

PRACTICE QUESTIONS

Databricks Certified Data Engineer Associate

SHARED BY [u/Placid-Pixel-7132](#) on aiexamcerts

EDITION Free Set 1

QUESTIONS 10

SHARED October 15, 2026

EXAM Databricks Certified Data Engineer Associate

Shared by [u/Placid-Pixel-7132](#) on aiexamcerts

10 practice questions for the Databricks Certified Data Engineer Associate exam, each followed by its answer and a short explanation. More free sets and the full practice exam:

www.aiexamcerts.com/databricks/databricks-certified-data-engineer-associate. Spotted a mistake? Email support@aiexamcerts.com with the exam name and the question number.

1. A team wants to run SQL queries on a small analytics warehouse with minimal administration, while data engineers keep using notebooks on job compute. Which pairing fits best?

- A One all-purpose cluster shared by every user and left running all day
- B A SQL warehouse for analytics plus job compute for scheduled pipelines
- C Job compute for analyst SQL queries plus a SQL warehouse for scheduled pipelines
- D An all-purpose cluster per analyst with auto termination disabled

Answer: B

A Databricks SQL warehouse is built for SQL analytics with managed scaling, and job compute runs scheduled pipelines and is cheaper than all-purpose clusters. Keeping one all-purpose cluster running all day wastes money. Giving every analyst an always-on cluster costs more, and swapping the two roles puts each workload on compute that is not designed for it.

2. Unity Catalog uses a three-level namespace of catalog, schema and table, so a table is referenced as catalog. _____ .table.

Answer: schema

The full name is catalog.schema.table, and the middle level is the schema, which groups related tables inside a catalog. Using the wrong level breaks name resolution and access grants. The fill-in names the middle level of the namespace.

3. A table is loaded from a folder of Parquet files, and the same command is run again by mistake. The team needs the rerun to load no duplicate files. Which command provides this behavior?

- A COPY INTO, which skips files that were already loaded
- B INSERT INTO ... SELECT, which appends every file again on each run
- C CREATE TABLE AS SELECT, which recreates the table each run
- D OPTIMIZE, which compacts the small files already in the table

Answer: A

COPY INTO tracks which source files it has loaded, so re-running it skips files that were already ingested. A plain INSERT from the folder appends the same files again. CREATE TABLE AS SELECT replaces the table, and OPTIMIZE only compacts existing files without loading anything.

4. Auto Loader cannot record an inferred schema, so every run must supply the full schema manually.
- A True
 - B False

Answer: B (False)

Auto Loader can infer the schema and store it in the schema location, which lets later runs reuse it and track changes. Supplying a schema is optional rather than required. The statement is false.

5. Which TWO statements about Delta Live Tables expectations are correct? (Choose two.)
- A An expectation can drop rows that violate a condition and record the count in pipeline metrics
 - B Expectations are checked only once, when the pipeline is first created
 - C An expectation can fail the update when a row violates it
 - D Expectations can only be defined in Scala
 - E Expectations change the source data files on storage

Answer: A, C

Expectations can warn, drop violating rows, or fail the update, and the pipeline metrics record how many rows violated each rule. They are evaluated on every update rather than once at creation. They can be written in SQL or Python and do not modify the source files.

6. A transformation filters rows and selects columns without moving data between executors. Which category does it belong to in Spark?
- A A wide transformation that triggers a shuffle
 - B A narrow transformation
 - C An action that returns results to the driver
 - D A checkpoint operation

Answer: B

Narrow transformations such as filter and select work on each partition independently, so no shuffle is needed. Wide transformations such as groupBy and join move data across partitions. Actions such as count trigger execution, and checkpoints save state.

7. A scheduled job failed in its third task after the first two succeeded. The team wants to rerun only the failed and downstream tasks. Which feature does this?
- A Repair run
 - B Run now with the default parameters
 - C Clone the job and run the copy
 - D Restart the job cluster

Answer: A

A repair run reruns only the failed task and the tasks that depend on it, which saves time and avoids repeating completed work. Run now starts every task again from the beginning, and cloning the job and running the copy does the same. Restarting the cluster does not rerun any tasks.

8. A job that a team wants to launch on demand with parameters from an external system is usually exposed through the Jobs REST API, using a _____ request to run it.

Answer: POST

Starting a job run is a POST request to the Jobs API run-now endpoint, which accepts parameters in the body. Read-only status checks use GET requests. The fill-in names the HTTP method for starting a run.

9. An auditor asks which upstream tables and notebooks feed a gold revenue table. Which Unity Catalog capability answers this?
- A A table comment
 - B Data lineage
 - C A dashboard refresh schedule
 - D A Delta VACUUM history

Answer: B

Unity Catalog data lineage captures how tables, columns and notebooks or jobs relate, so the auditor can trace upstream sources. Comments describe a table but do not record lineage automatically. Dashboard schedules and VACUUM history do not show dependencies.

10. A team must share a Unity Catalog table with an external partner who does not have a Databricks account. Which feature supports this?
- A Lakehouse Federation
 - B Partner Connect
 - C Delta Sharing
 - D Databricks Connect

Answer: C

Delta Sharing provides governed, read-only access to shared tables for recipients outside the organization, including those without a Databricks account, with the provider controlling what is shared. Lakehouse Federation queries external databases from Databricks, and Partner Connect links Databricks to partner tools. Databricks Connect lets IDEs run Spark code remotely, so none of them shares tables.