DATABRICKS ASSOCIATE

Contents

[Question 1 5](#_Toc154466713)

[Question 2 6](#_Toc154466714)

[Question 3 7](#_Toc154466715)

[Question 4 8](#_Toc154466716)

[Question 5 9](#_Toc154466717)

[Question 6 10](#_Toc154466718)

[Question 7 11](#_Toc154466719)

[Question 8 12](#_Toc154466720)

[Question 9 13](#_Toc154466721)

[Question 10 14](#_Toc154466722)

[Question 11 15](#_Toc154466723)

[Question 12 16](#_Toc154466724)

[Question 13 17](#_Toc154466725)

[Question 14 18](#_Toc154466726)

[Question 15 19](#_Toc154466727)

[Question 16 20](#_Toc154466728)

[Question 17 21](#_Toc154466729)

[Question 18 22](#_Toc154466730)

[Question 19 23](#_Toc154466731)

[Question 20: 24](#_Toc154466732)

[Question 21 25](#_Toc154466733)

[Question 22 26](#_Toc154466734)

[Question 23 27](#_Toc154466735)

[Question 24 28](#_Toc154466736)

[Question 25 29](#_Toc154466737)

[Question 26 30](#_Toc154466738)

[Question 27 31](#_Toc154466739)

[Question 28: 32](#_Toc154466740)

[Question 29: 33](#_Toc154466741)

[Question 30 34](#_Toc154466742)

[Question 31 35](#_Toc154466743)

[Question 32 36](#_Toc154466744)

[Question 33: 37](#_Toc154466745)

[Question 34: 38](#_Toc154466746)

[Question 35: 40](#_Toc154466747)

[Question 36 41](#_Toc154466748)

[Question 37 42](#_Toc154466749)

[Question 38: 43](#_Toc154466750)

[Question 39 44](#_Toc154466751)

[Question 40: 45](#_Toc154466752)

[Question 41 46](#_Toc154466753)

[Question 42: 47](#_Toc154466754)

[Question 43 48](#_Toc154466755)

[Question 44 49](#_Toc154466756)

[Question 45 50](#_Toc154466757)

[Question 46: 51](#_Toc154466758)

[Question 47 52](#_Toc154466759)

[Question 48 52](#_Toc154466760)

[Question 49 53](#_Toc154466761)

[Question 50 54](#_Toc154466762)

[Question 51 55](#_Toc154466763)

[Question 52 56](#_Toc154466764)

[Question 53 57](#_Toc154466765)

[Question 54 58](#_Toc154466766)

[Question 55: 59](#_Toc154466767)

[Question 56 60](#_Toc154466768)

[Question 57 61](#_Toc154466769)

[Question 58 62](#_Toc154466770)

[Question 59 63](#_Toc154466771)

[Question 60: 64](#_Toc154466772)

[Question 61 65](#_Toc154466773)

[Question 62 66](#_Toc154466774)

[Question 63 67](#_Toc154466775)

[Question 64 68](#_Toc154466776)

[Question 65 69](#_Toc154466777)

[Question 66 70](#_Toc154466778)

[Question 67 71](#_Toc154466779)

[Question 68 72](#_Toc154466780)

[Question 69 73](#_Toc154466781)

[Question 70 74](#_Toc154466782)

[Question 71 75](#_Toc154466783)

[Question 72 76](#_Toc154466784)

[Question 73 77](#_Toc154466785)

[Question 74 78](#_Toc154466786)

[Question 75 79](#_Toc154466787)

[Question 76 80](#_Toc154466788)

[Question 77 81](#_Toc154466789)

[Question 78 83](#_Toc154466790)

[Question 79 84](#_Toc154466791)

[Question 80 85](#_Toc154466792)

[Question 81 86](#_Toc154466793)

[Question 82 87](#_Toc154466794)

[Question 83 88](#_Toc154466795)

[Question 84 89](#_Toc154466796)

[Question 85 90](#_Toc154466797)

[Question 86 91](#_Toc154466798)

[Question 87 92](#_Toc154466799)

[Question 88 93](#_Toc154466800)

[Question 89 94](#_Toc154466801)

[Question 90 95](#_Toc154466802)

## Question 1

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**

**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>

## Question 2

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**
* **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>

## Question 3

In Delta Lake tables, which of the following is the primary format for the data files?

* **Delta**
* **Parquet**
* **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>

## Question 4

Which of the following locations hosts the Databricks web application ?

* **Data plane**
* **Control plane**
* **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>

## Question 5

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**

**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>

## Question 6

According to the Databricks Lakehouse architecture, which of the following is located in the customer's cloud account?

