

Databricks Certified Data Engineer Associate - Practice Exam 1 - Results

[◀ Return to review](#)



Attempt 2

All knowledge areas ▾

All questions ▾

Question 1: **Skipped**

Which of the following commands can a data engineer use to compact small data files of a Delta table into larger ones?

PARTITION BY

ZORDER BY

COMPACT

VACUUM

OPTIMIZE

(Correct)

Explanation

Delta Lake can improve the speed of read queries from a table. One way to improve this speed is by compacting small files into larger ones. You trigger compaction by running the `OPTIMIZE` command

Reference: <https://docs.databricks.com/sql/language-manual/delta-optimize.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 2: **Skipped**

A data engineer is trying to use Delta time travel to rollback a table to a previous version, but the data engineer received an error that the data files are no longer present.

Which of the following commands was run on the table that caused deleting the data files?

VACUUM

(Correct)

OPTIMIZE

ZORDER BY

DEEP CLONE

DELETE

Explanation

Running the VACUUM command on a Delta table deletes the unused data files older than a specified data retention period. As a result, you lose the ability to time travel back to any version older than that retention threshold.

Reference: <https://docs.databricks.com/sql/language-manual/delta-vacuum.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)

JSON Hive-specific format Both, Parquet and JSON**Explanation**

Delta Lake builds upon standard data formats. Delta lake table gets stored on the storage in one or more data files in Parquet format, along with transaction logs in JSON format.

Reference: <https://docs.databricks.com/delta/index.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 4: **Skipped**

Which of the following locations hosts the Databricks web application ?

 Data plane Control plane

(Correct)

 Databricks Filesystem Databricks-managed cluster Customer Cloud Account**Explanation**

According to the Databricks Lakehouse architecture, Databricks workspace is deployed in the control plane along with Databricks services like Databricks web application (UI), Cluster manager, workflow service, and notebooks.

Reference: <https://docs.databricks.com/getting-started/overview.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)

Question 5: **Skipped**

In Databricks Repos, which of the following operations a data engineer can use to update the local version of a repo from its remote Git repository ?

 Clone Commit Merge Push Pull

(Correct)

Explanation

The git Pull operation is used to fetch and download content from a remote repository and immediately update the local repository to match that content.

References:

- <https://docs.databricks.com/repos/index.html>
- <https://github.com/git-guides/git-pull>

Repos

Cluster virtual machines

(Correct)

Workflows

Explanation

When the customer sets up a Spark cluster, the cluster virtual machines are deployed in the data plane in the customer's cloud account.

Reference: <https://docs.databricks.com/getting-started/overview.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)

Question 7: **Skipped**

Which of the following best describes Databricks Lakehouse?

Single, flexible, high-performance system that supports data, analytics, and machine learning workloads.

(Correct)

Reliable data management system with transactional guarantees for organization's structured data.

Platform that helps reduce the costs of storing organization's open-format data files in the cloud.

Platform for developing increasingly complex machine learning workloads using a simple, SQL-based solution.

Platform that scales data lake workloads for organizations without investing on-premises hardware.

Explanation

Databricks Lakehouse is a unified analytics platform that combines the best elements of data lakes and data warehouses. So, in the Lakehouse, you can work on data engineering, analytics, and AI, all in one platform.

Reference: <https://www.databricks.com/glossary/data-lakehouse>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)

Question 8: **Skipped**

If the default notebook language is SQL, which of the following options a data engineer can use to run a Python code in this SQL Notebook ?

They need first to import the python module in a cell

This is not possible! They need to change the default language of the notebook to Python

Databricks detects cells language automatically, so they can write Python syntax in any cell

They can add `%language` magic command at the start of a cell to force language detection.

They can add `%python` at the start of a cell.

(Correct)

Explanation

By default, cells use the default language of the notebook. You can override the default language in a cell by using the language magic command at the beginning of a cell. The language can be either Python or Scala.

