

Free PDF Valid DEA-C02 - Valid SnowPro Advanced: Data Engineer (DEA-C02) Exam Tips



BTW, DOWNLOAD part of Dumpcollection DEA-C02 dumps from Cloud Storage: <https://drive.google.com/open?id=1dS4fNXzjlahU0KcTOBqAcod8sr3cynNO>

Dumpcollection provide training tools included Snowflake certification DEA-C02 exam study materials and simulation training questions and more importantly, we will provide you practice questions and answers which are very close with real certification exam. Selecting Dumpcollection can guarantee that you can in a short period of time to learn and to strengthen the professional knowledge of IT and pass Snowflake Certification DEA-C02 Exam with high score.

In order to make life better, attending Snowflake certification examinations will be the best choice for every IT workers. Passing DEA-C02 exam and obtaining a certification help candidates get salary raise and position promotion opportunities. It will be a fast and convenient road to success for the certification with our Snowflake DEA-C02 Practice Test Engine. As for our guaranteed pass policy, our products are too good a change to miss for ambitious people.

>> **Valid DEA-C02 Exam Tips <<**

Valid DEA-C02 Exam Tips - 100% High-quality Questions Pool

Dumpcollection has hired professionals to supervise the quality of the DEA-C02 PDF prep material. Laptops, tablets, and smartphones support the Snowflake DEA-C02 test questions PDF file. If any taker of the Snowflake DEA-C02 test prepares thoroughly with our exam product he will crack the exam of the credential on the first attempt.

Snowflake SnowPro Advanced: Data Engineer (DEA-C02) Sample Questions (Q264-Q269):

NEW QUESTION # 264

You are designing a data protection strategy for a Snowflake database. You need to implement dynamic data masking on the 'CREDIT CARD' column in the 'TRANSACTIONS' table. The requirement is that users with the 'FINANCE ADMIN' role should see the full credit card number, while all other users should see only the last four digits. You have the following masking policy:

What is the next step to apply this masking policy to the 'CREDIT CARD' column?

- A.
- B.
- C.
- D.
- E.

Answer: B

Explanation:

The correct syntax to apply a masking policy to a column in Snowflake is `SALTER TABLE ALTER COLUMN SET MASKING POLICY`'. Therefore, option B is the correct answer.

NEW QUESTION # 265

You accidentally truncated a large table named 'SALES DATA' in your 'REPORTING DB" database. You realize this happened 2 days ago, and your account has the default Time Travel retention of 1 day. You need to recover this table with minimal downtime. Analyze the situation and determine the best course of action, considering cost and recovery time.

- A. Raise a support ticket requesting data recovery from failsafe. Since data retention period has expired.
- B. Create a clone of the table using the 'AT' clause and a timestamp from 1 day ago. This would prevent any additional cost.
- C. **Immediately contact Snowflake Support to initiate a restore from Fail-safe, understanding that this process may take several hours or even days.**
- D. Because the data retention period has expired, the table is unrecoverable using Snowflake's built-in features; you must restore from an external backup solution if available.
- E. Increase the account-level to 2 days and then use the `UNDROP TABLE SALES_DATA` command.

Answer: C

Explanation:

Since the Time Travel window (1 day) has passed, the 'UNDROP' command and cloning using or 'BEFORE' clauses will not work (eliminating options B and D). While option C might be true if no external backups exist, the most appropriate first step is to contact Snowflake support (option A). This is because, even though Time Travel has expired, Fail-safe might still contain the data, offering a potential recovery path, although it is a longer process and not guaranteed. Option E is also a valid answer.

NEW QUESTION # 266

You're designing a Snowpark data transformation pipeline that requires running a Python function on each row of a large DataFrame. The Python function is computationally intensive and needs access to external libraries. Which of the following approaches will provide the BEST combination of performance, scalability, and resource utilization within the Snowpark architecture?

- A. Load the DataFrame into a Pandas DataFrame using and then apply the Python function using Pandas DataFrame operations.
- B. **Create a Snowpark UDF using `input_types=[StringType()], return_type=StringType()`' and apply it to the DataFrame using**
- C. **Create a Snowpark UDTF using `gudtf(output_schema=StructType([StructField('result', StringType())]))`, and apply it to the DataFrame using with a lateral flatten operation.**
- D. Define a stored procedure in Snowflake and use it to execute the Python code on each row by calling it in a loop.
- E. Use '`DataFrame.foreach(lambda row: my_python_function(row))`' to iterate through each row and apply the Python function.

Answer: B,C

Explanation:

Options B and D are the best choices. UDFs and UDTFs allow you to leverage Snowflake's compute resources for parallel processing. The function execution happens on Snowflake's servers, close to the data, minimizing data transfer. By specifying '`packages=['my_package']`', you ensure that the external libraries are available in the execution environment. A UDF is suitable for one-to-one row transformations, while a UDTF is more appropriate if the Python function needs to return multiple rows for each input row (one-to-many). Option A, `DataFrame.foreach`, is inefficient for large DataFrames as it processes rows sequentially. Option C, loading into Pandas, is also not ideal as it can lead to out-of-memory errors for very large DataFrames and transfers the data to the client machine. Option E, stored procedures with loops, is less scalable and efficient than UDFs or UDTFs.