* **Databricks web application**
* **Notebooks**
* **Repos**
* **Cluster virtual machines**
* **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>

## Question 7

Which of the following best describes Databricks Lakehouse?

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

## Question 8

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.**

**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 supported magic commands are: %python, %sql, %scala, and %r.

Reference: <https://docs.databricks.com/notebooks/notebooks-code.html>

## Question 9

Which of the following tasks is not supported by Databricks Repos, and must be performed in your Git provider?

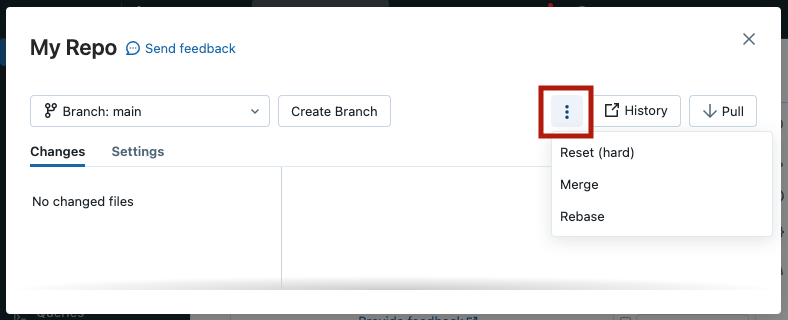
* **Clone, push to, or pull from a remote Git repository.**
* **Create and manage branches for development work.**
* **Create notebooks, and edit notebooks and other files.**
* **Visually compare differences upon commit.**
* **Delete branches**

**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>

## Question 10

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**
* **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>

## Question 11

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

* **0 days**
* **7 days**
* **30 days**
* **90 days**
* **365 days**

**Explanation**

By default, the retention threshold of the VACUUM command is 7 days. This means that 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>

## Question 12

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;**

**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

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 date 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**
* **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>

## Question 14

A data engineer has developed a code block to completely reprocess data based on the following if-condition in Python:

if process\_mode = "init" and not `:

print("Start processing ...")

This if-condition is returning an invalid syntax error.

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

* if process\_mode = "init" & not is\_table\_exist:

print("Start processing ...")

* if process\_mode = "init" and not is\_table\_exist = True:

print("Start processing ...")

* if process\_mode = "init" and is\_table\_exist = False:

print("Start processing ...")

* if (process\_mode = "init") and (not is\_table\_exist):

print("Start processing ...")

* if process\_mode == "init" and not is\_table\_exist:

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

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

CREATE TABLE employees

USING \_\_\_\_\_\_\_\_\_\_\_\_

OPTIONS (

url "jdbc:postgresql:dbserver",

dbtable "employees"

)

* **org.apache.spark.sql.jdbc**
* **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>

## Question 16

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

CREATE TABLE payments

COMMENT "This table contains sensitive information"

AS SELECT \* FROM bank\_transactions

CREATE TABLE payments

COMMENT("This table contains sensitive information")

AS SELECT \* FROM bank\_transactions

CREATE TABLE payments

AS SELECT \* FROM bank\_transactions

COMMENT "This table contains sensitive information"

CREATE TABLE payments

AS SELECT \* FROM bank\_transactions

COMMENT("This table contains sensitive information")

COMMENT("This table contains sensitive information")

CREATE TABLE payments

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>

## Question 17

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**
* **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>

## Question 18

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**
* **UPDATE**
* **COPY INTO**
* **cloud\_files**

**Explanation**

The events described in the question represent Change Data Capture (CDC) feed. CDC is logged at the source as events that contain both the data of the records along with metadata information:

1. Operation column indicating whether the specified record was inserted, updated, or deleted
2. Sequence column that is usually a timestamp indicating the order in which the changes happened

You can use the APPLY CHANGES INTO statement to use Delta Live Tables CDC functionality

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

## Question 19

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")**
* **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>

## Question 20:

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;**

* **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 }

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

## Question 21

When dropping a Delta table, which of the following explains why only the table's metadata will be deleted, while the data files will be kept in the storage ?

* **The table is deep cloned**
* **The table is external**
* **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>

## Question 22

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 ?

* SELECT \* FROM students\_course\_1

CROSS JOIN

SELECT \* FROM students\_course\_2

* SELECT \* FROM students\_course\_1

UNION

SELECT \* FROM students\_course\_2

* SELECT \* FROM students\_course\_1

INTERSECT

SELECT \* FROM students\_course\_2

* SELECT \* FROM students\_course\_1

OUTER JOIN

SELECT \* FROM students\_course\_2

* SELECT \* FROM students\_course\_1

INNER JOIN

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>

## Question 23

Given the following command:

CREATE DATABASE IF NOT EXISTS hr\_db ;

In which of the following locations will the **hr\_db** database be located?

