



# DEA-C01<sup>Q&As</sup>

SnowPro Advanced: Data Engineer Certification Exam

## Pass Snowflake DEA-C01 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass4itsure.com/dea-c01.html>

100% Passing Guarantee  
100% Money Back Assurance

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

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers





## QUESTION 1

A company has a production AWS account that runs company workloads. The company's security team created a security AWS account to store and analyze security logs from the production AWS account. The security logs in the production

AWS account are stored in Amazon CloudWatch Logs.

The company needs to use Amazon Kinesis Data Streams to deliver the security logs to the security AWS account.

Which solution will meet these requirements?

- A. Create a destination data stream in the production AWS account. In the security AWS account, create an IAM role that has cross-account permissions to Kinesis Data Streams in the production AWS account.
- B. Create a destination data stream in the security AWS account. Create an IAM role and a trust policy to grant CloudWatch Logs the permission to put data into the stream. Create a subscription filter in the security AWS account.
- C. Create a destination data stream in the production AWS account. In the production AWS account, create an IAM role that has cross-account permissions to Kinesis Data Streams in the security AWS account.
- D. Create a destination data stream in the security AWS account. Create an IAM role and a trust policy to grant CloudWatch Logs the permission to put data into the stream. Create a subscription filter in the production AWS account.

Correct Answer: D

Explanation: Amazon Kinesis Data Streams is a service that enables you to collect, process, and analyze real-time streaming data. You can use Kinesis Data Streams to ingest data from various sources, such as Amazon CloudWatch Logs, and deliver it to different destinations, such as Amazon S3 or Amazon Redshift. To use Kinesis Data Streams to deliver the security logs from the production AWS account to the security AWS account, you need to create a destination data stream in the security AWS account. This data stream will receive the log data from the CloudWatch Logs service in the production AWS account. To enable this cross-account data delivery, you need to create an IAM role and a trust policy in the security AWS account. The IAM role defines the permissions that the CloudWatch Logs service needs to put data into the destination data stream. The trust policy allows the production AWS account to assume the IAM role. Finally, you need to create a subscription filter in the production AWS account. A subscription filter defines the pattern to match log events and the destination to send the matching events. In this case, the destination is the destination data stream in the security AWS account. This solution meets the requirements of using Kinesis Data Streams to deliver the security logs to the security AWS account. The other options are either not possible or not optimal. You cannot create a destination data stream in the production AWS account, as this would not deliver the data to the security AWS account. You cannot create a subscription filter in the security AWS account, as this would not capture the log events from the production AWS account. References: Using Amazon Kinesis Data Streams with Amazon CloudWatch Logs AWS Certified Data Engineer - Associate DEA-C01 Complete Study Guide, Chapter 3: Data Ingestion and Transformation, Section 3.3: Amazon Kinesis Data Streams

---

## QUESTION 2

A company's data engineer needs to optimize the performance of table SQL queries. The company stores data in an Amazon Redshift cluster. The data engineer cannot increase the size of the cluster because of budget constraints. The company stores the data in multiple tables and loads the data by using the EVEN distribution style. Some tables are hundreds of gigabytes in size. Other tables are less than 10 MB in size.

Which solution will meet these requirements?



- A. Keep using the EVEN distribution style for all tables. Specify primary and foreign keys for all tables.
- B. Use the ALL distribution style for large tables. Specify primary and foreign keys for all tables.
- C. Use the ALL distribution style for rarely updated small tables. Specify primary and foreign keys for all tables.
- D. Specify a combination of distribution, sort, and partition keys for all tables.

Correct Answer: C

Explanation: This solution meets the requirements of optimizing the performance of table SQL queries without increasing the size of the cluster. By using the ALL distribution style for rarely updated small tables, you can ensure that the entire table is copied to every node in the cluster, which eliminates the need for data redistribution during joins. This can improve query performance significantly, especially for frequently joined dimension tables. However, using the ALL distribution style also increases the storage space and the load time, so it is only suitable for small tables that are not updated frequently or extensively. By specifying primary and foreign keys for all tables, you can help the query optimizer to generate better query plans and avoid unnecessary scans or joins. You can also use the AUTO distribution style to let Amazon Redshift choose the optimal distribution style based on the table size and the query patterns. References: Choose the best distribution style Distribution styles Working with data distribution styles

---

### QUESTION 3

A company has used an Amazon Redshift table that is named Orders for 6 months. The company performs weekly updates and deletes on the table. The table has an interleaved sort key on a column that contains AWS Regions.

The company wants to reclaim disk space so that the company will not run out of storage space. The company also wants to analyze the sort key column.

Which Amazon Redshift command will meet these requirements?

- A. VACUUM FULL Orders
- B. VACUUM DELETE ONLY Orders
- C. VACUUM REINDEX Orders
- D. VACUUM SORT ONLY Orders

Correct Answer: C

Amazon Redshift is a fully managed, petabyte-scale data warehouse service that enables fast and cost-effective analysis of large volumes of data. Amazon Redshift uses columnar storage, compression, and zone maps to optimize the storage and performance of data. However, over time, as data is inserted, updated, or deleted, the physical storage of data can become fragmented, resulting in wasted disk space and degraded query performance. To address this issue, Amazon Redshift provides the VACUUM command, which reclaims disk space and resorts rows in either a specified table or all tables in the current schema<sup>1</sup>. The VACUUM command has four options: FULL, DELETE ONLY, SORT ONLY, and REINDEX. The option that best meets the requirements of the question is VACUUM REINDEX, which re-sorts the rows in a table that has an interleaved sort key and rewrites the table to a new location on disk. An interleaved sort key is a type of sort key that gives equal weight to each column in the sort key, and stores the rows in a way that optimizes the performance of queries that filter by multiple columns in the sort key. However, as data is added or changed, the interleaved sort order can become skewed, resulting in suboptimal query performance. The VACUUM REINDEX option restores the optimal interleaved sort order and reclaims disk space by removing deleted rows. This option also analyzes the sort key column and updates the table statistics, which are used by the query optimizer to generate the most efficient query execution plan<sup>23</sup>. The other options are not optimal for the following reasons:



A. VACUUM FULL Orders. This option reclaims disk space by removing deleted rows and resorts the entire table. However, this option is not suitable for tables that have an interleaved sort key, as it does not restore the optimal interleaved

sort order. Moreover, this option is the most resource-intensive and time-consuming, as it rewrites the entire table to a new location on disk. B. VACUUM DELETE ONLY Orders. This option reclaims disk space by removing deleted rows, but

does not resort the table. This option is not suitable for tables that have any sort key, as it does not improve the query performance by restoring the sort order. Moreover, this option does not analyze the sort key column and update the table

statistics.

D. VACUUM SORT ONLY Orders. This option resorts the entire table, but does not reclaim disk space by removing deleted rows. This option is not suitable for tables that have an interleaved sort key, as it does not restore the optimal

interleaved sort order. Moreover, this option does not analyze the sort key column and update the table statistics.

References:

1: Amazon Redshift VACUUM

2: Amazon Redshift Interleaved Sorting

3: Amazon Redshift ANALYZE

---

#### QUESTION 4

A company uses an Amazon QuickSight dashboard to monitor usage of one of the company's applications. The company uses AWS Glue jobs to process data for the dashboard. The company stores the data in a single Amazon S3 bucket. The company adds new data every day.

A data engineer discovers that dashboard queries are becoming slower over time. The data engineer determines that the root cause of the slowing queries is long-running AWS Glue jobs.

Which actions should the data engineer take to improve the performance of the AWS Glue jobs? (Choose two.)

- A. Partition the data that is in the S3 bucket. Organize the data by year, month, and day.
- B. Increase the AWS Glue instance size by scaling up the worker type.
- C. Convert the AWS Glue schema to the DynamicFrame schema class.
- D. Adjust AWS Glue job scheduling frequency so the jobs run half as many times each day.
- E. Modify the 1AM role that grants access to AWS glue to grant access to all S3 features.

Correct Answer: AB

Explanation: Partitioning the data in the S3 bucket can improve the performance of AWS Glue jobs by reducing the amount of data that needs to be scanned and processed. By organizing the data by year, month, and day, the AWS Glue job can use partition pruning to filter out irrelevant data and only read the data that matches the query criteria. This can speed up the data processing and reduce the cost of running the AWS Glue job. Increasing the AWS Glue instance size by scaling up the worker type can also improve the performance of AWS Glue jobs by providing more memory and CPU



resources for the Spark execution engine. This can help the AWS Glue job handle larger data sets and complex transformations more efficiently. The other options are either incorrect or irrelevant, as they do not affect the performance of the AWS Glue jobs. Converting the AWS Glue schema to the DynamicFrame schema class does not improve the performance, but rather provides additional functionality and flexibility for data manipulation. Adjusting the AWS Glue job scheduling frequency does not improve the performance, but rather reduces the frequency of data updates. Modifying the IAM role that grants access to AWS Glue does not improve the performance, but rather affects the security and permissions of the AWS Glue service. References: Optimising Glue Scripts for Efficient Data Processing: Part 1 (Section: Partitioning Data in S3) Best practices to optimize cost and performance for AWS Glue streaming ETL jobs (Section: Development tools) Monitoring with AWS Glue job run insights (Section: Requirements) AWS Certified Data Engineer - Associate DEA-C01 Complete Study Guide (Chapter 5, page 133)

---

## QUESTION 5

A company uses Amazon Redshift for its data warehouse. The company must automate refresh schedules for Amazon Redshift materialized views.

Which solution will meet this requirement with the LEAST effort?

- A. Use Apache Airflow to refresh the materialized views.
- B. Use an AWS Lambda user-defined function (UDF) within Amazon Redshift to refresh the materialized views.
- C. Use the query editor v2 in Amazon Redshift to refresh the materialized views.
- D. Use an AWS Glue workflow to refresh the materialized views.

Correct Answer: C

Explanation: The query editor v2 in Amazon Redshift is a web-based tool that allows users to run SQL queries and scripts on Amazon Redshift clusters. The query editor v2 supports creating and managing materialized views, which are precomputed results of a query that can improve the performance of subsequent queries. The query editor v2 also supports scheduling queries to run at specified intervals, which can be used to refresh materialized views automatically. This solution requires the least effort, as it does not involve any additional services, coding, or configuration. The other solutions are more complex and require more operational overhead. Apache Airflow is an open-source platform for orchestrating workflows, which can be used to refresh materialized views, but it requires setting up and managing an Airflow environment, creating DAGs (directed acyclic graphs) to define the workflows, and integrating with Amazon Redshift. AWS Lambda is a serverless compute service that can run code in response to events, which can be used to refresh materialized views, but it requires creating and deploying Lambda functions, defining UDFs within Amazon Redshift, and triggering the functions using events or schedules. AWS Glue is a fully managed ETL service that can run jobs to transform and load data, which can be used to refresh materialized views, but it requires creating and configuring Glue jobs, defining Glue workflows to orchestrate the jobs, and scheduling the workflows using triggers. References: Query editor V2 Working with materialized views Scheduling queries [AWS Certified Data Engineer - Associate DEA-C01 Complete Study Guide]

[Latest DEA-C01 Dumps](#)

[DEA-C01 PDF Dumps](#)

[DEA-C01 VCE Dumps](#)