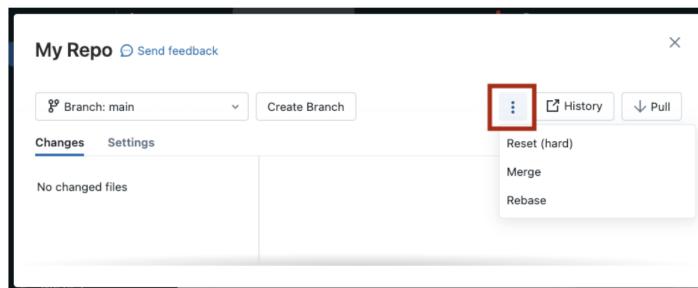
- Clone, push to, or pull from a remote Git repository.**
- Create and manage branches for development work.**
- Create notebooks, and edit notebooks and other files.**
- Visually compare differences upon commit.**
- Delete branches** (Correct)

Explanation

The following tasks are not supported by Databricks Repos, and must be performed in your Git provider:

- Create a pull request
- Delete branches
- Merge and rebase branches *

* **NOTE:** Recently, merge and rebase branches have become supported in Databricks Repos. However, this may still not be updated in the current exam version.



Reference: <https://docs.databricks.com/repos/index.html>

Study materials from our exam preparation course on Udemy:

- [Hands-on](#)

Question 10: **Skipped**

Which of the following statements is **Not** true about Delta Lake ?

- Delta Lake provides ACID transaction guarantees**
- Delta Lake provides scalable data and metadata handling**
- Delta Lake provides audit history and time travel**
- Delta Lake builds upon standard data formats: Parquet + XML** (Correct)
- Delta Lake supports unified streaming and batch data processing**

Explanation

It is not true that Delta Lake builds upon XML format. It builds upon Parquet and JSON formats

Reference: <https://docs.databricks.com/delta/index.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 11: **Skipped**

How long is the default retention period of the VACUUM command ?

- 0 days**

VACUUM operation will prevent you from deleting files less than 7 days old, just to ensure that no long-running operations are still referencing any of the files to be deleted.

Reference: <https://docs.databricks.com/sql/language-manual/delta-vacuum.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 12: **Skipped**

The data engineering team has a Delta table called **employees** that contains the employees personal information including their gross salaries.

Which of the following code blocks will keep in the table only the employees having a salary greater than 3000 ?

- DELETE FROM employees WHERE salary > 3000;**
- SELECT CASE WHEN salary <= 3000 THEN DELETE ELSE UPDATE END FROM employees;**
- UPDATE employees WHERE salary > 3000 WHEN MATCHED SELECT;**
- UPDATE employees WHERE salary <= 3000 WHEN MATCHED DELETE;**
- DELETE FROM employees WHERE salary <= 3000;** (Correct)

Explanation

In order to keep only the employees having a salary greater than 3000, we must delete the employees having salary less than or equal 3000. To do so, use the DELETE statement:

```
DELETE FROM table_name WHERE condition;
```

Reference: <https://docs.databricks.com/sql/language-manual/delta-delete-from.html>

Question 13: **Skipped**

A data engineer wants to create a relational object by pulling data from two tables. The relational object must be used by other data engineers in other sessions on the same cluster only. In order to save on storage costs, the data engineer wants to avoid copying and storing physical data.

Which of the following relational objects should the data engineer create?

- Temporary view**
- External table**
- Managed table**
- Global Temporary view** (Correct)
- View**

Explanation

In order to avoid copying and storing physical data, the data engineer must create a view object. A view in databricks is a virtual table that has no physical data. It's just a saved SQL query against actual tables.

The view type should be Global Temporary view that can be accessed in other sessions on the same cluster. Global Temporary views are tied to a cluster temporary database called global_temp.

Reference: <https://docs.databricks.com/sql/language-manual/sql-ref-syntax-ddl-create-view.html>

Study materials from our exam preparation course on Udemy:

Which of the following changes should be made to the code block to fix this error ?

1 | if process_mode = "init" & not is_table_exist:
2 | print("Start processing ...")

1 | if process_mode = "init" and not is_table_exist = True:
2 | print("Start processing ...")

1 | if process_mode = "init" and is_table_exist = False:
2 | print("Start processing ...")

1 | if (process_mode = "init") and (not is_table_exist):
2 | print("Start processing ...")

1 | if process_mode == "init" and not is_table_exist: (Correct)
2 | print("Start processing ...")