* **dbfs:/user/hive/warehouse**
* **dbfs:/user/hive/db\_hr**
* **dbfs:/user/hive/databases/db\_hr.db**
* **dbfs:/user/hive/databases**
* **dbfs:/user/hive**

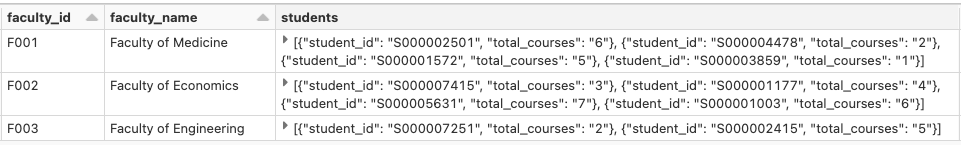
**Explanation**

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>

## Question 24

Given the following table **faculties**

****

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

SELECT

faculty\_id,

students,

\_\_\_\_\_\_\_\_\_\_\_ AS few\_courses\_students

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)**
* **CASE WHEN students.total\_courses < 3 THEN students**

**ELSE NULL**

**END**

**Explanation**

filter(input\_array, lamda\_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>

## Question 25

Given the following Structured Streaming query:

(spark.table("orders")

.withColumn("total\_after\_tax", col("total")+col("tax"))

.writeStream

.option("checkpointLocation", checkpointPath)

.outputMode("append")

.\_\_\_\_\_\_\_\_\_\_\_\_\_\_

.table("new\_orders")

)

Fill in the blank to make the query executes a micro-batch to process data every 2 minutes

* **trigger(once=”2 minutes”)**
* **trigger(processingTime=”2 minutes")**
* **processingTime(”2 minutes")**
* **trigger(”2 minutes")**
* **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>

## Question 26

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

* **DEEP CLONE**
* **Multi-hop architecture**
* **COPY INTO**
* **Spark Structured Streaming**
* **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>

## Question 27

Which of the following statements best describes Auto Loader ?

* **Auto loader allows applying Change Data Capture (CDC) feed to update tables based on changes captured in source data.**
* **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.**
* **Auto loader allows cloning a source Delta table to a target destination at a specific version.**
* **Auto loader defines data quality expectations on the contents of a dataset, and reports the records that violate these expectations in metrics.**
* **Auto loader enables efficient insert, update, deletes, and rollback capabilities by adding a storage layer that provides better data reliability to data lakes.**

**Explanation**

Auto Loader incrementally and efficiently processes new data files as they arrive in cloud storage.

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

## Question 28:

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**

**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>

## Question 29:

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.**
* **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>

## Question 30

Which of the following will utilize Gold tables as their source?

* **Silver tables**
* **Auto loader**
* **Bronze tables**
* **Dashboards**
* **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>

## Question 31

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")**
* **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>

## Question 32

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**
* **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, Kakfa stream, ...)

Reference:

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

## Question 33:

Given the following Structured Streaming query

(spark.readStream

.format("cloudFiles")

.option("cloudFiles.format", "json")

.load(ordersLocation)

.writeStream

.option("checkpointLocation", checkpointPath)

.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**
* **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>

## Question 34:

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

CREATE LIVE TABLE aggregated\_sales

AS

SELECT store\_id, sum(total)

FROM cleaned\_sales

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 ?

CREATE STREAMING TABLE aggregated\_sales

AS

  SELECT store\_id, sum(total)

  FROM LIVE.cleaned\_sales

  GROUP BY store\_id

CREATE TABLE aggregated\_sales

AS

  SELECT store\_id, sum(total)

  FROM LIVE.cleaned\_sales

  GROUP BY store\_id

CREATE LIVE TABLE aggregated\_sales

AS

  SELECT store\_id, sum(total)

  FROM LIVE.cleaned\_sales

  GROUP BY store\_id

CREATE STREAMING LIVE TABLE aggregated\_sales

AS

  SELECT store\_id, sum(total)

  FROM cleaned\_sales

  GROUP BY store\_id

CREATE STREAMING LIVE TABLE aggregated\_sales

AS

  SELECT store\_id, sum(total)

  FROM STREAM(cleaned\_sales)

  GROUP BY store\_id

**Explanation**

In DLT pipelines, we use the **CREATE LIVE TABLE** syntax to create a table with SQL. To query another live table, prepend the **LIVE.** keyword to the table name.

