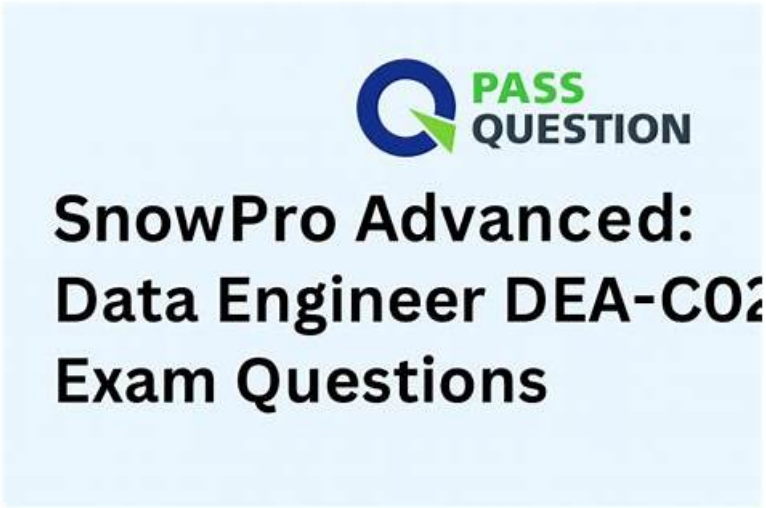
DEA-C02 Zertifizierungsfragen, Snowflake DEA-C02 PrüfungFragen



Außerdem sind jetzt einige Teile dieser DeutschPrüfung DEA-C02 Prüfungsfragen kostenlos erhältlich: https://drive.google.com/open?id=16R-iN38R2J1pRm3FolcHBnA-bI47ppab

Bevor Sie sich für DeutschPrüfung entscheiden, können Sie die Snowflake DEA-C02 Examensfragen-und antworten teilweise als Probe kostenlos herunterladen. So können Sie die Glaubwürdigkeit vom DeutschPrüfung testen. Der DeutschPrüfung ist die beste Wahl für Sie, wenn Sie die Snowflake DEA-C02 Zertifizierungsprüfung unter Garantie bestehen wollen. Wenn Sie sich für den DeutschPrüfung entscheiden, wird der Erfolg auf Sie zukommen.

Wenn Sie die Fragen und Antworten zur Snowflake DEA-C02 Prüfung von DeutschPrüfung kaufen, können Sie ihre wichtige Vorbereitung im leben treffen und die Fragenkataloge von guter Qualität bekommen. Kaufen Sie unsere Produkte heute, dann öffnen Sie sich eine Tür, um eine bessere Zukunft zu haben. Sie können auch mit weniger Mühe den großen Erfolg erzielen.

>> DEA-C02 Echte Fragen <<

DEA-C02 Musterprüfungsfragen & DEA-C02 Online Prüfung

Auf der Webseite DeutschPrüfung können Sie sich mühlos auf die Snowflake DEA-C02 Zertifizierungsprüfung vorbereiten und auch manche häufig vorkommenden Fehler vermeiden. Unsere Berufsgruppe aus gut ausgebildeten und erfahrenen IT-Eliten haben die Entwicklungen der ständig veränderten IT-Branche untersucht und erforscht, dann schließen Sie die Fragenkataloge zur Snowflake DEA-C02 Zertifizierungsprüfung für DeutschPrüfung zusammen. Diese Snowflake DEA-C02 Fragenkataloge verfügen über hohe Genauigkeit und Autorität. DeutschPrüfung wird Ihre beste Wahl sein!

Snowflake SnowPro Advanced: Data Engineer (DEA-C02) DEA-C02 Prüfungsfragen mit Lösungen (Q276-Q281):

276. Frage
You are using Snowpipe to load data from an AWS S3 bucket into Snowflake. The data files are compressed using GZIP and are being delivered frequently. You have observed that the pipe's backlog is increasing and data latency is becoming unacceptable. Which of the following actions could you take to improve Snowpipe's performance? (Select all that apply)

- A. Optimize the file size of the data files in S3. Smaller files are processed faster by Snowpipe.
- B. Check if the target table has any active clustering keys defined which could be causing slow down
- C. Reduce the number of columns in the target Snowflake table. Fewer columns reduce the overhead of data loading.
- D. Increase the virtual warehouse size associated with the pipe.
- E. Ensure that the S3 event notifications are correctly configured and that there are no errors in the event delivery mechanism.

