DATABRICKS-CERTIFIED-PR OFESSIONAL-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

QUESTION 1

A Delta Lake table was created with the below query:

Realizing that the original query had a typographical error, the below code was executed:

ALTER TABLE prod.sales_by_stor RENAME TO prod.sales_by_store

Which result will occur after running the second command?

A. The table reference in the metastore is updated and no data is changed.

B. The table name change is recorded in the Delta transaction log.

C. All related files and metadata are dropped and recreated in a single ACID transaction.

D. The table reference in the metastore is updated and all data files are moved.

E. A new Delta transaction log Is created for the renamed table.

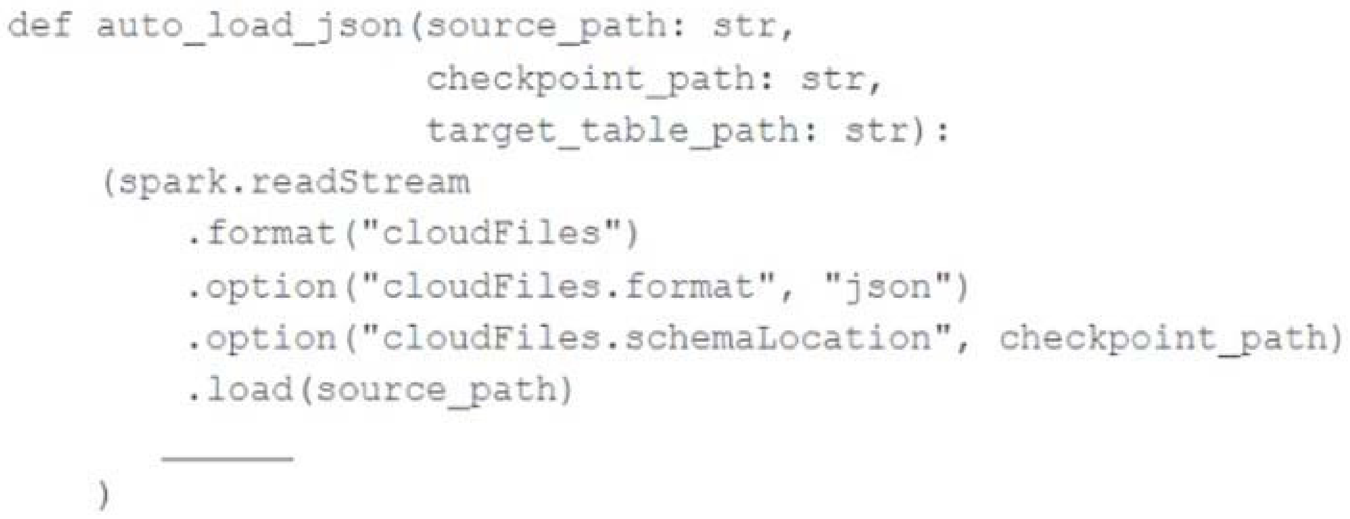
Correct Answer: A

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 result that will occur after running the second command is that the table reference in the metastore is updated and no data is changed. The metastore is a service that stores metadata about tables, such as their schema, location, properties, and partitions. The metastore allows users to access tables using SQL commands or Spark APIs without knowing their physical location or format. When renaming an external table using the ALTER TABLE RENAME TO command, only the table reference in the metastore is updated with the new name; no data files or directories are moved or changed in the storage system. The table will still point to the same location and use the same format as before. However, if renaming a managed table, which is a table whose metadata and data are both managed by Databricks, both the table reference in the metastore and the data files in the default warehouse directory are moved and renamed accordingly. Verified References: [Databricks Certified Data Engineer Professional], under "Delta Lake" section; Databricks Documentation, under "ALTER TABLE RENAME TO" section; Databricks Documentation, under "Metastore" section; Databricks Documentation, under "Managed and external tables" section.