Explanation

Python if statement looks like this in its simplest form:

```
1 | if <expr>:  
2 |   <statement>
```

Python supports the usual logical conditions from mathematics:

- Equals: `a == b`
- Not Equals: `a != b`
- `<`, `<=`, `>`, `>=`

To combine conditional statements, you can use the following logical operators:

- `and`
- `or`

The negation operator in Python is: `not`

Reference: https://www.w3schools.com/python/python_conditions.asp

Question 15: Skipped

Fill in the below blank to successfully create a table in Databricks using data from an existing PostgreSQL database:

```
1 | CREATE TABLE employees  
2 |   USING _____  
3 |   OPTIONS (  
4 |     url "jdbc:postgresql:dbserver",  
5 |     dbtable "employees"  
6 |   )
```

org.apache.spark.sql.jdbc (Correct)

postgresql

DELTA

dbserver

cloudfiles

Explanation

Using the JDBC library, Spark SQL can extract data from any existing relational database that supports JDBC. Examples include mysql, postgres, SQLite, and more.

Reference: <https://learn.microsoft.com/en-us/azure/databricks/external-data/jdbc>

Study materials from our exam preparation course on Udemy:

[Lecture](#)

Question 16: Skipped

Which of the following commands can a data engineer use to create a new table along with a comment ?

```
1 | COMMENT("This table contains sensitive information")
2 | CREATE TABLE payments
3 | AS SELECT * FROM bank_transactions
```

Explanation

The **CREATE TABLE** clause supports adding a descriptive comment for the table. This allows for easier discovery of table contents.

Syntax:

```
1 | CREATE TABLE table_name
2 | COMMENT "here is a comment"
3 | AS query
```

Reference: <https://docs.databricks.com/sql/language-manual/sql-ref-syntax-ddl-create-table-using.html>

Study materials from our exam preparation course on Udemy:

[Lecture](#)

Question 17: **Skipped**

A junior data engineer usually uses **INSERT INTO** command to write data into a Delta table. A senior data engineer suggested using another command that avoids writing of duplicate records.

Which of the following commands is the one suggested by the senior data engineer ?

- MERGE INTO** (Correct)
- APPLY CHANGES INTO**
- UPDATE**
- COPY INTO**
- INSERT OR OVERWRITE**

Explanation

MERGE INTO allows to merge a set of updates, insertions, and deletions based on a source table into a target Delta table. With **MERGE INTO**, you can avoid inserting the duplicate records when writing into Delta tables.

References:

- <https://docs.databricks.com/sql/language-manual/delta-merge-into.html>
- <https://docs.databricks.com/delta/merge.html#data-deduplication-when-writing-into-delta-tables>

Study materials from our exam preparation course on Udemy:

[Hands-on](#)

Question 18: **Skipped**

A data engineer is designing a Delta Live Tables pipeline. The source system generates files containing changes captured in the source data. Each change event has metadata indicating whether the specified record was inserted, updated, or deleted. In addition to a timestamp column indicating the order in which the changes happened. The data engineer needs to update a target table based on these change events.

Which of the following commands can the data engineer use to best solve this problem?

- MERGE INTO**
- APPLY CHANGES INTO** (Correct)
- UPDATE**
- COPY INTO**

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 19: **Skipped**

In PySpark, which of the following commands can you use to query the Delta table **employees** created in Spark SQL?

`pyspark.sql.read(SELECT * FROM employees)`

`spark.sql("employees")`

`spark.format("sql").read("employees")`

`spark.table("employees")`

(Correct)

Spark SQL tables can not be accessed from PySpark

Explanation

`spark.table()` function returns the specified Spark SQL table as a PySpark DataFrame

Reference:

https://spark.apache.org/docs/2.4.0/api/python/_modules/pyspark/sql/session.html#SparkSession.table

Study materials from our exam preparation course on Udemy:

[Hands-on](#)

Question 20: **Skipped**

Which of the following code blocks can a data engineer use to create a user defined function (UDF) ?

