and keep the replica out of the routing pool until it is able to handle requests. Liveness probes detect whether the process should be restarted, and startup probes protect slow-starting containers during initialization. A TCP readiness probe can test socket acceptance, but the HTTP readiness probe is the more application-aware choice for the HTTP service described.
Study Guide Alignment:Containerized Azure workloads: registry builds, App Service containers, Container Apps revision/scaling behavior, and AKS deployment choices.
Official Microsoft Learn References:AI-200 Study Guide | Health probes in Azure Container Apps

NEW QUESTION 5
You are developing a microservices-based application that uses Azure Container Apps The application consists of several containerized services that handle tasks, such as processing orders, managing inventory, and generating reports.
You must secure the container apps. All apps must reside in the same virtual network, share the same Dapr configuration, and share the same logging location. Apps must support the configuration of the amount of memory and compute resources available to containers.
You need to configure the Azure Container App
How should you complete the CLI command? To answer, select the appropriate options in the answer area.
NOTE: Each correct selection is worth one point.



A. Mastered
B. Not Mastered

Answer: A

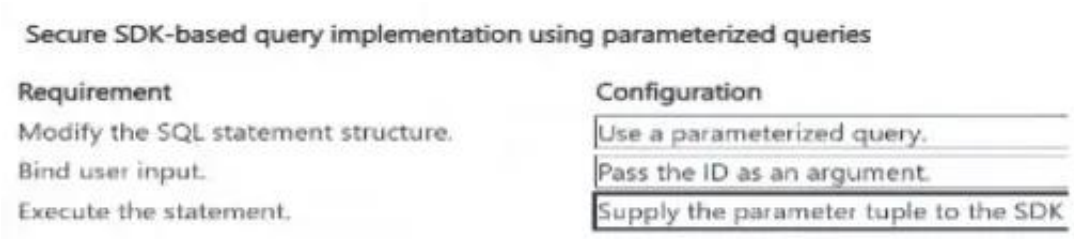
Explanation:
Verified Answer:`az containerapp env create ... --enable-workload-profiles`.
Detailed Explanation:The shared boundary for Container Apps is the managed environment. Apps in one environment can share the environment??s virtual network integration, observability destination, and Dapr-related environment configuration. Enabling workload profiles provides selectable compute profiles and resource sizing options for workloads in that environment. The CLI must therefore create a Container Apps environment and enable workload profiles. Creating an individual container app alone would not establish the common environment-level network, logging, and resource-profile boundary required by the scenario.
Study Guide Alignment:Containerized Azure workloads: registry builds, App Service containers, Container Apps revision/scaling behavior, and AKS deployment choices.
Official Microsoft Learn References:AI-200 Study Guide | Azure Container Apps environments

NEW QUESTION 6
A Python API retrieves a document from Azure Database for PostgreSQL by using a SQL statement. The API accepts the document ID from user input. The current implementation inserts the document ID directly into the SQL statement.
You need to secure the SQL statement execution by using a parameterized query. You must minimize the possibility of SQL injection.
How should you update the implementation to execute the SQL statement safely? To answer, move the appropriate configurations to the correct requirements. You may use each configuration once, more than once, or not at all. you may need to move the split bar between panes or scroll to view content.
NOTE: Each correct selection is worth one point.

A. Mastered
B. Not Mastered

Answer: A

Explanation:



Verified Answer:Modify SQL: use a parameterized query. Bind input: pass the ID as an argument. Execute: supply the parameter tuple to the SDK/driver method.
Detailed Explanation:The security boundary is created by separating SQL syntax from user-supplied values. The statement should contain a parameter placeholder rather than concatenated input, and the document ID should be supplied separately through the database driver??s parameter mechanism. The driver then performs the correct protocol-level binding and escaping. Manually escaping quotation marks is error-prone and does not provide the same injection resistance as true parameterization.
Study Guide Alignment:AI data-management workloads: Cosmos DB, PostgreSQL, caching, vector storage, vector retrieval, consistency, and connection optimization.
Official Microsoft Learn References:AI-200 Study Guide | Vector similarity search with Azure PostgreSQL

NEW QUESTION 7
You need to address the known issue resulting from vector similarity queries.
Which two actions should you perform? Each correct answer presents part of the solution. Choose two. NOTE: Each correct selection is worth one point.

