

## Certified Data Engineer Professional

Here you have the best Databricks Certified Data Engineer Professional practice exam questions

- You have 200 total questions to study from
- Each page has 5 questions, making a total of 40 pages
- You can navigate through the pages using the buttons at the bottom
- This questions were last updated on August 3, 2025
- This site is not affiliated with or endorsed by Databricks.

Question 1 of 200

---

An upstream system has been configured to pass the date for a given batch of data to the Databricks Jobs API as a parameter. The notebook to be scheduled will use this parameter to load data with the following code: df = spark.read.format("parquet").load(f"/mnt/source/(date)")

Which code block should be used to create the date Python variable used in the above code block?

A. date = spark.conf.get("date")

B. input\_dict = input()

```
date= input_dict["date"]
```

C. import sys

```
date = sys.argv[1]
```

D. date = dbutils.notebooks.getParam("date")

E. dbutils.widgets.text("date", "null")

```
date = dbutils.widgets.get("date")
```

[Answer](#)[Discussion](#)

Correct Answer: D

To create the date variable for use in the notebook, the code block should interact with the Databricks Jobs API to retrieve the parameter that has been passed by the upstream system. The appropriate method for this purpose is to use `dbutils.notebooks.getParam`, which is designed to retrieve parameters passed to the notebook from external sources like the Databricks Jobs API. This ensures that the value of 'date' is accurately obtained from the API, aligning with the requirements of the given scenario.

Question 2 of 200

---

The Databricks workspace administrator has configured interactive clusters for each of the data engineering groups. To control costs, clusters are set to terminate after 30 minutes of inactivity. Each user should be able to execute workloads against their assigned clusters at any time of the day.

Assuming users have been added to a workspace but not granted any permissions, which of the following describes the minimal permissions a user would need to start and attach to an already configured cluster.

- A. "Can Manage" privileges on the required cluster
- B. Workspace Admin privileges, cluster creation allowed, "Can Attach To" privileges on the required cluster
- C. Cluster creation allowed, "Can Attach To" privileges on the required cluster
- D. "Can Restart" privileges on the required cluster
- E. Cluster creation allowed, "Can Restart" privileges on the required cluster

[Answer](#)[Discussion](#)

Correct Answer: D

The minimal permissions a user needs to start and attach to an already configured cluster in Databricks are 'Can Restart' privileges on the required cluster. 'Can Restart' allows users to start a terminated cluster without unnecessary additional permissions.

### Question 3 of 200

---

When scheduling Structured Streaming jobs for production, which configuration automatically recovers from query failures and keeps costs low?

- A. Cluster: New Job Cluster;

Retries: Unlimited;

Maximum Concurrent Runs: Unlimited

- B. Cluster: New Job Cluster;

Retries: None;

Maximum Concurrent Runs: 1

- C. Cluster: Existing All-Purpose Cluster;

Retries: Unlimited;

Maximum Concurrent Runs: 1

- D. Cluster: New Job Cluster;

Retries: Unlimited;

Maximum Concurrent Runs: 1

- E. Cluster: Existing All-Purpose Cluster;

Retries: None;

Maximum Concurrent Runs: 1

Answer

Discussion

Correct Answer: D

To schedule Structured Streaming jobs for production with an automatic recovery from query failures while keeping costs low, you should use a new job cluster to ensure an isolated and streamlined environment, set retries to unlimited to handle and recover from any failures automatically, and limit maximum concurrent runs to 1 to avoid resource contention and ensure only one instance of the query runs at a time. This configuration efficiently manages resources and ensures the job's reliability and cost-effectiveness.

#### Question 4 of 200

---

The data engineering team has configured a Databricks SQL query and alert to monitor the values in a Delta Lake table. The recent\_sensor\_recordings table contains an identifying sensor\_id alongside the timestamp and temperature for the most recent 5 minutes of recordings.

The below query is used to create the alert:

```
SELECT MEAN(temperature), MAX(temperature), MIN(temperature)
FROM recent_sensor_recordings
GROUP BY sensor_id
```

The query is set to refresh each minute and always completes in less than 10 seconds. The alert is set to trigger when mean (temperature) > 120. Notifications are triggered to be sent at most every 1 minute.

If this alert raises notifications for 3 consecutive minutes and then stops, which statement must be true?

- A. The total average temperature across all sensors exceeded 120 on three consecutive executions of the query
- B. The recent\_sensor\_recordings table was unresponsive for three consecutive runs of the query
- C. The source query failed to update properly for three consecutive minutes and then restarted
- D. The maximum temperature recording for at least one sensor exceeded 120 on three consecutive executions of the query
- E. The average temperature recordings for at least one sensor exceeded 120 on three consecutive executions of the query

Answer

Discussion

Correct Answer: E

The alert is set to trigger when the mean temperature exceeds 120. Given that notifications were raised for three consecutive minutes and then stopped, it must be true that the average temperature recordings for at least one sensor exceeded 120 on three consecutive executions of the query.

## Question 5 of 200

---

A junior developer complains that the code in their notebook isn't producing the correct results in the development environment. A shared screenshot reveals that while they're using a notebook versioned with Databricks Repos, they're using a personal branch that contains old logic. The desired branch named dev-2.3.9 is not available from the branch selection dropdown.

Which approach will allow this developer to review the current logic for this notebook?

- A.** Use Repos to make a pull request use the Databricks REST API to update the current branch to dev-2.3.9
- B.** Use Repos to pull changes from the remote Git repository and select the dev-2.3.9 branch.
- C.** Use Repos to checkout the dev-2.3.9 branch and auto-resolve conflicts with the current branch
- D.** Merge all changes back to the main branch in the remote Git repository and clone the repo again
- E.** Use Repos to merge the current branch and the dev-2.3.9 branch, then make a pull request to sync with the remote repository

Answer

Discussion

Correct Answer: B

The correct approach is to use Repos to pull changes from the remote Git repository and select the dev-2.3.9 branch. This will allow the developer to access the most recent and desired logic for the notebook. Pulling changes from the remote repository ensures that the latest updates are incorporated and selecting the specific branch will provide the correct version of the code.

## Question 6 of 200

The security team is exploring whether or not the Databricks secrets module can be leveraged for connecting to an external database.

After testing the code with all Python variables being defined with strings, they upload the password to the secrets module and configure the correct permissions for the currently active user. They then modify their code to the following (leaving all other variables unchanged).

```
password = dbutils.secrets.get(scope="db_creds", key="jdbc_password")

print(password)

df = (spark
    .read
    .format("jdbc")
    .option("url", connection)
    .option("dbtable", tablename)
    .option("user", username)
    .option("password", password)
)
```

Which statement describes what will happen when the above code is executed?

- A. The connection to the external table will fail; the string "REDACTED" will be printed.
- B. An interactive input box will appear in the notebook; if the right password is provided, the connection will succeed and the encoded password will be saved to DBFS.
- C. An interactive input box will appear in the notebook; if the right password is provided, the connection will succeed and the password will be printed in plain text.
- D. The connection to the external table will succeed; the string value of password will be printed in plain text.
- E. The connection to the external table will succeed; the string "REDACTED" will be printed.

Answer

Discussion

Correct Answer: A

When using Databricks secrets with the `dbutils.secrets.get` method, the library handles secrets in a secure manner by redacting the output to avoid unintentional exposure. Therefore, when `print(password)` is executed, it will print the string 'REDACTED'. Since the connection details and permissions were configured correctly, the connection to the external table should succeed. Thus, the statement describing the result will be that the connection to the external table will succeed and the string 'REDACTED' will be printed.

Question 7 of 200

---

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")
)

```

Answer

Discussion

Correct Answer: A

To accomplish the task of saving predictions to a Delta Lake table with the ability to compare all predictions across time while minimizing potential compute costs, the provided code block in option A is suitable. It uses the `saveAsTable` method with the mode set to `append`, ensuring that predictions are continually added to the table without overwriting previous entries. This approach maintains a historical record of predictions as required and fits the batch processing context since the churn predictions are made at most once per day. It's important to note that in Databricks, databases and table creation typically default to Delta Lake format, so there's no need to explicitly specify the format when using `saveAsTable`.

Question 8 of 200

---

An upstream source writes Parquet data as hourly batches to directories named with the current date. A nightly batch job runs the following code to ingest all data from the previous day as indicated by the date variable:

```
(spark.read
    .format("parquet")
    .load(f"/mnt/raw_orders/{date}")
    .dropDuplicates(["customer_id", "order_id"])
    .write
    .mode("append")
    .saveAsTable("orders")
)
```

Assume that the fields customer\_id and order\_id serve as a composite key to uniquely identify each order.

If the upstream system is known to occasionally produce duplicate entries for a single order hours apart, which statement is correct?

- A. Each write to the orders table will only contain unique records, and only those records without duplicates in the target table will be written.
- B. Each write to the orders table will only contain unique records, but newly written records may have duplicates already present in the target table.
- C. Each write to the orders table will only contain unique records; if existing records with the same key are present in the target table, these records will be overwritten.
- D. Each write to the orders table will only contain unique records; if existing records with the same key are present in the target table, the operation will fail.
- E. Each write to the orders table will run deduplication over the union of new and existing records, ensuring no duplicate records are present.

Answer

Discussion

#### Correct Answer: B

Each write to the orders table will only contain unique records, but newly written records may have duplicates already present in the target table. The provided code uses the .dropDuplicates method, which removes duplicates in the current batch of data based on the specified keys (customer\_id and order\_id) before writing to the orders table. However, this method does not check for duplicates that might already exist in the target table from previous writes. Therefore, while each new batch of data will be de-duplicated, duplicates may still persist in the orders table if they were written in earlier batches.

#### Question 9 of 200

---

A junior member of the data engineering team is exploring the language interoperability of Databricks notebooks. The intended outcome of the below code is to register a view of all sales that occurred in countries on the continent of Africa that appear in the geo\_lookup table.

Before executing the code, running SHOW TABLES on the current database indicates the database contains only two tables: geo\_lookup and sales.

**Cmd 1**

```
%python  
countries_af = [x[0] for x in  
spark.table("geo_lookup").filter("continent='AF'").select("country").collect()]
```

**Cmd 2**

```
%sql  
CREATE VIEW sales_af AS  
SELECT *  
FROM sales  
WHERE city IN countries_af  
AND CONTINENT = "AF"
```

Which statement correctly describes the outcome of executing these command cells in order in an interactive notebook?

- A. Both commands will succeed. Executing show tables will show that countries\_af and sales\_af have been registered as views.
- B. Cmd 1 will succeed. Cmd 2 will search all accessible databases for a table or view named countries\_af: if this entity exists, Cmd 2 will succeed.
- C. Cmd 1 will succeed and Cmd 2 will fail. countries\_af will be a Python variable representing a PySpark DataFrame.
- D. Both commands will fail. No new variables, tables, or views will be created.
- E. Cmd 1 will succeed and Cmd 2 will fail. countries\_af will be a Python variable containing a list of strings.

Answer

Discussion

Correct Answer: E

Cmd 1 will succeed because it correctly uses PySpark to filter and collect the list of countries from the 'geo\_lookup' table where the continent is Africa ('AF'). This will result in 'countries\_af' being a Python list of country names. However, Cmd 2 will fail because the SQL command does not recognize the 'countries\_af' Python variable, leading to an error. SQL and Python operate in separate execution contexts within Databricks; therefore, Python variables are not accessible in SQL commands. Hence, 'countries\_af' being a Python list cannot be used within the SQL 'IN' clause in Cmd 2.

Question 10 of 200

---

A Delta table of weather records is partitioned by date and has the below schema: date DATE, device\_id INT, temp FLOAT, latitude FLOAT, longitude FLOAT

To find all the records from within the Arctic Circle, you execute a query with the below filter: latitude > 66.3

Which statement describes how the Delta engine identifies which files to load?

- A. All records are cached to an operational database and then the filter is applied
- B. The Parquet file footers are scanned for min and max statistics for the latitude column
- C. All records are cached to attached storage and then the filter is applied
- D. The Delta log is scanned for min and max statistics for the latitude column
- E. The Hive metastore is scanned for min and max statistics for the latitude column

Answer

Discussion

Correct Answer: D

The Delta engine identifies which files to load by scanning the Delta log for min and max statistics for the latitude column. Delta Lake captures statistics for each data file, including minimum and maximum values in each column, and uses these statistics to optimize query execution by determining which files may contain the data that matches the query filter.

Prev

Next

Examice delivers accurate questions, trusted answers, and the best value in certification exam preparation.



[Terms](#)   [Privacy](#)   [Refund](#)   [DMCA](#)   [Contact](#)

---

Copyright © 2025 Examice. All rights reserved.

## Question 11 of 200

The data engineering team has configured a job to process customer requests to be forgotten (have their data deleted). All user data that needs to be deleted is stored in Delta Lake tables using default table settings.

The team has decided to process all deletions from the previous week as a batch job at 1am each Sunday. The total duration of this job is less than one hour. Every Monday at 3am, a batch job executes a series of VACUUM commands on all Delta Lake tables throughout the organization.

The compliance officer has recently learned about Delta Lake's time travel functionality. They are concerned that this might allow continued access to deleted data.

Assuming all delete logic is correctly implemented, which statement correctly addresses this concern?

- A. Because the VACUUM command permanently deletes all files containing deleted records, deleted records may be accessible with time travel for around 24 hours.
- B. Because the default data retention threshold is 24 hours, data files containing deleted records will be retained until the VACUUM job is run the following day.
- C. Because Delta Lake time travel provides full access to the entire history of a table, deleted records can always be recreated by users with full admin privileges.
- D. Because Delta Lake's delete statements have ACID guarantees, deleted records will be permanently purged from all storage systems as soon as a delete job completes.
- E. Because the default data retention threshold is 7 days, data files containing deleted records will be retained until the VACUUM job is run 8 days later.

Answer

Discussion

Correct Answer: E

Because the default data retention threshold is 7 days, data files containing deleted records will be retained until the VACUUM job is run 8 days later. Delta Lake retains a 7-day history to support operations like time travel. As a result, even after deletion, the data remains accessible via time travel until the retention period expires and a VACUUM operation is performed to permanently remove the records.

Question 12 of 200

---

A junior data engineer has configured a workload that posts the following JSON to the Databricks REST API endpoint 2.0/jobs/create.

```
{  
  "name": "Ingest new data",  
  "existing_cluster_id": "6015-954420-peace720",  
  "notebook_task": {  
    "notebook_path": "/Prod/ingest.py"  
  }  
}
```

Assuming that all configurations and referenced resources are available, which statement describes the result of executing this workload three times?

- A. Three new jobs named "Ingest new data" will be defined in the workspace, and they will each run once daily.
- B. The logic defined in the referenced notebook will be executed three times on new clusters with the configurations of the provided cluster ID.
- C. Three new jobs named "Ingest new data" will be defined in the workspace, but no jobs will be executed.
- D. One new job named "Ingest new data" will be defined in the workspace, but it will not be executed.
- E. The logic defined in the referenced notebook will be executed three times on the referenced existing all purpose cluster.

Answer

Discussion

Correct Answer: C

When the workload is executed three times, it will create three separate jobs named "Ingest new data" in the workspace. Since the JSON definition does not include any scheduling information, these jobs will be defined but not executed. Each job creation will result in a unique job ID even though the names are the same, and they will only run if explicitly scheduled or triggered subsequently.

Question 13 of 200

---

An upstream system is emitting change data capture (CDC) logs that are being written to a cloud object storage directory. Each record in the log indicates the change type (insert, update, or delete) and the values for each field after the change. The source table has a primary key identified by the field pk\_id.

For auditing purposes, the data governance team wishes to maintain a full record of all values that have ever been valid in the source system. For analytical purposes, only the most recent value for each record needs to be recorded. The Databricks job to ingest these records occurs once per hour, but each individual record may have changed multiple times over the course of an hour.

Which solution meets these requirements?

- A. Create a separate history table for each pk\_id resolve the current state of the table by running a union all filtering the history tables for the most recent state.
- B. Use MERGE INTO to insert, update, or delete the most recent entry for each pk\_id into a bronze table, then propagate all changes throughout the system.
- C. Iterate through an ordered set of changes to the table, applying each in turn; rely on Delta Lake's versioning ability to create an audit log.
- D. Use Delta Lake's change data feed to automatically process CDC data from an external system, propagating all changes to all dependent tables in the Lakehouse.
- E. Ingest all log information into a bronze table; use MERGE INTO to insert, update, or delete the most recent entry for each pk\_id into a silver table to recreate the current table state.

Answer

Discussion

Correct Answer: E

The solution should involve ingesting all log information into a bronze table to preserve the raw change data capture (CDC) logs. Using the MERGE INTO operation will then allow the system to insert, update, or delete the most recent entry for each pk\_id into a silver table, effectively reconstructing the current table state. This approach ensures the bronze table maintains a full history of changes for auditing purposes, while the silver table provides the most recent values for analytical needs.

---

Question 14 of 200

---

An hourly batch job is configured to ingest data files from a cloud object storage container where each batch represent all records produced by the source system in a given hour. The batch job to process these records

into the Lakehouse is sufficiently delayed to ensure no late-arriving data is missed. The user\_id field represents a unique key for the data, which has the following schema: user\_id BIGINT, username STRING, user\_utc STRING, user\_region STRING, last\_login BIGINT, auto\_pay BOOLEAN, last\_updated BIGINT

New records are all ingested into a table named account\_history which maintains a full record of all data in the same schema as the source. The next table in the system is named account\_current and is implemented as a Type 1 table representing the most recent value for each unique user\_id.

Assuming there are millions of user accounts and tens of thousands of records processed hourly, which implementation can be used to efficiently update the described account\_current table as part of each hourly batch job?

- A. Use Auto Loader to subscribe to new files in the account\_history directory; configure a Structured Streaming trigger once job to batch update newly detected files into the account\_current table.
- B. Overwrite the account\_current table with each batch using the results of a query against the account\_history table grouping by user\_id and filtering for the max value of last\_updated.
- C. Filter records in account\_history using the last\_updated field and the most recent hour processed, as well as the max last\_login by user\_id write a merge statement to update or insert the most recent value for each user\_id.
- D. Use Delta Lake version history to get the difference between the latest version of account\_history and one version prior, then write these records to account\_current.
- E. Filter records in account\_history using the last\_updated field and the most recent hour processed, making sure to deduplicate on username; write a merge statement to update or insert the most recent value for each username.

Answer

Discussion

A table in the Lakehouse named customer\_churn\_params is used in churn prediction by the machine learning team. The table contains information about customers derived from a number of upstream sources. Currently, the data engineering team populates this table nightly by overwriting the table with the current valid values derived from upstream data sources.

The churn prediction model used by the ML team is fairly stable in production. The team is only interested in making predictions on records that have changed in the past 24 hours.

Which approach would simplify the identification of these changed records?

- A. Apply the churn model to all rows in the customer\_churn\_params table, but implement logic to perform an upsert into the predictions table that ignores rows where predictions have not changed.
- B. Convert the batch job to a Structured Streaming job using the complete output mode; configure a Structured Streaming job to read from the customer\_churn\_params table and incrementally predict against the churn model.
- C. Calculate the difference between the previous model predictions and the current customer\_churn\_params on a key identifying unique customers before making new predictions; only make predictions on those customers not in the previous predictions.
- D. Modify the overwrite logic to include a field populated by calling `spark.sql.functions.current_timestamp()` as data are being written; use this field to identify records written on a particular date.
- E. Replace the current overwrite logic with a merge statement to modify only those records that have changed; write logic to make predictions on the changed records identified by the change data feed.

Answer

Discussion

Correct Answer: E

To simplify the identification of changed records in the Lakehouse table customer\_churn\_params, it is most effective to replace the current overwrite logic with a merge statement. This way, only the records that have changed will be modified. The change data feed can then identify which records have been altered, allowing the machine learning team to make predictions only on those records. This approach minimizes unnecessary computations and processes only the relevant data, ensuring efficiency.

## Question 16 of 200

A table is registered with the following code:

```
CREATE TABLE recent_orders AS (
    SELECT a.user_id, a.email, b.order_id, b.order_date
    FROM
        (SELECT user_id, email
        FROM users) a
    INNER JOIN
        (SELECT user_id, order_id, order_date
        FROM orders
        WHERE order_date >= (current_date() - 7)) b
    ON a.user_id = b.user_id
)
```

Both users and orders are Delta Lake tables. Which statement describes the results of querying recent\_orders?

- A. All logic will execute at query time and return the result of joining the valid versions of the source tables at the time the query finishes.
- B. All logic will execute when the table is defined and store the result of joining tables to the DBFS; this stored data will be returned when the table is queried.
- C. Results will be computed and cached when the table is defined; these cached results will incrementally update as new records are inserted into source tables.
- D. All logic will execute at query time and return the result of joining the valid versions of the source tables at the time the query began.
- E. The versions of each source table will be stored in the table transaction log; query results will be saved to DBFS with each query.

Answer

Discussion

Correct Answer: B

When using a CREATE TABLE AS SELECT (CTAS) statement in Delta Lake, the logic executes at the time the table is defined. The result of the join operation is stored in the Databricks File System (DBFS). This means that the data resulting from the join operation at the time the table was created is stored, and this stored data will be returned whenever the table is queried. The CTAS statement does not recompute the join at query time. Instead, it queries the already stored result set.

Question 17 of 200

---

A production workload incrementally applies updates from an external Change Data Capture feed to a Delta Lake table as an always-on Structured Stream job. When data was initially migrated for this table, OPTIMIZE was executed and most data files were resized to 1 GB. Auto Optimize and Auto Compaction were both turned on for the streaming production job. Recent review of data files shows that most data files are under 64 MB, although each partition in the table contains at least 1 GB of data and the total table size is over 10 TB.

Which of the following likely explains these smaller file sizes?

- A.** Databricks has autotuned to a smaller target file size to reduce duration of MERGE operations
- B.** Z-order indices calculated on the table are preventing file compaction
- C.** Bloom filter indices calculated on the table are preventing file compaction
- D.** Databricks has autotuned to a smaller target file size based on the overall size of data in the table
- E.** Databricks has autotuned to a smaller target file size based on the amount of data in each partition

Answer

Discussion

Correct Answer: E

Databricks has likely autotuned to a smaller target file size based on the amount of data in each partition. Given that Auto Optimize and Auto Compaction are enabled, the system continuously manages file sizes to ensure efficient processing. The observed file sizes under 64 MB, despite each partition containing at least 1 GB of data and the overall table size exceeding 10 TB, suggest that the tuning is influenced by the per-partition data amounts to optimize the incremental updates efficiently.

Question 18 of 200

---

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.

Answer

Discussion

Correct Answer: A

When a microbatch of a stream-static join is processed, the most recent version of the static Delta table is used. This ensures that the data being joined is up-to-date with the latest available information in the static Delta table at the time of processing each microbatch.

Question 19 of 200

---

A junior data engineer has been asked to develop a streaming data pipeline with a grouped aggregation using DataFrame df. 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 DataFrame df has 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")

Answer

Discussion

Correct Answer: B

To compute the average humidity and temperature for each non-overlapping five-minute interval in a streaming data pipeline using a DataFrame, you need to group the data based on time windows. The window function in Structured Streaming is used to specify these kinds of time-based grouping. In this case, using `window('event_time', '5 minutes')` allows the data to be grouped into five-minute intervals based on the `event_time` column. This grouping is necessary to calculate the average values over these specific intervals.

Question 20 of 200

---

A data architect has designed a system in which two Structured Streaming jobs will concurrently write to a single bronze Delta table. Each job is subscribing to a different topic from an Apache Kafka source, but they will write data with the same schema. To keep the directory structure simple, a data engineer has decided to nest a checkpoint directory to be shared by both streams.

The proposed directory structure is displayed below:

```
./bronze
  ├── _checkpoint
  ├── _delta_log
  ├── year_week=2020_01
  └── year_week=2020_02
    ...
  ...
```

Which statement describes whether this checkpoint directory structure is valid for the given scenario and why?

- A. No; Delta Lake manages streaming checkpoints in the transaction log.
- B. Yes; both of the streams can share a single checkpoint directory.
- C. No; only one stream can write to a Delta Lake table.
- D. Yes; Delta Lake supports infinite concurrent writers.
- E. No; each of the streams needs to have its own checkpoint directory.

Answer

Discussion

Correct Answer: E

In a scenario where multiple Structured Streaming jobs are concurrently writing to a Delta table, each stream needs to maintain its own checkpoint directory. Checkpoints are used to store the streaming state and metadata, ensuring data consistency and recovery in case of failures. Sharing a single checkpoint directory between multiple streams can lead to data corruption and inconsistencies because the checkpointing mechanism is not designed to handle concurrent access by multiple streams. Therefore, the checkpoint directory structure depicted is not valid, and each stream should have its own checkpoint directory.

### Question 21 of 200

---

A Structured Streaming job deployed to production has been experiencing delays during peak hours of the day. At present, during normal execution, each microbatch of data is processed in less than 3 seconds. During peak hours of the day, execution time for each microbatch becomes very inconsistent, sometimes exceeding 30 seconds. The streaming write is currently configured with a trigger interval of 10 seconds.

Holding all other variables constant and assuming records need to be processed in less than 10 seconds, which adjustment will meet the requirement?

- A. Decrease the trigger interval to 5 seconds; triggering batches more frequently allows idle executors to begin processing the next batch while longer running tasks from previous batches finish.
- B. Increase the trigger interval to 30 seconds; setting the trigger interval near the maximum execution time observed for each batch is always best practice to ensure no records are dropped.
- C. The trigger interval cannot be modified without modifying the checkpoint directory; to maintain the current stream state, increase the number of shuffle partitions to maximize parallelism.
- D. Use the trigger once option and configure a Databricks job to execute the query every 10 seconds; this ensures all backlogged records are processed with each batch.
- E. Decrease the trigger interval to 5 seconds; triggering batches more frequently may prevent records from backing up and large batches from causing spill.

Answer

Discussion

Correct Answer: E

To ensure that records are processed in less than 10 seconds, the key is to handle microbatch processing more efficiently during peak hours. Decreasing the trigger interval to 5 seconds can help achieve this by triggering batches more frequently, which may prevent records from backing up and large batches from causing spill. This allows more consistent batch processing times and utilizes available resources effectively, reducing the risk of exceeding the 10-second processing requirement.

Question 22 of 200

---

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.

Answer

Discussion

Correct Answer: E

Delta Lake Auto Compaction involves checking if files can be further compacted after a write completes. If further compaction is possible, an asynchronous job runs to execute an OPTIMIZE command aiming for a default file size of 128 MB. This ensures better storage efficiency by minimizing the number of small files in a Delta table.

Question 23 of 200

---

Which statement characterizes the general programming model used by Spark Structured Streaming?

- A. Structured Streaming leverages the parallel processing of GPUs to achieve highly parallel data throughput.
- B. Structured Streaming is implemented as a messaging bus and is derived from Apache Kafka.
- C. Structured Streaming uses specialized hardware and I/O streams to achieve sub-second latency for data transfer.
- D. Structured Streaming models new data arriving in a data stream as new rows appended to an unbounded table.
- E. Structured Streaming relies on a distributed network of nodes that hold incremental state values for cached stages.

Answer

Discussion

Correct Answer: D

Structured Streaming models new data arriving in a data stream as new rows appended to an unbounded table. This concept is integral to understanding how Spark Structured Streaming operates.

It treats streaming data as an append-only table, allowing for continuous incremental processing, which is similar in operation to traditional batch processing but applied in a real-time context.

#### Question 24 of 200

---

Which configuration parameter directly affects the size of a spark-partition upon ingestion of data into Spark?

- A. `spark.sql.files.maxPartitionBytes`
- B. `spark.sql.autoBroadcastJoinThreshold`
- C. `spark.sql.files.openCostInBytes`
- D. `spark.sql.adaptive.coalescePartitions.minPartitionNum`
- E. `spark.sql.adaptive.advisoryPartitionSizeInBytes`

Answer

Discussion

Correct Answer: A

The configuration parameter '`spark.sql.files.maxPartitionBytes`' directly affects the size of a spark-partition upon ingestion of data into Spark. It defines the maximum number of bytes to pack into a single partition when reading files, which in turn influences the partition size.

---

#### Question 25 of 200

A Spark job is taking longer than expected. Using the Spark UI, a data engineer notes that the Min, Median, and Max Durations for tasks in a particular stage show the minimum and median time to complete a task as roughly the same, but the max duration for a task to be roughly 100 times as long as the minimum.

Which situation is causing increased duration of the overall job?

- A. Task queueing resulting from improper thread pool assignment.
- B. Spill resulting from attached volume storage being too small.
- C. Network latency due to some cluster nodes being in different regions from the source data
- D. Skew caused by more data being assigned to a subset of spark-partitions.
- E. Credential validation errors while pulling data from an external system.

Answer

Discussion

Correct Answer: D

When the minimum and median durations for tasks are roughly the same, but the maximum duration is significantly longer, it indicates that some tasks are taking much longer than others. This is typically due to data skew, where some partitions have more data than others. As a result, tasks processing the larger partitions take significantly more time, causing the overall job to take longer. Therefore, the increased duration of the overall job is caused by skew due to more data being assigned to a subset of Spark partitions.

Prev

Next

## Question 26 of 200

Each configuration below is identical to the extent that each cluster has 400 GB total of RAM, 160 total cores and only one Executor per VM.

Given a job with at least one wide transformation, which of the following cluster configurations will result in maximum performance?

- A. • Total VMs: 1
  - 400 GB per Executor
  - 160 Cores / Executor
- B. • Total VMs: 8
  - 50 GB per Executor
  - 20 Cores / Executor
- C. • Total VMs: 16
  - 25 GB per Executor
  - 10 Cores/Executor
- D. • Total VMs: 4
  - 100 GB per Executor
  - 40 Cores/Executor
- E. • Total VMs: 2
  - 200 GB per Executor
  - 80 Cores / Executor

[Answer](#)[Discussion](#)

Correct Answer: C

To maximize performance, it is crucial to balance the level of parallelism and the resources available to each Executor. Option C provides 16 VMs, with 25 GB of RAM and 10 cores per Executor. This configuration offers a

good level of parallelism, which is beneficial for handling wide transformations that require significant data shuffling across multiple nodes. The allocation of 25 GB of RAM per Executor ensures that each Executor can handle its tasks efficiently without being resource-starved. The higher number of VMs compared to other options also improves fault tolerance and workload distribution, which are key to optimizing performance in distributed computing environments.

Question 27 of 200

---

A junior data engineer on your team has implemented the following code block.

```
MERGE INTO events
USING new_events
ON events.event_id = new_events.event_id
WHEN NOT MATCHED
    INSERT *
```

The view `new_events` contains a batch of records with the same schema as the `events` Delta table. The `event_id` field serves as a unique key for this table.

When this query is executed, what will happen with new records that have the same `event_id` as an existing record?

- A. They are merged.
- B. They are ignored.
- C. They are updated.
- D. They are inserted.
- E. They are deleted.

Answer

Discussion

Correct Answer: B

The code snippet uses the `MERGE` statement to combine data from the '`new_events`' view into the '`events`' table based on the '`event_id`' field. The '`WHEN NOT MATCHED`' clause specifies that new records (those that do not have a matching '`event_id`' in the '`events`' table) should be inserted. There is no clause that defines what happens when existing records with the same '`event_id`' are matched. Therefore, according to the standard behavior of the `MERGE` statement without a '`WHEN MATCHED`' clause, existing records with matching '`event_id`' will be ignored and left unchanged.

A junior data engineer seeks to leverage Delta Lake's Change Data Feed functionality to create a Type 1 table representing all of the values that have ever been valid for all rows in a bronze table created with the property `delta.enableChangeDataFeed = true`. They plan to execute the following code as a daily job:

```
from pyspark.sql.functions import col  
  
(spark.read.format("delta")  
    .option("readChangeFeed", "true")  
    .option("startingVersion", 0)  
    .table("bronze")  
    .filter(col("_change_type").isin(["update_postimage", "insert"]))  
    .write  
    .mode("append")  
    .table("bronze_history_type1")  
)
```

Which statement describes the execution and results of running the above query multiple times?

- A. Each time the job is executed, newly updated records will be merged into the target table, overwriting previous values with the same primary keys.
- B. Each time the job is executed, the entire available history of inserted or updated records will be appended to the target table, resulting in many duplicate entries.
- C. Each time the job is executed, the target table will be overwritten using the entire history of inserted or updated records, giving the desired result.
- D. Each time the job is executed, the differences between the original and current versions are calculated; this may result in duplicate entries for some records.
- E. Each time the job is executed, only those records that have been inserted or updated since the last execution will be appended to the target table, giving the desired result.

Answer

Discussion

Correct Answer: B

The provided code uses Delta Lake's Change Data Feed (CDF) functionality to read and filter changes from the beginning (startingVersion, 0) each time it runs. It filters for update postimage and insert changes only and appends the filtered results to the target table. Due to starting from version 0 every time, the entire history of inserted or updated records will be read and appended to the target table on each execution, leading to many duplicate entries.

Question 29 of 200

---

A new data engineer notices that a critical field was omitted from an application that writes its Kafka source to Delta Lake. This happened even though the critical field was in the Kafka source. That field was further missing from data written to dependent, long-term storage. The retention threshold on the Kafka service is seven days. The pipeline has been in production for three months.

Which describes how Delta Lake can help to avoid data loss of this nature in the future?

- A. The Delta log and Structured Streaming checkpoints record the full history of the Kafka producer.
- B. Delta Lake schema evolution can retroactively calculate the correct value for newly added fields, as long as the data was in the original source.
- C. Delta Lake automatically checks that all fields present in the source data are included in the ingestion layer.
- D. Data can never be permanently dropped or deleted from Delta Lake, so data loss is not possible under any circumstance.
- E. Ingesting all raw data and metadata from Kafka to a bronze Delta table creates a permanent, replayable history of the data state.

Answer

Discussion

Correct Answer: E

Ingesting all raw data and metadata from Kafka to a bronze Delta table creates a permanent, replayable history of the data state, ensuring that no critical data is lost even if it was omitted initially in the application writing process. This provides a safeguard allowing the recovery and inclusion of all necessary fields from the original source data.

Question 30 of 200

---

A nightly job ingests data into a Delta Lake table using the following code:

```
from pyspark.sql.functions import current_timestamp, input_file_name, col
from pyspark.sql.column import Column

def ingest_daily_batch(time_col: Column, year:int, month:int, day:int):
    (spark.read
        .format("parquet")
        .load(f"/mnt/daily_batch/{year}/{month}/{day}")
        .select("*",
            time_col.alias("ingest_time"),
            input_file_name().alias("source_file")
        )
        .write
        .mode("append")
        .saveAsTable("bronze")
    )
```

The next step in the pipeline requires a function that returns an object that can be used to manipulate new records that have not yet been processed to the next table in the pipeline.

Which code snippet completes this function definition?

```
def new_records():
```

- A. `return spark.readStream.table("bronze")`
- B. `return spark.readStream.load("bronze")`
- C. `return (spark.read
 .table("bronze")
 .filter(col("ingest_time") == current_timestamp())
)`
- D. `return spark.read.option("readChangeFeed", "true").table ("bronze")`
- E. `return (spark.read
 .table("bronze")
 .filter(col("source_file") == f"/mnt/daily_batch/{year}/{month}/{day}")
)`

Answer

Discussion

Correct Answer: A

The function needs to return an object that can be used to manipulate new records that have not yet been processed. Since the ingest job appends to the 'bronze' table, using the `readStream` method on this table allows for reading new data incrementally. The option '`spark.readStream.table("bronze")`' is appropriate for this purpose as it ensures the function can handle new records as they are added to the 'bronze' table.

## Question 31 of 200

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.

[Answer](#)[Discussion](#)

Correct Answer: D

Setting types manually provides greater assurance of data quality enforcement. Relying on inferred schema can be risky due to the possibility of encountering unexpected data types, which may result in errors or inefficient processing. Specifying the schema manually ensures that the data is accurately interpreted and conforms to expected standards, which is crucial for maintaining data integrity in production environments.

Question 32 of 200

---

The data engineering team maintains the following code:

```
accountDF = spark.table("accounts")
orderDF = spark.table("orders")
itemDF = spark.table("items")

orderWithItemDF = (orderDF.join(
    itemDF,
    orderDF.itemID == itemDF.itemID)
.select(
    orderDF.accountID,
    orderDF.itemID,

    itemDF.itemName))

finalDF = (accountDF.join(
    orderWithItemDF,
    accountDF.accountID == orderWithItemDF.accountID)
.select(
    orderWithItemDF["*"],

    accountDF.city))

(finalDF.write
    .mode("overwrite")
    .table("enriched_itemized_orders_by_account"))
```

Assuming that this code produces logically correct results and the data in the source tables has been de-duplicated and validated, which statement describes what will occur when this code is executed?

- A. A batch job will update the enriched\_itemized\_orders\_by\_account table, replacing only those rows that have different values than the current version of the table, using accountID as the primary key.
- B. The enriched\_itemized\_orders\_by\_account table will be overwritten using the current valid version of data in each of the three tables referenced in the join logic.
- C. An incremental job will leverage information in the state store to identify unjoined rows in the source tables and write these rows to the enriched\_itemized\_orders\_by\_account table.
- D. An incremental job will detect if new rows have been written to any of the source tables; if new rows are detected, all results will be recalculated and used to overwrite the enriched\_itemized\_orders\_by\_account table.
- E. No computation will occur until enriched\_itemized\_orders\_by\_account is queried; upon query materialization, results will be calculated using the current valid version of data

in each of the three tables referenced in the join logic.

Answer

Discussion

Correct Answer: B

This code reads data from three source tables, joins them, and then writes the result to the 'enriched\_itemized\_orders\_by\_account' table using the '.mode("overwrite")' method. The '.mode("overwrite")' indicates that the existing table will be completely replaced with the new data generated from the current values in the source tables. Thus, the entire 'enriched\_itemized\_orders\_by\_account' table will be overwritten using the current valid version of data in each of the three tables referenced in the join logic.

Question 33 of 200

---

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. Storing 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.

Answer

Discussion

Correct Answer: 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. This approach ensures that security policies can be managed effectively, reduces risks associated with data access, and supports the collaboration needs of diverse teams by clearly delineating access boundaries.

Question 34 of 200

---

The data architect has mandated that all tables in the Lakehouse should be configured as external Delta Lake tables.

Which approach will ensure that this requirement is met?

- A.** Whenever a database is being created, make sure that the LOCATION keyword is used
- B.** When configuring an external data warehouse for all table storage, leverage Databricks for all ELT.
- C.** Whenever a table is being created, make sure that the LOCATION keyword is used.
- D.** When tables are created, make sure that the EXTERNAL keyword is used in the CREATE TABLE statement.
- E.** When the workspace is being configured, make sure that external cloud object storage has been mounted.

Answer

Discussion

Correct Answer: C

To ensure that all tables in the Lakehouse are configured as external Delta Lake tables, the LOCATION keyword should be used whenever a table is created. This approach specifies the external storage location for each table individually, ensuring compliance with the data architect's mandate.

Question 35 of 200

---

To reduce storage and compute costs, the data engineering team has been tasked with curating a series of aggregate tables leveraged by business intelligence dashboards, customer-facing applications, production machine learning models, and ad hoc analytical queries.

The data engineering team has been made aware of new requirements from a customer-facing application, which is the only downstream workload they manage entirely. As a result, an aggregate table used by numerous teams across the organization will need to have a number of fields renamed, and additional fields will also be added.

Which of the solutions addresses the situation while minimally interrupting other teams in the organization without increasing the number of tables that need to be managed?

- A. Send all users notice that the schema for the table will be changing; include in the communication the logic necessary to revert the new table schema to match historic queries.
- B. Configure a new table with all the requisite fields and new names and use this as the source for the customer-facing application; create a view that maintains the original data schema and table name by aliasing select fields from the new table.
- C. Create a new table with the required schema and new fields and use Delta Lake's deep clone functionality to sync up changes committed to one table to the corresponding table.
- D. Replace the current table definition with a logical view defined with the query logic currently writing the aggregate table; create a new table to power the customer-facing application.
- E. Add a table comment warning all users that the table schema and field names will be changing on a given date; overwrite the table in place to the specifications of the customer-facing application.

Answer

Discussion

Correct Answer: B

To accommodate schema changes while minimally disrupting other teams, the best approach is to configure a new table that includes the necessary fields and new names for the customer-facing application. Then, create a view that aliases select fields from the new table to maintain the original data schema and table name. This allows the data engineering team to meet the specific needs of the customer-facing application without altering the table structure that other teams rely on, thereby avoiding interruptions. Additionally, this method doesn't increase the number of tables that need to be managed, as the view serves to provide compatibility rather than adding another table requiring maintenance.

## Question 36 of 200

A Delta Lake table representing metadata about content posts from users has the following schema: user\_id LONG, post\_text STRING, post\_id STRING, longitude FLOAT, latitude FLOAT, post\_time TIMESTAMP, date DATE

This table is partitioned by the date column. A query is run with the following filter: longitude < 20 & longitude > -20

Which statement describes how data will be filtered?

- A. Statistics in the Delta Log will be used to identify partitions that might include files in the filtered range.
- B. No file skipping will occur because the optimizer does not know the relationship between the partition column and the longitude.
- C. The Delta Engine will use row-level statistics in the transaction log to identify the files that meet the filter criteria.
- D. Statistics in the Delta Log will be used to identify data files that might include records in the filtered range.
- E. The Delta Engine will scan the parquet file footers to identify each row that meets the filter criteria.

Answer

Discussion

Correct Answer: D

Statistics in the Delta Log will be used to identify data files that might include records in the filtered range. The Delta Log keeps statistics such as min and max values for columns, which helps in identifying the relevant data files quickly without scanning all files. This file skipping mechanism improves the efficiency of the query execution by focusing only on the files that potentially contain the needed data.

Question 37 of 200

---

A small company based in the United States has recently contracted a consulting firm in India to implement several new data engineering pipelines to power artificial intelligence applications. All the company's data is stored in regional cloud storage in the United States.

The workspace administrator at the company is uncertain about where the Databricks workspace used by the contractors should be deployed.

Assuming that all data governance considerations are accounted for, which statement accurately informs this decision?

- A. Databricks runs HDFS on cloud volume storage; as such, cloud virtual machines must be deployed in the region where the data is stored.
- B. Databricks workspaces do not rely on any regional infrastructure; as such, the decision should be made based upon what is most convenient for the workspace administrator.
- C. Cross-region reads and writes can incur significant costs and latency; whenever possible, compute should be deployed in the same region the data is stored.
- D. Databricks leverages user workstations as the driver during interactive development; as such, users should always use a workspace deployed in a region they are physically near.
- E. Databricks notebooks send all executable code from the user's browser to virtual machines over the open internet; whenever possible, choosing a workspace region near the end users is the most secure.

Answer

Discussion

Correct Answer: C

Cross-region reads and writes can incur significant costs and latency. Therefore, compute should be deployed in the same region where the data is stored to optimize performance and reduce costs.

---

Question 38 of 200

The downstream consumers of a Delta Lake table have been complaining about data quality issues impacting performance in their applications. Specifically, they have complained that invalid latitude and longitude values in the activity\_details table have been breaking their ability to use other geolocation processes.

A junior engineer has written the following code to add CHECK constraints to the Delta Lake table:

```
ALTER TABLE activity_details
ADD CONSTRAINT valid_coordinates
CHECK (
    latitude >= -90 AND
    latitude <= 90 AND
    longitude >= -180 AND
    longitude <= 180);
```

A senior engineer has confirmed the above logic is correct and the valid ranges for latitude and longitude are provided, but the code fails when executed.

Which statement explains the cause of this failure?

- A. Because another team uses this table to support a frequently running application, two-phase locking is preventing the operation from committing.
- B. The activity\_details table already exists; CHECK constraints can only be added during initial table creation.
- C. The activity\_details table already contains records that violate the constraints; all existing data must pass CHECK constraints in order to add them to an existing table.
- D. The activity\_details table already contains records; CHECK constraints can only be added prior to inserting values into a table.
- E. The current table schema does not contain the field valid\_coordinates; schema evolution will need to be enabled before altering the table to add a constraint.

Answer

Discussion

Correct Answer: C

The operation fails because the activity\_details table already contains records that violate the constraints. In Delta Lake, when adding a CHECK constraint to an existing table, all existing data must comply with the new constraints. If any records in the table do not meet the constraint conditions specified, the operation will fail. Therefore, before adding the CHECK constraints, it is necessary to ensure that all existing records meet the constraints or modify the records accordingly.

---

Which of the following is true of Delta Lake and the Lakehouse?

- A. Because Parquet compresses data row by row, strings will only be compressed when a character is repeated multiple times.
- B. Delta Lake automatically collects statistics on the first 32 columns of each table which are leveraged in data skipping based on query filters.
- C. Views in the Lakehouse maintain a valid cache of the most recent versions of source tables at all times.
- D. Primary and foreign key constraints can be leveraged to ensure duplicate values are never entered into a dimension table.
- E. Z-order can only be applied to numeric values stored in Delta Lake tables.

Answer

Discussion

Correct Answer: B

Delta Lake automatically collects statistics on the first 32 columns of each table which are leveraged in data skipping based on query filters. This helps to optimize query performance by allowing the system to quickly determine which parts of the dataset do not need to be read based on the query conditions.

---

Question 40 of 200

The view updates represents an incremental batch of all newly ingested data to be inserted or updated in the customers table.

The following logic is used to process these records.

```

MERGE INTO customers
USING (
    SELECT updates.customer_id as merge_key, updates.*
    FROM updates
)
UNION ALL

SELECT NULL as merge_key, updates.*
FROM updates JOIN customers
ON updates.customer_id = customers.customer_id
WHERE customers.current = true AND updates.address <> customers.address
) staged_updates
ON customers.customer_id = mergeKey
WHEN MATCHED AND customers.current = true AND customers.address <> staged_updates.address THEN
    UPDATE SET current = false, end_date = staged_updates.effective_date
WHEN NOT MATCHED THEN
    INSERT(customer_id, address, current, effective_date, end_date)
    VALUES(staged_updates.customer_id, staged_updates.address, true, staged_updates.effective_date,
null)

```

Which statement describes this implementation?

- A. The customers table is implemented as a Type 3 table; old values are maintained as a new column alongside the current value.
- B. The customers table is implemented as a Type 2 table; old values are maintained but marked as no longer current and new values are inserted.
- C. The customers table is implemented as a Type 0 table; all writes are append only with no changes to existing values.
- D. The customers table is implemented as a Type 1 table; old values are overwritten by new values and no history is maintained.
- E. The customers table is implemented as a Type 2 table; old values are overwritten and new customers are appended.

Answer

Discussion

Correct Answer: B

The customers table is implemented as a Type 2 table; old values are maintained but marked as no longer current and new values are inserted. This is evident from the provided SQL logic. When records are matched and there are changes in the address, the current record is marked as no longer current by setting 'current' to false and updating the 'end\_date'. New records are then inserted with the new values and marked as current. This behavior is characteristic of a Type 2 Slowly Changing Dimension (SCD), where changes in data are tracked by maintaining historical records.

## Question 41 of 200

The DevOps team has configured a production workload as a collection of notebooks scheduled to run daily using the Jobs UI. A new data engineering hire is onboarding to the team and has requested access to one of these notebooks to review the production logic.

What are the maximum notebook permissions that can be granted to the user without allowing accidental changes to production code or data?

A. Can Manage

B. Can Edit

C. No permissions

D. Can Read

E. Can Run

Answer

Discussion

Correct Answer: D

To ensure the new data engineering hire can review the production logic without making accidental changes to the production code or data, the 'Can Read' permission is the most appropriate. This level of access allows the user to view the notebook's contents without being able to modify or execute it, thereby maintaining the integrity of the production environment.

## Question 42 of 200

A table named user\_ltv is 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.

The user\_ltv table 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 ltv 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\_ltv.
- 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.

Answer

Discussion

Correct Answer: E

The view definition uses a case statement to check if the user is a member of the 'marketing' group. If they are a member, it returns the actual email; otherwise, it returns 'REDACTED'. Since the analyst is not a member of the marketing group, querying the view will return the email column with the value 'REDACTED' for each row. Therefore, the query will return the email and ltv columns, with the email column containing the string 'REDACTED' in each row.

#### Question 43 of 200

---

The data governance team has instituted a requirement that all tables containing Personal Identifiable Information (PII) must be clearly annotated. This includes adding column comments, table comments, and setting the custom table property "contains\_pii" = true.

The following SQL DDL statement is executed to create a new table:

```
CREATE TABLE dev.pii_test
(id INT, name STRING COMMENT "PII")
COMMENT "Contains PII"
TBLPROPERTIES ('contains_pii' = True)
```

Which command allows manual confirmation that these three requirements have been met?

- A. DESCRIBE EXTENDED dev.pii\_test
- B. DESCRIBE DETAIL dev.pii\_test
- C. SHOW TBLPROPERTIES dev.pii\_test
- D. DESCRIBE HISTORY dev.pii\_test
- E. SHOW TABLES dev

Answer

Discussion

Correct Answer: A

The correct command to manually confirm that all three requirements have been met, including column comments, table comments, and the custom table property, is 'DESCRIBE EXTENDED dev.pii\_test' as it provides a detailed description of the table including all its properties and comments.

Question 44 of 200

---

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.

Answer

Discussion

Correct Answer: E

Delta Lake maintains versions of data and files for enabling time travel and ensuring ACID transactions. Even after executing a DELETE command, the deleted records remain in the older versions of the table until a VACUUM command is executed to remove them. Thus, files containing deleted records may still be accessible through time travel until they are permanently removed by performing a VACUUM operation.

Question 45 of 200

---

An external object storage container has been mounted to the location /mnt/finance\_eda\_bucket.

The following logic was executed to create a database for the finance team:

```
CREATE DATABASE finance_eda_db
LOCATION '/mnt/finance_eda_bucket';
GRANT USAGE ON DATABASE finance_eda_db TO finance;
GRANT CREATE ON DATABASE finance_eda_db TO finance;
```

After the database was successfully created and permissions configured, a member of the finance team runs the following code:

```
CREATE TABLE finance_eda_db.tx_sales AS
SELECT *
FROM sales
WHERE state = "TX";
```

If all users on the finance team are members of the finance group, which statement describes how the tx\_sales table will be created?

- A. A logical table will persist the query plan to the Hive Metastore in the Databricks control plane.
- B. An external table will be created in the storage container mounted to /mnt/finance\_eda\_bucket.
- C. A logical table will persist the physical plan to the Hive Metastore in the Databricks control plane.
- D. An managed table will be created in the storage container mounted to /mnt/finance\_eda\_bucket.
- E. A managed table will be created in the DBFS root storage container.

Answer

Discussion

Correct Answer: D

The table tx\_sales will be created as a managed table because the LOCATION keyword was not specified during the table creation. In Databricks, if a LOCATION is provided during the creation of the database, all tables created in that database will inherit that location by default, unless a different LOCATION is provided during table creation. In this case, since no specific LOCATION was specified for the table, it will be created as a managed table in the location defined at the database level, which is /mnt/finance\_eda\_bucket.

## Question 46 of 200

Although the Databricks Utilities Secrets module provides tools to store sensitive credentials and avoid accidentally displaying them in plain text users should still be careful with which credentials are stored here and which users have access to using these secrets.

Which statement describes a limitation of Databricks Secrets?

- A. Because the SHA256 hash is used to obfuscate stored secrets, reversing this hash will display the value in plain text.
- B. Account administrators can see all secrets in plain text by logging on to the Databricks Accounts console.
- C. Secrets are stored in an administrators-only table within the Hive Metastore; database administrators have permission to query this table by default.
- D. Iterating through a stored secret and printing each character will display secret contents in plain text.
- E. The Databricks REST API can be used to list secrets in plain text if the personal access token has proper credentials.

Answer

Discussion

Correct Answer: E

The Databricks REST API can indeed be used to list secrets in plain text if the personal access token has proper credentials. This means that the security of the secrets is dependent not only on how they

are stored but also on the security of the access tokens and the permissions assigned to them. This presents a potential security risk if a personal access token is compromised, which aligns with the limitations described in the question.

#### Question 47 of 200

---

What statement is true regarding the retention of job run history?

- A. It is retained until you export or delete job run logs
- B. It is retained for 30 days, during which time you can deliver job run logs to DBFS or S3
- C. It is retained for 60 days, during which you can export notebook run results to HTML
- D. It is retained for 60 days, after which logs are archived
- E. It is retained for 90 days or until the run-id is re-used through custom run configuration

Answer

Discussion

Correct Answer: C

Job run history is retained for 60 days, during which you can export notebook run results to HTML. This aligns with standard practices for the retention and exportation of job run data.

---

#### Question 48 of 200

A data engineer, User A, has promoted a new pipeline to production by using the REST API to programmatically create several jobs. A DevOps engineer, User B, has configured an external orchestration tool to trigger job runs through the REST API. Both users authorized the REST API calls using their personal access tokens.

Which statement describes the contents of the workspace audit logs concerning these events?

- A. Because the REST API was used for job creation and triggering runs, a Service Principal will be automatically used to identify these events.
- B. Because User B last configured the jobs, their identity will be associated with both the job creation events and the job run events.
- C. Because these events are managed separately, User A will have their identity associated with the job creation events and User B will have their identity associated with the job run events.
- D. Because the REST API was used for job creation and triggering runs, user identity will not be captured in the audit logs.
- E. Because User A created the jobs, their identity will be associated with both the job creation events and the job run events.

Answer

Discussion

Correct Answer: C

When using the REST API to create and trigger jobs, the identity associated with each event is determined by who performed the action. In this case, User A created the jobs and thus their identity will be associated with the job creation events. User B configured the external orchestration tool to trigger the job runs, so their identity will be associated with the job run events. This separation ensures that audit logs accurately reflect who performed each specific action.

---

A user new to Databricks is trying to troubleshoot long execution times for some pipeline logic they are working on. Presently, the user is executing code cell-by-cell, using `display()` calls to confirm code is producing the logically correct results as new transformations are added to an operation. To get a measure of average time to execute, the user is running each cell multiple times interactively.

Which of the following adjustments will get a more accurate measure of how code is likely to perform in production?

- A. Scala is the only language that can be accurately tested using interactive notebooks; because the best performance is achieved by using Scala code compiled to JARs, all PySpark and Spark SQL logic should be refactored.
- B. The only way to meaningfully troubleshoot code execution times in development notebooks is to use production-sized data and production-sized clusters with Run All execution.
- C. Production code development should only be done using an IDE; executing code against a local build of open source Spark and Delta Lake will provide the most accurate benchmarks for how code will perform in production.
- D. Calling `display()` forces a job to trigger, while many transformations will only add to the logical query plan; because of caching, repeated execution of the same logic does not provide meaningful results.
- E. The Jobs UI should be leveraged to occasionally run the notebook as a job and track execution time during incremental code development because Photon can only be enabled on clusters launched for scheduled jobs.

Answer

Discussion

Correct Answer: B

To get a more accurate measure of how code is likely to perform in production, you need to mimic the production environment as closely as possible. This means using production-sized data and production-sized clusters. Interactive execution in notebooks can be helpful during development, but without the same scale of data and resources, it won't provide an accurate representation of

production performance. Running all cells together also ensures that any dependencies and sequential processing are respected, providing a more realistic performance measurement.

### Question 50 of 200

---

A production cluster has 3 executor nodes and uses the same virtual machine type for the driver and executor.

When evaluating the Ganglia Metrics for this cluster, which indicator would signal a bottleneck caused by code executing on the driver?

- A. The five Minute Load Average remains consistent/flat
  - B. Bytes Received never exceeds 80 million bytes per second
  - C. Total Disk Space remains constant
  - D. Network I/O never spikes
- E. Overall cluster CPU utilization is around 25%

Answer

Discussion

Correct Answer: E

A bottleneck caused by code executing on the driver would be indicated by overall cluster CPU utilization being around 25%. This suggests that the driver node is likely overburdened, as it is consuming most of its CPU resources while the executor nodes are underutilized. In a properly balanced cluster, the CPU utilization should be spread more evenly across the nodes. If the driver is the bottleneck, it would prevent the executors from being effectively utilized, hence the low overall CPU utilization.

## Question 51 of 200

---

Where in the Spark UI can one diagnose a performance problem induced by not leveraging predicate push-down?

- A. In the Executor's log file, by grepping for "predicate push-down"
- B. In the Stage's Detail screen, in the Completed Stages table, by noting the size of data read from the Input column
- C. In the Storage Detail screen, by noting which RDDs are not stored on disk
- D. In the Delta Lake transaction log, by noting the column statistics
- E. In the Query Detail screen, by interpreting the Physical Plan

[Answer](#)[Discussion](#)

Correct Answer: B

Predicate push-down optimization reduces the amount of data read from the data source by applying filters early in the query process. To diagnose a performance problem caused by not leveraging predicate push-down, one should look at the size of the data read. The Stage's Detail screen in the Completed Stages table shows the size of data read from the Input column, which would reveal if too much data is being read because predicate push-down is not being used.

## Question 52 of 200

---

Review the following error traceback:

```
-----  
AnalysisException                                     Traceback (most recent call last)  
<command-3293767849433948> in <module>  
----> 1 display(df.select(3*"heartrate"))  
  
/databricks/spark/python/pyspark/sql/dataframe.py in select(self, *cols)  
    1690         [Row(name='Alice', age=12), Row(name='Bob', age=15)]  
    1691     """  
-> 1692         jdf = self._jdf.select(self._jcols(*cols))  
    1693         return DataFrame(jdf, self.sql_ctx)  
    1694  
  
/databricks/spark/python/lib/py4j-0.10.9-src.zip/py4j/java_gateway.py in __call__(self, *args)  
    1302             answer = self.gateway_client.send_command(command)  
-> 1304             return_value = get_return_value(  
    1305                 answer, self.gateway_client, self.target_id, self.name)  
    1306  
  
/databricks/spark/python/pyspark/sql/utils.py in deco(*a, **kw)  
    121         # Hide where the exception came from that shows a non-Pythonic  
    122         # JVM exception message.  
--> 123         raise converted from None  
    124     else:  
    125         raise  
  
AnalysisException: cannot resolve ``heartrateheartrateheartrate`` given input columns:  
[spark_catalog.database.table.device_id, spark_catalog.database.table.heartrate,  
spark_catalog.database.table.mrn, spark_catalog.database.table.time];  
'Project ['heartrateheartrateheartrate]  
+- SubqueryAlias spark_catalog.database.table  
  +- Relation[device_id#75L,heartrate#76,mrn#77L,time#78] parquet
```

Which statement describes the error being raised?

- A. The code executed was PySpark but was executed in a Scala notebook.
- B. There is no column in the table named heartrateheartrateheartrate
- C. There is a type error because a column object cannot be multiplied.
- D. There is a type error because a DataFrame object cannot be multiplied.

- E. There is a syntax error because the heartrate column is not correctly identified as a column.

Answer

Discussion

Correct Answer: B

The error message clearly states that the column heartrateheartrateheartrate cannot be resolved based on the input columns provided. This indicates that there is no column with the name heartrateheartrateheartrate in the table. The issue arises from the invalid column name generated by the expression `3*'heartrate'`, which results in `'heartrateheartrateheartrate'`. The DataFrame is expected to have a column named `heartrate`, not a string repetition of the column name.

Question 53 of 200

---

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

- A. sbt
- B. CRANC. npm
- C. pip
- D. Wheels
- E. jars

Answer

Discussion

Correct Answer: D

Wheels are the official built-package format for Python. They are a newer standard for packaging Python code, which includes compiled extensions and metadata, making them suitable for distributing custom Python packages. Databricks supports the use of wheels for installing custom Python code packages.

### Question 54 of 200

---

Which Python variable contains a list of directories to be searched when trying to locate required modules?

A. importlib.resource\_path

B. sys.path

C. os.path

D. pypi.path

E. pylib.source

Answer

Discussion

Correct Answer: B

The Python variable that contains a list of directories to be searched when trying to locate required modules is `sys.path`. This variable is part of the `sys` module and is a built-in list that includes the directory of the script being run, the PYTHONPATH, and the default installation-dependent directories.

## Question 55 of 200

---

Incorporating unit tests into a PySpark application requires upfront attention to the design of your jobs, or a potentially significant refactoring of existing code.

Which statement describes a main benefit that offset this additional effort?

- A. Improves the quality of your data
- B. Validates a complete use case of your application
- C. Troubleshooting is easier since all steps are isolated and tested individually
- D. Yields faster deployment and execution times
- E. Ensures that all steps interact correctly to achieve the desired end result

Answer

Discussion

Correct Answer: C

Incorporating unit tests into a PySpark application involves isolating and testing each step individually. This approach makes troubleshooting easier because you can identify and fix issues within specific units of code rather than debugging the entire application. Each part of the application is validated separately, ensuring robustness and stability of the code.

Prev

Next

## Question 56 of 200

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

[Answer](#)[Discussion](#)

## Correct Answer: A

Integration testing is a type of software testing where individual components of an application are combined and tested as a group. This type of testing aims to validate the interactions between subsystems or modules of an application to ensure that they work together as expected. The other options describe different aspects or types of testing: automated frameworks may or may not be used in integration testing (B), manual intervention is not inherently required (C), validating an application use case is broader than integration testing (D), and validating individual elements refers more to unit testing (E).

## Question 57 of 200

---

Which REST API call can be used to review the notebooks configured to run as tasks in a multi-task job?

A. /jobs/runs/list

B. /jobs/runs/get-output

C. /jobs/runs/get

D. /jobs/get

E. /jobs/list

Answer

Discussion

Correct Answer: D

The endpoint /jobs/get is used to retrieve details about a job, including the notebooks configured as tasks within a multi-task job. It provides the job's settings and includes information about the tasks associated with the job, making it the correct choice for reviewing configured notebooks.

---

## Question 58 of 200

---

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.

Answer

Discussion

Correct Answer: A

All logic expressed in the notebook associated with tasks A and B will have been successfully completed, with some operations in task C potentially completing before it fails. In this scenario, individual tasks execute until failure, and changes made by successful tasks are retained. However, changes in the failing task depend on the point of failure, meaning some operations within task C could have been completed before the failure occurred.

Question 59 of 200

A Delta Lake table was created with the below query:

```
CREATE TABLE prod.sales_by_stor
USING DELTA
LOCATION "/mnt/prod/sales_by_store"
```

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.

Answer

Discussion

Correct Answer: A

When you execute the command to rename a Delta Lake table, the table reference in the metastore is updated to reflect the new name. This operation does not involve moving or altering the data files themselves, and thus no actual data is changed. The metadata remains intact and consistent. The purpose of this command is purely administrative, to correct or update the table name in the metastore.

Question 60 of 200

---

The data engineering team maintains a table of aggregate statistics through batch nightly updates. This includes total sales for the previous day alongside totals and averages for a variety of time periods including the 7 previous days, year-to-date, and quarter-to-date. This table is named store\_sales\_summary and the schema is as follows:

```
store_id INT, total_sales_qtd FLOAT, avg_daily_sales_qtd FLOAT, total_sales_ytd FLOAT,  
avg_daily_sales_ytd FLOAT, previous_day_sales FLOAT, total_sales_7d FLOAT, avg_daily_sales_7d  
FLOAT, updated TIMESTAMP
```

The table `daily_store_sales` contains all the information needed to update `store_sales_summary`. The schema for this table is: `store_id INT, sales_date DATE, total_sales FLOAT`

If `daily_store_sales` is implemented as a Type 1 table and the `total_sales` column might be adjusted after manual data auditing, which approach is the safest to generate accurate reports in the `store_sales_summary` table?

- A. Implement the appropriate aggregate logic as a batch read against the `daily_store_sales` table and overwrite the `store_sales_summary` table with each Update.
- B. Implement the appropriate aggregate logic as a batch read against the `daily_store_sales` table and append new rows nightly to the `store_sales_summary` table.
- C. Implement the appropriate aggregate logic as a batch read against the `daily_store_sales` table and use upsert logic to update results in the `store_sales_summary` table.
- D. Implement the appropriate aggregate logic as a Structured Streaming read against the `daily_store_sales` table and use upsert logic to update results in the `store_sales_summary` table.
- E. Use Structured Streaming to subscribe to the change data feed for `daily_store_sales` and apply changes to the aggregates in the `store_sales_summary` table with each update.

Answer

Discussion

Correct Answer: A

The safest approach to generate accurate reports in the `store_sales_summary` table is to implement the appropriate aggregate logic as a batch read against the `daily_store_sales` table and overwrite the `store_sales_summary` table with each update. This ensures that all data is recalculated based on the most recent and accurate information, effectively handling any adjustments made to the `total_sales` column after manual data auditing. This method avoids complications that can arise from incremental updates or upserts and eliminates any potential inconsistencies that could result from partial updates.

## Question 61 of 200

A member of the data engineering team has submitted a short notebook that they wish to schedule as part of a larger data pipeline. Assume that the commands provided below produce the logically correct results when run as presented.

**Cmd 1**

```
rawDF = spark.table("raw_data")
```

**Cmd 2**

```
rawDF.printSchema()
```

**Cmd 3**

```
flattenedDF = rawDF.select("*", "values.*")
```

**Cmd 4**

```
finalDF = flattenedDF.drop("values")
```

**Cmd 5**

```
finalDF.explain()
```

**Cmd 6**

```
display(finalDF)
```

**Cmd 7**

```
finalDF.write.mode("append").saveAsTable("flat_data")
```

Which command should be removed from the notebook before scheduling it as a job?

A. Cmd 2

B. Cmd 3

C. Cmd 4

D. Cmd 5

E. Cmd 6

Answer

Discussion

Correct Answer: E

When scheduling a job, interactive commands designed for exploration and visualization, such as `display()`, should be removed. The `display()` function is intended for use in the notebook interface to visualize data during development and is not suitable for automated jobs in production. Removing Cmd 6 avoids unnecessary computation and potential issues in a production environment.

Question 62 of 200

---

The business reporting team requires that data for their dashboards be updated every hour. The total processing time for the pipeline that extracts transforms, and loads the data for their pipeline runs in 10 minutes.

Assuming normal operating conditions, which configuration will meet their service-level agreement requirements with the lowest cost?

A. Manually trigger a job anytime the business reporting team refreshes their dashboards

B. Schedule a job to execute the pipeline once an hour on a new job cluster

C. Schedule a Structured Streaming job with a trigger interval of 60 minutes

D. Schedule a job to execute the pipeline once an hour on a dedicated interactive cluster

E. Configure a job that executes every time new data lands in a given directory

Answer

Discussion

Correct Answer: B

The best option to meet the service-level agreement requirements with the lowest cost is to schedule a job to execute the pipeline once an hour on a new job cluster. This approach ensures that the data is updated every hour, meeting the requirement. Additionally, using a job cluster that is started and stopped for each run is more cost-effective than keeping a cluster running continuously, as it only incurs compute costs for the actual processing time, which is 10 minutes per hour.

Question 63 of 200

---

A Databricks SQL dashboard has been configured to monitor the total number of records present in a collection of Delta Lake tables using the following query pattern:

SELECT COUNT (\*) FROM table -

Which of the following describes how results are generated each time the dashboard is updated?

- A. The total count of rows is calculated by scanning all data files
- B. The total count of rows will be returned from cached results unless REFRESH is run
- C. The total count of records is calculated from the Delta transaction logs
- D. The total count of records is calculated from the parquet file metadata
- E. The total count of records is calculated from the Hive metastore

Answer

Discussion

Correct Answer: C

The total count of records is calculated from the Delta transaction logs. Delta Lake maintains a transaction log that records details about every change made to a table, including the number of records in each file. This allows the system to efficiently calculate the total number of records without needing to scan all the data files.

A Delta Lake table was created with the below query:

```
CREATE TABLE prod.sales_by_store
AS (
    SELECT *
    FROM prod.sales a
    INNER JOIN prod.store b
    ON a.store_id = b.store_id
)
```

Consider the following query:

```
DROP TABLE prod.sales_by_store -
```

If this statement is executed by a workspace admin, which result will occur?

- A. Nothing will occur until a COMMIT command is executed.
- B. The table will be removed from the catalog but the data will remain in storage.
- C. The table will be removed from the catalog and the data will be deleted.
- D. An error will occur because Delta Lake prevents the deletion of production data.
- E. Data will be marked as deleted but still recoverable with Time Travel.

Answer

Discussion

Correct Answer: C

In Delta Lake, when a DROP TABLE command is executed on a managed table, it will remove both the metadata entry for the table from the catalog and the data in storage associated with that table. This means that both the table definition and the data itself are deleted. Therefore, the correct answer is that the table will be removed from the catalog and the data will be deleted.

Question 65 of 200

---

Two of the most common data locations on Databricks are the DBFS root storage and external object storage mounted with dbutils.fs.mount().

Which of the following statements is correct?

- A. DBFS is a file system protocol that allows users to interact with files stored in object storage using syntax and guarantees similar to Unix file systems.
- B. By default, both the DBFS root and mounted data sources are only accessible to workspace administrators.
- C. The DBFS root is the most secure location to store data, because mounted storage volumes must have full public read and write permissions.
- D. Neither the DBFS root nor mounted storage can be accessed when using %sh in a Databricks notebook.
- E. The DBFS root stores files in ephemeral block volumes attached to the driver, while mounted directories will always persist saved data to external storage between sessions.

Answer

Discussion

Correct Answer: A

DBFS is a file system protocol that allows users to interact with files stored in object storage using syntax and guarantees similar to Unix file systems. It provides familiar file system semantics and is designed to function like a traditional file system, which aligns with POSIX-like behaviors seen in Unix file systems.

Prev

Next



Examice delivers accurate questions, trusted answers, and the best value in certification exam preparation.



[Terms](#)    [Privacy](#)    [Refund](#)    [DMCA](#)    [Contact](#)

---

Copyright © 2025 Examice. All rights reserved.

## Question 66 of 200

The following code has been migrated to a Databricks notebook from a legacy workload:

```
@sh
git clone https://github.com/foo/data_loader;
python ./data_loader/run.py;
mv ./output /dbfs/mnt/new_data
```

The code executes successfully and provides the logically correct results, however, it takes over 20 minutes to extract and load around 1 GB of data.

Which statement is a possible explanation for this behavior?

- A. %sh triggers a cluster restart to collect and install Git. Most of the latency is related to cluster startup time.
- B. Instead of cloning, the code should use %sh pip install so that the Python code can get executed in parallel across all nodes in a cluster.
- C. %sh does not distribute file moving operations; the final line of code should be updated to use %fs instead.
- D. Python will always execute slower than Scala on Databricks. The run.py script should be refactored to Scala.
- E. %sh executes shell code on the driver node. The code does not take advantage of the worker nodes or Databricks optimized Spark.

Answer

Discussion

Correct Answer: E

The %sh magic command in Databricks executes shell code on the driver node. This means that the commands run within this block do not utilize the distributed computing capabilities of Databricks, which leverages multiple

nodes for parallel processing. As a result, the operations such as the Git clone, Python script execution, and file move operations are limited to the resources of the driver node only. This lack of parallelism and distribution is a likely reason for the extended execution time for handling 1 GB of data.

### Question 67 of 200

---

The data science team has requested assistance in accelerating queries on free form text from user reviews. The data is currently stored in Parquet with the below schema:

```
item_id INT, user_id INT, review_id INT, rating FLOAT, review STRING
```

The review column contains the full text of the review left by the user. Specifically, the data science team is looking to identify if any of 30 key words exist in this field.

A junior data engineer suggests converting this data to Delta Lake will improve query performance.

Which response to the junior data engineer's suggestion is correct?

- A. Delta Lake statistics are not optimized for free text fields with high cardinality.
- B. Text data cannot be stored with Delta Lake.
- C. ZORDER ON review will need to be run to see performance gains.
- D. The Delta log creates a term matrix for free text fields to support selective filtering.
- E. Delta Lake statistics are only collected on the first 4 columns in a table.

Answer

Discussion

Correct Answer: A

Delta Lake statistics are not optimized for free text fields with high cardinality. The high cardinality of free-form text fields like the 'review' column means that each review text is unique or nearly unique. Consequently, Delta Lake's statistics and data skipping optimizations, which are most effective for columns with a limited set of distinct values, do not significantly improve the performance of queries that search for specific keywords within the text.

## Question 68 of 200

---

Assuming that the Databricks CLI has been installed and configured correctly, which Databricks CLI command can be used to upload a custom Python Wheel to object storage mounted with the DBFS for use with a production job?

A. configure

B. fs

C. jobs

D. libraries

E. workspace

Answer

Discussion

Correct Answer: B

The correct command for uploading a custom Python Wheel to object storage mounted with the DBFS for use with a production job is 'fs'. The 'fs' command in the Databricks CLI is used to interact with the Databricks File System (DBFS). Specifically, the 'fs cp' subcommand can be used to copy files from the local file system to DBFS, which includes uploading custom Python Wheels.

## Question 69 of 200

---

The business intelligence team has a dashboard configured to track various summary metrics for retail stores. This includes total sales for the previous day alongside totals and averages for a variety of time periods. The fields required to populate this dashboard have the following schema:

```
store_id INT, total_sales_qtd FLOAT, avg_daily_sales_qtd FLOAT, total_sales_ytd  
FLOAT, avg_daily_sales_ytd FLOAT, previous_day_sales FLOAT, total_sales_7d FLOAT,  
avg_daily_sales_7d FLOAT, updated TIMESTAMP
```

For demand forecasting, the Lakehouse contains a validated table of all itemized sales updated incrementally in near real-time. This table, named products\_per\_order, includes the following fields:

```
store_id INT, order_id INT, product_id INT, quantity INT, price FLOAT,  
order_timestamp TIMESTAMP
```

Because reporting on long-term sales trends is less volatile, analysts using the new dashboard only require data to be refreshed once daily. Because the dashboard will be queried interactively by many users throughout a normal business day, it should return results quickly and reduce total compute associated with each materialization.

Which solution meets the expectations of the end users while controlling and limiting possible costs?

- A. Populate the dashboard by configuring a nightly batch job to save the required values as a table overwritten with each update.
- B. Use Structured Streaming to configure a live dashboard against the products\_per\_order table within a Databricks notebook.
- C. Configure a webhook to execute an incremental read against products\_per\_order each time the dashboard is refreshed.
- D. Use the Delta Cache to persist the products\_per\_order table in memory to quickly update the dashboard with each query.
- E. Define a view against the products\_per\_order table and define the dashboard against this view.

Answer

Discussion

Correct Answer: A

The best solution is to configure a nightly batch job to save the required values as a table overwritten with each update. This approach meets the requirement for daily refreshes and ensures that the data is precomputed and stored in a format that allows for quick querying. Since the dashboard only needs to be refreshed once daily, a batch job is more cost-effective and efficient compared to real-time streaming or querying live data, which would require more compute resources and incur higher costs. Additionally, this method avoids the need for constant computation, thus reducing the overall system load during business hours when the dashboard is accessed by many users.

## Question 70 of 200

---

A data ingestion task requires a one-TB JSON dataset to be written out to Parquet with a target part-file size of 512 MB. Because Parquet is being used instead of Delta Lake, built-in file-sizing features such as Auto-Optimize & Auto-Compaction cannot be used.

Which strategy will yield the best performance without shuffling data?

- A. Set `spark.sql.files.maxPartitionBytes` to 512 MB, ingest the data, execute the narrow transformations, and then write to parquet.
- B. Set `spark.sql.shuffle.partitions` to 2,048 partitions ( $1\text{TB} * 1024 * 1024 / 512$ ), ingest the data, execute the narrow transformations, optimize the data by sorting it (which automatically repartitions the data), and then write to parquet.
- C. Set `spark.sql.adaptive.advisoryPartitionSizeInBytes` to 512 MB bytes, ingest the data, execute the narrow transformations, coalesce to 2,048 partitions ( $1\text{TB} * 1024 * 1024 / 512$ ), and then write to parquet.
- D. Ingest the data, execute the narrow transformations, repartition to 2,048 partitions ( $1\text{TB} * 1024 * 1024 / 512$ ), and then write to parquet.
- E. Set `spark.sql.shuffle.partitions` to 512, ingest the data, execute the narrow transformations, and then write to parquet.

Answer

Discussion

Correct Answer: D

Ingesting the data, executing the narrow transformations, and then repartitioning to 2,048 partitions ensures that each partition will approximate the target part-file size of 512 MB when writing to Parquet. This strategy directly controls the number of output partitions without requiring any shuffling, which can be expensive in terms of performance.

Prev

Next



Examice delivers accurate questions, trusted answers, and the best value in certification exam preparation.



[Terms](#)   [Privacy](#)   [Refund](#)   [DMCA](#)   [Contact](#)

---

Copyright © 2025 Examice. All rights reserved.

## Question 71 of 200

A junior data engineer has been asked to develop a streaming data pipeline with a grouped aggregation using DataFrame df. The pipeline needs to calculate the average humidity and average temperature for each non-overlapping five-minute interval. Incremental state information should be maintained for 10 minutes for late-arriving data.

Streaming DataFrame df has the following schema:

"device\_id INT, event\_time TIMESTAMP, temp FLOAT, humidity FLOAT"

Code block:

```
df._____
    .groupBy(
        window("event_time", "5 minutes").alias("time"),
        "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. withWatermark("event\_time", "10 minutes")
- B. awaitArrival("event\_time", "10 minutes")
- C. await("event\_time + '10 minutes'")
- D. slidingWindow("event\_time", "10 minutes")

- E. `delayWrite("event_time", "10 minutes")`

Answer

Discussion

Correct Answer: A

In a streaming data pipeline using Spark Structured Streaming, to handle and manage late-arriving data while performing aggregations, the `withWatermark` function is used. This function defines how long the framework should wait for late data to arrive before it is considered too late. The watermark value, in this case, 10 minutes, specifies the threshold for lateness. Therefore, the code block should use `withWatermark` to ensure incremental state information is maintained for 10 minutes for late-arriving data.

Question 72 of 200

---

A data team's Structured Streaming job is configured to calculate running aggregates for item sales to update a downstream marketing dashboard. The marketing team has introduced a new promotion, and they would like to add a new field to track the number of times this promotion code is used for each item. A junior data engineer suggests updating the existing query as follows. Note that proposed changes are in bold.

Original query:

```
df.groupBy("item")
    .agg(count("item").alias("total_count"),
          mean("sale_price").alias("avg_price"))
    .writeStream
    .outputMode("complete")
    .option("checkpointLocation", "/item_agg/__checkpoint")
    .start("/item_agg")
```

Proposed query:

```
df.groupBy("item")
    .agg(count("item").alias("total_count"),
          mean("sale_price").alias("avg_price"))
    .writeStream
    .outputMode("complete")
    .option("checkpointLocation", "/item_agg/__checkpoint")
    .start("/item_agg")
```

Proposed query:

```
.start("/item_agg")
```

Which step must also be completed to put the proposed query into production?

- A. Specify a new checkpointLocation
- B. Increase the shuffle partitions to account for additional aggregates
- C. Run REFRESH TABLE delta.'/item\_agg'
- D. Register the data in the "/item\_agg" directory to the Hive metastore
- E. Remove .option('mergeSchema', 'true') from the streaming write

Answer

Discussion

Correct Answer: A

A

Question 73 of 200

---

A Structured Streaming job deployed to production has been resulting in higher than expected cloud storage costs. At present, during normal execution, each microbatch of data is processed in less than 3s; at least 12 times per minute, a microbatch is processed that contains 0 records. The streaming write was configured using the default trigger settings. The production job is currently scheduled alongside many other Databricks jobs in a workspace with instance pools provisioned to reduce start-up time for jobs with batch execution.

Holding all other variables constant and assuming records need to be processed in less than 10 minutes, which adjustment will meet the requirement?

- A. Set the trigger interval to 3 seconds; the default trigger interval is consuming too many records per batch, resulting in spill to disk that can increase volume costs.
- B. Increase the number of shuffle partitions to maximize parallelism, since the trigger interval cannot be modified without modifying the checkpoint directory.
- C. Set the trigger interval to 10 minutes; each batch calls APIs in the source storage account, so decreasing trigger frequency to maximum allowable threshold should minimize this cost.

- D. Set the trigger interval to 500 milliseconds; setting a small but non-zero trigger interval ensures that the source is not queried too frequently.
- E. Use the trigger once option and configure a Databricks job to execute the query every 10 minutes; this approach minimizes costs for both compute and storage.

Answer

Discussion

Correct Answer: E

Setting the trigger interval to 10 minutes would block a VM permanently and could lead to inefficient use of resources. Using the 'trigger once' option and scheduling the job to execute every 10 minutes will minimize costs for both compute and storage. This method allows efficient use of instance pools by grabbing a VM only when needed, running the job, and returning the VM to the pool, ensuring no unnecessary resource usage while still processing records within the required 10-minute window.

Question 74 of 200

---

Which statement describes the correct use of `pyspark.sql.functions.broadcast`?

- A. It marks a column as having low enough cardinality to properly map distinct values to available partitions, allowing a broadcast join.
- B. It marks a column as small enough to store in memory on all executors, allowing a broadcast join.
- C. It caches a copy of the indicated table on attached storage volumes for all active clusters within a Databricks workspace.
- D. It marks a DataFrame as small enough to store in memory on all executors, allowing a broadcast join.
- E. It caches a copy of the indicated table on all nodes in the cluster for use in all future queries during the cluster lifetime.

Answer

Discussion

Correct Answer: D

The correct statement is that it marks a DataFrame as small enough to store in memory on all executors, allowing a broadcast join. This optimization technique is used in PySpark to speed up the join process by broadcasting the smaller DataFrame to all executor nodes, thereby avoiding the need to shuffle the larger DataFrame across the cluster.

Question 75 of 200

---

A data engineer is configuring a pipeline that will potentially see late-arriving, duplicate records.

In addition to de-duplicating records within the batch, which of the following approaches allows the data engineer to deduplicate data against previously processed records as it is inserted into a Delta table?

- A. Set the configuration delta.deduplicate = true.
- B. VACUUM the Delta table after each batch completes.
- C. Perform an insert-only merge with a matching condition on a unique key.
- D. Perform a full outer join on a unique key and overwrite existing data.
- E. Rely on Delta Lake schema enforcement to prevent duplicate records.

Answer

Discussion

Correct Answer: C

To efficiently deduplicate data against previously processed records as it is inserted into a Delta table, performing an insert-only merge with a matching condition on a unique key is the appropriate approach. This technique allows the data engineer to perform upsert operations, meaning that if an incoming record matches an existing record based on the unique key, the existing record can be updated or ignored to handle duplicates. If there is no match, the new record will be inserted. This ensures that duplicates are managed both within the current batch and against previously processed records.

## Question 76 of 200

A data pipeline uses Structured Streaming to ingest data from Apache Kafka to Delta Lake. Data is being stored in a bronze table, and includes the Kafka-generated timestamp, key, and value. Three months after the pipeline is deployed, the data engineering team has noticed some latency issues during certain times of the day.

A senior data engineer updates the Delta Table's schema and ingestion logic to include the current timestamp (as recorded by Apache Spark) as well as the Kafka topic and partition. The team plans to use these additional metadata fields to diagnose the transient processing delays.

Which limitation will the team face while diagnosing this problem?

- A. New fields will not be computed for historic records.
- B. Spark cannot capture the topic and partition fields from a Kafka source.
- C. New fields cannot be added to a production Delta table.
- D. Updating the table schema will invalidate the Delta transaction log metadata.
- E. Updating the table schema requires a default value provided for each field added.

[Answer](#)[Discussion](#)

Correct Answer: A

When updating the schema of a Delta table to include new fields, these fields will only be populated for new records ingested after the schema update. The new fields cannot be retroactively computed or added to historic records already stored in the Delta table. Therefore, while the new metadata fields like the current timestamp, Kafka topic, and partition will be available for diagnosing issues in data ingested after the update, they will not exist for the historical data, limiting the diagnostic scope to newly ingested records.

## Question 77 of 200

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)`

Answer

Discussion

Correct Answer: E

To enable near real-time workloads using Databricks Auto Loader with automatic schema detection and evolution, the function should be set up to utilize streaming writes. This requires the use of .writeStream for continuous processing, specifying the checkpoint location for fault tolerance, and setting mergeSchema to True to allow automatic schema evolution. Option E accurately reflects this setup by including these necessary configurations.

Question 78 of 200

---

The data engineering team maintains the following code:

```
import pyspark.sql.functions as F

(spark.table("silver_customer_sales")
 .groupBy("customer_id")
 .agg(
     F.min("sale_date").alias("first_transaction_date"),
     F.max("sale_date").alias("last_transaction_date"),
     F.mean("sale_total").alias("average_sales"),
     F.countDistinct("order_id").alias("total_orders"),
     F.sum("sale_total").alias("lifetime_value")
 ) .write
 .mode("overwrite")
 .table("gold_customer_lifetime_sales_summary")
)
```

Assuming that this code produces logically correct results and the data in the source table has been de-duplicated and validated, which statement describes what will occur when this code is executed?

- A. The silver\_customer\_sales table will be overwritten by aggregated values calculated from all records in the gold\_customer\_lifetime\_sales\_summary table as a batch job.
- B. A batch job will update the gold\_customer\_lifetime\_sales\_summary table, replacing only those rows that have different values than the current version of the table, using customer\_id as the primary key.
- C. The gold\_customer\_lifetime\_sales\_summary table will be overwritten by aggregated values calculated from all records in the silver\_customer\_sales table as a batch job.

- D. An incremental job will leverage running information in the state store to update aggregate values in the gold\_customer\_lifetime\_sales\_summary table.
- E. An incremental job will detect if new rows have been written to the silver\_customer\_sales table; if new rows are detected, all aggregates will be recalculated and used to overwrite the gold\_customer\_lifetime\_sales\_summary table.

Answer

Discussion

Correct Answer: C

When this code executes, it performs a batch aggregation operation on the 'silver\_customer\_sales' table grouped by 'customer\_id'. It calculates the first and last transaction dates, the average sales, the total number of distinct orders, and the lifetime value of sales for each customer. The .mode('overwrite') operation specifies that the output table 'gold\_customer\_lifetime\_sales\_summary' should be overwritten with the result of this aggregation. This means that every time this code runs, it replaces the existing 'gold\_customer\_lifetime\_sales\_summary' table with a new version that reflects the aggregate calculations of the 'silver\_customer\_sales' table.

Question 79 of 200

---

The data architect has mandated that all tables in the Lakehouse should be configured as external (also known as "unmanaged") Delta Lake tables.

Which approach will ensure that this requirement is met?

- A. When a database is being created, make sure that the LOCATION keyword is used.
- B. When configuring an external data warehouse for all table storage, leverage Databricks for all ELT.
- C. When data is saved to a table, make sure that a full file path is specified alongside the Delta format.
- D. When tables are created, make sure that the EXTERNAL keyword is used in the CREATE TABLE statement.
- E. When the workspace is being configured, make sure that external cloud object storage has been mounted.

Answer

Discussion

Correct Answer: C

To ensure that tables in the Lakehouse are configured as external (unmanaged) Delta Lake tables, it is necessary to specify the full file path when saving data to a table. This step ensures that the data is stored in the location specified by the user, rather than the default directory managed by Delta Lake. By doing so, the table will be considered external since the data resides outside the managed storage, fulfilling the requirement set by the data architect.

Question 80 of 200

---

The marketing team is looking to share data in an aggregate table with the sales organization, but the field names used by the teams do not match, and a number of marketing-specific fields have not been approved for the sales org.

Which of the following solutions addresses the situation while emphasizing simplicity?

- A. Create a view on the marketing table selecting only those fields approved for the sales team; alias the names of any fields that should be standardized to the sales naming conventions.
- B. Create a new table with the required schema and use Delta Lake's DEEP CLONE functionality to sync up changes committed to one table to the corresponding table.
- C. Use a CTAS statement to create a derivative table from the marketing table; configure a production job to propagate changes.
- D. Add a parallel table write to the current production pipeline, updating a new sales table that varies as required from the marketing table.
- E. Instruct the marketing team to download results as a CSV and email them to the sales organization.

Answer

Discussion

Correct Answer: A

Creating a view on the marketing table is the best solution for sharing data with the sales organization while keeping it simple. By selecting only the fields approved for the sales team and aliasing names to match the sales naming conventions, this approach ensures that no unauthorized data is shared and that the naming is consistent. Additionally, views are easy to create and maintain, and they always reflect the most up-to-date data without the need for complex synchronization processes or additional data storage.

## Question 81 of 200

A CHECK constraint has been successfully added to the Delta table named activity\_details using the following logic:

```
ALTER TABLE activity_details
ADD CONSTRAINT valid_coordinates
CHECK (
    latitude >= -90 AND
    latitude <= 90 AND
    longitude >= -180 AND
    longitude <= 180);
```

A batch job is attempting to insert new records to the table, including a record where latitude = 45.50 and longitude = 212.67.

Which statement describes the outcome of this batch insert?

- A. The write will fail when the violating record is reached; any records previously processed will be recorded to the target table.
- B. The write will fail completely because of the constraint violation and no records will be inserted into the target table.
- C. The write will insert all records except those that violate the table constraints; the violating records will be recorded to a quarantine table.
- D. The write will include all records in the target table; any violations will be indicated in the boolean column named valid\_coordinates.
- E. The write will insert all records except those that violate the table constraints; the violating records will be reported in a warning log.

Answer

Discussion

Correct Answer: B

When inserting records into a Delta table with a CHECK constraint, if any record violates the constraint, the entire write operation will fail. This ensures the atomicity of the transaction, meaning that either all records are inserted, or none are inserted. In this case, the record with a longitude of 212.67 violates the constraint that longitude must be between -180 and 180 degrees. Hence, the entire batch insert will fail, and no records will be inserted into the target table.

Question 82 of 200

---

A junior data engineer has manually configured a series of jobs using the Databricks Jobs UI. Upon reviewing their work, the engineer realizes that they are listed as the "Owner" for each job. They attempt to transfer "Owner" privileges to the "DevOps" group, but cannot successfully accomplish this task.

Which statement explains what is preventing this privilege transfer?

- A. Databricks jobs must have exactly one owner; "Owner" privileges cannot be assigned to a group.
- B. The creator of a Databricks job will always have "Owner" privileges; this configuration cannot be changed.
- C. Other than the default "admins" group, only individual users can be granted privileges on jobs.
- D. A user can only transfer job ownership to a group if they are also a member of that group.
- E. Only workspace administrators can grant "Owner" privileges to a group.

Answer

Discussion

Correct Answer: A

Databricks jobs must have exactly one owner; 'Owner' privileges cannot be assigned to a group. This ensures the job has a single point of accountability and control. While other permissions can be delegated using Access Control Lists (ACLs), the 'Owner' role itself cannot be transferred to a group.

### Question 83 of 200

---

All records from an Apache Kafka producer are being ingested into a single Delta Lake table with the following schema:

key BINARY, value BINARY, topic STRING, partition LONG, offset LONG, timestamp LONG

There are 5 unique topics being ingested. Only the "registration" topic contains Personal Identifiable Information (PII). The company wishes to restrict access to PII. The company also wishes to only retain records containing PII in this table for 14 days after initial ingestion. However, for non-PII information, it would like to retain these records indefinitely.

Which of the following solutions meets the requirements?

- A. All data should be deleted biweekly; Delta Lake's time travel functionality should be leveraged to maintain a history of non-PII information.
- B. Data should be partitioned by the registration field, allowing ACLs and delete statements to be set for the PII directory.
- C. Because the value field is stored as binary data, this information is not considered PII and no special precautions should be taken.
- D. Separate object storage containers should be specified based on the partition field, allowing isolation at the storage level.
- E. Data should be partitioned by the topic field, allowing ACLs and delete statements to leverage partition boundaries.

Answer

Discussion

Correct Answer: E

Data should be partitioned by the topic field because this allows for access control lists (ACLs) and delete statements to be applied effectively. By partitioning by the topic, you can isolate the 'registration' topic, which contains PII, and set retention policies to delete records after 14 days. Non-PII data can remain in other partitions indefinitely, thus meeting both the privacy and data retention requirements.

#### Question 84 of 200

---

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.

Answer

Discussion

Correct Answer: D

Group members have been granted USAGE and SELECT privileges on the prod database. The USAGE privilege allows them to access the database context and list tables, while the SELECT privilege allows them to run queries on tables and views. They do not have permissions to create or modify tables or views. Thus, they are only able to query all tables and views in the prod database, but cannot create or edit anything in the database.

Question 85 of 200

---

A distributed team of data analysts share computing resources on an interactive cluster with autoscaling configured. In order to better manage costs and query throughput, the workspace administrator is hoping to evaluate whether cluster upscaling is caused by many concurrent users or resource-intensive queries.

In which location can one review the timeline for cluster resizing events?

A. Workspace audit logs

B. Driver's log file

C. Ganglia

D. Cluster Event Log

E. Executor's log file

Answer

Discussion

Correct Answer: D

To evaluate the timeline for cluster resizing events, one should review the Cluster Event Log. The Cluster Event Log provides detailed information about various events affecting the cluster, including resizing events. It displays the timestamp, event type (e.g., CLUSTER\_RESIZED), and relevant details for each event, which is essential for understanding the causes behind autoscaling events.

## Question 86 of 200

When evaluating the Ganglia Metrics for a given cluster with 3 executor nodes, which indicator would signal proper utilization of the VM's resources?

- A. The five Minute Load Average remains consistent/flat
- B. Bytes Received never exceeds 80 million bytes per second
- C. Network I/O never spikes
- D. Total Disk Space remains constant
- E. CPU Utilization is around 75%

[Answer](#)[Discussion](#)

Correct Answer: E

CPU Utilization is a primary indicator of how well the VMs' computational resources are being used. An ideal CPU utilization of around 75% suggests that the CPUs are being effectively utilized without being overburdened, providing room for handling additional load and minimizing the risks of bottlenecks. This level of usage implies that tasks are being processed efficiently, making it a key metric for proper resource utilization in a distributed computing environment.

## Question 87 of 200

---

Which of the following technologies can be used to identify key areas of text when parsing Spark Driver log4j output?

- A. Regex
- B. Julia
- C. pyspspark.ml.feature
- D. Scala Datasets
- E. C++

Answer

Discussion

Correct Answer: A

Regex (regular expressions) is widely used for pattern matching and text processing, making it highly suitable for identifying key areas in log files such as Spark Driver log4j output. By defining specific patterns, regex allows you to search for and extract relevant information like error messages, timestamps, and log levels efficiently. Other options like Julia, pyspspark.ml.feature, Scala Datasets, and C++ are not primarily designed for text parsing tasks of this nature.

---

## Question 88 of 200

You are testing a collection of mathematical functions, one of which calculates the area under a curve as described by another function.

```
assert(myIntegrate(lambda x: x*x, 0, 3) [0] == 9)
```

Which kind of test would the above line exemplify?

- A. Unit
- B. Manual
- C. Functional
- D. Integration
- E. End-to-end

Answer

Discussion

**Correct Answer: A**

The test checks a single function, `myIntegrate`, and verifies its output for a specific input, `lambda x: x*x`, within a specified range from 0 to 3. This is characteristic of a unit test, which is designed to test individual components or functions in isolation to ensure they perform as expected.

Question 89 of 200

---

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 task A fails during a scheduled run, which statement describes the results of this run?

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

Answer

Discussion

Correct Answer: D

In Databricks job execution, if a task that other tasks depend on fails, the dependent tasks will not be executed. Since Tasks B and C depend on Task A and Task A failed, Tasks B and C will be skipped. Some logic expressed in Task A may have been committed before the failure occurred.

Question 90 of 200

---

Which statement regarding Spark configuration on the Databricks platform is true?

- A. The Databricks REST API can be used to modify the Spark configuration properties for an interactive cluster without interrupting jobs currently running on the cluster.
  - B. Spark configurations set within a notebook will affect all SparkSessions attached to the same interactive cluster.
  - C. Spark configuration properties can only be set for an interactive cluster by creating a global init script.
- D. Spark configuration properties set for an interactive cluster with the Clusters UI will impact all notebooks attached to that cluster.
- E. When the same Spark configuration property is set for an interactive cluster and a notebook attached to that cluster, the notebook setting will always be ignored.

Answer

Discussion

Correct Answer: D

On the Databricks platform, Spark configuration properties set using the Clusters UI will apply to the entire interactive cluster, impacting all notebooks attached to that cluster. This is because configurations at the cluster level ensure that consistent settings are applied across all SparkSessions within the cluster, providing a uniform environment for all running notebooks.

Prev

Next

## Question 91 of 200

A developer has successfully configured their credentials for Databricks Repos and cloned a remote Git repository. They do not have privileges to make changes to the main branch, which is the only branch currently visible in their workspace.

Which approach allows this user to share their code updates without the risk of overwriting the work of their teammates?

- A. Use Repos to checkout all changes and send the git diff log to the team.
- B. Use Repos to create a fork of the remote repository, commit all changes, and make a pull request on the source repository.
- C. Use Repos to pull changes from the remote Git repository; commit and push changes to a branch that appeared as changes were pulled.
- D. Use Repos to merge all differences and make a pull request back to the remote repository.
- E. Use Repos to create a new branch, commit all changes, and push changes to the remote Git repository.

[Answer](#)[Discussion](#)

Correct Answer: E

To share code updates without the risk of overwriting teammates' work, the developer should use Repos to create a new branch, commit all changes, and push changes to the remote Git repository. This approach allows the developer to keep the main branch untouched while working on their updates. After pushing the changes to the new branch, they can then create a pull request, enabling their teammates to review the changes before they are merged into the main branch. This method ensures the main branch remains stable and prevents any accidental overwrites.

Question 92 of 200

---

In order to prevent accidental commits to production data, a senior data engineer has instituted a policy that all development work will reference clones of Delta Lake tables. After testing both DEEP and SHALLOW CLONE, development tables are created using SHALLOW CLONE.

A few weeks after initial table creation, the cloned versions of several tables implemented as Type 1 Slowly Changing Dimension (SCD) stop working. The transaction logs for the source tables show that VACUUM was run the day before.

Which statement describes why the cloned tables are no longer working?