`CREATE FUNCTION plus_one(value INTEGER)
RETURN value +1`

`CREATE UDF plus_one(value INTEGER)
RETURNS INTEGER
RETURN value +1;`

`CREATE UDF plus_one(value INTEGER)
RETURN value +1;`

`CREATE FUNCTION plus_one(value INTEGER)
RETURNS INTEGER
RETURN value +1;`

(Correct)

`CREATE FUNCTION plus_one(value INTEGER)
RETURNS INTEGER
value +1;`

Explanation

The correct syntax to create a UDF is:

```
1 | CREATE [OR REPLACE] FUNCTION function_name ( [ parameter_name data_type  
[,...] ] )  
2 | RETURNS data_type  
3 | RETURN { expression | query }
```

- The table is external (Correct)
- The user running the command has no permission to delete the data files
- The table is managed
- Delta prevents deleting files less than retention threshold, just to ensure that no long-running operations are still referencing any of the files to be deleted

Explanation

External (unmanaged) tables are tables whose data is stored in an external storage path by using a `LOCATION` clause.

When you run `DROP TABLE` on an external table, only the table's metadata is deleted, while the underlying data files are kept.

Reference: <https://docs.databricks.com/lakehouse/data-objects.html#what-is-an-unmanaged-table>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 22: **Skipped**

Given the two tables `students_course_1` and `students_course_2`. Which of the following commands can a data engineer use to get all the students from the above two tables without duplicate records ?

- 1 | SELECT * FROM students_course_1
2 | CROSS JOIN
3 | SELECT * FROM students_course_2
- 1 | SELECT * FROM students_course_1
2 | UNION
3 | SELECT * FROM students_course_2 (Correct)
- 1 | SELECT * FROM students_course_1
2 | INTERSECT
3 | SELECT * FROM students_course_2
- 1 | SELECT * FROM students_course_1
2 | OUTER JOIN
3 | SELECT * FROM students_course_2
- 1 | SELECT * FROM students_course_1
2 | INNER JOIN
3 | SELECT * FROM students_course_2

Explanation

With `UNION`, you can return the result of subquery1 plus the rows of subquery2

Syntax:

```
1 | subquery1
2 | UNION [ ALL | DISTINCT ]
3 | subquery2
```

- If `ALL` is specified duplicate rows are preserved.
- If `DISTINCT` is specified the result does not contain any duplicate rows. This is the default.

Note that both subqueries must have the same number of columns and share a least common type for each respective column.

Reference: <https://docs.databricks.com/sql/language-manual/sql-ref-syntax-qry-select-setops.html>

Study materials from our exam preparation course on Udemy:

- [Hands-on](#)

Question 23: **Skipped**

dbfs:/user/hive/databases

dbfs:/user/hive

Explanation

Since we are creating the database here without specifying a **LOCATION** clause, the database will be created in the default warehouse directory under dbfs:/user/hive/warehouse

Reference: <https://docs.databricks.com/sql/language-manual/sql-ref-syntax-ddl-create-schema.html>

Study materials from our exam preparation course on Udemy:

[Lecture](#)

[Hands-on](#)

Question 24: **Skipped**

Given the following table **faculties**

faculty_id	faculty_name	students
F001	Faculty of Medicine	[({"student_id": "S000002501", "total_courses": "6"}, {"student_id": "S000004478", "total_courses": "2"}, {"student_id": "S0000015721", "total_courses": "5"}, {"student_id": "S000003859", "total_courses": "1"}]
F002	Faculty of Economics	[({"student_id": "S000007415", "total_courses": "3"}, {"student_id": "S000001177", "total_courses": "4"}, {"student_id": "S000005631", "total_courses": "7"}, {"student_id": "S000001003", "total_courses": "6"}]
F003	Faculty of Engineering	[({"student_id": "S000007251", "total_courses": "2"}, {"student_id": "S000002415", "total_courses": "5"})]

Fill in the below blank to get the students enrolled in less than 3 courses from the array column **students**

```

1 | SELECT
2 |   faculty_id,
3 |   students,
4 |   _____ AS few_courses_students
5 | FROM faculties

```

TRANSFORM (students, total_courses < 3)

TRANSFORM (students, i -> i.total_courses < 3)

FILTER (students, total_courses < 3)

FILTER (students, i -> i.total_courses < 3) (Correct)

**CASE WHEN students.total_courses < 3 THEN students
ELSE NULL
END**

Explanation

`filter(input_array, lambda_function)` is a higher order function that returns an output array from an input array by extracting elements for which the predicate of a lambda function holds.

