



VCE & PDF

PassApply.com

<https://www.passapply.com/databricks-certified-professional-data-engineer>.

2024 Latest passapply DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER PDF and VCE dumps Download

DATABRICKS-CERTIFIED- PR OFESIONAL-DATA-ENGINEER^{Q&As}

Databricks Certified Professional Data Engineer Exam

**Pass Databricks DATABRICKS-CERTIFIED-
PROFESSIONAL-DATA-ENGINEER Exam with 100%
Guarantee**

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

<https://www.passapply.com/databricks-certified-professional-data-engineer.html>

100% Passing Guarantee
100% Money Back Assurance

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



VCE & PDF

PassApply.com

<https://www.passapply.com/databricks-certified-professional-data-engineer.>

2024 Latest passapply DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER PDF and VCE dumps Download

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





QUESTION 1

A Databricks job has been configured with 3 tasks, each of which is a Databricks notebook. Task A does not depend on other tasks. Tasks B and C run in parallel, with each having a serial dependency on task A.

If tasks A and B complete successfully but task C fails during a scheduled run, which statement describes the resulting state?

- A. All logic expressed in the notebook associated with tasks A and B will have been successfully completed; some operations in task C may have completed successfully.
- B. All logic expressed in the notebook associated with tasks A and B will have been successfully completed; any changes made in task C will be rolled back due to task failure.
- C. All logic expressed in the notebook associated with task A will have been successfully completed; tasks B and C will not commit any changes because of stage failure.
- D. Because all tasks are managed as a dependency graph, no changes will be committed to the Lakehouse until all tasks have successfully been completed.
- E. Unless all tasks complete successfully, no changes will be committed to the Lakehouse; because task C failed, all commits will be rolled back automatically.

Correct Answer: A

The query uses the CREATE TABLE USING DELTA syntax to create a Delta Lake table from an existing Parquet file stored in DBFS. The query also uses the LOCATION keyword to specify the path to the Parquet file as /mnt/finance_eda_bucket/tx_sales.parquet. By using the LOCATION keyword, the query creates an external table, which is a table that is stored outside of the default warehouse directory and whose metadata is not managed by Databricks. An external table can be created from an existing directory in a cloud storage system, such as DBFS or S3, that contains data files in a supported format, such as Parquet or CSV. The resulting state after running the second command is that an external table will be created in the storage container mounted to /mnt/finance_eda_bucket with the new name prod.sales_by_store. The command will not change any data or move any files in the storage container; it will only update the table reference in the metastore and create a new Delta transaction log for the renamed table. Verified References: [Databricks Certified Data Engineer Professional], under "Delta Lake" section; Databricks Documentation, under "ALTER TABLE RENAME TO" section; Databricks Documentation, under "Create an external table" section.

QUESTION 2

The data science team has created and logged a production model using MLflow. The following code correctly imports and applies the production model to output the predictions as a new DataFrame named preds with the schema "customer_id LONG, predictions DOUBLE, date DATE".



```
from pyspark.sql.functions import current_date

model = mlflow.pyfunc.spark_udf(spark, model_uri="models:/churn/prod")
df = spark.table("customers")
columns = ["account_age", "time_since_last_seen", "app_rating"]
preds = (df.select(
    "customer_id",
    model(*columns).alias("predictions"),
    current_date().alias("date")
))
```

The data science team would like predictions saved to a Delta Lake table with the ability to compare all predictions across time. Churn predictions will be made at most once per day. Which code block accomplishes this task while minimizing potential compute costs?