**CREATE LIVE TABLE** aggregated\_sales

AS

SELECT store\_id, sum(total)

FROM **LIVE.**cleaned\_sales

GROUP BY store\_id

Reference: <https://docs.databricks.com/workflows/delta-live-tables/delta-live-tables-sql-ref.html>

## Question 35:

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**
* **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>

## Question 36

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

* **Single-node clusters**
* **Multi-nodes clusters**
* **On-premises clusters**
* **SQL warehouses**
* **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>

## Question 37

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 ideal services**
* **Minimizes the total running time of the warehouse**
* **Provides higher security by automatically stopping unused ports of the warehouse**
* **Increases the availability of the warehouse by automatically stopping long-running SQL queries**
* **Databricks SQL does not have Auto Stop feature**

**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>

## Question 38:

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

* **Slack**
* **Webhook**
* **SMS**
* **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>

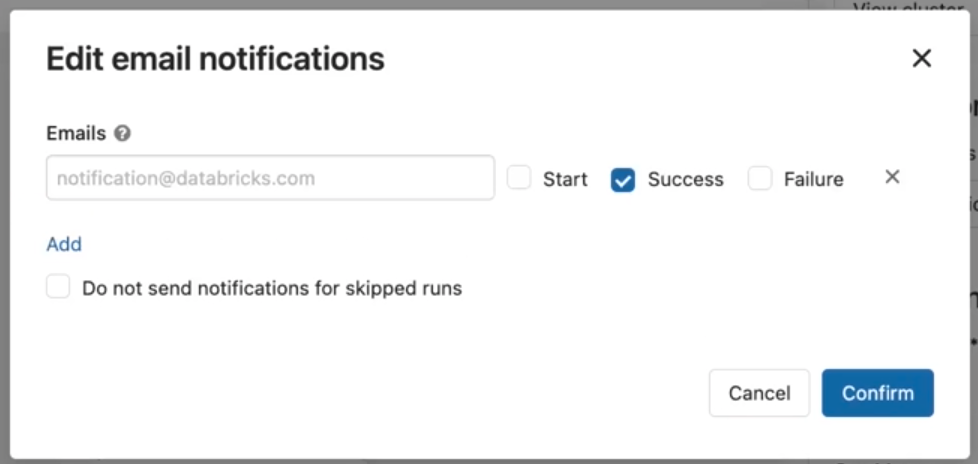
## Question 39

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**
* **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.



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

## Question 40:

A data engineer wants to increase the cluster size of an existing Databricks SQL warehouse.

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

* **Improves the latency of the queries execution**
* **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>

## Question 41

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**

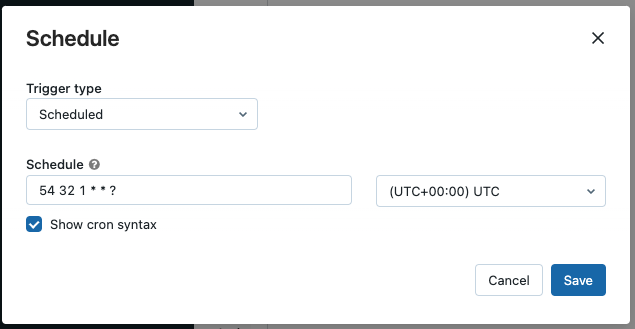
**It’s an expression to represent the retry policy of a job**

**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>

## Question 42:

The data engineer team has a DLT pipeline that updates all the tables at defined intervals until manually stopped. The compute resources terminate when the pipeline is stopped.

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.**
* **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>

## Question 43

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**

**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>

## Question 44

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**
* **GRANT ALL PRIVILEGES ON TABLE hr\_team TO employees**
* **GRANT SELECT, MODIFY, CREATE, READ\_METADATA ON TABLE employees TO hr\_team**

**Explanation**

ALL PRIVILEGES is used to grant full permissions on an object to a user or group of users. It is translated into all the below privileges:

* SELECT
* CREATE
* MODIFY
* USAGE
* READ\_METADATA

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

## Question 45

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**
* **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>

## Question 46:

One of the foundational technologies provided by the Databricks Lakehouse Platform is an open-source, file-based storage format that brings reliability to data lakes.

Which of the following technologies is being described in the above statement?

* **Delta Lives Tables (DLT)**
* **Delta Lake**
* **Apache Spark**
* **Unity Catalog**
* **Photon**

**Explanation**

Delta Lake is an open source technology that extends Parquet data files with a file-based transaction log for ACID transactions that brings reliability to data lakes.

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

## Question 47

Which of the following commands can a data engineer use to purge stale data files of a Delta table?