Example:

Extracting odd numbers from an input array of integers:

```
SELECT filter(array(1, 2, 3, 4), i -> i % 2 == 1);
```

output: [1, 3]

References:

- <https://docs.databricks.com/sql/language-manual/functions/filter.html>
- <https://docs.databricks.com/optimizations/higher-order-lambda-functions.html>

Study materials from our exam preparation course on Udemy:

[Hands-on](#)

Question 25: **Skipped**

<input type="radio"/> trigger(processingTime="2 minutes")	(Correct)
<input type="radio"/> processingTime("2 minutes")	
<input type="radio"/> trigger("2 minutes")	
<input type="radio"/> trigger()	

Explanation

In Spark Structured Streaming, in order to process data in micro-batches at the user-specified intervals, you can use `processingTime` keyword. It allows to specify a time duration as a string.

Reference: <https://docs.databricks.com/structured-streaming/triggers.html#configure-structured-streaming-trigger-intervals>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 26: **Skipped**

Which of the following is used by Auto Loader to load data incrementally?

<input type="radio"/> DEEP CLONE	
<input type="radio"/> Multi-hop architecture	
<input type="radio"/> COPY INTO	
<input type="radio"/> Spark Structured Streaming	(Correct)
<input type="radio"/> Databricks SQL	

Explanation

Auto Loader is based on Spark Structured Streaming. It provides a Structured Streaming source called `cloudFiles`.

Reference: <https://docs.databricks.com/ingestion/auto-loader/index.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 27: **Skipped**

Which of the following statements best describes Auto Loader ?

<input type="radio"/> Auto loader allows applying Change Data Capture (CDC) feed to update tables based on changes captured in source data.	
<input type="radio"/> Auto loader monitors a source location, in which files accumulate, to identify and ingest only new arriving files with each command run. While the files that have already been ingested in previous runs are skipped.	(Correct)
<input type="radio"/> Auto loader allows cloning a source Delta table to a target destination at a specific version.	
<input type="radio"/> Auto loader defines data quality expectations on the contents of a dataset, and reports the records that violate these expectations in metrics.	
<input type="radio"/> Auto loader enables efficient insert, update, deletes, and rollback capabilities by adding a storage layer that provides better data reliability to data lakes.	

Question 28: **Skipped**

A data engineer has defined the following data quality constraint in a Delta Live Tables pipeline:

`CONSTRAINT valid_id EXPECT (id IS NOT NULL) _____`

Fill in the above blank so records violating this constraint will be added to the target table, and reported in metrics

ON VIOLATION ADD ROW

ON VIOLATION FAIL UPDATE

ON VIOLATION SUCCESS UPDATE

ON VIOLATION NULL

There is no need to add ON VIOLATION clause. By default, records violating the constraint will be kept, and reported as invalid in the event log (Correct)

Explanation

By default, records that violate the expectation are added to the target dataset along with valid records, but violations will be reported in the event log

Reference:

<https://learn.microsoft.com/en-us/azure/databricks/workflows/delta-live-tables/delta-live-tables-expectations>

Study materials from our exam preparation course on Udemy:

- [Hands-on](#)

Question 29: **Skipped**

The data engineer team has a DLT pipeline that updates all the tables once and then stops. The compute resources of the pipeline continue running to allow for quick testing.

Which of the following best describes the execution modes of this DLT pipeline ?

The DLT pipeline executes in Continuous Pipeline mode under Production mode.

The DLT pipeline executes in Continuous Pipeline mode under Development mode.

The DLT pipeline executes in Triggered Pipeline mode under Production mode.

The DLT pipeline executes in Triggered Pipeline mode under Development mode. (Correct)

More information is needed to determine the correct response

Explanation

Triggered pipelines update each table with whatever data is currently available and then they shut down.

In Development mode, the Delta Live Tables system ease the development process by

- Reusing a cluster to avoid the overhead of restarts. The cluster runs for two hours when development mode is enabled.
- Disabling pipeline retries so you can immediately detect and fix errors.

