DATABRICKS-CERTIFIED-PR OFESSIONAL-DATA-ENGINEER$^{Q\&As}$

Databricks Certified Professional Data Engineer Exam

Pass Databricks DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER Exam with 100% Guarantee

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

QUESTION 1

A Delta table of weather records is partitioned by date and has the below schema:

date DATE, device_id INT, temp FLOAT, latitude FLOAT, longitude FLOAT

To find all the records from within the Arctic Circle, you execute a query with the below filter:

latitude > 66.3

Which statement describes how the Delta engine identifies which files to load?

A. All records are cached to an operational database and then the filter is applied

B. The Parquet file footers are scanned for min and max statistics for the latitude column

C. All records are cached to attached storage and then the filter is applied

D. The Delta log is scanned for min and max statistics for the latitude column

E. The Hive metastore is scanned for min and max statistics for the latitude column

Correct Answer: D

Explanation: This is the correct answer because Delta Lake uses a transaction log to store metadata about each table, including min and max statistics for each column in each data file. The Delta engine can use this information to quickly

identify which files to load based on a filter condition, without scanning the entire table or the file footers. This is called data skipping and it can improve query performance significantly. Verified References:

[Databricks Certified Data Engineer Professional], under "Delta Lake" section; [Databricks Documentation], under "Optimizations - Data Skipping" section.

QUESTION 2

A Structured Streaming job deployed to production has been experiencing delays during peak hours of the day. At present, during normal execution, each microbatch of data is processed in less than 3 seconds. During peak hours of the day, execution time for each microbatch becomes very inconsistent, sometimes exceeding 30 seconds. The streaming write is currently configured with a trigger interval of 10 seconds.

Holding all other variables constant and assuming records need to be processed in less than 10 seconds, which adjustment will meet the requirement?

A. Decrease the trigger interval to 5 seconds; triggering batches more frequently allows idle executors to begin processing the next batch while longer running tasks from previous batches finish.

B. Increase the trigger interval to 30 seconds; setting the trigger interval near the maximum execution time observed for each batch is always best practice to ensure no records are dropped.

C. The trigger interval cannot be modified without modifying the checkpoint directory; to maintain the current stream state, increase the number of shuffle partitions to maximize parallelism.

D. Use the trigger once option and configure a Databricks job to execute the query every 10 seconds; this ensures all backlogged records are processed with each batch.

E. Decrease the trigger interval to 5 seconds; triggering batches more frequently may prevent records from backing up and large batches from causing spill.

Correct Answer: D

Explanation: This is the correct answer because it can meet the requirement of processing records in less than 10 seconds without modifying the checkpoint directory or dropping records. The trigger once option is a special type of trigger that runs the streaming query only once and terminates after processing all available data. This option can be useful for scenarios where you want to run streaming queries on demand or periodically, rather than continuously. By using the trigger once option and configuring a Databricks job to execute the query every 10 seconds, you can ensure that all backlogged records are processed with each batch and avoid inconsistent execution times. Verified References: [Databricks Certified Data Engineer Professional], under "Structured Streaming" section; Databricks Documentation, under "Trigger Once" section.

QUESTION 3

Although the Databricks Utilities Secrets module provides tools to store sensitive credentials and avoid accidentally displaying them in plain text users should still be careful with which credentials are stored here and which users have access to using these secrets.

Which statement describes a limitation of Databricks Secrets?

A. Because the SHA256 hash is used to obfuscate stored secrets, reversing this hash will display the value in plain text.

B. Account administrators can see all secrets in plain text by loggingon to the Databricks Accounts console.

C. Secrets are stored in an administrators-only table within the Hive Metastore; database administrators have permission to query this table by default.

D. Iterating through a stored secret and printing each character will display secret contents in plain text.

E. The Databricks REST API can be used to list secrets in plain text if the personal access token has proper credentials.

Correct Answer: E

Explanation: This is the correct answer because it describes a limitation of Databricks Secrets. Databricks Secrets is a module that provides tools to store sensitive credentials and avoid accidentally displaying them in plain text. Databricks Secrets allows creating secret scopes, which are collections of secrets that can be accessed by users or groups. Databricks Secrets also allows creating and managing secrets using the Databricks CLI or the Databricks REST API. However, a limitation of Databricks Secrets is that the Databricks REST API can be used to list secrets in plain text if the personal access token has proper credentials. Therefore, users should still be careful with which credentials are stored in Databricks Secrets and which users have access to using these secrets. Verified References: [Databricks Certified Data Engineer Professional], under "Databricks Workspace" section; Databricks Documentation, under "List secrets" section.

QUESTION 4

When scheduling Structured Streaming jobs for production, which configuration automatically recovers from query failures and keeps costs low?

A. Cluster: New Job Cluster; Retries: Unlimited; Maximum Concurrent Runs: Unlimited

B. Cluster: New Job Cluster; Retries: None; Maximum Concurrent Runs: 1

C. Cluster: Existing All-Purpose Cluster; Retries: Unlimited; Maximum Concurrent Runs: 1

D. Cluster: Existing All-Purpose Cluster; Retries: Unlimited; Maximum Concurrent Runs: 1

E. Cluster: Existing All-Purpose Cluster; Retries: None; Maximum Concurrent Runs: 1

Correct Answer: B

Explanation: This is the best configuration for scheduling Structured Streaming jobs for production, as it automatically recovers from query failures and keeps costs low. A new job cluster is created for each run of the job and terminated when the job completes, which saves costs and avoids resource contention. Retries are not needed for Structured Streaming jobs, as they can automatically recover from failures using checkpointing and write-ahead logs. Maximum concurrent runs should be set to 1 to avoid duplicate output or data loss. Verified References: Databricks Certified Data Engineer Professional, under "Monitoring and Logging" section; Databricks Documentation, under "Schedule streaming jobs" section.

QUESTION 5

A production cluster has 3 executor nodes and uses the same virtual machine type for the driver and executor.

When evaluating the Ganglia Metrics for this cluster, which indicator would signal a bottleneck caused by code executing on the driver?

A. The five Minute Load Average remains consistent/flat

B. Bytes Received never exceeds 80 million bytes per second

C. Total Disk Space remains constant

D. Network I/O never spikes

E. Overall cluster CPU utilization is around 25%

Correct Answer: E

Explanation: This is the correct answer because it indicates a bottleneck caused by code executing on the driver. A bottleneck is a situation where the performance or capacity of a system is limited by a single component or resource. A bottleneck can cause slow execution, high latency, or low throughput. A production cluster has 3 executor nodes and uses the same virtual machine type for the driver and executor. When evaluating the Ganglia Metrics for this cluster, one can look for indicators that show how the cluster resources are being utilized, such as CPU, memory, disk, or network. If the overall cluster CPU utilization is around 25%, it means that only one out of the four nodes (driver + 3 executors) is using its full CPU capacity, while the other three nodes are idle or underutilized. This suggests that the code executing on the driver is taking too long or consuming too much CPU resources, preventing the executors from receiving tasks or data to process. This can happen when the code has driver-side operations that are not parallelized or distributed, such as collecting large amounts of data to the driver, performing complex calculations on the driver, or using non-Spark libraries on the driver. Verified References: [Databricks Certified Data Engineer Professional], under "Spark Core" section; Databricks Documentation, under "View cluster status and event logs - Ganglia metrics" section; Databricks Documentation, under "Avoid collecting large RDDs" section.