A. Add a composite index on the vector fields and metadata properties of the container.
B. Reduce the numeric precision of the vector embeddings where supported.
C. Set the account consistency level to Strong.
D. Change the vector index type from flat to quantizedFlat or diskANN.

Answer: BD

Explanation:
The objective is to reduce RU consumption from vector similarity queries. Microsoft guidance identifies vector numeric precision and vector-index selection as major cost and performance levers. Using a lower supported vector precision can reduce storage and processing cost, while quantizedFlat and DiskANN are designed to reduce latency and RU consumption compared with flat search for appropriate data sizes. Strong consistency would increase resource cost rather than solve vector-search efficiency. A regular composite index is not a substitute for the vector index. Option B should be read as reducing vector numeric precision, not changing a generic ??indexing precision?? setting.
Study Guide Alignment: AI data-management workloads: Cosmos DB, PostgreSQL, caching, vector storage, vector retrieval, consistency, and connection optimization.
Official Microsoft Learn References: AI-200 Study Guide | Vector search in Azure Cosmos DB | Optimize Cosmos DB vector search performance

NEW QUESTION 8
You are developing an AI application that retrieves database credentials from Key Vault by using the Azure SDK for Python.
The application must use managed identity for authentication.

You review the following code segment that retrieves a secret from Key Vault:

```
01  from azure.identity import DefaultAzureCredential
02  from azure.keyvault.secrets import SecretClient
03
04  credential = DefaultAzureCredential()
05  client = SecretClient(
06      vault_url="https://vault.vault.azure.net/",
07      credential=credential
08  )
09
10  secret = client.get_secret("dbPassword", version="123")
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No. NOTE: Each correct selection is worth one point.

Secret retrieval behavior

Statement	Yes	No
The code retrieves a specific version of the dbPassword secret.	○	○
The code authenticates to Key Vault without storing client secrets in the application when deployed to an Azure hosted environment that has a managed identity enabled.	○	○
If dbPassword is rotated and a new version is created, subsequent calls to this code will retrieve the new version.	○	○

A. Mastered
B. Not Mastered

Answer: A

Explanation:
The get_secret call supplies an explicit version value, so it retrieves that particular version of dbPassword. DefaultAzureCredential can authenticate to Key Vault by using the Azure-hosted application??s managed identity without embedding a client secret in the application. Because the code requests a fixed version, later secret rotation creates a newer version but subsequent executions of this exact code continue to request the specified old version. To follow the latest version automatically, the version parameter must be omitted.
Study Guide Alignment: Security and operations: Key Vault, App Configuration, managed identity, OpenTelemetry, Azure Monitor, and KQL-based troubleshooting.
Official Microsoft Learn References: AI-200 Study Guide | Managed identities for Azure resources | Use Key Vault references for App Service and Functions

NEW QUESTION 9
You deploy an API to Azure Container Apps.
The solution must provide the following functionality:
• Support the concurrent activation of multiple application versions.
• Allocate a specific percentage of incoming requests to a secondary version
You need to configure revision behavior.
Which configurations should you use? To answer, move the appropriate configurations to the correct requirements. You may use each configuration once, more than once, or not at all. You may need to move the split bar between panes or scroll to view content.
NOTE: Each correct selection is worth one point.

Configurations

- Traffic splitting
- Traffic isolation
- Single revision mode
- Multiple revision mode

Revision management and traffic routing in Azure Container Apps

Requirement	Configuration
Test a new revision without affecting production traffic.	
Route 20 percent of traffic to a new revision.	

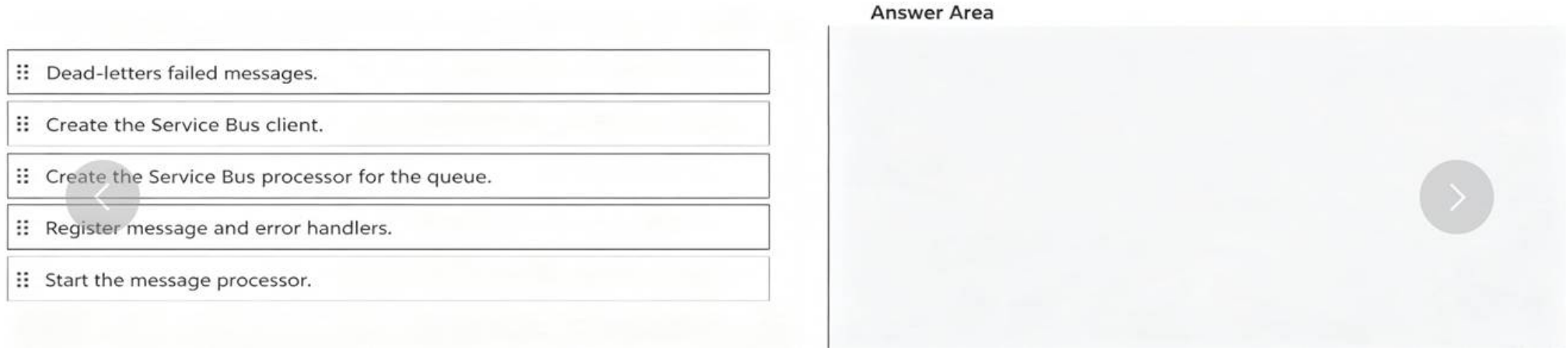
A. Mastered
B. Not Mastered

Answer: A

Explanation:
Azure Container Apps supports side-by-side application versions through multiple revision mode. Once multiple revisions can be active, ingress traffic can be distributed by percentage across those revisions. This is the required combination for canary or controlled rollout scenarios. Single revision mode deactivates the previous revision as a new one becomes active, while traffic splitting is the specific feature that assigns a percentage of incoming requests to each active revision.
Study Guide Alignment: Containerized Azure workloads: registry builds, App Service containers, Container Apps revision/scaling behavior, and AKS deployment choices.

Official Microsoft Learn References: AI-200 Study Guide | Azure Container Apps revisions

NEW QUESTION 10
You are provisioning and configuring a Service Bus processor for AI batch jobs.
The processor must connect to an existing queue, register handlers for message and error processing, and then begin receiving messages.