Reference:

<https://docs.databricks.com/workflows/delta-live-tables/delta-live-tables-concepts.html>

Study materials from our exam preparation course on Udemy:

Dashboards (Correct)

Streaming jobs

Explanation

Gold tables provide business level aggregates often used for reporting and dashboarding, or even for Machine learning

Reference:

<https://www.databricks.com/glossary/medallion-architecture>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)

Question 31: **Skipped**

Which of the following code blocks can a data engineer use to query the existing streaming table **events** ?

`spark.readStream("events")`

`spark.read`
`.table("events")`

`spark.readStream`
`.table("events")` (Correct)

`spark.readStream()`
`.table("events")`

`spark.stream`
`.read("events")`

Explanation

Delta Lake is deeply integrated with Spark Structured Streaming. You can load tables as a stream using:

`spark.readStream.table(table_name)`

Reference: <https://docs.databricks.com/structured-streaming/delta-lake.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 32: **Skipped**

In multi-hop architecture, which of the following statements best describes the Bronze layer ?

It maintains data that powers analytics, machine learning, and production applications

It maintains raw data ingested from various sources (Correct)

It represents a filtered, cleaned, and enriched version of data

It provides business-level aggregated version of data

It provides a more refined view of the data.

Explanation

Bronze tables contain data in its rawest format ingested from various sources (e.g., JSON files, Operational Databaes, Kafka stream, ...)

```

1 |     .option("cloudFiles")
2 |     .format("cloudFiles")
3 |     .option("cloudFiles.format", "json")
4 |     .load(ordersLocation)
5 | 
6 |     .writeStream
7 |     .option("checkpointLocation", checkpointPath)
8 |     .table("uncleanedOrders")

```

Which of the following best describe the purpose of this query in a multi-hop architecture?

The query is performing raw data ingestion into a Bronze table (Correct)

The query is performing a hop from a Bronze table to a Silver table

The query is performing a hop from Silver table to a Gold table

The query is performing data transfer from a Gold table into a production application

This query is performing data quality controls prior to Silver layer

Explanation

The query here is using Autoloader (**cloudFiles**) to load raw **json** data from **ordersLocation** into the Bronze table **uncleanedOrders**

References:

- <https://www.databricks.com/glossary/medallion-architecture>
- <https://docs.databricks.com/ingestion/auto-loader/index.html>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 34: Skipped

A data engineer has the following query in a Delta Live Tables pipeline:

```

1 | CREATE LIVE TABLE aggregated_sales
2 | AS
3 |   SELECT store_id, sum(total)
4 |   FROM cleaned_sales
5 |   GROUP BY store_id

```

The pipeline is failing to start due to an error in this query

Which of the following changes should be made to this query to successfully start the DLT pipeline ?

```

1 | CREATE STREAMING TABLE aggregated_sales
2 | AS
3 |   SELECT store_id, sum(total)
4 |   FROM LIVE.cleaned_sales
5 |   GROUP BY store_id

```

```

1 | CREATE TABLE aggregated_sales
2 | AS
3 |   SELECT store_id, sum(total)
4 |   FROM LIVE.cleaned_sales
5 |   GROUP BY store_id

```

```

1 | CREATE LIVE TABLE aggregated_sales
2 | AS
3 |   SELECT store_id, sum(total)
4 |   FROM LIVE.cleaned_sales
5 |   GROUP BY store_id

```

(Correct)

```

1 | CREATE STREAMING LIVE TABLE aggregated_sales
2 | AS
3 |   SELECT store_id, sum(total)
4 |   FROM cleaned_sales
5 |   GROUP BY store_id

```

```

1 | CREATE STREAMING LIVE TABLE aggregated_sales
2 | AS
3 |   SELECT store_id, sum(total)
4 |   FROM STREAM(cleaned_sales)
5 |   GROUP BY store_id

```

Explanation

Question 35: **Skipped**

A data engineer has defined the following data quality constraint in a Delta Live Tables pipeline:

`CONSTRAINT valid_id EXPECT (id IS NOT NULL) _____`

Fill in the above blank so records violating this constraint will be dropped, and reported in metrics