QUESTION 2

In order to facilitate near real-time workloads, a data engineer is creating a helper function to leverage the schema detection and evolution functionality of Databricks Auto Loader. The desired function will automatically detect the schema of the source directly, incrementally process JSON files as they arrive in a source directory, and automatically evolve the schema of the table when new fields are detected.

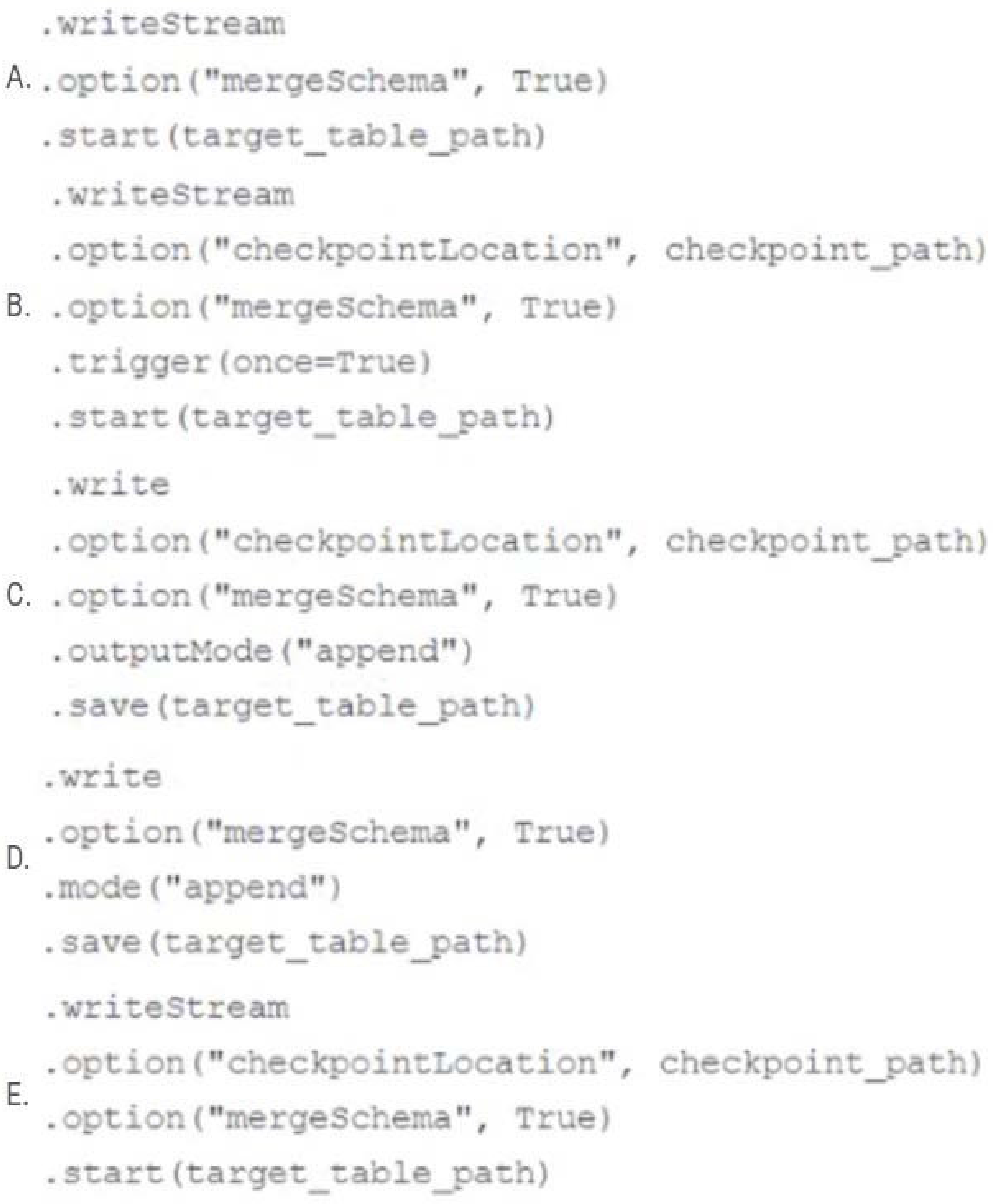
The function is displayed below with a blank:



```
def auto_load_json(source_path: str,
                   checkpoint_path: str,
                   target_table_path: str):
    (spark.readStream
        .format("cloudFiles")
        .option("cloudFiles.format", "json")
        .option("cloudFiles.schemaLocation", checkpoint_path)
        .load(source_path)

       ______
    )
```