You need to provision and configure the Service Bus processor for message and error handling.
Which four 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.



A. Mastered
B. Not Mastered

Answer: A

Explanation:
The processor depends on a namespace client/connection, so the client is created first. Next, the queue-specific processor or receiver is created from that client. Message and error callbacks must be registered before processing starts so incoming messages and failures have defined handlers. Starting the processor is therefore the final step. Dead-lettering failed messages is an optional application settlement decision and is not a prerequisite for constructing and starting the processor.
Study Guide Alignment: Azure service integration: Service Bus, Event Grid, Azure Functions triggers
/bindings, and event-driven processing.
Official Microsoft Learn References: AI-200 Study Guide | Service Bus message transfers, locks, and settlement

NEW QUESTION 10
You optimize an AI inference API that uses Redis caching.
You must reduce the risk of serving outdated data while minimizing cache management overhead. You need to implement the caching strategy that satisfies the requirements. What should you do?

A. Reset expiration on each read.
B. Disable expiration for cache keys.
C. Set eviction policy to allkeys-lru.
D. Trigger invalidation when source data changes

Answer: D

Explanation:
If the priority is to minimize stale results, source-driven invalidation is the direct control: when the underlying record changes, delete the corresponding cache entry so the next request repopulates it from current data. Sliding expiration can keep a frequently accessed stale item alive, disabling expiration is worse, and an allkeys-LRU policy evicts according to memory pressure rather than data freshness. Invalidation therefore best addresses correctness without requiring constant cache-wide maintenance.
Study Guide Alignment: AI data-management workloads: Cosmos DB, PostgreSQL, caching, vector storage, vector retrieval, consistency, and connection optimization.
Official Microsoft Learn References: AI-200 Study Guide | Azure Managed Redis documentation

NEW QUESTION 11
A Python API running in ACA must send distributed traces to Azure Monitor. The API creates spans. However, no traces appear in Azure Monitor.
You need to configure the OpenTelemetry SDK pipeline to export traces to Azure Monitor.
What should you do? To answer, move the appropriate actions to the correct requirements. You may use each action once, more than once, or not at all. You may need to move the split bar between panes or scroll to view content.
NOTE: Each correct selection is worth one point.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
OpenTelemetry tracing is a pipeline: the application obtains a tracer from a configured provider, creates spans, passes ended spans through a span processor, and exports them using the Azure Monitor exporter. Creating spans alone is insufficient; without the provider/processor/exporter chain, no telemetry reaches Azure Monitor. Log sampling and legacy Application Insights TrackEvent calls are unrelated to the required OpenTelemetry trace-export path.
Study Guide Alignment: Security and operations: Key Vault, App Configuration, managed identity, OpenTelemetry, Azure Monitor, and KQL-based troubleshooting.
Official Microsoft Learn References: AI-200 Study Guide | Enable Azure Monitor OpenTelemetry

NEW QUESTION 14
You are designing a solution that will use two Azure Functions apps: App1 and App2. App1 is Windows based and will be deployed as code. App2 is Linux based and will be deployed as a container image. Estimates show that the duration of the request processing for both apps will range from 1 to 10 minutes.
You plan to implement App1 and App2 by using the hosting plan to satisfy the following requirements:
• Request processing can complete within the estimated time range.
• The autoscaling behavior is event driven.
• The upper scaling limit is maximized.
You need to create the hosting plan for the implementation.
Which hosting plan should you create? To answer, move the appropriate hosting plans to the correct apps. You may use each hosting plan once, more than once, or not at all. You may..

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Explanation
pp1: Consumption. App2: Premium.
Detailed Explanation: For App1, the Windows code-based Consumption plan satisfies event-driven autoscaling, allows execution up to the stated ten-minute range, and has the highest scaling ceiling among the listed event-driven choices for this workload. App2 is deployed as a Linux container image; among Premium, Dedicated, and Consumption, Premium is the event-driven plan that supports Linux containerized Functions and permits long-running execution. Dedicated hosting is not the requested event-driven autoscale model. The original Premium/Premium answer therefore did not maximize App1??s upper scaling limit.
Study Guide Alignment: Azure service integration: Service Bus, Event Grid, Azure Functions triggers/bindings, and event-driven processing.
Official Microsoft Learn References: AI-200 Study Guide | Azure Functions scale and hosting

NEW QUESTION 17
You are reviewing the message processing code in a backend worker service.
You review the following Python code that initializes and starts a Service Bus processor