- ON VIOLATION DROP ROW** (Correct)
- ON VIOLATION FAIL UPDATE**
- ON VIOLATION DELETE ROW**
- ON VIOLATION DISCARD ROW**
- There is no need to add ON VIOLATION clause. By default, records violating the constraint will be discarded, and reported as invalid in the event log**

Explanation

With **ON VIOLATION DROP ROW**, records that violate the expectation are dropped, and violations are reported in the event log

Reference:

<https://learn.microsoft.com/en-us/azure/databricks/workflows/delta-live-tables/delta-live-tables-expectations>

Study materials from our exam preparation course on Udemy:

- [Hands-on](#)

Question 36: **Skipped**

Which of the following compute resources is available in Databricks SQL ?

- Single-node clusters**
- Multi-nodes clusters**
- On-premises clusters**
- SQL warehouses** (Correct)
- SQL engines**

Explanation

Compute resources are infrastructure resources that provide processing capabilities in the cloud. A SQL warehouse is a compute resource that lets you run SQL commands on data objects within Databricks SQL.

Reference: <https://docs.databricks.com/sql/admin/sql-endpoints.html>

Study materials from our exam preparation course on Udemy:

[Hands-on](#)

Question 37: **Skipped**

Which of the following is the benefit of using the Auto Stop feature of Databricks SQL warehouses ?

- Improves the performance of the warehouse by automatically stopping**

Explanation

The Auto Stop feature stops the warehouse if it's idle for a specified number of minutes.

Reference: <https://docs.databricks.com/sql/admin/sql-endpoints.html>

Study materials from our exam preparation course on Udemy:

- [Hands-on](#)

Question 38: **Skipped**

Which of the following alert destinations is **Not** supported in Databricks SQL ?

Slack

Webhook

SMS

(Correct)

Microsoft Teams

Email

Explanation

SMS is not supported as an alert destination in Databricks SQL . While, email, webhook, Slack, and Microsoft Teams are supported alert destinations in Databricks SQL.

Reference: <https://docs.databricks.com/sql/admin/alert-destinations.html>

Study materials from our exam preparation course on Udemy:

- [Hands-on](#)

Question 39: **Skipped**

A data engineering team has a long-running multi-tasks Job. The team members need to be notified when the run of this job completes.

Which of the following approaches can be used to send emails to the team members when the job completes ?

They can use Job API to programmatically send emails according to each task status

They can configure email notifications settings in the job page (Correct)

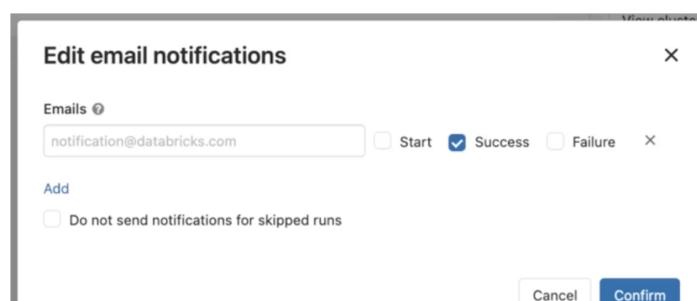
There is no way to notify users when the job completes

Only Job owner can be configured to be notified when the job completes

They can configure email notifications settings per notebook in the task page

Explanation

Databricks Jobs supports email notifications to be notified in the case of job start, success, or failure. Simply, click **Edit email notifications** from the details panel in the Job page. From there, you can add one or more email addresses.



Which of the following is the benefit of increasing the cluster size of Databricks SQL warehouses ?

- Improves the latency of the queries execution (Correct)
- Speeds up the start up time of the SQL warehouse
- Reduces cost since large clusters use Spot instances
- The cluster size of SQL warehouses is not configurable. Instead, they can increase the number of clusters
- The cluster size can not be changed for existing SQL warehouses. Instead, they can enable the auto-scaling option.

Explanation

Cluster Size represents the number of cluster workers and size of compute resources available to run your queries and dashboards. To reduce query latency, you can increase the cluster size.