- A. Because Type 1 changes overwrite existing records, Delta Lake cannot guarantee data consistency for cloned tables.
- B. Running VACUUM automatically invalidates any shallow clones of a table; DEEP CLONE should always be used when a cloned table will be repeatedly queried.
- C. Tables created with SHALLOW CLONE are automatically deleted after their default retention threshold of 7 days.
- D. The metadata created by the CLONE operation is referencing data files that were purged as invalid by the VACUUM command.
- E. The data files compacted by VACUUM are not tracked by the cloned metadata; running REFRESH on the cloned table will pull in recent changes.

Answer

Discussion

Correct Answer: D

Shallow clones in Delta Lake only duplicate the metadata of the table being cloned, without copying the actual data files. These clones reference the source table's data files. When the VACUUM command is executed, it removes data files that are no longer referenced by the source table's transaction log. However, the shallow clones still reference these purged data files. As a result, the metadata created by the CLONE operation in the shallow clones is referencing data files that were purged by the VACUUM command, causing the cloned tables to stop working.

#### Question 93 of 200

---

You are performing a join operation to combine values from a static userLookup table with a streaming DataFrame streamingDF.

Which code block attempts to perform an invalid stream-static join?

- A. `userLookup.join(streamingDF, ["userid"], how="inner")`
- B. `streamingDF.join(userLookup, ["user_id"], how="outer")`
- C. `streamingDF.join(userLookup, ["user_id"], how="left")`
- D. `streamingDF.join(userLookup, ["userid"], how="inner")`
- E. `userLookup.join(streamingDF, ["user_id"], how="right")`

Answer

Discussion

Correct Answer: B

Performing an outer join between a streaming DataFrame and a static DataFrame is invalid. This is because outer joins involve matching rows from both DataFrames, and if there are unmatched rows from the streaming DataFrame, it will not have the necessary data at all times since streaming data continuously evolves. Hence, it cannot return a complete set of results for unmatched data.

#### Question 94 of 200

---

Spill occurs as a result of executing various wide transformations. However, diagnosing spill requires one to proactively look for key indicators.

Where in the Spark UI are two of the primary indicators that a partition is spilling to disk?

- A. Query's detail screen and Job's detail screen
- B. Stage's detail screen and Executor's log files
- C. Driver's and Executor's log files
- D. Executor's detail screen and Executor's log files
- E. Stage's detail screen and Query's detail screen

Answer

Discussion

Correct Answer: B

The Stage's detail screen provides key metrics about each stage of a job, including the amount of data that has been spilled to disk. High numbers in the "Spill (Memory)" or "Spill (Disk)" columns are indicators of spill. Additionally, the Executor's log files contain messages like "Spilling UnsafeExternalSorter to disk" or "Task memory spill", which indicate that a partition is spilling to disk because the task ran out of memory.

---

Question 95 of 200

---

A task orchestrator has been configured to run two hourly tasks. First, an outside system writes Parquet data to a directory mounted at /mnt/raw\_orders/. After this data is written, a Databricks job containing the following code is executed:

```
(spark.readStream
    .format("parquet")
    .load("/mnt/raw_orders/")
    .withWatermark("time", "2 hours")
    .dropDuplicates(["customer_id", "order_id"])
    .writeStream
    .trigger(once=True)
    .table("orders")
)
```

Assume that the fields customer\_id and order\_id serve as a composite key to uniquely identify each order, and that the time field indicates when the record was queued in the source system.

If the upstream system is known to occasionally enqueue duplicate entries for a single order hours apart, which statement is correct?

- A. Duplicate records enqueued more than 2 hours apart may be retained and the orders table may contain duplicate records with the same customer\_id and order\_id.
- B. All records will be held in the state store for 2 hours before being deduplicated and committed to the orders table.
- C. The orders table will contain only the most recent 2 hours of records and no duplicates will be present.
- D. Duplicate records arriving more than 2 hours apart will be dropped, but duplicates that arrive in the same batch may both be written to the orders table.
- E. The orders table will not contain duplicates, but records arriving more than 2 hours late will be ignored and missing from the table.

Answer

Discussion

Correct Answer: A

The provided Databricks job reads a stream of Parquet data, applies a watermark on the 'time' field, and deduplicates records based on 'customer\_id' and 'order\_id' only within the specified 2-hour window. This means that if duplicate records are enqueued more than 2 hours apart, they will not be recognized as duplicates and will both be written to the 'orders' table. Hence, the table may contain duplicate records with the same 'customer\_id' and 'order\_id' if they were queued more than 2 hours apart.

## Question 96 of 200

A junior data engineer is migrating a workload from a relational database system to the Databricks Lakehouse. The source system uses a star schema, leveraging foreign key constraints and multi-table inserts to validate records on write.

Which consideration will impact the decisions made by the engineer while migrating this workload?

- A. Databricks only allows foreign key constraints on hashed identifiers, which avoid collisions in highly-parallel writes.
- B. Databricks supports Spark SQL and JDBC; all logic can be directly migrated from the source system without refactoring.
- C. Committing to multiple tables simultaneously requires taking out multiple table locks and can lead to a state of deadlock.
- D. All Delta Lake transactions are ACID compliant against a single table, and Databricks does not enforce foreign key constraints.
- E. Foreign keys must reference a primary key field; multi-table inserts must leverage Delta Lake's upsert functionality.

[Answer](#)[Discussion](#)

Correct Answer: D

One major consideration while migrating a workload to Databricks Lakehouse is that all Delta Lake transactions are ACID compliant against a single table only, and Databricks does not enforce foreign key constraints. This means that the engineer will need to manage referential integrity manually, as the system will not automatically enforce these constraints.

## Question 97 of 200

---

A data architect has heard about Delta Lake's built-in versioning and time travel capabilities. For auditing purposes, they have a requirement to maintain a full record of all valid street addresses as they appear in the customers table.

The architect is interested in implementing a Type 1 table, overwriting existing records with new values and relying on Delta Lake time travel to support long-term auditing. A data engineer on the project feels that a Type 2 table will provide better performance and scalability.

Which piece of information is critical to this decision?

- A.** Data corruption can occur if a query fails in a partially completed state because Type 2 tables require setting multiple fields in a single update.
- B.** Shallow clones can be combined with Type 1 tables to accelerate historic queries for long-term versioning.
- C.** Delta Lake time travel cannot be used to query previous versions of these tables because Type 1 changes modify data files in place.
- D.** Delta Lake time travel does not scale well in cost or latency to provide a long-term versioning solution.
- E.** Delta Lake only supports Type 0 tables; once records are inserted to a Delta Lake table, they cannot be modified.

Answer

Discussion

Correct Answer: D

Delta Lake time travel does not scale well in cost or latency to provide a long-term versioning solution. This is because each write operation generates a new version of the data, leading to substantial storage consumption over time. Additionally, querying older versions involves scanning numerous files, resulting in increased query latency. Therefore, for long-term versioning, Type 2 tables, which track changes using separate records instead of versioning, might be a more suitable and efficient choice.

---

Question 98 of 200

A table named user\_ltv is 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.

The user\_ltv table has the following schema:

```
email STRING, age INT, ltv INT
```

The following view definition is executed:

```
CREATE VIEW user_ltv_no_minors AS
SELECT email, age, ltv
FROM user_ltv
WHERE
CASE
    WHEN is_member("auditing") THEN TRUE
    ELSE age >= 18
END
```

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

```
SELECT * FROM user_ltv_no_minors
```

Which statement describes the results returned by this query?

- A. All columns will be displayed normally for those records that have an age greater than 17; records not meeting this condition will be omitted.
- B. All age values less than 18 will be returned as null values, all other columns will be returned with the values in user\_ltv.
- C. All values for the age column will be returned as null values, all other columns will be returned with the values in user\_ltv.
- D. All records from all columns will be displayed with the values in user\_ltv.
- E. All columns will be displayed normally for those records that have an age greater than 18; records not meeting this condition will be omitted.

Answer

Discussion

Correct Answer: A

The query returns all columns of the table for records where the age is 18 or older because of the condition 'age >= 18'. Therefore, records with an age value less than 18 will be omitted. This means that records will only be displayed normally if the age is greater than or equal to 18.

---

The data governance team is reviewing code used for deleting records for compliance with GDPR. The following logic has been implemented to propagate delete requests from the user\_lookup table to the user\_aggregates table.

```
(spark.read
    .format("delta")
    .option("readChangeData", True)
    .option("startingTimestamp", '2021-08-22 00:00:00')
    .option("endingTimestamp", '2021-08-29 00:00:00')
    .table("user_lookup")
    .createOrReplaceTempView("changes"))

spark.sql("""
    DELETE FROM user_aggregates
    WHERE user_id IN (
        SELECT user_id
        FROM changes
        WHERE _change_type='delete'
    )
""")
"""
```

Assuming that user\_id is a unique identifying key and that all users that have requested deletion have been removed from the user\_lookup table, which statement describes whether successfully executing the above logic guarantees that the records to be deleted from the user\_aggregates table are no longer accessible and why?

- A. No; the Delta Lake DELETE command only provides ACID guarantees when combined with the MERGE INTO command.
- B. No; files containing deleted records may still be accessible with time travel until a VACUUM command is used to remove invalidated data files.
- C. Yes; the change data feed uses foreign keys to ensure delete consistency throughout the Lakehouse.
- D. Yes; Delta Lake ACID guarantees provide assurance that the DELETE command succeeded fully and permanently purged these records.
- E. No; the change data feed only tracks inserts and updates, not deleted records.

Answer

Discussion

Correct Answer: B

No; files containing deleted records may still be accessible with time travel until a VACUUM command is used to remove invalidated data files. In Delta Lake, data deletion does not immediately remove the data files containing

the deleted records. The deleted data can still be accessed through Delta Lake's time travel feature until the VACUUM command is executed to remove the invalidated data files permanently.

#### Question 100 of 200

---

The data engineering team has been tasked with configuring connections to an external database that does not have a supported native connector with Databricks. The external database already has data security configured by group membership. These groups map directly to user groups already created in Databricks that represent various teams within the company.

A new login credential has been created for each group in the external database. The Databricks Utilities Secrets module will be used to make these credentials available to Databricks users.

Assuming that all the credentials are configured correctly on the external database and group membership is properly configured on Databricks, which statement describes how teams can be granted the minimum necessary access to using these credentials?

- A. "Manage" permissions should be set on a secret key mapped to those credentials that will be used by a given team.
- B. "Read" permissions should be set on a secret key mapped to those credentials that will be used by a given team.
- C. "Read" permissions should be set on a secret scope containing only those credentials that will be used by a given team.
- D. "Manage" permissions should be set on a secret scope containing only those credentials that will be used by a given team.

No additional configuration is necessary as long as all users are configured as administrators in the workspace where secrets have been added.

Answer

Discussion

Correct Answer: C

In Databricks, secret scopes are used to manage and organize secrets effectively. By setting 'Read' permissions on a secret scope containing the credentials, the team is granted access to only the necessary credentials without providing additional privileges. This approach ensures that teams have the minimum necessary access and maintains a scalable and organized method of handling secrets. 'Manage' permissions would grant more access than needed and permissions set at the secret key level lack the organizational benefits provided by secret scopes.

## Question 101 of 200

Which indicators would you look for in the Spark UI's Storage tab to signal that a cached table is not performing optimally? Assume you are using Spark's MEMORY\_ONLY storage level.

- A. Size on Disk is < Size in Memory
- B. The RDD Block Name includes the "\*" annotation signaling a failure to cache
- C. Size on Disk is > 0
- D. The number of Cached Partitions > the number of Spark Partitions
- E. On Heap Memory Usage is within 75% of Off Heap Memory Usage

[Answer](#)[Discussion](#)

Correct Answer: C

When using Spark's MEMORY\_ONLY storage level, the data should ideally be fully cached in memory. If the Size on Disk is greater than 0, it indicates that some data has spilled to disk, which can degrade performance because reading from disk is slower than reading from memory. Therefore, having any data on disk means the cache is not performing optimally.

Question 102 of 200

---

What is the first line of a Databricks Python notebook when viewed in a text editor?

- A. %python
- B. // Databricks notebook source
- C. # Databricks notebook source
- D. -- Databricks notebook source
- E. # MAGIC %python

Answer

Discussion

Correct Answer: C

The first line of a Databricks Python notebook when viewed in a text editor is '# Databricks notebook source'. The '#' symbol is used for comments in Python, and this comment line indicates the start of the notebook's source code in the text file.

Question 103 of 200

---

Which statement describes a key benefit of an end-to-end test?

- A. Makes it easier to automate your test suite
- B. Pinpoints errors in the building blocks of your application
- C. Provides testing coverage for all code paths and branches
- D. Closely simulates real world usage of your application
- E. Ensures code is optimized for a real-life workflow

Answer

Discussion

Correct Answer: D

A key benefit of an end-to-end test is that it closely simulates real world usage of your application. End-to-end testing ensures that the flow of an application performs as expected by testing the complete application in a real-world environment, including interactions with databases, network, and other hardware or software components.

Question 104 of 200

---

The Databricks CLI is used to trigger a run of an existing job by passing the job\_id parameter. The response that the job run request has been submitted successfully includes a field run\_id.

Which statement describes what the number alongside this field represents?

- A. The job\_id and number of times the job has been run are concatenated and returned.
  - B. The total number of jobs that have been run in the workspace.
  - C. The number of times the job definition has been run in this workspace.
  - D. The job\_id is returned in this field.
- E. The globally unique ID of the newly triggered run.

Answer

Discussion

Correct Answer: E

The number alongside the run\_id field represents the globally unique ID of the newly triggered run. Each execution of a job in Databricks is assigned a unique run\_id, which helps in tracking and referencing that specific job run.

Question 105 of 200

---

The data science team has created and logged a production model using MLflow. The model accepts a list of column names and returns a new column of type DOUBLE.

The following code correctly imports the production model, loads the customers table containing the customer\_id key column into a DataFrame, and defines the feature columns needed for the model.

```
model = mlflow.pyfunc.spark_udf(spark, model_uri="models:/churn/prod")
df = spark.table("customers")
columns = ["account_age", "time_since_last_seen", "app_rating"]
```

Which code block will output a DataFrame with the schema "customer\_id LONG, predictions DOUBLE"?

- A. df.map(lambda x: model(x[columns])).select("customer\_id, predictions")
- B. df.select("customer\_id", model(\*columns).alias("predictions"))
- C. model.predict(df, columns)
- D. df.select("customer\_id", pandas\_udf(model, columns).alias("predictions"))
- E. df.apply(model, columns).select("customer\_id, predictions")

Answer

Discussion

Correct Answer: B

The correct code block uses Spark SQL to apply the UDF to the selected columns and alias the result as 'predictions'. Specifically, df.select("customer\_id", model(\*columns).alias("predictions")) selects the customer\_id column and the predictions generated by the model. The model function, which is a Spark UDF, is applied to the specified columns, and the output is named predictions. This approach correctly combines the customer\_id column with the newly generated predictions column, resulting in a DataFrame with the desired schema: customer\_id LONG, predictions DOUBLE.

## Question 106 of 200

A nightly batch job is configured to ingest all data files from a cloud object storage container where records are stored in a nested directory structure YYYY/MM/DD. The data for each date represents all records that were processed by the source system on that date, noting that some records may be delayed as they await moderator approval. Each entry represents a user review of a product and has the following schema:

user\_id STRING, review\_id BIGINT, product\_id BIGINT, review\_timestamp TIMESTAMP, review\_text STRING

The ingestion job is configured to append all data for the previous date to a target table reviews\_raw with an identical schema to the source system. The next step in the pipeline is a batch write to propagate all new records inserted into reviews\_raw to a table where data is fully deduplicated, validated, and enriched.

Which solution minimizes the compute costs to propagate this batch of data?

- A. Perform a batch read on the reviews\_raw table and perform an insert-only merge using the natural composite key user\_id, review\_id, product\_id, review\_timestamp.
- B. Configure a Structured Streaming read against the reviews\_raw table using the trigger once execution mode to process new records as a batch job.
- C. Use Delta Lake version history to get the difference between the latest version of reviews\_raw and one version prior, then write these records to the next table.
- D. Filter all records in the reviews\_raw table based on the review\_timestamp; batch append those records produced in the last 48 hours.
- E. Reprocess all records in reviews\_raw and overwrite the next table in the pipeline.

[Answer](#)[Discussion](#)

Using Delta Lake version history to get the difference between the latest version of reviews\_raw and the one version prior minimizes compute costs. This approach only processes the newly added data since the last batch, avoiding the need to process the entire dataset or maintain a complex streaming setup, thus offering a cost-effective solution for this specific batch processing job.

#### Question 107 of 200

---

Which statement describes Delta Lake optimized writes?

- A. Before a Jobs cluster terminates, OPTIMIZE is executed on all tables modified during the most recent job.
- B. 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.
- C. 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.
- D. Optimized writes use logical partitions instead of directory partitions; because partition boundaries are only represented in metadata, fewer small files are written.
- E. A shuffle occurs prior to writing to try to group similar data together resulting in fewer files instead of each executor writing multiple files based on directory partitions.

Answer

Discussion

Correct Answer: E

Optimized writes in Delta Lake involve a shuffle operation prior to writing the data. This shuffle groups similar data together, which results in fewer files being written instead of having each executor write multiple small files based on directory partitions. This process aims to improve the efficiency of file management and subsequent reads, although it may increase write latency due to the shuffle.

Question 108 of 200

---

Which statement describes the default execution mode for Databricks Auto Loader?

- A. Cloud vendor-specific queue storage and notification services are configured to track newly arriving files; the target table is materialized by directly querying all valid files in the source directory.
- B. New files are identified by listing the input directory; the target table is materialized by directly querying all valid files in the source directory.
- C. Webhooks trigger a Databricks job to run anytime new data arrives in a source directory; new data are automatically merged into target tables using rules inferred from the data.
- D. New files are identified by listing the input directory; new files are incrementally and idempotently loaded into the target Delta Lake table.
- E. Cloud vendor-specific queue storage and notification services are configured to track newly arriving files; new files are incrementally and idempotently loaded into the target Delta Lake table.

Answer

Discussion

Correct Answer: D

The default execution mode for Databricks Auto Loader identifies new files by listing the input directory. This mode allows new files to be incrementally and idempotently loaded into the target Delta Lake table, which ensures data consistency and efficiency in processing.

---

Question 109 of 200

A Delta Lake table representing metadata about content posts from users has the following schema:

user\_id LONG, post\_text STRING, post\_id STRING, longitude FLOAT, latitude FLOAT, post\_time TIMESTAMP, date DATE

Based on the above schema, which column is a good candidate for partitioning the Delta Table?

A. post\_time

B. latitude

C. post\_id

D. user\_id

E. date

Answer

Discussion

Correct Answer: E

A good candidate for partitioning a Delta Lake table is the column 'date'. Partitioning on the 'date' column is advantageous because it organizes the data based on the date, which is very useful for queries that filter by time periods. This approach can significantly improve query performance in analytics workloads that involve time-series data. Additionally, partitioning by 'date' helps manage the size of partitions effectively, as each partition will contain data for a specific date, resulting in more efficient reads and writes.

Question 110 of 200

A large company seeks to implement a near real-time solution involving hundreds of pipelines with parallel updates of many tables with extremely high volume and high velocity data.

Which of the following solutions would you implement to achieve this requirement?

- A. Use Databricks High Concurrency clusters, which leverage optimized cloud storage connections to maximize data throughput.
- B. Partition ingestion tables by a small time duration to allow for many data files to be written in parallel.
- C. Configure Databricks to save all data to attached SSD volumes instead of object storage, increasing file I/O significantly.
- D. Isolate Delta Lake tables in their own storage containers to avoid API limits imposed by cloud vendors.
- E. Store all tables in a single database to ensure that the Databricks Catalyst Metastore can load balance overall throughput.

Answer

Discussion

Correct Answer: A

In a scenario involving hundreds of pipelines with parallel updates to many tables handling extremely high volume and high velocity data, it is crucial to choose a solution that optimizes resource utilization and manages concurrent operations efficiently. Databricks High Concurrency clusters are specifically designed to handle multiple concurrent users and workloads. They provide fine-grained sharing of cluster resources and are optimized for operations such as running multiple parallel queries and updates, making them well-suited for this requirement. By leveraging optimized cloud storage connections, they can maximize data throughput, ensuring efficient processing of high-volume data.

Prev

Next

## Question 111 of 200

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

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

[Answer](#)[Discussion](#)

Correct Answer: C

To install a Python package scoped at the notebook level to all nodes in the currently active cluster, you should use the %pip install command in a notebook cell. This command is provided by Databricks specifically for this purpose, allowing you to install and manage Python packages directly within the notebook, ensuring the installation is applied across all nodes of the cluster associated with the notebook.

## Question 112 of 200

Each configuration below is identical to the extent that each cluster has 400 GB total of RAM 160 total cores and only one Executor per VM.

Given an extremely long-running job for which completion must be guaranteed, which cluster configuration will be able to guarantee completion of the job in light of one or more VM failures?

- A. • Total VMs: 8
  - 50 GB per Executor
  - 20 Cores / Executor
- B. • Total VMs: 16
  - 25 GB per Executor
  - 10 Cores / Executor
- C. • Total VMs: 1
  - 400 GB per Executor
  - 160 Cores/Executor
- D. • Total VMs: 4
  - 100 GB per Executor
  - 40 Cores / Executor
- E. • Total VMs: 2
  - 200 GB per Executor
  - 80 Cores / Executor

Answer

Discussion

Correct Answer: B

Ensuring the completion of an extremely long-running job in the event of one or more VM failures requires a cluster configuration that distributes the workload across the highest number of VMs. This way, the impact of losing any single VM is minimized and the remaining VMs can continue the job. Option B, with 16 VMs, each with 25 GB per Executor and 10 cores per Executor, offers the best setup for resilience and redundancy. More VMs mean that the job can still likely proceed even if several VMs fail, ensuring the completion of the job with minimal interruption.

Question 113 of 200

---

A Delta Lake table in the Lakehouse named `customer_churn_params` is used in churn prediction by the machine learning team. The table contains information about customers derived from a number of upstream sources. Currently, the data engineering team populates this table nightly by overwriting the table with the current valid values derived from upstream data sources.

Immediately after each update succeeds, the data engineering team would like to determine the difference between the new version and the previous version of the table.

Given the current implementation, which method can be used?

- A. Execute a query to calculate the difference between the new version and the previous version using Delta Lake's built-in versioning and time travel functionality.
- B. Parse the Delta Lake transaction log to identify all newly written data files.
- C. Parse the Spark event logs to identify those rows that were updated, inserted, or deleted.
- D. Execute `DESCRIBE HISTORY customer_churn_params` to obtain the full operation metrics for the update, including a log of all records that have been added or modified.
- E. Use Delta Lake's change data feed to identify those records that have been updated, inserted, or deleted.

Answer

Discussion

Correct Answer: A

Delta Lake's built-in versioning and time travel functionality allows you to query the state of the data at different points in time. By executing a query that compares the state of the table between two different versions, you can accurately determine the difference between the new version and the previous version. This method directly leverages Delta Lake's capabilities for maintaining historical data versions and simplifies the process of identifying changes made during the update.

---

Question 114 of 200

A data team's Structured Streaming job is configured to calculate running aggregates for item sales to update a downstream marketing dashboard. The marketing team has introduced a new promotion, and they would like to add a new field to track the number of times this promotion code is used for each item. A junior data engineer suggests updating the existing query as follows. Note that proposed changes are in bold.

Original query:

```
df.groupBy("item")
    .agg(count("item").alias("total_count"),
          mean("sale_price").alias("avg_price"))
    .writeStream
    .outputMode("complete")
    .option("checkpointLocation", "/item_agg/_checkpoint")
    .start("/item_agg")
```

Proposed query:

```
df.groupBy("item")
    .agg(count("item").alias("total_count"),
          mean("sale_price").alias("avg_price"),
          count("promo_code = 'NEW_MEMBER'").alias("new_member_promo"))
    .writeStream
    .outputMode("complete")
    .option('mergeSchema', 'true')
    .option("checkpointLocation", "/item_agg/_checkpoint")
    .start("/item_agg")
```

Which step must also be completed to put the proposed query into production?

- A. Specify a new checkpointLocation
- B. Remove .option('mergeSchema', 'true') from the streaming write
- C. Increase the shuffle partitions to account for additional aggregates
- D. Run REFRESH TABLE delta.'/item\_agg'

Answer

Discussion

Correct Answer: A

To ensure a smooth transition when updating the schema of a streaming job, it is crucial to specify a new checkpoint location. This precaution ensures that the streaming query starts afresh with the updated schema, thus preventing any potential conflicts or issues arising from mismatches between the old and new schemas.