```
from azure.servicebus import ServiceBusClient, ServiceBusReceiveMode

01 import json
02 from azure.servicebus import ServiceBusClient, ServiceBusReceiveMode
03 from azure.identity import DefaultAzureCredential
04 credential = DefaultAzureCredential()
05 with client.get_queue_receiver(
06     queue_name="inference-requests",
```

Service Bus processor behavior and SDK defaults

Statement	Yes	No
Messages are removed from the queue as soon as they are received by the processor.	○	○
If message processing fails due to a transient service error, the message can be retried by another consumer.	○	○
Messages with invalid JSON payloads are retried until the maximum delivery count is reached.	○	○

A. Mastered
B. Not Mastered

Answer: A

Explanation:

The default Service Bus receive behavior is PeekLock rather than receive-and-delete, so merely receiving a message does not remove it from the queue; it must be settled successfully. If processing fails and the message remains unsettled or the lock is lost, it can become available for another delivery. Invalid JSON, however, is an application-level content problem: Service Bus does not inspect the payload as JSON and automatically retry it to MaxDeliveryCount solely because deserialization fails. The application??s settlement/error path determines what happens.
Study Guide Alignment: Azure service integration: Service Bus, Event Grid, Azure Functions triggers/bindings, and event-driven processing.
Official Microsoft Learn References: AI-200 Study Guide | Service Bus message transfers, locks, and settlement | Service Bus messages, routing, and correlation

NEW QUESTION 22
You are developing an AI search API that caches semantic search results in Redis.
Search results must remain cached for 10 minutes. If the underlying data changes, cached entries must NOT be returned.
You need to implement a cache aside strategy to ensure data consistency.
Which two actions should you perform? Each correct answer presents part of the solution. Choose two
NOTE: Each correct selection is worth one point.

A. Configure a 10-minute Time to Live on each key.
B. Implement sliding expiration based on key access.
C. Configure a cache notification for key space events.
D. Delete related cache keys when the source data changes

Answer: AD

Explanation:

The cache-aside design needs both bounded lifetime and explicit invalidation. A ten-minute TTL ensures cached search results do not persist indefinitely, while deleting related keys when source data changes prevents known-stale entries from being returned during that ten-minute window. Sliding expiration would extend lifetime based on access and can preserve stale data. Keyspace notifications report Redis key events; they do not replace application logic that invalidates entries when the authoritative source changes.
Study Guide Alignment: AI data-management workloads: Cosmos DB, PostgreSQL, caching, vector storage, vector retrieval, consistency, and connection optimization.
Official Microsoft Learn References: AI-200 Study Guide | Azure Managed Redis documentation

NEW QUESTION 23
You are designing an Azure Database for PostgreSQL table for semantic search. Queries frequently filter on the created_at column. The schema must support vector similarity search and reliable date filtering You need to ensure that the table meets the requirements.
Which two actions should you perform? Each correct answer presents part of the solution. Choose two NOTE: Each correct selection is worth one point.

A. Store embeddings in a pgvector column.
B. Store embeddings in a varchar column.
C. Use a typed timestamp column for created_at.
D. Store created_at as a free-form string.

Answer: AC

Explanation:
A semantic-search schema should use the pgvector vector type for embeddings so PostgreSQL can apply vector distance operators and vector indexes. The creation timestamp should use an actual timestamp data type so comparisons, range predicates, ordering, and indexes use date semantics rather than string ordering. Storing vectors or dates as free-form character data would sacrifice type validation and make similarity or date filtering less efficient and less reliable.
Study Guide Alignment: AI data-management workloads: Cosmos DB, PostgreSQL, caching, vector storage, vector retrieval, consistency, and connection optimization.
Official Microsoft Learn References: AI-200 Study Guide | Vector similarity search with Azure PostgreSQL

NEW QUESTION 25
You have an Azure web app that uses Azure Cosmos DB as a data store. You create a Cosmos DB container by running the following PowerShell script:

```
$resourceGroupName = "testResourceGroup"
$accountName = "testCosmosAccount"
$databaseName = "testDatabase"
$containerName = "testContainer"
$partitionKeyPath = "/EmployeeId"
$autoscaleMaxThroughput = 5000

New-AzCosmosDBSqlContainer `
   -ResourceGroupName $resourceGroupName `
   -AccountName $accountName `
   -DatabaseName $databaseName `
   -Name $containerName `
   -PartitionKeyKind Hash `
   -PartitionKeyPath $partitionKeyPath `
   -AutoscaleMaxThroughput $autoscaleMaxThroughput
```

You create the following queries that target the container:

```
SELECT * FROM c WHERE c.EmployeeId > '12345'
SELECT * FROM c WHERE c.UserId = '12345'
```

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 Ares

Statements	Yes	No
The minimum throughput for the container is 400 RU/s	○	○
The first query statement is an in-partition query.	○	○
The second query statement is a cross-partition query.	○	○

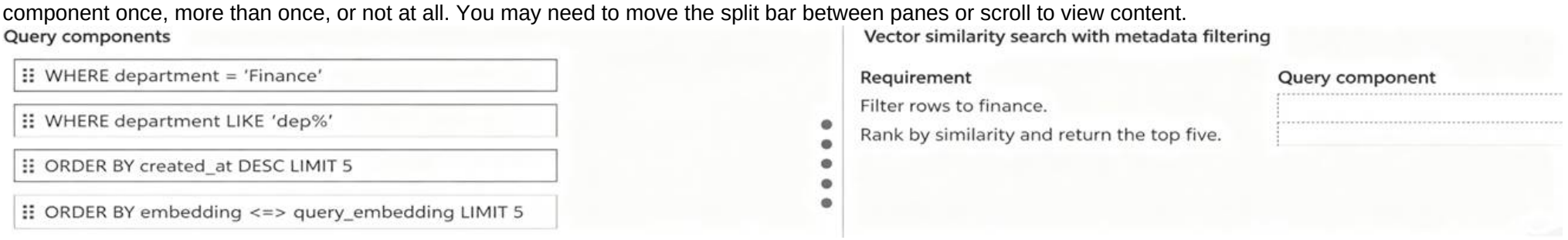
A. Mastered
B. Not Mastered

Answer: A

Explanation:
The question maps directly to the AI-200 objective ??Develop AI solutions by using Azure Cosmos DB for NoSQL,?? which includes running queries and optimizing query performance and Request Unit (RU) consumption.
Statement 1: No — The minimum throughput is not 400 RU/s.
The PowerShell command provisions the container with:
-AutoscaleMaxThroughput 5000
Azure Cosmos DB autoscale operates between approximately 10% and 100% of the configured maximum throughput . Microsoft documentation specifically gives the example of an autoscale container provisioned with 5,000 RU/s scaling between 500 RU/s and 5,000 RU/s . Therefore, for this container, the minimum operating autoscale throughput is 500 RU/s , not 400 RU/s.

Therefore:
??The minimum throughput for the container is 400 RU/s.?? # No Statement 2: No — The first query is not an in-partition query. The container uses:
/EmployeeId
as its partition key. The first query is:
SELECT * FROM c WHERE c.EmployeeId > ' 12345 '
Although the query references the partition key, it uses a range predicate (>) , not an equality predicate. Microsoft explicitly states that a range filter on a partition key is not scoped to a single physical partition . To qualify as an in-partition query, the filter must identify the applicable partition, typically through an equality predicate such as:
WHERE c.EmployeeId = ' 12345 '
Microsoft ' s documentation provides essentially the same example: a query using DeviceId > ... against a container partitioned by DeviceId is not an in-partition query .
Therefore:
??The first query statement is an in-partition query.?? # No Statement 3: Yes — The second query is a cross-partition query. The second query is:
SELECT * FROM c WHERE c.UserId = ' 12345 '
The container ' s partition key is /EmployeeId, not /UserId. Because the query contains no filter on the partition key , Azure Cosmos DB cannot route it to one logical partition. It must fan out the query across the applicable physical partitions and combine the results.
Microsoft describes this behavior directly: when a query does not contain a filter on the partition key, it must execute across the partitions.
Therefore:
??The second query statement is a cross-partition query.?? # Yes

NEW QUESTION 28
A semantic search application queries Azure Database for PostgreSQL and stores document embeddings and metadata in a table with the following columns:
• embedding (pgvector)
• department
• created_at