Which response correctly fills in the blank to meet the specified requirements?



A.
```
.writeStream
.option("mergeSchema", True)
.start(target_table_path)
```

B.
```
.writeStream
.option("checkpointLocation", checkpoint_path)
.option("mergeSchema", True)
.trigger(once=True)
.start(target_table_path)
```

C.
```
.write
.option("checkpointLocation", checkpoint_path)
.option("mergeSchema", True)
.outputMode("append")
.save(target_table_path)
```

D.
```
.write
.option("mergeSchema", True)
.mode("append")
.save(target_table_path)
```

E.
```
.writeStream
.option("checkpointLocation", checkpoint_path)
.option("mergeSchema", True)
.start(target_table_path)
```

A. Option A

B. Option B

C. Option C

D. Option D

E. Option E

Correct Answer: B

Option B correctly fills in the blank to meet the specified requirements. Option B uses the "cloudFiles.schemaLocation" option, which is required for the schema detection and evolution functionality of Databricks Auto Loader. Additionally,

option B uses the "mergeSchema" option, which is required for the schema evolution functionality of Databricks Auto Loader. Finally, option B uses the "writeStream" method, which is required for the incremental processing of JSON files as

they arrive in a source directory. The other options are incorrect because they either omit the required options, use the wrong method, or use the wrong format. References:

Configure schema inference and evolution in Auto Loader:

https://docs.databricks.com/en/ingestion/auto-loader/schema.html Write streaming data: https://docs.databricks.com/spark/latest/structured-streaming/writing-streaming-data.html

QUESTION 3

The data architect has decided that once data has been ingested from external sources into the

Databricks Lakehouse, table access controls will be leveraged to manage permissions for all production tables and views.

The following logic was executed to grant privileges for interactive queries on a production database to the core engineering group.

GRANT USAGE ON DATABASE prod TO eng;

GRANT SELECT ON DATABASE prod TO eng;

Assuming these are the only privileges that have been granted to the eng group and that these users are not workspace administrators, which statement describes their privileges?

A. Group members have full permissions on the prod database and can also assign permissions to other users or groups.

B. Group members are able to list all tables in the prod database but are not able to see the results of any queries on those tables.

C. Group members are able to query and modify all tables and views in the prod database, but cannot create new tables or views.

D. Group members are able to query all tables and views in the prod database, but cannot create or edit anything in the database.

E. Group members are able to create, query, and modify all tables and views in the prod database, but cannot define custom functions.

Correct Answer: D

The GRANT USAGE ON DATABASE prod TO eng command grants the eng group the permission to use the prod database, which means they can list and access the tables and views in the database. The GRANT SELECT ON DATABASE prod TO eng command grants the eng group the permission to select data from the tables and views in the prod database, which means they can query the data using SQL or DataFrame API. However, these commands do not grant the eng group any other permissions, such as creating, modifying, or deleting tables and views, or defining custom functions. Therefore, the eng group members are able to query all tables and views in the prod database, but cannot create or edit anything in the database. References: Grant privileges on a database: https://docs.databricks.com/en/security/auth-authz/table-acls/grant-privileges-database.html Privileges you can grant on Hive metastore objects: https://docs.databricks.com/en/security/auth-authz/table-acls/privileges.html

QUESTION 4

A junior data engineer is working to implement logic for a Lakehouse table named silver_device_recordings. The source data contains 100 unique fields in a highly nested JSON structure.