NEW QUESTION # 267

You are designing a data pipeline that requires applying a complex scoring algorithm to customer data in Snowflake. This algorithm involves multiple steps, including feature engineering, model loading, and prediction. You want to encapsulate this logic within a

reusable component and apply it to incoming data streams efficiently. Which of the following approaches is most suitable and scalable for implementing this scoring logic as a UDF/UDTF, considering real-time data processing and low latency requirements?

- A. A Python UDTF using Snowpark, leveraging external libraries like 'torch' for accelerated calculations and ML model inference by GPU.
- B. A Java UDTF that leverages a custom Java library for feature engineering and model prediction, deployed as a JAR file to Snowflake's internal stage.
- C. A Python UDF that loads a pre-trained machine learning model (e.g., using scikit-learn) and performs predictions on the input data.
- D. A JavaScript UDF that uses basic JavaScript functions to perform the entire scoring algorithm without external dependencies.
- E. A SQL UDF containing a series of nested CASE statements to implement the entire scoring algorithm.

Answer: A

Explanation:

Python UDTFs in Snowpark provide a powerful and scalable way to perform complex scoring algorithms, especially when combined with the GPU. Snowpark optimizes data processing within Snowflake's engine, and integration with Anaconda allows for using machine learning libraries such as scikit-learn or Pytorch for model loading and prediction and external libraries like 'torch' for accelerated calculations and ML model inference by GPU. SQL UDFs are not suitable for complex algorithms. JavaScript UDFs lack the necessary functionality and performance for advanced scoring. While Java UDTFs can be used, managing JAR files and potentially less efficient integration can be disadvantages. Using Python with SNOWPARK for GPU is suitable for real-time scoring and low latency.

NEW QUESTION # 268

You are designing a data sharing solution in Snowflake where a provider account shares a view with a consumer account. The view is based on a table that undergoes frequent DML operations (inserts, updates, deletes). The consumer account needs to see a consistent snapshot of the data, even during these DML operations. Which of the following strategies, or combination of strategies, would be MOST effective in ensuring data consistency from the consumer's perspective, and what considerations should be made?

- A. Creating a stream on the base table in the provider account and building a view on top of the stream. This way, changes are only reflected when the stream is consumed, allowing for batch processing and controlled updates in the consumer account.
- B. Creating a materialized view in the provider account and sharing that materialized view. This adds compute costs to the provider but ensures a consistent snapshot for the consumer account. The materialized view needs to be refreshed periodically, based on the rate of DML changes.
- C. Using Snowflake's Time Travel feature by querying the view with a specific 'AT' or 'BEFORE' clause in the consumer account. The provider account needs to inform the consumer account of a specific timestamp that guarantees consistency, adding administrative overhead.
- D. A and B
- E. Creating a standard view in the provider account and relying on Snowflake's inherent transactional consistency. The consumer account will always see a consistent snapshot of the data as it existed at the beginning of their query execution. No additional configurations are necessary.

Answer: E

Explanation:

Snowflake's architecture inherently provides transactional consistency. When the consumer account queries the shared view, they will see a consistent snapshot of the data as it existed at the beginning of their query execution. No additional mechanisms like Time Travel (A) or materialized views (B) are strictly necessary to ensure consistency. While streams (D) can be useful for change data capture, they don't directly guarantee consistency for a standard view shared with a consumer. Time travel in this case would also require significant coordination overhead.

NEW QUESTION # 269

.....

Now we can say that SnowPro Advanced: Data Engineer (DEA-C02) (DEA-C02) exam questions are real and top-notch Snowflake DEA-C02 exam questions that you can expect in the upcoming Snowflake DEA-C02 exam. In this way, you can easily pass the SnowPro Advanced: Data Engineer (DEA-C02) (DEA-C02) exam with good scores. The countless DEA-C02 Exam

candidates have passed their dream Snowflake DEA-C02 certification exam and they all got help from real, valid, and updated DEA-C02 practice questions. You can also trust on Dumpcollection and start preparation with confidence.

DEA-C02 Reliable Exam Test: https://www.dumpcollection.com/DEA-C02_braindumps.html

The ability of DEA-C02 Reliable Exam Test - SnowPro Advanced: Data Engineer (DEA-C02) latest valid dumps will kill all negative words and gives doubters a heavy punch, Success in the DEA-C02 exam is the basic requirement to get the a good job, You can use our DEA-C02 exam materials to study independently, Snowflake Valid DEA-C02 Exam Tips So the choice is important, Snowflake DEA-C02 certification exam opens the doors for starting a bright career in the sector.

I passed the exam just by using this dump, so it is enough for you to pass DEA-C02 your exam, good luck to you, After downloading the media files, you will be guided through creating a project from scratch to finished draft.

Access Real Dumpcollection Snowflake DEA-C02 Exam Questions Easily in dumps PDF Form

The ability of SnowPro Advanced: Data Engineer (DEA-C02) latest valid dumps will kill all negative words and gives doubters a heavy punch, Success in the DEA-C02 Exam is the basic requirement to get the a good job.

You can use our DEA-C02 exam materials to study independently, So the choice is important, Snowflake DEA-C02 certification exam opens the doors for starting a bright career in the sector.

DOWNLOAD the newest Dumpcollection DEA-C02 PDF dumps from Cloud Storage for free: <https://drive.google.com/open?id=1dS4fNxZlahU0KcTOBqAcod8sr3cynNO>