

PRACTICE QUESTIONS

Snowflake

SnowPro Advanced: MLOps Engineer

SHARED BY [u/patient_kiwi_67](#) on aiexamcerts

EDITION Free Set 2

QUESTIONS 10

SHARED October 15, 2026

EXAM Snowflake SnowPro Advanced: MLOps Engineer

Shared by [u/patient_kiwi_67](#) on aiexamcerts

10 practice questions for the Snowflake SnowPro Advanced: MLOps Engineer exam, each followed by its answer and a short explanation. More free sets and the full practice exam:

www.aiexamcerts.com/snowflake/snowpro-advanced-mlops-engineer. Spotted a mistake? Email support@aiexamcerts.com with the exam name and the question number.

1. Which Snowpark DataFrame approach keeps computation in Snowflake rather than pulling rows to the client?
 - A Iterating over each row in a Python loop running on the client machine, one record at a time
 - B Lazy DataFrame transformations that are pushed down and executed in Snowflake
 - C Writing every row to a local CSV file before the next transformation runs
 - D Converting the DataFrame to a pandas object with `to_pandas` before every step

Answer: B

Snowpark DataFrames build a query plan that Snowflake executes when an action runs, so the work stays on the warehouse. Calling `to_pandas` early downloads the rows to the client, which is the tempting shortcut for a Python developer. Row loops and local CSV files also move the data out of Snowflake.

2. Snowflake Time Travel retention is fixed at one day for every table in every Snowflake edition.
 - A True
 - B False

Answer: B (False)

Retention for Time Travel is configurable per table and per database or schema through the `DATA_RETENTION_TIME_IN_DAYS` parameter, within limits that depend on the edition. A fixed one-day rule would make retention planning impossible. The statement is false.

3. What does a feature view define in Snowflake Feature Store?
 - A A saved worksheet used to browse the results of a query, shared with other analysts in the account
 - B A set of features computed from a source query, together with its refresh behavior
 - C A full physical copy of the source table at the time of creation
 - D A role that grants read access to a group of feature tables

Answer: B

A feature view groups feature definitions with the query behind them and how often they refresh, which is what makes features reusable. A worksheet is a notebook-style query surface, a role controls access and a full copy is not required by the definition. Feature views are logical definitions over source data.

4. The data split used to tune a model and compare candidate models, kept separate from the final holdout set, is called the _____ set.

Answer: validation (also accepted: validation set)

The validation set guides model selection while the final test or holdout set gives an unbiased estimate. Confusing it with the test set is a common cause of over-optimistic reporting.

5. Which two practices reduce the risk of data leakage between training and evaluation? (Choose two.)

- A Fit preprocessing steps inside the training fold only
- B Shuffle the test rows after every training epoch is complete
- C Use the test set column names as additional model features
- D Copy the test set into the training table for the final model
- E Split by entity or time before any fitting of transformers

Answer: A, E

Splitting first and fitting preprocessing only on training data keeps evaluation data out of every learned step. Shuffling test rows changes nothing about leakage, and copying the test set into training or using test-set columns as features both leak evaluation data directly.

6. Why do teams use a feature store rather than recomputing features inside each training notebook?

- A To remove the need for any data quality tests on the source tables
- B To run every training job on CPU instances rather than on GPUs, which keeps the cost of each run low
- C To share one tested definition of each feature between training and inference
- D To keep raw source tables in a compressed format for storage savings

Answer: C

A shared feature definition reduces duplicated logic and the training-serving skew that comes from computing the same feature two different ways. Compression is a storage matter, and a feature store does not remove the need for quality tests. Choice of CPU or GPU is an infrastructure decision unrelated to feature reuse.

7. A new model version expects a feature that was renamed in production. What is the most likely failure?
- A Training speeds up because the feature count falls after the rename, which reduces memory use
 - B Inference fails or silently fills nulls because the input schema no longer matches
 - C The warehouse renames the feature back to its original name in storage
 - D The registry deletes the model automatically once the rename happens

Answer: B

A schema mismatch is a common deployment failure, and it can surface as an error or as null-filled inputs that quietly degrade predictions. A smaller feature count does not speed training in any way that matters here. The registry does not delete models, and warehouses do not rename columns on their own.

8. Rolling back a deployment usually means pointing traffic back to a previous model version still available in the registry.
- A True
 - B False

Answer: A (True)

Keeping previous versions available lets a team restore a known-good model quickly by changing the reference used by the serving layer. Deleting old versions removes that option. The statement is true.

9. What does data drift mean for a deployed model?
- A A change in the distribution of input features compared with the data used for training
 - B A change in the model's source code between two releases, including its test and feature code
 - C A drop in warehouse credit consumption over a long period
 - D A decrease in the number of tables that a schema contains

Answer: A

Data drift means the inputs now look different from what the model learned from, which can degrade predictions even when the model code is unchanged. Code changes, credit trends and table counts are operational facts and do not describe a change in input distributions.

10. Predictions look stable, but business results have worsened. Which conclusion is best supported?
- A The model is working correctly, so the problem must lie in the business process
 - B The input-outcome relationship may have changed, so the model needs review against recent labels
 - C The model weights have been corrupted in storage and need restoring
 - D The warehouse needs more memory so that predictions finish sooner, which would explain the drop in outcomes

Answer: B

Stable predictions alongside worse outcomes suggests concept drift or a business change that the model no longer reflects, which requires comparing predictions with recent labels. Assuming the model is fine skips the check, and corrupted weights or memory limits would show up as errors or changed predictions rather than steady ones.