- ☐ A. `preds.write.mode("append").saveAsTable("churn_preds")`
- ☐ B. `preds.write.format("delta").save("/preds/churn_preds")`
- ☐ C.

```
(preds.writeStream
    .outputMode("overwrite")
    .option("checkpointPath", "/_checkpoints/churn_preds")
    .start("/preds/churn_preds")
)
```
- ☐ D.

```
(preds.write
    .format("delta")
    .mode("overwrite")
    .saveAsTable("churn_preds")
)
```
- ☐ E.

```
(preds.writeStream
    .outputMode("append")
    .option("checkpointPath", "/_checkpoints/churn_preds")
    .table("churn_preds")
)
```

A. Option A

B. Option B

C. Option C

D. Option D

E. Option E

Correct Answer: A



QUESTION 3

Which statement describes Delta Lake Auto Compaction?

- A. An asynchronous job runs after the write completes to detect if files could be further compacted; if yes, an optimize job is executed toward a default of 1 GB.
- B. Before a Jobs cluster terminates, optimize is executed on all tables modified during the most recent job.
- C. Optimized writes use logical partitions instead of directory partitions; because partition boundaries are only represented in metadata, fewer small files are written.
- D. Data is queued in a messaging bus instead of committing data directly to memory; all data is committed from the messaging bus in one batch once the job is complete.
- E. An asynchronous job runs after the write completes to detect if files could be further compacted; if yes, an optimize job is executed toward a default of 128 MB.

Correct Answer: E

This is the correct answer because it describes the behavior of Delta Lake Auto Compaction, which is a feature that automatically optimizes the layout of Delta Lake tables by coalescing small files into larger ones. Auto Compaction runs as an asynchronous job after a write to a table has succeeded and checks if files within a partition can be further compacted. If yes, it runs an optimize job with a default target file size of 128 MB. Auto Compaction only compacts files that have not been compacted previously. Verified References: [Databricks Certified Data Engineer Professional], under "Delta Lake" section; Databricks Documentation, under "Auto Compaction for Delta Lake on Databricks" section. "Auto compaction occurs after a write to a table has succeeded and runs synchronously on the cluster that has performed the write. Auto compaction only compacts files that haven't been compacted previously." <https://learn.microsoft.com/en-us/azure/databricks/delta/tune-file-size>

QUESTION 4

The data governance team is reviewing code used for deleting records for compliance with GDPR. They note the following logic is used to delete records from the Delta Lake table named users.

```
DELETE FROM users
WHERE user_id IN
(SELECT user_id FROM delete_requests)
```

Assuming that user_id is a unique identifying key and that delete_requests contains all users that have requested deletion, which statement describes whether successfully executing the above logic guarantees that the records to be deleted are no longer accessible and why?

- A. Yes; Delta Lake ACID guarantees provide assurance that the delete command succeeded fully and permanently purged these records.
- B. No; the Delta cache may return records from previous versions of the table until the cluster is restarted.



- C. Yes; the Delta cache immediately updates to reflect the latest data files recorded to disk.
- D. No; the Delta Lake delete command only provides ACID guarantees when combined with the merge into command.
- E. No; files containing deleted records may still be accessible with time travel until a vacuum command is used to remove invalidated data files.

Correct Answer: E

The code uses the DELETE FROM command to delete records from the users table that match a condition based on a join with another table called delete_requests, which contains all users that have requested deletion. The DELETE FROM command deletes records from a Delta Lake table by creating a new version of the table that does not contain the deleted records. However, this does not guarantee that the records to be deleted are no longer accessible, because Delta Lake supports time travel, which allows querying previous versions of the table using a timestamp or version number. Therefore, files containing deleted records may still be accessible with time travel until a vacuum command is used to remove invalidated data files from physical storage. Verified References: [Databricks Certified Data Engineer Professional], under "Delta Lake" section; Databricks Documentation, under "Delete from a table" section; Databricks Documentation, under "Remove files no longer referenced by a Delta table" section.

QUESTION 5

Which distribution does Databricks support for installing custom Python code packages?

- A. sbt
- B. CRAN
- C. CRAM
- D. nom
- E. Wheels
- F. jars

Correct Answer: D

[Latest DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER Dumps](#)

[DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER Exam Questions](#)

[DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER Braindumps](#)