This step is particularly important when introducing new fields, as existing state data might not be compatible with the new schema.

Question 115 of 200

---

When using CLI or REST API to get results from jobs with multiple tasks, which statement correctly describes the response structure?

- A. Each run of a job will have a unique job\_id; all tasks within this job will have a unique job\_id
- B. Each run of a job will have a unique job\_id; all tasks within this job will have a unique task\_id
- C. Each run of a job will have a unique orchestration\_id; all tasks within this job will have a unique run\_id
- D. Each run of a job will have a unique run\_id; all tasks within this job will have a unique task\_id
- E. Each run of a job will have a unique run\_id; all tasks within this job will also have a unique run\_id

Answer

Discussion

Correct Answer: D

When using CLI or REST API to get results from jobs with multiple tasks, each run of a job is typically assigned a unique run\_id to identify that specific execution of the job. Additionally, each individual task within that job run is assigned a unique task\_id to differentiate between the various tasks. This structure allows for clear identification and retrieval of data related to specific job runs and their associated tasks.

## Question 116 of 200

The data engineering team is configuring environments for development, testing, and production before beginning migration on a new data pipeline. The team requires extensive testing on both the code and data resulting from code execution, and the team wants to develop and test against data as similar to production data as possible.

A junior data engineer suggests that production data can be mounted to the development and testing environments, allowing pre-production code to execute against production data. Because all users have admin privileges in the development environment, the junior data engineer has offered to configure permissions and mount this data for the team.

Which statement captures best practices for this situation?

- A.** All development, testing, and production code and data should exist in a single, unified workspace; creating separate environments for testing and development complicates administrative overhead.
- B.** In environments where interactive code will be executed, production data should only be accessible with read permissions; creating isolated databases for each environment further reduces risks.
- C.** As long as code in the development environment declares USE dev\_db at the top of each notebook, there is no possibility of inadvertently committing changes back to production data sources.
- D.** Because Delta Lake versions all data and supports time travel, it is not possible for user error or malicious actors to permanently delete production data; as such, it is generally safe to mount production data anywhere.
- E.** Because access to production data will always be verified using passthrough credentials, it is safe to mount data to any Databricks development environment.

Answer

Discussion

Correct Answer: B

In environments where interactive code will be executed, production data should only be accessible with read permissions; creating isolated databases for each environment further reduces risks. This ensures that development and testing environments do not inadvertently alter production data, ensuring data integrity and security.

Question 117 of 200

---

A data engineer, User A, has promoted a pipeline to production by using the REST API to programmatically create several jobs. A DevOps engineer, User B, has configured an external orchestration tool to trigger job runs through the REST API. Both users authorized the REST API calls using their personal access tokens.

A workspace admin, User C, inherits responsibility for managing this pipeline. User C uses the Databricks Jobs UI to take "Owner" privileges of each job. Jobs continue to be triggered using the credentials and tooling configured by User B.

An application has been configured to collect and parse run information returned by the REST API. Which statement describes the value returned in the creator\_user\_name field?

- A.** Once User C takes "Owner" privileges, their email address will appear in this field; prior to this, User A's email address will appear in this field.
- B.** User B's email address will always appear in this field, as their credentials are always used to trigger the run.
- C.** User A's email address will always appear in this field, as they still own the underlying notebooks.
- D.** Once User C takes "Owner" privileges, their email address will appear in this field; prior to this, User B's email address will appear in this field.
- E.** User C will only ever appear in this field if they manually trigger the job, otherwise it will indicate User B.

Answer

Discussion

Correct Answer: B

The value in the creator\_user\_name field will reflect the user whose credentials are used to trigger the job run. Since User B's credentials are always used to trigger the runs through the external orchestration tool, User B's email address will always appear in this field.

Question 118 of 200

---

A member of the data engineering team has submitted a short notebook that they wish to schedule as part of a larger data pipeline. Assume that the commands provided below produce the logically correct results when run as presented.

Cmd 1

```
rawDF = spark.table("raw_data")
```

Cmd 2

```
rawDF.printSchema()
```

Cmd 3

```
flattenedDF = rawDF.select("*", "values.*")
```

Cmd 4

```
finalDF = flattenedDF.drop("values")
```

Cmd 5

```
display(finalDF)
```

Cmd 6

```
finalDF.write.mode("append").saveAsTable("flat_data")
```

Which command should be removed from the notebook before scheduling it as a job?

A. Cmd 2

B. Cmd 3

C. Cmd 4

D. Cmd 5

Answer

Discussion

Correct Answer: D

Question 119 of 200

---

Which statement regarding Spark configuration on the Databricks platform is true?

- A. The Databricks REST API can be used to modify the Spark configuration properties for an interactive cluster without interrupting jobs currently running on the cluster.
- B. Spark configurations set within a notebook will affect all SparkSessions attached to the same interactive cluster.
- C. When the same Spark configuration property is set for an interactive cluster and a notebook attached to that cluster, the notebook setting will always be ignored.
- D. Spark configuration properties set for an interactive cluster with the Clusters UI will impact all notebooks attached to that cluster.

Answer

Discussion

Correct Answer: D

Spark configuration properties set for an interactive cluster with the Clusters UI will impact all notebooks attached to that cluster. This is because the cluster-level configuration applies globally to any notebooks or jobs running on that cluster. Therefore, any setting specified in the Clusters UI will affect the environment in which all notebooks and jobs operate, ensuring consistency across workloads.

Question 120 of 200

---

The business reporting team requires that data for their dashboards be updated every hour. The total processing time for the pipeline that extracts, transforms, and loads the data for their pipeline runs in 10 minutes.

Assuming normal operating conditions, which configuration will meet their service-level agreement requirements with the lowest cost?

- A. Configure a job that executes every time new data lands in a given directory
- B. Schedule a job to execute the pipeline once an hour on a new job cluster
- C. Schedule a Structured Streaming job with a trigger interval of 60 minutes
- D. Schedule a job to execute the pipeline once an hour on a dedicated interactive cluster

Answer

Discussion

Correct Answer: B

Prev

Next

## Question 121 of 200

A Databricks SQL dashboard has been configured to monitor the total number of records present in a collection of Delta Lake tables using the following query pattern:

```
SELECT COUNT (*) FROM table -
```

Which of the following describes how results are generated each time the dashboard is updated?

- A. The total count of rows is calculated by scanning all data files
- B. The total count of rows will be returned from cached results unless REFRESH is run
- C. The total count of records is calculated from the Delta transaction logs
- D. The total count of records is calculated from the parquet file metadata

[Answer](#)[Discussion](#)

Correct Answer: C

The total count of records is calculated from the Delta transaction logs. Delta Lake optimizes COUNT(\*) queries by storing the row counts in the transaction logs, thereby avoiding the need for a full table scan and significantly improving query performance.

## Question 122 of 200

A Delta Lake table was created with the below query:

```
CREATE TABLE prod.sales_by_store
AS (
    SELECT *
    FROM prod.sales a
    INNER JOIN prod.store b
    ON a.store_id = b.store_id
)
```

Consider the following query:

```
DROP TABLE prod.sales_by_store -
```

If this statement is executed by a workspace admin, which result will occur?

- A. Data will be marked as deleted but still recoverable with Time Travel.
- B. The table will be removed from the catalog but the data will remain in storage.
- C. The table will be removed from the catalog and the data will be deleted.
- D. An error will occur because Delta Lake prevents the deletion of production data.

Answer

Discussion

Correct Answer: C

When the DROP TABLE statement is executed on a managed table in Delta Lake, both the table metadata and the physical data itself are deleted. The given query does not specify a location for the table, which implies it's a managed table. Therefore, dropping the table will result in the removal of both the table from the catalog and the data stored.

Question 123 of 200

---

A developer has successfully configured their credentials for Databricks Repos and cloned a remote Git repository. They do not have privileges to make changes to the main branch, which is the only branch currently visible in their workspace.

Which approach allows this user to share their code updates without the risk of overwriting the work of their teammates?

- A. Use Repos to create a new branch, commit all changes, and push changes to the remote Git repository.
- B. Use Repos to create a fork of the remote repository, commit all changes, and make a pull request on the source repository.
- C. Use Repos to pull changes from the remote Git repository; commit and push changes to a branch that appeared as changes were pulled.
- D. Use Repos to merge all differences and make a pull request back to the remote repository.

Answer

Discussion

Correct Answer: A

The user should create a new branch in Repos to make their changes. By creating a new branch, they can commit and push their updates to the remote Git repository without affecting the main branch. This approach ensures that their updates are isolated from the main branch, reducing the risk of overwriting teammates' work. Once the changes are ready, they can be merged into the main branch through a pull request, allowing for review and collaboration.

Question 124 of 200

---

The security team is exploring whether or not the Databricks secrets module can be leveraged for connecting to an external database.

After testing the code with all Python variables being defined with strings, they upload the password to the secrets module and configure the correct permissions for the currently active user. They then modify their code to the following (leaving all other variables unchanged).

```
password = dbutils.secrets.get(scope="db_creds", key="jdbc_password")

print(password)

df = (spark
    .read
    .format("jdbc")
    .option("url", connection)
    .option("dbtable", tablename)
    .option("user", username)
    .option("password", password)
)
```

Which statement describes what will happen when the above code is executed?

- A. The connection to the external table will succeed; the string "REDACTED" will be printed.
- B. An interactive input box will appear in the notebook; if the right password is provided, the connection will succeed and the encoded password will be saved to DBFS.
- C. An interactive input box will appear in the notebook; if the right password is provided, the connection will succeed and the password will be printed in plain text.
- D. The connection to the external table will succeed; the string value of password will be printed in plain text.

Answer

Discussion

Correct Answer: A

The code uses Databricks utilities to fetch and use a secret value from the secrets module. When fetching a password from the secrets scope and printing it, Databricks redacts the output to avoid exposing sensitive information. Therefore, when the script executes, sensitive information like passwords stored in the secret scope will not be shown in plain text; instead, 'REDACTED' will be printed to safeguard the secret. Thus, the correct outcome is that the connection to the external database will succeed, and 'REDACTED' will be printed.

The data science team has created and logged a production model using MLflow. The model accepts a list of column names and returns a new column of type DOUBLE.

The following code correctly imports the production model, loads the customers table containing the customer\_id key column into a DataFrame, and defines the feature columns needed for the model.

```
model = mlflow.pyfunc.spark_udf (spark,  
model_uri="models:/churn/prod")  
  
df = spark.table("customers")  
  
columns = ["account_age", "time_since_last_seen", "app_rating"]
```

Which code block will output a DataFrame with the schema "customer\_id LONG, predictions DOUBLE"?

- A. df.map(lambda x: model(x[columns])).select("customer\_id, predictions")
- B. df.select("customer\_id",  
model(\*columns).alias("predictions"))
- C. model.predict(df, columns)
- D. df.apply(model, columns).select("customer\_id, predictions")

Answer

Discussion

Correct Answer: B

The correct code block uses the `select` method to specify columns from the DataFrame and applies the model to the selected columns (`columns`). By aliasing the output of the model as `predictions`, it ensures that the resulting DataFrame will have the schema `customer\_id LONG, predictions DOUBLE`. This aligns with PySpark's DataFrame API for transforming and applying models to specific DataFrame columns, which is appropriate in the Databricks environment.

Prev

Next

## Question 131 of 200

An upstream system is emitting change data capture (CDC) logs that are being written to a cloud object storage directory. Each record in the log indicates the change type (insert, update, or delete) and the values for each field after the change. The source table has a primary key identified by the field pk\_id.

For auditing purposes, the data governance team wishes to maintain a full record of all values that have ever been valid in the source system. For analytical purposes, only the most recent value for each record needs to be recorded. The Databricks job to ingest these records occurs once per hour, but each individual record may have changed multiple times over the course of an hour.

Which solution meets these requirements?

- A. Iterate through an ordered set of changes to the table, applying each in turn to create the current state of the table, (insert, update, delete), timestamp of change, and the values.
- B. Use merge into to insert, update, or delete the most recent entry for each pk\_id into a table, then propagate all changes throughout the system.
- C. Deduplicate records in each batch by pk\_id and overwrite the target table.
- D. Use Delta Lake's change data feed to automatically process CDC data from an external system, propagating all changes to all dependent tables in the Lakehouse.

[Answer](#)[Discussion](#)

Correct Answer: D

The correct solution for maintaining both a comprehensive historical log and ensuring the most recent state for analytical purposes is to use Delta Lake's change data feed. This feature automates the processing of CDC data from external systems by capturing all changes (inserts, updates, deletes) and propagating them to all dependent tables in the Lakehouse. This approach ensures a detailed history is maintained while efficiently updating the current state for analytics.

An hourly batch job is configured to ingest data files from a cloud object storage container where each batch represent all records produced by the source system in a given hour. The batch job to process these records into the Lakehouse is sufficiently delayed to ensure no late-arriving data is missed. The user\_id field represents a unique key for the data, which has the following schema:

```
user_id BIGINT, username STRING, user_utc STRING, user_region STRING, last_login BIGINT, auto_pay  
BOOLEAN, last_updated BIGINT
```

New records are all ingested into a table named account\_history which maintains a full record of all data in the same schema as the source. The next table in the system is named account\_current and is implemented as a Type 1 table representing the most recent value for each unique user\_id.

Which implementation can be used to efficiently update the described account\_current table as part of each hourly batch job assuming there are millions of user accounts and tens of thousands of records processed hourly?

- A. Filter records in account\_history using the last\_updated field and the most recent hour processed, making sure to deduplicate on username; write a merge statement to update or insert the most recent value for each username.
- B. Use Auto Loader to subscribe to new files in the account\_history directory; configure a Structured Streaming trigger available job to batch update newly detected files into the account\_current table.
- C. Overwrite the account\_current table with each batch using the results of a query against the account\_history table grouping by user\_id and filtering for the max value of last\_updated.
- D. Filter records in account\_history using the last\_updated field and the most recent hour processed, as well as the max last\_login by user\_id write a merge statement to update or insert the most recent value for each user\_id.

Answer

Discussion

Correct Answer: D

To efficiently update the account\_current table as described, filtering records in account\_history using the last\_updated field for the most recent hour processed, and filtering for the max last\_login by user\_id, followed by a merge statement to update or insert the most recent value for each user\_id, is the optimal approach. This ensures that only the latest records based on user\_id, which is the unique identifier, are considered. This method maintains data integrity and ensures that the account\_current table always reflects the most recent data for each user.

Question 133 of 200

---

The business intelligence team has a dashboard configured to track various summary metrics for retail stores. This includes total sales for the previous day alongside totals and averages for a variety of time periods. The fields required to populate this dashboard have the following schema:

```
store_id INT, total_sales_qtd FLOAT, avg_daily_sales_qtd FLOAT, total_sales_ytd  
FLOAT, avg_daily_sales_ytd FLOAT, previous_day_sales FLOAT, total_sales_7d FLOAT,  
avg_daily_sales_7d FLOAT, updated TIMESTAMP
```

For demand forecasting, the Lakehouse contains a validated table of all itemized sales updated incrementally in near real-time. This table, named products\_per\_order, includes the following fields:

```
store_id INT, order_id INT, product_id INT, quantity INT, price FLOAT,  
order_timestamp TIMESTAMP
```

Because reporting on long-term sales trends is less volatile, analysts using the new dashboard only require data to be refreshed once daily. Because the dashboard will be queried interactively by many users throughout a normal business day, it should return results quickly and reduce total compute associated with each materialization.

Which solution meets the expectations of the end users while controlling and limiting possible costs?

- A. Populate the dashboard by configuring a nightly batch job to save the required values as a table overwritten with each update.
- B. Use Structured Streaming to configure a live dashboard against the products\_per\_order table within a Databricks notebook.
- C. Define a view against the products\_per\_order table and define the dashboard against this view.
- D. Use the Delta Cache to persist the products\_per\_order table in memory to quickly update the dashboard with each query.

Answer

Discussion

Correct Answer: A

Given that the dashboard only needs to be refreshed once daily, configuring a nightly batch job to extract and save the required values for the dashboard is the most efficient solution. This approach ensures that the data is up-to-date with minimal compute resources involved, as the data extraction and updating occur only once a day. Options involving live dashboards, views, or in-memory tables would increase compute costs and complexity without providing significant benefits for a dashboard that does not require real-time updates.

Question 134 of 200

---

A Delta lake table with CDF enabled table in the Lakehouse named `customer_churn_params` is used in churn prediction by the machine learning team. The table contains information about customers derived from a number of upstream sources. Currently, the data engineering team populates this table nightly by overwriting the table with the current valid values derived from upstream data sources.

The churn prediction model used by the ML team is fairly stable in production. The team is only interested in making predictions on records that have changed in the past 24 hours.

Which approach would simplify the identification of these changed records?

- A. Apply the churn model to all rows in the `customer_churn_params` table, but implement logic to perform an upsert into the predictions table that ignores rows where predictions have not changed.
- B. Convert the batch job to a Structured Streaming job using the complete output mode; configure a Structured Streaming job to read from the `customer_churn_params` table and incrementally predict against the churn model.
- C. Replace the current overwrite logic with a merge statement to modify only those records that have changed; write logic to make predictions on the changed records identified by the change data feed.
- D. Modify the overwrite logic to include a field populated by calling `spark.sql.functions.current_timestamp()` as data are being written; use this field to identify records written on a particular date.

Answer

Discussion

Correct Answer: C

Question 135 of 200

---

A view is registered with the following code:

```
CREATE VIEW recent_orders AS (
    SELECT a.user_id, a.email, b.order_id, b.order_date
    FROM
        (SELECT user_id, email
        FROM users) a
    INNER JOIN
        (SELECT user_id, order_id, order_date
        FROM orders
        WHERE order_date >= (current_date() - 7)) b
    ON a.user_id = b.user_id
)
```

Both users and orders are Delta Lake tables.

Which statement describes the results of querying recent\_orders?

- A. All logic will execute when the view is defined and store the result of joining tables to the DBFS; this stored data will be returned when the view is queried.
- B. Results will be computed and cached when the view is defined; these cached results will incrementally update as new records are inserted into source tables.
- C. All logic will execute at query time and return the result of joining the valid versions of the source tables at the time the query finishes.
- D. All logic will execute at query time and return the result of joining the valid versions of the source tables at the time the query began.

Answer

Discussion

Correct Answer: C

A SQL view is a virtual table that provides a way to represent the results of a query as a table. When you query an SQL view, the underlying query is executed against the current state of the data in the source tables at the time the query finishes. This ensures that the most recent data is always retrieved, reflecting any changes that might have occurred in the source tables. Therefore, the correct answer is that all logic will execute at query time and return the result of joining the valid versions of the source tables at the time the query finishes.

## Question 136 of 200

A data ingestion task requires a one-TB JSON dataset to be written out to Parquet with a target part-file size of 512 MB. Because Parquet is being used instead of Delta Lake, built-in file-sizing features such as Auto-Optimize & Auto-Compaction cannot be used.

Which strategy will yield the best performance without shuffling data?

- A. Set `spark.sql.files.maxPartitionBytes` to 512 MB, ingest the data, execute the narrow transformations, and then write to parquet.
- B. Set `spark.sql.shuffle.partitions` to 2,048 partitions ( $1\text{TB} \times 1024 \times 1024 / 512$ ), ingest the data, execute the narrow transformations, optimize the data by sorting it (which automatically repartitions the data), and then write to parquet.
- C. Set `spark.sql.adaptive.advisoryPartitionSizeInBytes` to 512 MB bytes, ingest the data, execute the narrow transformations, coalesce to 2,048 partitions ( $1\text{TB} \times 1024 \times 1024 / 512$ ), and then write to parquet.
- D. Ingest the data, execute the narrow transformations, repartition to 2,048 partitions ( $1\text{TB} \times 1024 \times 1024 / 512$ ), and then write to parquet.

[Answer](#)[Discussion](#)

#### Correct Answer: D

To achieve the best performance without shuffling data while ensuring the desired part-file size of 512 MB, the ideal strategy is to ingest the data, execute the narrow transformations, and then repartition the data to 2,048 partitions. This approach directly sets the number of partitions based on the target size (1TB divided by 512MB), ensuring that the data is correctly partitioned for writing out to Parquet. By setting the partitions to 2,048, each partition should be approximately 512 MB, leading to efficient processing without the need for additional shuffling.

#### Question 137 of 200

---

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

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

Answer

Discussion

#### Correct Answer: B

Each microbatch of a stream-static join uses the most recent version of the static Delta table as of the job's initialization. This means that the static Delta table does not update with each new microbatch; it remains constant for the duration of the streaming job. This approach allows for consistent joins during the entire streaming process without the need for updating the static dataset continuously.

Question 138 of 200

---

A junior data engineer has been asked to develop a streaming data pipeline with a grouped aggregation using DataFrame df. 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 DataFrame df has 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")
```

Which line of code 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. lag("event\_time", "10 minutes").alias("time")

Answer

Discussion

Correct Answer: B

Question 139 of 200

---

A Structured Streaming job deployed to production has been resulting in higher than expected cloud storage costs. At present, during normal execution, each microbatch of data is processed in less than 3s; at least 12 times per minute, a microbatch is processed that contains 0 records. The streaming write was configured using the default trigger settings. The production job is currently scheduled alongside many other Databricks jobs in a workspace with instance pools provisioned to reduce start-up time for jobs with batch execution.

Holding all other variables constant and assuming records need to be processed in less than 10 minutes, which adjustment will meet the requirement?

- A. Set the trigger interval to 3 seconds; the default trigger interval is consuming too many records per batch, resulting in spill to disk that can increase volume costs.
- B. Use the trigger once option and configure a Databricks job to execute the query every 10 minutes; this approach minimizes costs for both compute and storage.
- C. Set the trigger interval to 10 minutes; each batch calls APIs in the source storage account, so decreasing trigger frequency to maximum allowable threshold should minimize this cost.
- D. Set the trigger interval to 500 milliseconds; setting a small but non-zero trigger interval ensures that the source is not queried too frequently.

Answer

Discussion

Correct Answer: C

Setting the trigger interval to 10 minutes will ensure that the microbatches are processed at a frequency that minimizes the number of times the source storage account APIs are called, thereby reducing the associated costs. This adjustment will meet the requirement of processing records in less than 10 minutes and reduce the occurrences of processing microbatches with zero records.

Which statement describes Delta Lake optimized writes?

- A. Before a Jobs cluster terminates, OPTIMIZE is executed on all tables modified during the most recent job.
- B. 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.
- C. A shuffle occurs prior to writing to try to group similar data together resulting in fewer files instead of each executor writing multiple files based on directory partitions.
- D. Optimized writes use logical partitions instead of directory partitions; because partition boundaries are only represented in metadata, fewer small files are written.

Answer

Discussion

Correct Answer: C

A shuffle occurs prior to writing to try to group similar data together resulting in fewer files instead of each executor writing multiple files based on directory partitions. This description aligns with the concept of Delta Lake optimized writes, where data is reorganized before writing to reduce the number of output files and improve efficiency.

Prev

Next

## Question 141 of 200

---

Which statement characterizes the general programming model used by Spark Structured Streaming?

- A. Structured Streaming leverages the parallel processing of GPUs to achieve highly parallel data throughput.
- B. Structured Streaming is implemented as a messaging bus and is derived from Apache Kafka.
- C. Structured Streaming relies on a distributed network of nodes that hold incremental state values for cached stages.
- D. Structured Streaming models new data arriving in a data stream as new rows appended to an unbounded table.

[Answer](#)[Discussion](#)

Correct Answer: D

Structured Streaming models new data arriving in a data stream as new rows appended to an unbounded table. This mechanism allows Spark to seamlessly integrate batch and streaming operations, treating the streaming data as a continuously growing table, which it processes incrementally.

Question 142 of 200

---

Which configuration parameter directly affects the size of a spark-partition upon ingestion of data into Spark?

- A. `spark.sql.files.maxPartitionBytes`
- B. `spark.sql.autoBroadcastJoinThreshold`
- C. `spark.sql.adaptive.advisoryPartitionSizeInBytes`
- D. `spark.sql.adaptive.coalescePartitions.minPartitionNum`

Answer

Discussion

Correct Answer: A

The configuration parameter '`spark.sql.files.maxPartitionBytes`' directly affects the size of a spark-partition upon ingestion of data into Spark. This parameter specifies the maximum number of bytes to be packed into a single partition when reading files, thereby controlling the partition size directly.

---

Question 143 of 200

A Spark job is taking longer than expected. Using the Spark UI, a data engineer notes that the Min, Median, and Max Durations for tasks in a particular stage show the minimum and median time to complete a task as roughly the same, but the max duration for a task to be roughly 100 times as long as the minimum.

Which situation is causing increased duration of the overall job?

- A. Task queueing resulting from improper thread pool assignment.

- B.** Spill resulting from attached volume storage being too small.
  - C.** Network latency due to some cluster nodes being in different regions from the source data
- D. Skew caused by more data being assigned to a subset of spark-partitions.

Answer

Discussion

Correct Answer: D

When the minimum and median times for task completion are roughly the same, but there is a significant outlier with the maximum duration being much higher, it suggests that a small number of tasks are taking disproportionately longer to complete. This is indicative of skew, where more data is assigned to a subset of Spark partitions, resulting in those partitions taking significantly longer to process. This would cause the overall job duration to increase as the job cannot complete until all tasks are finished. Therefore, the increased duration is being caused by skew due to uneven data distribution.

Question 144 of 200

---

Each configuration below is identical to the extent that each cluster has 400 GB total of RAM, 160 total cores and only one Executor per VM.

Given an extremely long-running job for which completion must be guaranteed, which cluster configuration will be able to guarantee completion of the job in light of one or more VM failures?

- A.**
    - Total VMs: 8
      - 50 GB per Executor
      - 20 Cores / Executor
- B.** • Total VMs: 16

- 25 GB per Executor
- 10 Cores / Executor

C. • Total VMs: 1

- 400 GB per Executor
- 160 Cores/Executor

D. • Total VMs: 4

- 100 GB per Executor
- 40 Cores / Executor

Answer

Discussion

Correct Answer: B

Question 145 of 200

---

A task orchestrator has been configured to run two hourly tasks. First, an outside system writes Parquet data to a directory mounted at /mnt/raw\_orders/. After this data is written, a Databricks job containing the following code is executed:

```
(spark.readStream  
    .format("parquet")  
    .load("/mnt/raw_orders/")  
    .withWatermark("time", "2 hours")  
    .dropDuplicates(["customer_id", "order_id"])  
    .writeStream  
    .trigger(once=True)  
    .table("orders")  
)
```

Assume that the fields customer\_id and order\_id serve as a composite key to uniquely identify each order, and that the time field indicates when the record was queued in the source system.

If the upstream system is known to occasionally enqueue duplicate entries for a single order hours apart, which statement is correct?

- A. Duplicate records enqueued more than 2 hours apart may be retained and the orders table may contain duplicate records with the same customer\_id and order\_id.
- B. All records will be held in the state store for 2 hours before being deduplicated and committed to the orders table.
- C. The orders table will contain only the most recent 2 hours of records and no duplicates will be present.
- D. The orders table will not contain duplicates, but records arriving more than 2 hours late will be ignored and missing from the table.

Answer

Discussion

Correct Answer: A

### Question 146 of 200

---

A data engineer is configuring a pipeline that will potentially see late-arriving, duplicate records.

In addition to de-duplicating records within the batch, which of the following approaches allows the data engineer to deduplicate data against previously processed records as it is inserted into a Delta table?

- A. Rely on Delta Lake schema enforcement to prevent duplicate records.
- B. VACUUM the Delta table after each batch completes.
- C. Perform an insert-only merge with a matching condition on a unique key.
- D. Perform a full outer join on a unique key and overwrite existing data.

[Answer](#)[Discussion](#)

Correct Answer: C

### Question 147 of 200

---

A junior data engineer seeks to leverage Delta Lake's Change Data Feed functionality to create a Type 1 table representing all of the values that have ever been valid for all rows in a bronze table created with

the property delta.enableChangeDataFeed = true. They plan to execute the following code as a daily job:

```
from pyspark.sql.functions import col  
  
(spark.read.format("delta")  
    .option("readChangeFeed", "true")  
    .option("startingVersion", 0)  
    .table("bronze")  
    .filter(col("_change_type").isin(["update_postimage", "insert"]))  
    .write  
    .mode("append")  
    .table("bronze_history_type1")  
)
```

Which statement describes the execution and results of running the above query multiple times?

- A. Each time the job is executed, newly updated records will be merged into the target table, overwriting previous values with the same primary keys.
- B. Each time the job is executed, the entire available history of inserted or updated records will be appended to the target table, resulting in many duplicate entries.
- C. Each time the job is executed, only those records that have been inserted or updated since the last execution will be appended to the target table, giving the desired result.
- D. Each time the job is executed, the differences between the original and current versions are calculated; this may result in duplicate entries for some records.

Answer

Discussion

Correct Answer: B

A DLT pipeline includes the following streaming tables:

- raw\_iot ingests raw device measurement data from a heart rate tracking device.
- bpm\_stats incrementally computes user statistics based on BPM measurements from raw\_iot.

How can the data engineer configure this pipeline to be able to retain manually deleted or updated records in the raw\_iot table, while recomputing the downstream table bpm\_stats table when a pipeline update is run?

- A. Set the pipelines.reset.allowed property to false on raw\_iot
- B. Set the skipChangeCommits flag to true on raw\_iot
- C. Set the pipelines.reset.allowed property to false on bpm\_stats
- D. Set the skipChangeCommits flag to true on bpm\_stats

Answer

Discussion

Correct Answer: A

To retain manually deleted or updated records in the raw\_iot table and ensure the downstream table bpm\_stats is recomputed when a pipeline update is run, the data engineer needs to set the pipelines.reset.allowed property to false on raw\_iot. This prevents the raw\_iot table from being reset during pipeline updates, ensuring that the manual changes are preserved.

Question 149 of 200

A data pipeline uses Structured Streaming to ingest data from Apache Kafka to Delta Lake. Data is being stored in a bronze table, and includes the Kafka-generated timestamp, key, and value. Three months after the pipeline is deployed, the data engineering team has noticed some latency issues during certain times of the day.

A senior data engineer updates the Delta Table's schema and ingestion logic to include the current timestamp (as recorded by Apache Spark) as well as the Kafka topic and partition. The team plans to use these additional metadata fields to diagnose the transient processing delays.

Which limitation will the team face while diagnosing this problem?

- A. New fields will not be computed for historic records.
- B. Spark cannot capture the topic and partition fields from a Kafka source.
- C. Updating the table schema requires a default value provided for each field added.
- D. Updating the table schema will invalidate the Delta transaction log metadata.

Answer

Discussion

Correct Answer: A

Question 150 of 200

A nightly job ingests data into a Delta Lake table using the following code:

```
from pyspark.sql.functions import current_timestamp, input_file_name, col
from pyspark.sql.column import Column

def ingest_daily_batch(time_col: Column, year:int, month:int, day:int):
    (spark.read
        .format("parquet")
        .load(f"/mnt/daily_batch/{year}/{month}/{day}")
        .select("*",
            time_col.alias("ingest_time"),
            input_file_name().alias("source_file")
        )
        .write
        .mode("append")
        .saveAsTable("bronze")
    )
```

The next step in the pipeline requires a function that returns an object that can be used to manipulate new records that have not yet been processed to the next table in the pipeline.

Which code snippet completes this function definition?

```
def new_records():
```

- A. `return spark.readStream.table("bronze")`
- B. `return spark.read.option("readChangeFeed", "true").table ("bronze")`
- C. `return (spark.read  
 .table("bronze")  
 .filter(col("ingest_time") == current_timestamp())  
)`
- D. `return (spark.read  
 .table("bronze")  
 .filter(col("source_file") == f"/mnt/daily_batch/{year}/{month}/{day}")  
)`

Answer

Discussion

Correct Answer: B

To effectively identify and manipulate new records in a Delta Lake table, you should use the Change Data Feed (CDF) feature. This will allow the function to read only the changes (inserts and updates) made to the 'bronze' table since the last processing. This is accomplished by using the read option 'readChangeFeed' set to 'true'. This method is efficient for processing new records that have not yet been moved to the next table in the pipeline, meeting the goal of the function defined in the question.

Prev

Next

## Question 151 of 200

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

[Answer](#)[Discussion](#)

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. Manually setting schema types ensures that the data adheres to specific formatting and standards, which is crucial when the data is being used in production environments for monitoring dashboards and models.

### Question 152 of 200

---

The data engineering team maintains the following code:

```
accountDF = spark.table("accounts")
orderDF = spark.table("orders")
itemDF = spark.table("items")

orderWithItemDF = (orderDF.join(
    itemDF,
    orderDF.itemID == itemDF.itemID)
.select(
    orderDF.accountID,
    orderDF.itemID,
    itemDF.itemName))

finalDF = (accountDF.join(
    orderWithItemDF,
    accountDF.accountID == orderWithItemDF.accountID)
.select(
    orderWithItemDF["*"],
    accountDF.city))

(finalDF.write
 .mode("overwrite")
 .table("enriched_itemized_orders_by_account"))
```

Assuming that this code produces logically correct results and the data in the source tables has been de-duplicated and validated, which statement describes what will occur when this code is executed?

- A. A batch job will update the enriched\_itemized\_orders\_by\_account table, replacing only those rows that have different values than the current version of the table, using accountID as the primary key.
- B. The enriched\_itemized\_orders\_by\_account table will be overwritten using the current valid version of data in each of the three tables referenced in the join logic.

- C. No computation will occur until enriched\_itemized\_orders\_by\_account is queried; upon query materialization, results will be calculated using the current valid version of data in each of the three tables referenced in the join logic.
- D. An incremental job will detect if new rows have been written to any of the source tables; if new rows are detected, all results will be recalculated and used to overwrite the enriched\_itemized\_orders\_by\_account table.

Answer

Discussion

Correct Answer: B

The code uses the `.write.mode('overwrite')` method when writing the final DataFrame to the table `'enriched_itemized_orders_by_account'`. This method will overwrite the existing table with the new data, thus ensuring the table is completely updated with the latest data from the three source tables (`'accounts'`, `'orders'`, `'items'`). Therefore, the `enriched_itemized_orders_by_account` table will be overwritten using the current valid version of data in each of the three tables referenced in the join logic.

Question 153 of 200

---

The data engineering team is configuring environments for development, testing, and production before beginning migration on a new data pipeline. The team requires extensive testing on both the code and data resulting from code execution, and the team wants to develop and test against data as similar to production data as possible.

A junior data engineer suggests that production data can be mounted to the development and testing environments, allowing pre-production code to execute against production data. Because all users have admin privileges in the development environment, the junior data engineer has offered to configure permissions and mount this data for the team.

Which statement captures best practices for this situation?

- A. All development, testing, and production code and data should exist in a single, unified workspace; creating separate environments for testing and development complicates administrative overhead.

B. In environments where interactive code will be executed, production data should only be accessible with read permissions; creating isolated databases for each environment further reduces risks.

C. Because access to production data will always be verified using passthrough credentials, it is safe to mount data to any Databricks development environment.

D. Because Delta Lake versions all data and supports time travel, it is not possible for user error or malicious actors to permanently delete production data; as such, it is generally safe to mount production data anywhere.

Answer

Discussion

Correct Answer: B

Question 154 of 200

---

The data architect has mandated that all tables in the Lakehouse should be configured as external Delta Lake tables.

Which approach will ensure that this requirement is met?

A. Whenever a database is being created, make sure that the LOCATION keyword is used.

B. When the workspace is being configured, make sure that external cloud object storage has been mounted.

C. Whenever a table is being created, make sure that the LOCATION keyword is used.

D. When tables are created, make sure that the UNMANAGED keyword is used in the CREATE TABLE statement.

Answer

Discussion

Correct Answer: C

Question 155 of 200

---

The marketing team is looking to share data in an aggregate table with the sales organization, but the field names used by the teams do not match, and a number of marketing-specific fields have not been approved for the sales org.

Which of the following solutions addresses the situation while emphasizing simplicity?

- A. Create a view on the marketing table selecting only those fields approved for the sales team; alias the names of any fields that should be standardized to the sales naming conventions.
- B. Create a new table with the required schema and use Delta Lake's DEEP CLONE functionality to sync up changes committed to one table to the corresponding table.
- C. Use a CTAS statement to create a derivative table from the marketing table; configure a production job to propagate changes.
- D. Add a parallel table write to the current production pipeline, updating a new sales table that varies as required from the marketing table.

Answer

Discussion

Correct Answer: A

## Question 156 of 200

A Delta Lake table representing metadata about content posts from users has the following schema:

user\_id LONG, post\_text STRING, post\_id STRING, longitude FLOAT, latitude FLOAT, post\_time TIMESTAMP,  
date DATE

This table is partitioned by the date column. A query is run with the following filter:

longitude < 20 & longitude > -20

Which statement describes how data will be filtered?

- A.** Statistics in the Delta Log will be used to identify partitions that might include files in the filtered range.
- B.** No file skipping will occur because the optimizer does not know the relationship between the partition column and the longitude.
- C.** The Delta Engine will scan the parquet file footers to identify each row that meets the filter criteria.
- D.** Statistics in the Delta Log will be used to identify data files that might include records in the filtered range.

[Answer](#)[Discussion](#)

Correct Answer: D

When a Delta Lake table is partitioned by a column different from the filtered column (in this case, 'date' for partitioning and 'longitude' for filtering), Delta Lake uses statistics stored in the Delta Log to identify data files that might include records within the specified filter range. This allows for efficient data skipping

during query execution. Therefore, the correct statement is that statistics in the Delta Log will be used to identify data files that might include records in the filtered range.

Question 157 of 200

---

A small company based in the United States has recently contracted a consulting firm in India to implement several new data engineering pipelines to power artificial intelligence applications. All the company's data is stored in regional cloud storage in the United States.

The workspace administrator at the company is uncertain about where the Databricks workspace used by the contractors should be deployed.

Assuming that all data governance considerations are accounted for, which statement accurately informs this decision?

- A. Databricks runs HDFS on cloud volume storage; as such, cloud virtual machines must be deployed in the region where the data is stored.
- B. Databricks workspaces do not rely on any regional infrastructure; as such, the decision should be made based upon what is most convenient for the workspace administrator.
- C. Cross-region reads and writes can incur significant costs and latency; whenever possible, compute should be deployed in the same region the data is stored.
- D. Databricks notebooks send all executable code from the user's browser to virtual machines over the open internet; whenever possible, choosing a workspace region near the end users is the most secure.

Answer

Discussion

Correct Answer: C

Cross-region reads and writes can incur significant costs and latency. Whenever possible, compute should be deployed in the same region where the data is stored to minimize these issues. This will ensure better performance and cost-efficiency for the data engineering pipelines being implemented by the consulting firm.

Question 158 of 200

---

A CHECK constraint has been successfully added to the Delta table named activity\_details using the following logic:

```
ALTER TABLE activity_details
ADD CONSTRAINT valid_coordinates
CHECK (
    latitude >= -90 AND
    latitude <= 90 AND
    longitude >= -180 AND
    longitude <= 180);
```

A batch job is attempting to insert new records to the table, including a record where latitude = 45.50 and longitude = 212.67.

Which statement describes the outcome of this batch insert?

- A. The write will insert all records except those that violate the table constraints; the violating records will be reported in a warning log.
- B. The write will fail completely because of the constraint violation and no records will be inserted into the target table.
- C. The write will insert all records except those that violate the table constraints; the violating records will be recorded to a quarantine table.
- D. The write will include all records in the target table; any violations will be indicated in the boolean column named valid\_coordinates.

Answer

Discussion

Correct Answer: B

A CHECK constraint enforces specific rules on the data entering a table. In this particular case, the constraint ensures that the latitude must be between -90 and 90 and that the longitude must be between

-180 and 180. Since the record with latitude = 45.50 and longitude = 212.67 violates the longitude constraint, the entire batch insert will fail, and no records will be inserted into the target table. Constraints are designed to prevent any invalid data from entering the table, ensuring the integrity of the data.

#### Question 159 of 200

---

A junior data engineer is migrating a workload from a relational database system to the Databricks Lakehouse. The source system uses a star schema, leveraging foreign key constraints and multi-table inserts to validate records on write.

Which consideration will impact the decisions made by the engineer while migrating this workload?

- A.** Databricks only allows foreign key constraints on hashed identifiers, which avoid collisions in highly-parallel writes.
- B.** Foreign keys must reference a primary key field; multi-table inserts must leverage Delta Lake's upsert functionality.
- C.** Committing to multiple tables simultaneously requires taking out multiple table locks and can lead to a state of deadlock.
- D.** All Delta Lake transactions are ACID compliant against a single table, and Databricks does not enforce foreign key constraints.

Answer

Discussion

Correct Answer: D

When migrating a workload from a relational database system to the Databricks Lakehouse, it is important to note that Databricks does not enforce foreign key constraints. While Databricks leverages Delta Lake, which ensures ACID compliance for transactions within a single table, it does not support foreign key constraints. The lack of foreign key enforcement will impact how the engineer handles data integrity and relationships between tables during the migration.

Question 160 of 200

---

A data architect has heard about Delta Lake's built-in versioning and time travel capabilities. For auditing purposes, they have a requirement to maintain a full record of all valid street addresses as they appear in the customers table.

The architect is interested in implementing a Type 1 table, overwriting existing records with new values and relying on Delta Lake time travel to support long-term auditing. A data engineer on the project feels that a Type 2 table will provide better performance and scalability.

Which piece of information is critical to this decision?

- A.** Data corruption can occur if a query fails in a partially completed state because Type 2 tables require setting multiple fields in a single update.
- B.** Shallow clones can be combined with Type 1 tables to accelerate historic queries for long-term versioning.
- C.** Delta Lake time travel cannot be used to query previous versions of these tables because Type 1 changes modify data files in place.
- D.** Delta Lake time travel does not scale well in cost or latency to provide a long-term versioning solution.

Answer

Discussion

Correct Answer: D

Delta Lake time travel does not scale well in cost or latency to provide a long-term versioning solution. While Delta Lake's built-in versioning and time travel capabilities are powerful, they can become expensive and slow when used to maintain a long-term record of historical data. A Type 1 table, which constantly overwrites records, would require frequent time travel queries for historical data, leading to potential performance and scalability issues. Therefore, knowing the limitations of Delta Lake time travel in terms of cost and latency is critical to deciding between Type 1 and Type 2 tables for this use case.

## Question 161 of 200

A data engineer wants to join a stream of advertisement impressions (when an ad was shown) with another stream of user clicks on advertisements to correlate when impressions led to monetizable clicks.

In the code below, Impressions is a streaming DataFrame with a watermark ("event\_time", "10 minutes")

```
.groupBy(  
window("event_time", "5 minutes"),  
"id")  
.count()  
).withWatermark("event_time", 2 hours)  
impressions.join(clicks, expr("clickAdId = impressionAdId"), "inner")
```

The data engineer notices the query slowing down significantly.

Which solution would improve the performance?

- A. Joining on event time constraint: clickTime  $\geq$  impressionTime AND clickTime  $\leq$  impressionTime interval 1 hour
- B. Joining on event time constraint: clickTime + 3 hours  $<$  impressionTime - 2 hours
- C. Joining on event time constraint: clickTime == impressionTime using a leftOuter join
- D. Joining on event time constraint: clickTime  $\geq$  impressionTime - interval 3 hours and removing watermarks

Answer

Discussion

Correct Answer: A

Question 162 of 200

---

A junior data engineer has manually configured a series of jobs using the Databricks Jobs UI. Upon reviewing their work, the engineer realizes that they are listed as the "Owner" for each job. They attempt to transfer "Owner" privileges to the "DevOps" group, but cannot successfully accomplish this task.

Which statement explains what is preventing this privilege transfer?

- A. Databricks jobs must have exactly one owner; "Owner" privileges cannot be assigned to a group.
- B. The creator of a Databricks job will always have "Owner" privileges; this configuration cannot be changed.
- C. Only workspace administrators can grant "Owner" privileges to a group.
- D. A user can only transfer job ownership to a group if they are also a member of that group.

Answer

Discussion

Correct Answer: A

Databricks jobs must have exactly one owner; 'Owner' privileges cannot be assigned to a group. This design ensures clear accountability and management for the job.

Question 163 of 200

---

A table named user\_ltv is 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.

The user\_ltv table has the following schema:

email STRING, age INT, ltv INT

The following view definition is executed:

```

CREATE VIEW user_ltv_no_minors AS
SELECT email, age, ltv
FROM user_ltv
WHERE
CASE
    WHEN is_member("auditing") THEN TRUE
    ELSE age >= 18
END

```

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

```
SELECT * FROM user_ltv_no_minors
```

Which statement describes the results returned by this query?

- A. All columns will be displayed normally for those records that have an age greater than 17; records not meeting this condition will be omitted.
- B. All age values less than 18 will be returned as null values, all other columns will be returned with the values in user\_ltv.
- C. All values for the age column will be returned as null values, all other columns will be returned with the values in user\_ltv.
- D. All columns will be displayed normally for those records that have an age greater than 18; records not meeting this condition will be omitted.

Answer

Discussion

Correct Answer: A

The view definition includes a condition that filters records where the user's age must be greater than or equal to 18 for users who are not members of the 'auditing' group. Therefore, only records with an age of 18 or older will be included in the view for non-auditing users. This condition is correctly described in option A, which states that all columns will be displayed normally for records with an age greater than 17 (equivalent to age  $\geq 18$ ), and records not meeting this condition (age  $< 18$ ) will be omitted.

---

All records from an Apache Kafka producer are being ingested into a single Delta Lake table with the following schema:

key BINARY, value BINARY, topic STRING, partition LONG, offset LONG, timestamp LONG

There are 5 unique topics being ingested. Only the "registration" topic contains Personal Identifiable Information (PII). The company wishes to restrict access to PII. The company also wishes to only retain records containing PII in this table for 14 days after initial ingestion. However, for non-PII information, it would like to retain these records indefinitely.

Which solution meets the requirements?

- A. All data should be deleted biweekly; Delta Lake's time travel functionality should be leveraged to maintain a history of non-PII information.
- B. Data should be partitioned by the registration field, allowing ACLs and delete statements to be set for the PII directory.
- C. Data should be partitioned by the topic field, allowing ACLs and delete statements to leverage partition boundaries.
- D. Separate object storage containers should be specified based on the partition field, allowing isolation at the storage level.

Answer

Discussion

Correct Answer: C

Data should be partitioned by the topic field, allowing ACLs and delete statements to leverage partition boundaries. Partitioning by topic enables efficient management of access control and retention policies specifically for each topic. This way, the sensitive 'registration' topic containing PII can be managed separately to comply with the retention requirement of 14 days, while other topics can be retained indefinitely without performance degradation.

---

Question 165 of 200

The data governance team is reviewing code used for deleting records for compliance with GDPR. The following logic has been implemented to propagate delete requests from the user\_lookup table to the user\_aggregates table.