The application must return the top five most similar documents for a given query embedding only from the finance department. You need to implement semantic retrieval with metadata filtering.
Which query components should you select? To answer, move the appropriate query components to the correct requirements. You may use each query component once, more than once, or not at all. You may need to move the split bar between panes or scroll to view content.



A. Mastered
B. Not Mastered

Answer: A

Explanation:
The metadata predicate must restrict the candidate rows to the Finance department, so the `WHERE department = ' finance ' ` component supplies the required filter. The pgvector cosine-distance operator ` < = > ` orders rows by vector distance to the supplied query embedding, and `LIMIT 5` keeps only the five nearest matches. Ordering by creation date would rank recency rather than semantic similarity, and a wildcard department filter would not satisfy the Finance-only requirement.
Study Guide Alignment: AI data-management workloads: Cosmos DB, PostgreSQL, caching, vector storage, vector retrieval, consistency, and connection optimization.
Official Microsoft Learn References: AI-200 Study Guide | Vector similarity search with Azure PostgreSQL

NEW QUESTION 33
You are reviewing an Azure Function app that processes incoming order requests for a company. The function must:
• Accept order submissions from an external client application.
• Require controlled access for security
• Return a response containing the processed request payload.
You review the following code segment:

```
01  import azure.functions as func
02  app = func.FunctionApp(http_auth_level=func.AuthLevel.FUNCTION)
03    @app.route(route="processorder", methods=["POST"])
04    def processorder(req: func.HttpRequest) -> func.HttpResponse:
05      result = req.get_body().decode("utf-8")
06      return func.HttpResponse(result)
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.
NOTE: Each correct selection is worth one point.

Code-level analysis of HTTP trigger configuration

Statement	Yes	No
The function requires a function key for invocation.	○	○
The function supports HTTP GET requests.	○	○
The response returns the request body.	○	○
The function validates the request body before returning a response.	○	○

A. Mastered
B. Not Mastered

Answer: A

Explanation:

auth_level=FUNCTION` means invocation requires a function key unless a stronger platform authentication layer is configured, so the first statement is true. The route explicitly permits POST,
not GET. The function reads `req.get_body()` and returns that value in the HttpResponse, so the response contains the request body. No schema, type, required-field, or other payload validation is performed before the response is constructed, making the final statement false.
Study Guide Alignment: Azure service integration: Service Bus, Event Grid, Azure Functions triggers /bindings, and event-driven processing.
Official Microsoft Learn References: AI-200 Study Guide | Azure Functions HTTP trigger

NEW QUESTION 37
You develop a message-processing service deployed to Azure Container Apps. The service reads messages from an Azure Service Bus queue. The solution must minimize costs by ensuring NO compute resources are consumed when the queue is empty. You need to configure scaling for the service. Which two actions should you perform? Each correct answer presents part of the solution. NOTE: Each correct selection is worth one point.

