

Databricks-Certified-Professional-Data-Engineer Frequent Updates | Databricks-Certified-Professional- Data-Engineer Valid Mock Exam



One of the top features of Databricks Databricks-Certified-Professional-Data-Engineer exam dumps is the Databricks-Certified-Professional-Data-Engineer exam passing a money-back guarantee. In other words, your investments with TestInsides Links to an external site. Databricks Databricks Certified Professional Data Engineer Exam exam questions are secured with the 100 Databricks Certified Professional Data Engineer Exam Databricks-Certified-Professional-Data-Engineer Exam passing a money-back guarantee. Due to any reason, if you did not succeed in the final Databricks-Certified-Professional-Data-Engineer exam despite using TestInsides Databricks-Certified-Professional-Data-Engineer pdf questions and practice tests, we will return your whole payment without any deduction.

The Databricks Databricks-Certified-Professional-Data-Engineer Exam is intended for data engineers with experience in designing and implementing data solutions using Databricks. Candidates for this certification should have a good understanding of data engineering concepts, data processing frameworks, and programming languages such as Python and SQL. They should also be familiar with cloud platforms such as AWS, Azure, and Google Cloud Platform.

>> Databricks-Certified-Professional-Data-Engineer Frequent Updates <<

Databricks-Certified-Professional-Data-Engineer Valid Mock Exam & Databricks-Certified-Professional-Data-Engineer Actualtest

Databricks-Certified-Professional-Data-Engineer practice materials stand the test of time and harsh market, convey their sense of proficiency with passing rate up to 98 to 100 percent. They are 100 percent guaranteed Databricks-Certified-Professional-Data-Engineer practice materials. And our content of them are based on real exam by whittling down superfluous knowledge without delinquent mistakes. Our Databricks-Certified-Professional-Data-Engineer practice materials comprise of a number of academic questions for your practice, which are interlinked and helpful for your exam. So their perfection is unquestionable.

Databricks is an American-based technology company that develops software that manages big data via Apache Spark. Their software helps different organizations to process and analyze large data sets efficiently. Therefore, there is a need to certify professionals to handle these technologies. The Databricks Certified Professional Data Engineer (Databricks-Certified-Professional-Data-Engineer) Certification Exam is designed to validate the knowledge and proficiency of data engineers in designing and building effective and scalable data engineering solutions on the Databricks platform.

Databricks Certified Professional Data Engineer Exam Sample Questions (Q75-Q80):

NEW QUESTION # 75

Review the following error traceback:

□

Which statement describes the error being raised?

- A. There is no column in the table named heartrateheartrateheartrate
- **B. There is a syntax error because the heartrate column is not correctly identified as a column.**
- C. There is a type error because a column object cannot be multiplied.
- D. The code executed was PvSoark but was executed in a Scala notebook.
- E. There is a type error because a DataFrame object cannot be multiplied.

Answer: B

Explanation:

The error being raised is an AnalysisException, which is a type of exception that occurs when Spark SQL cannot analyze or execute a query due to some logical or semantic error¹. In this case, the error message indicates that the query cannot resolve the column name 'heartrateheartrateheartrate' given the input columns 'heartrate' and 'age'. This means that there is no column in the table named 'heartrateheartrateheartrate', and the query is invalid. A possible cause of this error is a typo or a copy-paste mistake in the query. To fix this error, the query should use a valid column name that exists in the table, such as 'heartrate'. References: AnalysisException

NEW QUESTION # 76

What is a method of installing a Python package scoped at the notebook level to all nodes in the currently active cluster?

- A. Install libraries from PyPi using the cluster UI
- **B. Use %pip install in a notebook cell**
- C. Run source env/bin/activate in a notebook setup script
- D. Use &sh install in a notebook cell

Answer: B

Explanation:

In Databricks notebooks, you can use the %pip install command in a notebook cell to install a Python package. This will install the package on all nodes in the currently active cluster at the notebook level. It is a feature provided by Databricks to facilitate the installation of Python libraries for the notebook environment specifically.

NEW QUESTION # 77

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_recording 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. Because Delta Lake uses Parquet for data storage, data types can be easily evolved by just modifying file footer information in place.
- **B. 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.**
- C. 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.
- D. Schema inference and evolution on Databricks ensure that inferred types will always accurately match the data types used by downstream systems.
- E. 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.

Answer: B

Explanation:

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.

NEW QUESTION # 78

A data engineer wants to automate job monitoring and recovery in Databricks using the Jobs API. They need to list all jobs, identify a failed job, and rerun it.

Which sequence of API actions should the data engineer perform?

- A. Use the jobs/list endpoint to list jobs, check job run statuses with jobs/runs/list, and rerun a failed job using jobs/run-now.
- B. Use the jobs/cancel endpoint to remove failed jobs, then recreate them with jobs/create and run the new ones.
- C. Use the jobs/get endpoint to retrieve job details, then use jobs/update to rerun failed jobs.
- D. Use the jobs/list endpoint to list jobs, then use the jobs/create endpoint to create a new job, and run the new job using jobs/run-now.

Answer: A

Explanation:

Comprehensive and Detailed Explanation From Exact Extract of Databricks Data Engineer Documents:

The Databricks Jobs REST API provides several endpoints for automation. The correct monitoring and rerun flow uses three specific calls:

GET /api/2.1/jobs/list - Lists all available jobs within the workspace.

GET /api/2.1/jobs/runs/list - Returns all runs for a specific job, including their current state (e.g., TERMINATED, FAILED).

POST /api/2.1/jobs/run-now - Immediately triggers a rerun of the specified job.

This sequence aligns with Databricks' prescribed automation model for job observability and recovery. Using jobs/update modifies metadata but does not rerun jobs, and jobs/create is only used for creating new jobs, not rerunning failed ones. Cancelling and recreating jobs introduces unnecessary duplication. Therefore, option A is the correct automated recovery workflow.

NEW QUESTION # 79

A user wants to use DLT expectations to validate that a derived table report contains all records from the source, included in the table validation_copy.

The user attempts and fails to accomplish this by adding an expectation to the report table definition.

Which approach would allow using DLT expectations to validate all expected records are present in this table?

- A. Define a view that performs a left outer join on validation_copy and report, and reference this view in DLT expectations for the report table
- B. Define a temporary table that perform a left outer join on validation_copy and report, and define an expectation that no report key values are null
- C. Define a SQL UDF that performs a left outer join on two tables, and check if this returns null values for report key values in a DLT expectation for the report table.
- D. Define a function that performs a left outer join on validation_copy and report and report, and check against the result in a DLT expectation for the report table

Answer: A

Explanation:

To validate that all records from the source are included in the derived table, creating a view that performs a left outer join between the validation_copy table and the report table is effective. The view can highlight any discrepancies, such as null values in the report table's key columns, indicating missing records. This view can then be referenced in DLT (Delta Live Tables) expectations for the report table to ensure data integrity. This approach allows for a comprehensive comparison between the source and the derived table.

References:

* Databricks Documentation on Delta Live Tables and Expectations: Delta Live Tables Expectations

• • • • •

- [illegible]