Reference: <https://docs.databricks.com/sql/admin/sql-endpoints.html#cluster-size-1>

Study materials from our exam preparation course on Udemy:

[Hands-on](#)

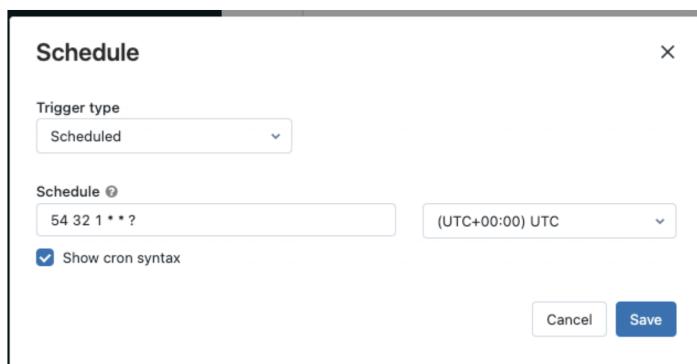
Question 41: **Skipped**

Which of the following describes Cron syntax in Databricks Jobs ?

- It's an expression to represent the maximum concurrent runs of a job
- It's an expression to represent complex job schedule that can be defined programmatically (Correct)
- It's an expression to represent the retry policy of a job
- It's an expression to describe the email notification events (start, success, failure)
- It's an expression to represent the run timeout of a job

Explanation

To define a schedule for a Databricks job, you can either interactively specify the period and starting time, or write a Cron Syntax expression. The Cron Syntax allows to represent complex job schedule that can be defined programmatically



Reference: <https://docs.databricks.com/workflows/jobs/jobs.html#schedule-a-job>

Study materials from our exam preparation course on Udemy:

[Hands-on](#)

The DLT pipeline executes in Triggered Pipeline mode under Production mode.

The DLT pipeline executes in Triggered Pipeline mode under Development mode.

More information is needed to determine the correct response

Explanation

Continuous pipelines update tables continuously as input data changes. Once an update is started, it continues to run until the pipeline is shut down.

In Production mode, the Delta Live Tables system:

- Terminates the cluster immediately when the pipeline is stopped.
- Restarts the cluster for recoverable errors (e.g., memory leak or stale credentials).
- Retries execution in case of specific errors (e.g., a failure to start a cluster)

Reference:

<https://docs.databricks.com/workflows/delta-live-tables/delta-live-tables-concepts.html>

Study materials from our exam preparation course on Udemy:

- [Hands-on](#)

Question 43: **Skipped**

Which part of the Databricks Platform can a data engineer use to grant permissions on tables to users ?

Data Studio

Cluster event log

Workflows

DBFS

Data Explorer

(Correct)

Explanation

Data Explorer in Databricks SQL allows you to manage data object permissions. This includes granting privileges on tables and databases to users or groups of users.

Reference: <https://docs.databricks.com/security/access-control/data-acl.html#data-explorer>

Study materials from our exam preparation course on Udemy:

- [Hands-on](#)

Question 44: **Skipped**

Which of the following commands can a data engineer use to grant full permissions to the HR team on the table **employees** ?

GRANT FULL PRIVILEGES ON TABLE employees TO hr_team

GRANT FULL PRIVILEGES ON TABLE hr_team TO employees

GRANT ALL PRIVILEGES ON TABLE employees TO hr_team

(Correct)

GRANT ALL PRIVILEGES ON TABLE hr_team TO employees

GRANT SELECT, MODIFY, CREATE, READ_METADATA ON TABLE employees TO hr_team

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)

Question 45: **Skipped**

A data engineer uses the following SQL query:

```
GRANT MODIFY ON TABLE employees TO hr_team
```

Which of the following describes the ability given by the **MODIFY** privilege ?

- It gives the ability to add data from the table
- It gives the ability to delete data from the table
- It gives the ability to modify data in the table
- All the above abilities are given by the **MODIFY** privilege (Correct)
- None of these options correctly describe the ability given by the **MODIFY** privilege

Explanation

The **MODIFY** privilege gives the ability to add, delete, and modify data to or from an object.

Reference: <https://docs.databricks.com/security/access-control/table-acls/object-privileges.html#privileges>

Study materials from our exam preparation course on Udemy:

- [Lecture](#)
- [Hands-on](#)