DUMPSBOSS.

Databricks Certified Data Engineer Associate Exam

Databricks Databricks-Certified-Data-Engineer-Associate

Version Demo

Total Demo Questions: 7

Total Premium Questions: 109

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

QUESTION NO: 1

A data engineer needs to determine whether to use the built-in Databricks Notebooks versioning or version their project using Databricks Repos.

Which of the following is an advantage of using Databricks Repos over the Databricks Notebooks versioning?

A. Databricks Repos automatically saves development progress

B. Databricks Repos supports the use of multiple branches

C. Databricks Repos allows users to revert to previous versions of a notebook

D. Databricks Repos provides the ability to comment on specific changes

E. Databricks Repos is wholly housed within the Databricks Lakehouse Platform

ANSWER: B

Explanation:

Databricks Repos is a visual Git client and API in Databricks that supports common Git operations such as cloning, committing, pushing, pulling, and branch management. Databricks Notebooks versioning is a legacy feature that allows users to link notebooks to GitHub repositories and perform basic Git operations. However, Databricks Notebooks versioning does not support the use of multiple branches for development work, which is an advantage of using Databricks Repos. With Databricks Repos, users can create and manage branches for different features, experiments, or bug fixes, and merge, rebase, or resolve conflicts between them. Databricks recommends using a separate branch for each notebook and following data science and engineering code development best practices using Git for version control, collaboration, and CI/CD. Reference: Git integration with Databricks

Repos - Azure Databricks | Microsoft Learn, Git version control for notebooks (legacy) | Databricks on AWS, Databricks Repos Is Now Generally Available - New ˜Files' Feature in ¦, Databricks Repos - What it is and how we can use it | Adatis.

QUESTION NO: 2

Which two components function in the DB platform architecture's control plane? (Choose two.)

A. Virtual Machines

B. Compute Orchestration

C. Serverless Compute

D. Compute

E. Unity Catalog

ANSWER: B E

Explanation:

QUESTION NO: 3

A data engineer is attempting to drop a Spark SQL table my_table. The data engineer wants to delete all table metadata and data.

They run the following command:

DROP TABLE IF EXISTS my_table

While the object no longer appears when they run SHOW TABLES, the data files still exist.

Which of the following describes why the data files still exist and the metadata files were deleted?

A. The table's data was larger than 10 GB

B. The table's data was smaller than 10 GB

C. The table was external

D. The table did not have a location

E. The table was managed

ANSWER: C

Explanation:

An external table is a table that is defined in the metastore and points to an existing location in the storage system. When you drop an external table, only the metadata is deleted from the metastore, but the data files are not deleted from the storage system. This is because external tables are meant to be shared by multiple applications and users, and dropping them should not affect the data availability. On the other hand, a managed table is a table that is defined in the metastore and also managed by the metastore. When you drop a managed table, both the metadata and the data files are deleted from the metastore and the storage system, respectively. This is because managed tables are meant to be exclusive to the application or user that created them, and dropping them should free up the storage space. Therefore, the correct answer is C, because the table was external and only the metadata was deleted when the table was dropped. Reference: Databricks Documentation -

Managed and External Tables, Databricks Documentation - Drop Table

---

QUESTION NO: 4

The Delta transaction log for the ˜students' tables is shown using the ˜DESCRIBE HISTORY students' command. A Data Engineer needs to query the table as it existed before the UPDATE operation listed in the log.

	version	timestamp	operation
1	8	2024-04-22T14:33:31.000	OPTIMIZE
2	7	2024-04-22T14:33:16.000	MERGE
3	6	2024-04-22T14:33:06.000	DELETE
4	5	2024-04-22T14:32:58.000	UPDATE
5	4	2024-04-22T14:32:47.000	WRITE
6	3	2024-04-22T14:32:44.000	WRITE
7	2	2024-04-22T14:32:23.000	WRITE
8	1	2024-04-22T14:32:20.000	WRITE
9	0	2024-04-22T14:31:49.000	CREATE TABLE

Which command should the Data Engineer use to achieve this? (Choose two.)