The silver_device_recordings table will be used downstream to power several production monitoring dashboards and a production model. At present, 45 of the 100 fields are being used in at least one of these applications.

The data engineer is trying to determine the best approach for dealing with schema declaration given the highly-nested structure of the data and the numerous fields.

Which of the following accurately presents information about Delta Lake and Databricks that may impact their decision-making process?

A. The Tungsten encoding used by Databricks is optimized for storing string data; newly-added native support for querying JSON strings means that string types are always most efficient.

B. Because Delta Lake uses Parquet for data storage, data types can be easily evolved by just modifying file footer information in place.

C. Human labor in writing code is the largest cost associated with data engineering workloads; as such, automating table declaration logic should be a priority in all migration workloads.

D. Because Databricks will infer schema using types that allow all observed data to be processed, setting types manually provides greater assurance of data quality enforcement.

E. Schema inference and evolution on .Databricks ensure that inferred types will always accurately match the data types used by downstream systems.

Correct Answer: D

This is the correct answer because it accurately presents information about Delta Lake and Databricks that may impact the decision-making process of a junior data engineer who is trying to determine the best approach for dealing with schema declaration given the highly-nested structure of the data and the numerous fields. Delta Lake and Databricks support schema inference and evolution, which means that they can automatically infer the schema of a table from the source data and allow adding new columns or changing column types without affecting existing queries or pipelines. However, schema inference and evolution may not always be desirable or reliable, especially when dealing with complex or nested data structures or when enforcing data quality and consistency across different systems. Therefore, setting types manually can provide greater assurance of data quality enforcement and avoid potential errors or conflicts due to incompatible or unexpected data types. Verified References: [Databricks Certified Data Engineer Professional], under "Delta Lake" section; Databricks Documentation, under "Schema inference and partition of streaming DataFrames/ Datasets" section.

QUESTION 5

The data engineering team is migrating an enterprise system with thousands of tables and views into the Lakehouse. They plan to implement the target architecture using a series of bronze, silver, and gold tables. Bronze tables will almost exclusively be used by production data engineering workloads, while silver tables will be used to support both data engineering and machine learning workloads. Gold tables will largely serve business intelligence and reporting purposes. While personal identifying information (PII) exists in all tiers of data, pseudonymization and anonymization rules are in place for all data at the silver and gold levels.

The organization is interested in reducing security concerns while maximizing the ability to collaborate across diverse teams.

Which statement exemplifies best practices for implementing this system?

A. Isolating tables in separate databases based on data quality tiers allows for easy permissions management through database ACLs and allows physical separation of default storage locations for managed tables.

B. Because databases on Databricks are merely a logical construct, choices around database organization do not impact security or discoverability in the Lakehouse.

C. Storinq all production tables in a single database provides a unified view of all data assets available throughout the Lakehouse, simplifying discoverability by granting all users view privileges on this database.

D. Working in the default Databricks database provides the greatest security when working with managed tables, as these will be created in the DBFS root.

E. Because all tables must live in the same storage containers used for the database they\\'re created in, organizations should be prepared to create between dozens and thousands of databases depending on their data isolation requirements.

Correct Answer: A

This is the correct answer because it exemplifies best practices for implementing this system. By isolating tables in separate databases based on data quality tiers, such as bronze, silver, and gold, the data engineering team can achieve several benefits. First, they can easily manage permissions for different users and groups through database ACLs, which allow granting or revoking access to databases, tables, or views. Second, they can physically separate the default storage locations for managed tables in each database, which can improve performance and reduce costs. Third, they can provide a clear and consistent naming convention for the tables in each database, which can improve discoverability and usability. Verified References: [Databricks Certified Data Engineer Professional], under "Lakehouse" section; Databricks Documentation, under "Database object privileges" section.