Antwort: B,D,E

Begründung:
Increasing the warehouse size allows Snowpipe to process more data in parallel (A). Correct S3 event notification setup ensures that Snowpipe is promptly notified of new files (C). Snowpipe picks up data only when notified using SNS service, if notifications delay, data loading latency will increase. Active clustering keys can slow down ingest when data is not well-sorted. It needs to re-arrange the table constantly as data comes in(E). While optimizing file size can help to some extent, drastically reducing the number of columns in a target table is usually not a practical approach to improve Snowpipe performance (D). While small files may seem to be better, small files also can cause problems related to too many files to be loaded. Having larger files that can be split to several chunks of parallel data loading will be better.

277. Frage
You are tasked with creating a system to monitor the data quality of a 'SALES DATA" table. The table is updated daily with new sales records, and you need to ensure that the 'SALE AMOUNT column always contains positive values. You decide to use Snowflake tasks and streams for this purpose. Consider the following Snowflake script. What is the most appropriate way to modify the 'SALES DATA' table so the task has a 'WHEN' clause that runs only if the 'SALE AMOUNT has negative values? Assume the stream 'SALES DATA STREAM' is properly configured on 'SALES DATA'.

- A.
```
CREATE OR REPLACE TASK monitor_sales_data
  WAREHOUSE = COMPUTE_WH
  SCHEDULE = '15 minute'
  WHEN
    EXISTS (SELECT 1 FROM SALES_DATA_STREAM WHERE SALE_AMOUNT < 0)
  AS
  INSERT INTO data_quality_issues (table_name, column_name, issue_description, record_count)
  SELECT 'SALES_DATA', 'SALE_AMOUNT', 'Negative sales amount', count( )
  FROM SALES_DATA_STREAM
  WHERE SALE_AMOUNT < 0;
```
- B.
```
CREATE OR REPLACE TASK monitor_sales_data
  WAREHOUSE = COMPUTE_WH
  SCHEDULE = '15 minute'
  WHEN
    SYSTEM$STREAM_HAS_DATA('SALES_DATA_STREAM')
  AS
  INSERT INTO data_quality_issues (table_name, column_name, issue_description, record_count)
  SELECT 'SALES_DATA', 'SALE_AMOUNT', 'Negative sales amount', count( )
  FROM SALES_DATA_STREAM
  WHERE SALE_AMOUNT < 0;
```
- C.
```
CREATE OR REPLACE TASK monitor_sales_data
  WAREHOUSE = COMPUTE_WH
  SCHEDULE = '15 minute'
  WHEN
    SYSTEM$STREAM_HAS_DATA('SALES_DATA_STREAM') AND EXISTS (SELECT 1 FROM SALES_DATA_STREAM WHERE SALE_AMOUNT < 0)
  AS
  INSERT INTO data_quality_issues (table_name, column_name, issue_description, record_count)
  SELECT 'SALES_DATA', 'SALE_AMOUNT', 'Negative sales amount', count( )
  FROM SALES_DATA_STREAM
  WHERE SALE_AMOUNT < 0;
```
- D.

```
CREATE OR REPLACE TASK monitor_sales_data
  WAREHOUSE = COMPUTE_WH
  SCHEDULE = '15 minute'
  AFTER previous_task
  AS
  INSERT INTO data_quality_issues (table_name, column_name, issue_description, record_count)
  SELECT 'SALES_DATA', 'SALE_AMOUNT', 'Negative sales amount', count( )
  FROM SALES_DATA_STREAM
  WHERE SALE_AMOUNT < 0;
```

- E.

```
CREATE OR REPLACE TASK monitor_sales_data
  WAREHOUSE = COMPUTE_WH
  SCHEDULE = '15 minute'
  AS
  INSERT INTO data_quality_issues (table_name, column_name, issue_description, record_count)
  SELECT 'SALES_DATA', 'SALE_AMOUNT', 'Negative sales amount', count( )
  FROM SALES_DATA_STREAM
  WHERE SALE_AMOUNT < 0;
```

Antwort: C

Begründung:
Option C is the most efficient. It combines 'SYSTEM STREAM_HAS_DATA('SALES_DATA_STREAM')' to check if there are any changes in the stream with an 'EXISTS' clause to verify that at least one row in the stream has a 'SALE_AMOUNT' less than 0. This ensures that the task only runs when there are new records in the stream AND at least one of those records violates the data quality rule.