A. SELECT * FROM students@v4

B. SELECT * FROM students TIMESTAMP AS OF ˜2024-04-22T 14:32:47.000+00:00'

C. SELECT * FROM students FROM HISTORY VERSION AS OF 3

D. SELECT * FROM students VERSION AS OF 5

E. SELECT * FROM students TIMESTAMP AS OF ˜2024-04-22T 14:32:58.000+00:00'

ANSWER: A B

Explanation:

QUESTION NO: 5

Which of the following Structured Streaming queries is performing a hop from a Silver table to a Gold table?

A)

```
(spark.readStream.load(rawSalesLocation)
    .writeStream
    .option("checkpointLocation", checkpointPath)
    .outputMode("append")
    .table("newSales")
)
```

B)

```
(spark.read.load(rawSalesLocation)
    .writeStream
    .option("checkpointLocation", checkpointPath)
    .outputMode("append")
    .table("newSales")
)
```

C)

```
(spark.table("sales")
    .withColumn("avgPrice", col("sales") / col("units"))
    .writeStream
    .option("checkpointLocation", checkpointPath)
    .outputMode("append")
    .table("newSales")
)
```

D)

```
(spark.table("sales")
    .filter(col("units") > 0)
    .writeStream
    .option("checkpointLocation", checkpointPath)
    .outputMode("append")
    .table("newSales")
)
```

E)

```
(spark.table("sales")
    .groupBy("store")
    .agg(sum("sales"))
    .writeStream
    .option("checkpointLocation", checkpointPath)
    .outputMode("complete")
    .table("newSales")
)
```

A. Option A

B. Option B

C. Option C

D. Option D

E. Option E

ANSWER: E

Explanation:

The best practice is to use "Complete" as output mode instead of "append" when working with aggregated tables. Since gold layer is work final aggregated tables, the only option with output mode as complete is option E.

QUESTION NO: 6

Which of the following describes a benefit of creating an external table from Parquet rather than CSV when using a CREATE TABLE AS SELECT statement?

A. Parquet files can be partitioned

B. CREATE TABLE AS SELECT statements cannot be used on files

C. Parquet files have a well-defined schema

D. Parquet files have the ability to be optimized

E. Parquet files will become Delta tables

ANSWER: C

Explanation:

Option C is the correct answer because Parquet files have a well-defined schema that is embedded within the data itself. This means that the data types and column names of the Parquet files are automatically detected and preserved when creating an external table from them. This also enables the use of SQL and other structured query languages to access and analyze the data. CSV files, on the other hand, do not have a schema embedded in them, and require specifying the schema explicitly or inferring it from the data when creating an external table from them. This can lead to errors or inconsistencies in the data types and column names, and also increase the processing time and complexity.

Reference: CREATE TABLE AS SELECT, Parquet Files, CSV Files, Parquet vs. CSV

QUESTION NO: 7

Which of the following data lakehouse features results in improved data quality over a traditional data lake?

A. A data lakehouse provides storage solutions for structured and unstructured data.

B. A data lakehouse supports ACID-compliant transactions.

C. A data lakehouse allows the use of SQL queries to examine data.

D. A data lakehouse stores data in open formats.

E. A data lakehouse enables machine learning and artificial Intelligence workloads.

ANSWER: B

Explanation:

: A data lakehouse is a data management architecture that combines the flexibility, cost-efficiency, and scale of data lakes with the data management and ACID transactions of data warehouses, enabling business intelligence (BI) and machine learning (ML) on all data12. One of the key features of a data lakehouse is that it supports ACID-compliant transactions, which means that it ensures data integrity, consistency, and isolation across concurrent read and write operations3. This feature results in improved data quality over a traditional data lake, which does not support transactions and may suffer from data corruption, duplication, or inconsistency due to concurrent or streaming data ingestion and processing . Reference: 1: What is a Data Lakehouse? - Databricks 2: What is a Data

Lakehouse? Definition, features & benefits. - Qlik 3: ACID Transactions - Databricks : [Data Lake vs

Data Warehouse: Key Differences] : [Data Lakehouse: The Future of Data Engineering]