```

(spark.read
  .format("delta")
  .option("readChangeData", True)
  .option("startingTimestamp", '2021-08-22 00:00:00')
  .option("endingTimestamp", '2021-08-29 00:00:00')
  .table("user_lookup")
  .createOrReplaceTempView("changes"))

spark.sql("""
    DELETE FROM user_aggregates
    WHERE user_id IN (
        SELECT user_id
        FROM changes
        WHERE _change_type='delete'
    )
""")
"""
)

```

Assuming that user\_id is a unique identifying key and that all users that have requested deletion have been removed from the user\_lookup table, which statement describes whether successfully executing the above logic guarantees that the records to be deleted from the user\_aggregates table are no longer accessible and why?

- A. No; the Delta Lake DELETE command only provides ACID guarantees when combined with the MERGE INTO command.
- B. No; files containing deleted records may still be accessible with time travel until a VACUUM command is used to remove invalidated data files.
- C. No; the change data feed only tracks inserts and updates, not deleted records.
- D. Yes; Delta Lake ACID guarantees provide assurance that the DELETE command succeeded fully and permanently purged these records.

Answer

Discussion

Correct Answer: B

Successfully executing the above logic does not guarantee that the records to be deleted from the user\_aggregates table are no longer accessible. In Delta Lake, deleted records can still be accessible through the time travel feature until a VACUUM command is used to permanently remove the invalidated data files. Thus, to ensure that the records are not accessible, the VACUUM command must be executed after the DELETE command.

## Question 166 of 200

An external object storage container has been mounted to the location /mnt/finance\_eda\_bucket.

The following logic was executed to create a database for the finance team:

```
CREATE DATABASE finance_eda_db
LOCATION '/mnt/finance_eda_bucket';
GRANT USAGE ON DATABASE finance_eda_db TO finance;
GRANT CREATE ON DATABASE finance_eda_db TO finance;
```

After the database was successfully created and permissions configured, a member of the finance team runs the following code:

```
CREATE TABLE finance_eda_db.tx_sales AS
SELECT *
FROM sales
WHERE state = "TX";
```

If all users on the finance team are members of the finance group, which statement describes how the tx\_sales table will be created?

- A. A logical table will persist the query plan to the Hive Metastore in the Databricks control plane.
- B. An external table will be created in the storage container mounted to /mnt/finance\_eda\_bucket.
- C. A managed table will be created in the DBFS root storage container.
- D. An managed table will be created in the storage container mounted to /mnt/finance\_eda\_bucket.

Answer

Discussion

Correct Answer: D

The table will be created as a managed table in the storage container mounted to /mnt/finance\_eda\_bucket. In Databricks, the default behavior when creating a table without specifying an external location is to create a managed table, which means that Databricks will manage both the data and the metadata for this table. Since the database 'finance\_eda\_db' has a specified location '/mnt/finance\_eda\_bucket', any managed tables created in this database will have their data stored in this location.

Question 167 of 200

---

The data engineering team has been tasked with configuring connections to an external database that does not have a supported native connector with Databricks. The external database already has data security configured by group membership. These groups map directly to user groups already created in Databricks that represent various teams within the company.

A new login credential has been created for each group in the external database. The Databricks Utilities Secrets module will be used to make these credentials available to Databricks users.

Assuming that all the credentials are configured correctly on the external database and group membership is properly configured on Databricks, which statement describes how teams can be granted the minimum necessary access to using these credentials?

- A. No additional configuration is necessary as long as all users are configured as administrators in the workspace where secrets have been added.
- B. "Read" permissions should be set on a secret key mapped to those credentials that will be used by a given team.
- C. "Read" permissions should be set on a secret scope containing only those credentials that will be used by a given team.

- D. "Manage" permissions should be set on a secret scope containing only those credentials that will be used by a given team.

Answer

Discussion

Correct Answer: C

To grant teams the minimum necessary access to using the credentials, 'Read' permissions should be set on a secret scope containing only those credentials that will be used by a given team. By configuring the secret scope at the team level, access is securely managed and restricted to the appropriate group, ensuring that credentials are only accessible to the correct users. This approach maintains the principle of least privilege and aligns with security best practices.

Question 168 of 200

---

What is the retention of job run history?

- A. It is retained until you export or delete job run logs
- B. It is retained for 30 days, during which time you can deliver job run logs to DBFS or S3
- C. It is retained for 60 days, during which you can export notebook run results to HTML
- D. It is retained for 60 days, after which logs are archived

Answer

Discussion

Correct Answer: D

The retention of job run history is 60 days, after which logs are archived. This means the logs are kept for a period of 60 days and then subsequently moved to an archive location.

Question 169 of 200

---

A data engineer, User A, has promoted a new pipeline to production by using the REST API to programmatically create several jobs. A DevOps engineer, User B, has configured an external orchestration tool to trigger job runs through the REST API. Both users authorized the REST API calls using their personal access tokens.

Which statement describes the contents of the workspace audit logs concerning these events?

- A.** Because the REST API was used for job creation and triggering runs, a Service Principal will be automatically used to identify these events.
- B.** Because User A created the jobs, their identity will be associated with both the job creation events and the job run events.
- C.** Because these events are managed separately, User A will have their identity associated with the job creation events and User B will have their identity associated with the job run events.
- D.** Because the REST API was used for job creation and triggering runs, user identity will not be captured in the audit logs.

Answer

Discussion

Correct Answer: C

When User A and User B use their personal access tokens to authenticate their REST API calls, their individual actions are logged under their respective identities. User A's creation of the jobs will be logged with their identity, and User B's orchestration of the job runs will be logged with their identity. This

separation ensures that audit logs accurately attribute actions to the correct user, maintaining accountability and traceability for activities performed within the workspace.

Question 170 of 200

---

A distributed team of data analysts share computing resources on an interactive cluster with autoscaling configured. In order to better manage costs and query throughput, the workspace administrator is hoping to evaluate whether cluster upscaling is caused by many concurrent users or resource-intensive queries.

In which location can one review the timeline for cluster resizing events?

A. Workspace audit logs

B. Driver's log file

C. Ganglia

D. Cluster Event Log

Answer

Discussion

Correct Answer: D

To evaluate cluster resizing events and understand their timeline, the Cluster Event Log is the appropriate location. This log specifically records events related to the cluster's operational changes, including upscaling and downscaling activities. This is essential for tracing and analyzing both user actions and query impacts on resource allocation.

## Question 171 of 200

When evaluating the Ganglia Metrics for a given cluster with 3 executor nodes, which indicator would signal proper utilization of the VM's resources?

- A. The five Minute Load Average remains consistent/flat
- B. CPU Utilization is around 75%
- C. Network I/O never spikes
- D. Total Disk Space remains constant

[Answer](#)[Discussion](#)

Correct Answer: B

When evaluating the Ganglia Metrics for a cluster, CPU utilization being around 75% is a strong indicator that the VM's resources are being effectively utilized. This level of utilization suggests that the CPU is actively handling processes without being either idle or overburdened, thus indicating proper resource utilization and balancing of workloads across the cluster.

## Question 172 of 200

The data engineer is using Spark's MEMORY\_ONLY storage level.

Which indicators should the data engineer look for in the Spark UI's Storage tab to signal that a cached table is not performing optimally?

- A. On Heap Memory Usage is within 75% of Off Heap Memory Usage

- B. The RDD Block Name includes the "\*" annotation signaling a failure to cache

- C. Size on Disk is > 0

- D. The number of Cached Partitions > the number of Spark Partitions

Answer

Discussion

Correct Answer: C

If the Spark MEMORY\_ONLY storage level is being used, any data that is spilled to disk indicates that there is insufficient memory to store all the data in memory, which directly contradicts the purpose of using MEMORY\_ONLY. Therefore, if the Size on Disk is greater than 0, it signals that the cached table is not performing optimally.

Question 173 of 200

Review the following error traceback:

```
-----  
AnalysisException                                     Traceback (most recent call last)  
<command-3293767849433948> in <module>  
----> 1 display(df.select(3*"heartrate"))  
  
/databricks/spark/python/pyspark/sql/dataframe.py in select(self, *cols)  
    1690         [Row(name='Alice', age=12), Row(name='Bob', age=15)]  
    1691         """  
-> 1692         jdf = self._jdf.select(self._jcols(*cols))  
    1693         return DataFrame(jdf, self.sql_ctx)  
    1694  
  
/databricks/spark/python/lib/py4j-0.10.9-src.zip/py4j/java_gateway.py in __call__(self, *args)  
    1302  
    1303         answer = self.gateway_client.send_command(command)  
-> 1304         return_value = get_return_value(  
    1305             answer, self.gateway_client, self.target_id, self.name)  
    1306  
  
/databricks/spark/python/pyspark/sql/utils.py in deco(*a, **kw)  
    121             # Hide where the exception came from that shows a non-Pythonic  
    122             # JVM exception message.  
--> 123             raise converted from None  
    124         else:  
    125             raise  
  
AnalysisException: cannot resolve '`heartrateheartrateheartrate`' given input columns:  
[spark_catalog.database.device_id, spark_catalog.database.table.heartrate,  
spark_catalog.database.mrn, spark_catalog.database.table.time];  
'Project ['heartrateheartrateheartrate']  
+- SubqueryAlias spark_catalog.database.table  
  +- Relation[device_id#75L,heartrate#76,mrn#77L,time#78] parquet
```

Which statement describes the error being raised?

- A. There is a syntax error because the heartrate column is not correctly identified as a column.
- B. There is no column in the table named heartrateheartrateheartrate
- C. There is a type error because a column object cannot be multiplied.
- D. There is a type error because a DataFrame object cannot be multiplied.

Answer

Discussion

Correct Answer: B

Question 174 of 200

---

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

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

Answer

Discussion

Correct Answer: C

To install a Python package scoped at the notebook level in a Databricks environment, the appropriate method is to use %pip install in a notebook cell. This command ensures the package is installed only in the context of

the current notebook session and is available to all nodes in the active cluster.

### Question 175 of 200

---

What is the first line of a Databricks Python notebook when viewed in a text editor?

- A. %python
- B. // Databricks notebook source
- C. # Databricks notebook source
- D. -- Databricks notebook source

Answer

Discussion

Correct Answer: C

The first line of a Databricks Python notebook when viewed in a text editor is '# Databricks notebook source'. This line is a comment indicating that the contents are from a Databricks notebook.

Prev

Next



Examice delivers accurate questions, trusted answers, and the best value in certification exam preparation.



Question 176 of 200

---

Incorporating unit tests into a PySpark application requires upfront attention to the design of your jobs, or a potentially significant refactoring of existing code.

Which benefit offsets this additional effort?

- A. Improves the quality of your data
- B. Validates a complete use case of your application
- C. Troubleshooting is easier since all steps are isolated and tested individually
- D. Ensures that all steps interact correctly to achieve the desired end result

[Answer](#)[Discussion](#)

Correct Answer: C

Troubleshooting is easier since all steps are isolated and tested individually. Unit tests break down the application into smaller, testable parts which makes it easier to identify and fix errors at each step, rather than debugging a larger, more complex system.

Question 177 of 200

---

What describes integration testing?

- A. It validates an application use case.
- B. It validates behavior of individual elements of an application,
- C. It requires an automated testing framework.
- D. It validates interactions between subsystems of your application.

Answer

Discussion

Correct Answer: D

Integration testing validates interactions between subsystems of your application. It focuses on ensuring that different components or modules of an application work together as expected when integrated.

Question 178 of 200

---

The Databricks CLI is used to trigger a run of an existing job by passing the job\_id parameter. The response that the job run request has been submitted successfully includes a field run\_id.

Which statement describes what the number alongside this field represents?

- A. The job\_id and number of times the job has been run are concatenated and returned.
- B. The globally unique ID of the newly triggered run.
- C. The number of times the job definition has been run in this workspace.
- D. The job\_id is returned in this field.

Answer

Discussion

Correct Answer: B

Question 179 of 200

---

A Databricks job has been configured with three 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.

What will be the resulting state if tasks A and B complete successfully but task C fails during a scheduled run?

- 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. Unless all tasks complete successfully, no changes will be committed to the Lakehouse; because task C failed, all commits will be rolled back automatically.
- C. Because all tasks are managed as a dependency graph, no changes will be committed to the Lakehouse until all tasks have successfully been completed.
- D. 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.

Answer

Discussion

Correct Answer: D

When a Databricks job is set up with multiple tasks where some tasks are dependent on others, the execution and success of tasks are crucial for the final outcome. Since task A does not depend on any other tasks, it will run independently and its completion will stand regardless of subsequent tasks' outcomes. Tasks B and C, however, depend on the successful completion of task A. Since task B is completed successfully, its logic will be executed and changes will be committed. Task C fails during execution; therefore, any partial or completed operations specific to it will be rolled back. However, since task C failing doesn't affect the commitment of changes from tasks A and B, all logic from A and B will be successfully executed and any changes from task C will be rolled back.

When scheduling Structured Streaming jobs for production, which configuration automatically recovers from query failures and keeps costs low?

A. Cluster: New Job Cluster;

Retries: Unlimited;

Maximum Concurrent Runs: 1

B. Cluster: New Job Cluster;

Retries: Unlimited;

Maximum Concurrent Runs: Unlimited

C. Cluster: Existing All-Purpose Cluster;

Retries: Unlimited;

Maximum Concurrent Runs: 1

D. Cluster: New Job Cluster;

Retries: None;

Maximum Concurrent Runs: 1

Answer

Discussion

Correct Answer: A

Prev

Next

## Question 181 of 200

A Delta Lake table was created with the below query:

```
CREATE TABLE prod.sales_by_stor
USING DELTA
LOCATION "/mnt/prod/sales_by_store"
```

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.
- B. All related files and metadata are dropped and recreated in a single ACID transaction.
- C. The table name change is recorded in the Delta transaction log.
- D. A new Delta transaction log is created for the renamed table.

Answer

Discussion

Correct Answer: A

## Question 182 of 200

The data engineering team has configured a Databricks SQL query and alert to monitor the values in a Delta Lake table. The recent\_sensor\_recordings table contains an identifying sensor\_id alongside the timestamp and temperature for the most recent 5 minutes of recordings.

The below query is used to create the alert:

```
SELECT MEAN(temperature), MAX(temperature), MIN(temperature)
FROM recent_sensor_recordings
GROUP BY sensor_id
```

The query is set to refresh each minute and always completes in less than 10 seconds. The alert is set to trigger when mean (temperature) > 120. Notifications are triggered to be sent at most every 1 minute.

If this alert raises notifications for 3 consecutive minutes and then stops, which statement must be true?

- A. The total average temperature across all sensors exceeded 120 on three consecutive executions of the query
- B. The average temperature recordings for at least one sensor exceeded 120 on three consecutive executions of the query
- C. The source query failed to update properly for three consecutive minutes and then restarted
- D. The maximum temperature recording for at least one sensor exceeded 120 on three consecutive executions of the query

Answer

Discussion

Correct Answer: B

Question 183 of 200

---

The Databricks workspace administrator has configured interactive clusters for each of the data engineering groups. To control costs, clusters are set to terminate after 30 minutes of inactivity. Each user should be able to execute workloads against their assigned clusters at any time of the day.

Assuming users have been added to a workspace but not granted any permissions, which of the following describes the minimal permissions a user would need to start and attach to an already configured cluster.

- A. "Can Manage" privileges on the required cluster

- B.** Cluster creation allowed, "Can Restart" privileges on the required cluster
- C.** Cluster creation allowed, "Can Attach To" privileges on the required cluster
- D.** "Can Restart" privileges on the required cluster

Answer

Discussion

Correct Answer: D

Question 184 of 200

---

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("append")
    .option("checkpointPath", "/_checkpoints/churn_preds")
    .table("churn_preds")
)
```
  
- D. 

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

Answer

Discussion

Correct Answer: A

Question 185 of 200

---

A Delta table of weather records is partitioned by date and has the below schema:

date DATE, device\_id INT, temp FLOAT, latitude FLOAT, longitude FLOAT

To find all the records from within the Arctic Circle, you execute a query with the below filter:

latitude > 66.3

Which statement describes how the Delta engine identifies which files to load?

- A. All records are cached to an operational database and then the filter is applied
  
- B. The Parquet file footers are scanned for min and max statistics for the latitude column
  
- C. The Hive metastore is scanned for min and max statistics for the latitude column
  
- D. The Delta log is scanned for min and max statistics for the latitude column

Answer

Discussion

Correct Answer: D

## Question 186 of 200

A junior data engineer has configured a workload that posts the following JSON to the Databricks REST API endpoint 2.0/jobs/create.

```
{  
  "name": "Ingest new data",  
  "existing_cluster_id": "6015-954420-peace720",  
  "notebook_task": {  
    "notebook_path": "/Prod/ingest.py"  
  }  
}
```

Assuming that all configurations and referenced resources are available, which statement describes the result of executing this workload three times?

- A. The logic defined in the referenced notebook will be executed three times on the referenced existing all purpose cluster.
- B. The logic defined in the referenced notebook will be executed three times on new clusters with the configurations of the provided cluster ID.
- C. Three new jobs named "Ingest new data" will be defined in the workspace, but no jobs will be executed.
- D. One new job named "Ingest new data" will be defined in the workspace, but it will not be executed.

[Answer](#)[Discussion](#)

Correct Answer: C

## Question 187 of 200

---

A view is registered with the following code:

```
CREATE VIEW recent_orders AS (
    SELECT a.user_id, a.email, b.order_id, b.order_date
    FROM
        (SELECT user_id, email
        FROM users) a
    INNER JOIN
        (SELECT user_id, order_id, order_date
        FROM orders
        WHERE order_date >= (current_date() - 7)) b
    ON a.user_id = b.user_id
)
```

Both users and orders are Delta Lake tables.

Which statement describes the results of querying recent\_orders?

- A. The versions of each source table will be stored in the table transaction log; query results will be saved to DBFS with each query.
- B. All logic will execute when the table is defined and store the result of joining tables to the DBFS; this stored data will be returned when the table is queried.
- C. All logic will execute at query time and return the result of joining the valid versions of the source tables at the time the query finishes.
- D. All logic will execute at query time and return the result of joining the valid versions of the source tables at the time the query began.

Answer

Discussion

Correct Answer: D

---

Question 188 of 200

A data engineer is performing a join operation to combine values from a static userLookup table with a streaming DataFrame streamingDF.

Which code block attempts to perform an invalid stream-static join?

- A. `userLookup.join(streamingDF, ["user_id"], how="right")`
- B. `streamingDF.join(userLookup, ["user_id"], how="inner")`
- C. `userLookup.join(streamingDF, ["user_id"], how="inner")`
- D. `userLookup.join(streamingDF, ["user_id"], how="left")`

Answer

Discussion

Correct Answer: D

#### Question 189 of 200

---

A data architect has designed a system in which two Structured Streaming jobs will concurrently write to a single bronze Delta table. Each job is subscribing to a different topic from an Apache Kafka source, but they will write data with the same schema. To keep the directory structure simple, a data engineer has decided to nest a checkpoint directory to be shared by both streams.

The proposed directory structure is displayed below:

```
./bronze
└── _checkpoint
    ├── _delta_log
    ├── year_week=2020_01
    ├── year_week=2020_02
    └── ...
```

Which statement describes whether this checkpoint directory structure is valid for the given scenario and why?

- A. No; Delta Lake manages streaming checkpoints in the transaction log.
- B. Yes; both of the streams can share a single checkpoint directory.
- C. No; only one stream can write to a Delta Lake table.

D. No; each of the streams needs to have its own checkpoint directory.

Answer

Discussion

Correct Answer: D

Question 190 of 200

---

Which statement describes the default execution mode for Databricks Auto Loader?

- A. Cloud vendor-specific queue storage and notification services are configured to track newly arriving files; new files are incrementally and idempotently loaded into the target Delta Lake table.
- B. New files are identified by listing the input directory; the target table is materialized by directly querying all valid files in the source directory.
- C. Webhooks trigger a Databricks job to run anytime new data arrives in a source directory; new data are automatically merged into target tables using rules inferred from the data.
- D. New files are identified by listing the input directory; new files are incrementally and idempotently loaded into the target Delta Lake table.

Answer

Discussion

Correct Answer: D

Prev

Next

## Question 191 of 200

An upstream source writes Parquet data as hourly batches to directories named with the current date. A nightly batch job runs the following code to ingest all data from the previous day as indicated by the date variable:

```
(spark.read
    .format("parquet")
    .load(f"/mnt/raw_orders/{date}")
    .dropDuplicates(["customer_id", "order_id"])
    .write
    .mode("append")
    .saveAsTable("orders")
)
```

Assume that the fields customer\_id and order\_id serve as a composite key to uniquely identify each order.

If the upstream system is known to occasionally produce duplicate entries for a single order hours apart, which statement is correct?

- A. Each write to the orders table will only contain unique records, and only those records without duplicates in the target table will be written.
- B. Each write to the orders table will only contain unique records, but newly written records may have duplicates already present in the target table.
- C. Each write to the orders table will only contain unique records; if existing records with the same key are present in the target table, these records will be overwritten.
- D. Each write to the orders table will run deduplication over the union of new and existing records, ensuring no duplicate records are present.

[Answer](#)[Discussion](#)

Correct Answer: B

Question 192 of 200

---

A junior data engineer on your team has implemented the following code block.

```
MERGE INTO events
USING new_events
ON events.event_id = new_events.event_id
WHEN NOT MATCHED
    INSERT *
```

The view new\_events contains a batch of records with the same schema as the events Delta table. The event\_id field serves as a unique key for this table.

When this query is executed, what will happen with new records that have the same event\_id as an existing record?

- A. They are merged.
- B. They are ignored.
- C. They are updated.
- D. They are inserted.

Answer

Discussion

Correct Answer: B

Question 193 of 200

---

A team of data engineers are adding tables to a DLT pipeline that contain repetitive expectations for many of the same data quality checks. One member of the team suggests reusing these data quality rules across all tables defined for this pipeline.

What approach would allow them to do this?

- A. Add data quality constraints to tables in this pipeline using an external job with access to pipeline configuration files.
- B. Use global Python variables to make expectations visible across DLT notebooks included in the same pipeline.
- C. Maintain data quality rules in a separate Databricks notebook that each DLT notebook or file can import as a library.
- D. Maintain data quality rules in a Delta table outside of this pipeline's target schema, providing the schema name as a pipeline parameter.

Answer

Discussion

Correct Answer: D

Question 194 of 200

---

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.

```
CREATE LIVE TABLE report (
    CONSTRAINT no_missing_records EXPECT (key IN validation_copy)
)
AS SELECT <...>
```

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

- A. Define a temporary table that performs a left outer join on validation\_copy and report, and define an expectation that no report key values are null
- B. 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
- C. 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

- D. Define a function that performs a left outer join on validation\_copy and report, and check against the result in a DLT expectation for the report table

Answer

Discussion

Correct Answer: A

Question 195 of 200

---

A user new to Databricks is trying to troubleshoot long execution times for some pipeline logic they are working on. Presently, the user is executing code cell-by-cell, using `display()` calls to confirm code is producing the logically correct results as new transformations are added to an operation. To get a measure of average time to execute, the user is running each cell multiple times interactively.

Which of the following adjustments will get a more accurate measure of how code is likely to perform in production?

- A. The Jobs UI should be leveraged to occasionally run the notebook as a job and track execution time during incremental code development because Photon can only be enabled on clusters launched for scheduled jobs.
- B. The only way to meaningfully troubleshoot code execution times in development notebooks is to use production-sized data and production-sized clusters with Run All execution.
- C. Production code development should only be done using an IDE; executing code against a local build of open source Spark and Delta Lake will provide the most accurate benchmarks for how code will perform in production.
- D. Calling `display()` forces a job to trigger, while many transformations will only add to the logical query plan; because of caching, repeated execution of the same logic does not provide meaningful results.

Answer

Discussion

Correct Answer: B

## Question 196 of 200

Where in the Spark UI can one diagnose a performance problem induced by not leveraging predicate push-down?

- A. In the Executor's log file, by grepping for "predicate push-down"
- B. In the Stage's Detail screen, in the Completed Stages table, by noting the size of data read from the Input column
- C. In the Query Detail screen, by interpreting the Physical Plan
- D. In the Delta Lake transaction log, by noting the column statistics

[Answer](#)[Discussion](#)

Correct Answer: C

## Question 197 of 200

A data engineer needs to capture pipeline settings from an existing setting in the workspace, and use them to create and version a JSON file to create a new pipeline.

Which command should the data engineer enter in a web terminal configured with the Databricks CLI?

- A. Use list pipelines to get the specs for all pipelines; get the pipeline spec from the returned results; parse and use this to create a pipeline

- B.** Stop the existing pipeline; use the returned settings in a reset command
- C.** Use the get command to capture the settings for the existing pipeline; remove the pipeline\_id and rename the pipeline; use this in a create command
- D.** Use the clone command to create a copy of an existing pipeline; use the get JSON command to get the pipeline definition; save this to git

Answer

Discussion

Correct Answer: C

Question 198 of 200

---

Which REST API call can be used to review the notebooks configured to run as tasks in a multi-task job?

- A.** /jobs/runs/list
- B.** /jobs/list
- C.** /jobs/runs/get
- D.** /jobs/get

Answer

Discussion

Correct Answer: D

Question 199 of 200

---

A Data Engineer wants to run unit tests using common Python testing frameworks on Python functions defined across several Databricks notebooks currently used in production.

How can the data engineer run unit tests against functions that work with data in production?

- A. Define and import unit test functions from a separate Databricks notebook
- B. Define and unit test functions using Files in Repos
- C. Run unit tests against non-production data that closely mirrors production
- D. Define unit tests and functions within the same notebook

Answer

Discussion

Correct Answer: B

---

Question 200 of 200

---

A data engineer wants to refactor the following DLT code, which includes multiple table definitions with very similar code.

```
@dlt.table(name=f"t1_dataset")
def t1_dataset():
    return spark.read.table(t1)

@dlt.table(name=f"t2_dataset")
def t2_dataset():
    return spark.read.table(t2)

@dlt.table(name=f"t3_dataset")
def t3_dataset():
    return spark.read.table(t3)

...
```

In an attempt to programmatically create these tables using a parameterized table definition, the data engineer writes the following code.

```
tables = ["t1", "t2", "t3"]

for t in tables:
    @dlt.table(name=f"{t}_dataset")
        def new_table():
            return spark.read.table(t)
```

The pipeline runs an update with this refactored code, but generates a different DAG showing incorrect configuration values for these tables.

How can the data engineer fix this?

- A. Wrap the for loop inside another table definition, using generalized names and properties to replace with those from the inner table definition.
- B. Convert the list of configuration values to a dictionary of table settings, using table names as keys.
- C. Move the table definition into a separate function, and make calls to this function using different input parameters inside the for loop.
- D. Load the configuration values for these tables from a separate file, located at a path provided by a pipeline parameter.

Answer

Discussion

Correct Answer: C

Prev

Next >



Examice delivers accurate questions, trusted answers, and the best value in certification exam preparation.