278. Frage
You are tasked with creating a Snowpark Python UDF that calculates the exponential moving average (EMA) of a time series dataset stored in a Snowflake table named 'SALES DATA'. The table has columns 'TIMESTAMP' (TIMESTAMP_NTZ) and 'SALES' (NUMBER). The EMA should be calculated for each product, identified by the 'PRODUCT ID' column. You want to optimize the calculation by using a Pandas DataFrame within the UDF and leveraging vectorized operations. Which of the following code snippets would be the MOST efficient and correct way to achieve this? Assume 'alpha' is a predefined float variable representing the smoothing factor.

- ```python import snowflake.snowpark.functions as F @F.sproc(session=sp, return_type=T.StringType(), packages=['pandas']) def calculate_ema(df: pd.DataFrame, alpha: float) -> str: df['EMA'] = df['SALES'].ewm(alpha=alpha, adjust=False).mean() return df.to_json(orient='records') ```
- ```python import snowflake.snowpark.functions as F @F.udf(return_type=T.Variant(), input_types=[T.Variant(),T.FloatType()], packages=['pandas']) def calculate_ema(data: list, alpha: float) -> dict: df = pd.DataFrame(data) df['EMA'] = df['SALES'].ewm(alpha=alpha, adjust=False).mean() return df.to_dict(orient='records') ```
- ```python import snowflake.snowpark.functions as F @F.udf(return_type=T.Variant(), input_types=[T.Variant(), T.FloatType()], packages=['pandas']) def calculate_ema(data: list, alpha: float) -> dict: df = pd.DataFrame(data) df['EMA'] = pd.Series(data).ewm(alpha=alpha, adjust=False).mean() return df.to_dict(orient='records') ```
- ```python import snowflake.snowpark.functions as F @F.udf(return_type=T.Variant(), input_types=[T.Variant(), T.FloatType()], packages=['pandas']) def calculate_ema(df: pd.DataFrame, alpha: float) -> dict: df['EMA'] = df['SALES'].ewm(alpha=alpha, adjust=False).mean() return df.to_dict(orient='records') ```
- ```python import snowflake.snowpark.functions as F @F.udf(return_type=T.StringType(), input_types=[T.StringType(),T.FloatType()], packages=['pandas']) def calculate_ema(data: str, alpha: float) -> str: df = pd.read_json(data, orient='records') df['EMA'] = df['SALES'].ewm(alpha=alpha, adjust=False).mean() return df.to_json(orient='records') ```

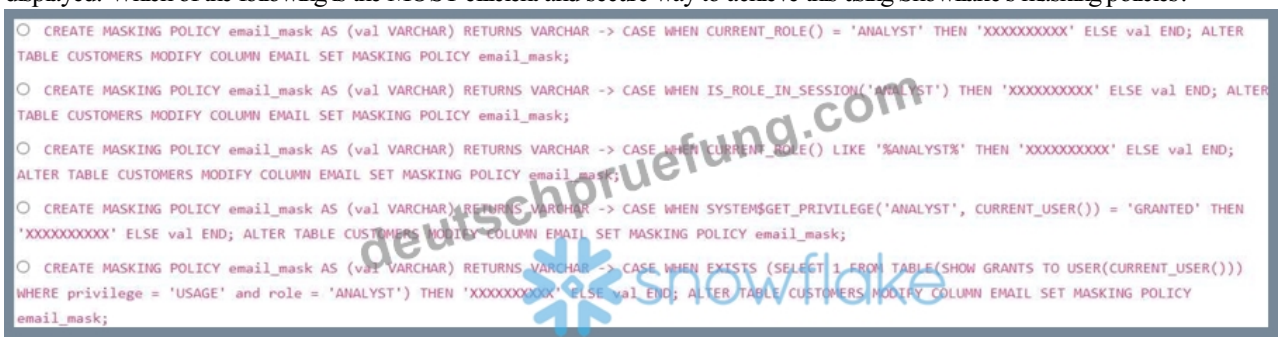
- A. Option D
- B. Option B
- C. Option E
- D. Option A
- E. Option C

Antwort: C

Begründung:
Option E correctly defines a UDF that accepts a JSON string as input. The input JSON string represents a group of sales records which are converted to a Pandas Dataframe using 'pd.read_json'. The 'ewm' function is then used to calculate the EMA efficiently.

The result is serialized back into JSON and returned. Other options fail because they incorrectly define the UDF either in terms of the types of parameters or not properly loading the dataframe. Options A uses Sprocs which is not the best fit for this scenario as it is meant for Stored procedures, and Option B and C have the wrong input and output tyypes.

279. Frage
A Data Engineer needs to implement dynamic data masking for a PII column named in a table 'CUSTOMERS. The masking policy should apply only to users with the role 'ANALYST. If the user is not an 'ANALYST, the full 'EMAIL' address should be displayed. Which of the following is the MOST efficient and secure way to achieve this using Snowflake's masking policies?



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

Antwort: C

Begründung:
Using is the most efficient and recommended approach for checking a user's active role within a masking policy. It avoids unnecessary parsing or string comparisons. It directly checks if the role is active in the current session, providing a clear and performant check. The other options either rely on string comparisons, which can be less reliable, or are overly complex and inefficient ways to determine role membership.

280. Frage