A. A.Increase the scaling rule to allow for the maximum running replica count.B.Configure the scaling rule to allow for the termination of all active replicas.C.Configure a Kubernetes Event-driven Autoscaler rule that monitors queue length.D.Enable HTTP ingress concurrency scaling.

Answer: BC

NEW QUESTION 41
You plan to deploy a container to an Azure App Service API app named api1. You host the source code for api1 in a GitHub repository. The container uses the API key at runtime to connect to a backend service.
The container must be able to retrieve the API key at runtime without exposing it in the source repository or Git commit history.
You need to ensure that the API key remains outside of Git commit history and is available to the container at runtime.
Solution: Store the API key as a GitHub repository secret.
Does the solution meet the goal?

A. Yes
B. No

Answer: B

Explanation:
Correct:
* Store the API key in Azure Key Vault and reference it from an App Service application setting.
Storing the API key in Azure Key Vault and referencing it via App Service application settings is the recommended, secure approach. This strategy completely removes sensitive credentials from your GitHub repository and Git commit history while injecting them safely into your container environment at runtime.
Incorrect:
* Embed the API key as a hardcoded environment variable in the Dockerfile.
* Store the API key as a GitHub repository secret.
Reference:
https://www.qservicesit.com/full-stack-applications-on-azure

NEW QUESTION 44
You plan to deploy a container to an Azure App Service API app named api1. You host the source code for api1 in a GitHub repository. The container uses the API key at runtime to connect to a backend service.
The container must be able to retrieve the API key at runtime without exposing it in the source repository or Git commit history.
You need to ensure that the API key remains outside of Git commit history and is available to the container at runtime.
Solution: Embed the API key as a hardcoded environment variable in the Dockerfile.
Does the solution meet the goal?

A. Yes
B. No

Answer: B

Explanation:
Correct:
* Store the API key in Azure Key Vault and reference it from an App Service application setting.
Storing the API key in Azure Key Vault and referencing it via App Service application settings is the recommended, secure approach. This strategy completely

removes sensitive credentials from your GitHub repository and Git commit history while injecting them safely into your container environment at runtime.
Incorrect:
* Embed the API key as a hardcoded environment variable in the Dockerfile.
* Store the API key as a GitHub repository secret.
Reference:
https://www.qservicesit.com/full-stack-applications-on-azure

NEW QUESTION 46
You are developing an AI search API that caches semantic search results in Redis.
Search results must remain cached for 10 minutes. If the underlying data changes, cached entries must NOT be returned.
You need to implement a cache-aside strategy to ensure data consistency.
Which two actions should you perform? Each correct answer presents part of the solution.
NOTE: Each correct selection is worth one point.

A. Configure a cache notification for key space event
B. Delete related cache keys when the source data change
C. Implement sliding expiration based on key acces
D. Configure a 10-minute Time to Live on each key.

Answer: BD

Explanation:
Delete keys on mutation. Implement an event-driven or database-hook system to purge related Redis keys the moment the source data updates.
Use absolute expiration. Set a strict, un-extendable Time-To-Live (TTL) of 10 minutes when writing to the cache, ensuring the data naturally expires even if a deletion event is missed.
Incorrect:
[Not C]
Implementing a sliding expiration is not a good step for this specific architecture because it violates the requirement that entries must not be returned if the underlying data changes.
Reference:
https://medium.com/real-world-net/net-caching-strategies-compared-redis-vs-memory-vs-hybrid-b8b8be1e708d

NEW QUESTION 51
.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questons and Answers in PDF Format

AI-200 Practice Exam Features:

* AI-200 Questions and Answers Updated Frequently

* AI-200 Practice Questions Verified by Expert Senior Certified Staff

* AI-200 Most Realistic Questions that Guarantee you a Pass on Your FirstTry

* AI-200 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year