DATABRICKS-CERTIFIED-PROFESSIONAL-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.pass4itsure.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 table nameduser_ltvis being used to create a view that will be used by data analysts on various teams. Users in the workspace are configured into groups, which are used for setting up data access using ACLs.

Theuser_ltvtable has the following schema:

email STRING, age INT, ltv INT

The following view definition is executed:

```
CREATE VIEW email_ltv AS
SELECT
CASE WHEN
  is_member('marketing') THEN email
  ELSE 'REDACTED'
END AS email,
ltv
FROM user_ltv
```

An analyst who is not a member of the marketing group executes the following query:

SELECT * FROM email_ltv

Which statement describes the results returned by this query?

A. Three columns will be returned, but one column will be named "redacted" and contain only null values.

B. Only the email and itv columns will be returned; the email column will contain all null values.

C. The email and ltv columns will be returned with the values in user itv.

D. The email, age. and ltv columns will be returned with the values in user ltv.

E. Only the email and ltv columns will be returned; the email column will contain the string "REDACTED" in each row.

Correct Answer: E

Explanation: The code creates a view called email_ltv that selects the email and ltv columns from a table called user_ltv, which has the following schema: email STRING, age INT, ltv INT. The code alsouses the CASE WHEN expression to replace the email values with the string "REDACTED" if the user is not a member of the marketing group. The user who executes the query is not a member of the marketing group, so they will only see the email and ltv columns, and the email column will contain the string "REDACTED" in each row. Verified References: [Databricks Certified Data Engineer Professional], under "Lakehouse" section; Databricks Documentation, under "CASE expression" section.

QUESTION 2

A junior data engineer has been asked to develop a streaming data pipeline with a grouped aggregation using DataFramedf. The pipeline needs to calculate the average humidity and average temperature for each non-overlapping five-minute interval. Events are recorded once per minute per device.

Streaming DataFramedfhas the following schema:

"device_id INT, event_time TIMESTAMP, temp FLOAT, humidity FLOAT"

Code block:

```
df.withWatermark("event_time", "10 minutes")
    .groupBy(
        _____,
        "device_id"
    )
    .agg(
        avg("temp").alias("avg_temp"),
        avg("humidity").alias("avg_humidity")
    )
    .writeStream
    .format("delta")
    .saveAsTable("sensor_avg")
```

Choose the response that correctly fills in the blank within the code block to complete this task.

A. to_interval("event_time", "5 minutes").alias("time")

B. window("event_time", "5 minutes").alias("time")

C. "event_time"

D. window("event_time", "10 minutes").alias("time")

E. lag("event_time", "10 minutes").alias("time")

Correct Answer: B

Explanation: This is the correct answer because the window function is used to group streaming data by time intervals. The window function takes two arguments: a time column and a window duration. The window duration specifies how long each window is, and must be a multiple of 1second. In this case, the window duration is "5 minutes", which means each window will cover a non-overlapping five-minute interval. The window function also returns a struct column with two fields: start and end, which represent the start and end time of each window. The alias function is used to rename the struct column as "time". Verified References: [Databricks Certified Data Engineer Professional], under "Structured Streaming" section; Databricks Documentation, under "WINDOW" section.

QUESTION 3

Which statement regarding stream-static joins and static Delta tables is correct?

A. Each microbatch of a stream-static join will use the most recent version of the static Delta table as of each microbatch.

B. Each microbatch of a stream-static join will use the most recent version of the static Delta table as of the job\\'s initialization.

C. The checkpoint directory will be used to track state information for the unique keys present in the join.

D. Stream-static joins cannot use static Delta tables because of consistency issues.

E. The checkpoint directory will be used to track updates to the static Delta table.

Correct Answer: A

Explanation: This is the correct answer because stream-static joins are supported by Structured Streaming when one of the tables is a static Delta table. A static Delta table is a Delta table that is not updated by any concurrent writes, such as appends or merges, during the execution of a streaming query. In this case, each microbatch of a stream-static join will use the most recent version of the static Delta table as of each microbatch, which means it will reflect any changes made to the static Delta table before the start of each microbatch. Verified References:[Databricks Certified Data Engineer Professional], under "Structured Streaming" section; Databricks Documentation, under "Stream and static joins" section.

QUESTION 4

Which statement describes integration testing?

A. Validates interactions between subsystems of your application

B. Requires an automated testing framework

C. Requires manual intervention

D. Validates an application use case

E. Validates behavior of individual elements of your application

Correct Answer: A

Explanation: This is the correct answer because it describes integration testing. Integration testing is a type of testing that validates interactions between subsystems of your application, such as modules, components, or services. Integration testing ensures that the subsystems work together as expected and produce the correct outputs or results. Integration testing can be done at different levels of granularity, such as component integration testing, system integration testing, or end-to-end testing. Integration testing can help detect errors or bugs that may not be found by unit testing, which only validates behavior of individual elements of your application. Verified References: [Databricks Certified Data Engineer Professional], under "Testing" section; Databricks Documentation, under "Integration testing" section.

QUESTION 5

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 namedpredswith 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)

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

B. Option

C. Option

D. Option

E. Option

Correct Answer: C

Explanation: This is the correct answer because it will save the predictions to a Delta Lake table with the ability to compare all predictions across time. The code uses the mergeInto method to perform an upsert operation, which means it will insert new records or update existing records based on the customer_id and date columns. This way, the table will always contain the latest predictions for each customer and date, and also keep the history of previous predictions. The code also uses a new job cluster to run the job, which will minimize the compute costs as it will be created and terminated for each run. Verified References: [Databricks Certified Data Engineer Professional], under "Delta Lake" section; Databricks Documentation, under "Upsert into a table using merge" section.

DATABRICKS-CERTIFIED- DATABRICKS-CERTIFIED- DATABRICKS-CERTIFIED-