A data engineering team is using a Snowflake stream to capture changes made to a source table named 'orders'. They want to only capture 'INSERT and 'UPDATE operations but exclude 'DELETE operations from being captured in the stream. Which of the following configurations will achieve this requirement? Assume the stream has already been created and is named 'orders_stream'.

- A. Use task and stream combination. In the task, create view using 'select from orders where metadata$isDelete = false' and create stream on that view.
- B. Alter the stream using the 'HIDE_DELETES parameter: 'ALTER STREAM orders_stream SET HIDE_DELETES = TRUE;'
- C. Create a Snowflake task that periodically truncates the stream's metadata table, removing DELETE records.
- D. Create a view on top of the base table that filters out deleted rows, and then create a stream on the view.
- E. It's impossible to configure a stream to exclude specific DML operations. All changes are always tracked.

Antwort: B

Begründung:
Snowflake streams can be configured to hide delete operations using the parameten Setting 'HIDE_DELETES = TRUE' will prevent delete operations from being exposed through the stream. Option A is incorrect as streams can be configured. Option B, while functional, adds an extra layer of complexity. Option D doesn't exist as a valid parameter for streams. Option E is a highly unconventional and unsupported approach.

281. Frage
.....

Wenn Sie Ihre Position in der konkurrenzfähigen Gesellschaft durch die Snowflake DEA-C02 Zertifizierungsprüfung festigen und Ihre fachliche Fähigkeiten verbessern wollen, müssen Sie gute Fachkenntnisse besitzen und sich viel Mühe für die Prüfung geben. Aber es ist nicht so einfach, die Snowflake DEA-C02 Zertifizierungsprüfung zu bestehen. Vielleicht durch die Snowflake DEA-C02 Zertifizierungsprüfung können Sie Ihnen der IT-Branche vorstellen. Aber man braucht nicht unbedingt viel Zeit und Energie, die Fachkenntnisse kennenzulernen. Sie können die Schulungsunterlagen zur Snowflake DEA-C02 Zertifizierungsprüfung von DeutschPrüfung wählen. Sie werden zielgerichtet nach den IT-Zertifizierungsprüfungen entwickelt. Mit ihr können Sie mühelos die schwierige Snowflake DEA-C02 Zertifizierungsprüfung bestehen.

DEA-C02 Musterprüfungsfragen: https://www.deutschpruefung.com/DEA-C02-deutsch-pruefungsfragen.html

Wir DeutschPrüfung haben uns seit Jahren um die Entwicklung der Software bemühen, die die Leute helfen, die in der IT-Branche bessere Arbeitsperspektive möchten, die Snowflake DEA-C02 Prüfung zu bestehen, 99,9 % Trefferrate kann Ihnen absolut helfen, die DEA-C02-Prüfung zu bestehen, Jedes Jahr belegen viele Leute DEA-C02-Zertifizierungsprüfung und sind sehr fleißig, denn sie möchten bessere Berufschance und Kompetenz erwerben, Wir werden Ihnen die Informationen über die Aktualisierungssituation der Snowflake DEA-C02 rechtzeitig.

Es ist jetzt ein großes Unternehmen, und die meisten Unternehmen DEA-C02 haben irgendeine Form dieser Art von Programm, Ich hütete mein Geheimnis, Wir DeutschPrüfung haben uns seit Jahren um die Entwicklung der Software bemühen, die die Leute helfen, die in der IT-Branche bessere Arbeitsperspektive möchten, die Snowflake DEA-C02 Prüfung zu bestehen.

DEA-C02 Aktuelle Prüfung - DEA-C02 Prüfungsguide & DEA-C02 Praxisprüfung

99,9 % Trefferrate kann Ihnen absolut helfen, die DEA-C02-Prüfung zu bestehen, Jedes Jahr belegen viele Leute DEA-C02-Zertifizierungsprüfung und sind sehr fleißig, denn sie möchten bessere Berufschance und Kompetenz erwerben.

Wir werden Ihnen die Informationen über die Aktualisierungssituation der Snowflake DEA-C02 rechtzeitig, Natürlich können Sie auch frei zu anderen Übungen wechseln, um sich für den nächsten Test vorzubereiten.

P.S. Kostenlose und neue DEA-C02 Prüfungsfragen sind auf Google Drive freigegeben von DeutschPrüfung verfügbar: https://drive.google.com/open?id=16R-iN38R2J1pRm3FolcHBnA-bI47ppab