* + **DELETE**
  + **GARBAGE COLLECTION**
  + **CLEAN**
  + **VACUUM**
  + **OPTIMIZE**

**Explanation**

The VACUUM command deletes the unused data files older than a specified data retention period.

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

## Question 48

In Databricks Repos, which of the following operations a data engineer can use to save local changes of a repo to its remote repository ?

* **Create Pull Request**
* **Commit & Pull**
* **Commit & Push**
* **Merge & Push**
* **Merge & Pull**

**Explanation**

Commit & Push is used to save the changes on a local repo, then uploads this local repo content to the remote repository.

References:

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

## Question 49

In Delta Lake tables, which of the following is the primary format for the transaction log files?

* **Delta**
* **Parquet**
* **JSON**
* **Hive-specific format**
* **XML**

**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>

## Question 50

Which of the following functionalities can be performed in Databricks Repos ?

**Create pull requests**

**Create new remote Git repositories**

**Delete branches**

**Create CI/CD pipelines**

**Pull from a remote Git repository**

**Explanation**

Databricks Repos supports git Pull operation. It is used to fetch and download content from a remote repository and immediately update the local repo to match that content.

References:

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

## Question 51

Which of the following locations completely hosts the customer data ?

* **Customer's cloud account**
* **Control plane**
* **Databricks account**
* **Databricks-managed cluster**
* **Repos**

**Explanation**

According to the Databricks Lakehouse architecture, the storage account hosting the customer data is provisioned in the data plane in the Databricks customer's cloud account.

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

## Question 52

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

* **They need first to import the SQL library in a cell**
* **This is not possible! They need to change the default language of the notebook to SQL**
* **Databricks detects cells language automatically, so they can write SQL syntax in any cell**
* **They can add %language magic command at the start of a cell to force language detection.**
* **They can add %sql at the start of a cell.**

**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 supported magic commands are: %python, %sql, %scala, and %r.

Reference: <https://docs.databricks.com/notebooks/notebooks-code.html>

## Question 53

A junior data engineer uses the built-in Databricks Notebooks versioning for source control. A senior data engineer recommended using Databricks Repos instead.

Which of the following could explain why Databricks Repos is recommended instead of Databricks Notebooks versioning?

* **Databricks Repos supports creating and managing branches for development work.**
* **Databricks Repos automatically tracks the changes and keeps the history.**
* **Databricks Repos allows users to resolve merge conflicts**
* **Databricks Repos allows users to restore previous versions of a notebook**
* **All of these advantages explain why Databricks Repos is recommended instead of Notebooks versioning**

**Explanation**

One advantage of Databricks Repos over the built-in Databricks Notebooks versioning is that Databricks Repos supports creating and managing branches for development work.

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

## Question 54

Which of the following services provides a data warehousing experience to its users?

* **Databricks SQL**
* **Databricks Machine Learning**
* **Data Science and Engineering Workspace**
* **Unity Catalog**
* **Delta Lives Tables (DLT)**

**Explanation**

Databricks SQL (DB SQL) is a data warehouse on the Databricks Lakehouse Platform that lets you run all your SQL and BI applications at scale.

Reference: <https://www.databricks.com/product/databricks-sql>

## Question 55:

A data engineer noticed that there are unused data files in the directory of a Delta table. They executed the VACUUM command on this table; however, only some of those unused data files have been deleted.

Which of the following could explain why only some of the unused data files have been deleted after running the VACUUM command ?

* **The deleted data files were larger than the default size threshold. While the remaining files are smaller than the default size threshold and can not be deleted.**
* **The deleted data files were smaller than the default size threshold. While the remaining files are larger than the default size threshold and can not be deleted.**
* **The deleted data files were older than the default retention threshold. While the remaining files are newer than the default retention threshold and can not be deleted.**
* **The deleted data files were newer than the default retention threshold. While the remaining files are older than the default retention threshold and can not be deleted.**
* **More information is needed to determine the correct answer**

**Explanation**

Running the VACUUM command on a Delta table deletes the unused data files older than a specified data retention period. Unused files newer than the default retention threshold are kept untouched.

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

## Question 56

The data engineering team has a Delta table called **products** that contains products’ details including the net price.

Which of the following code blocks will apply a 50% discount on all the products where the price is greater than 1000 and save the new price to the table?

* **UPDATE products SET price = price \* 0.5 WHERE price >= 1000;**
* **SELECT price \* 0.5 AS new\_price FROM products WHERE price > 1000;**
* **MERGE INTO products WHERE price < 1000 WHEN MATCHED UPDATE price = price \* 0.5;**
* **UPDATE products SET price = price \* 0.5 WHERE price > 1000;**
* **MERGE INTO products WHERE price > 1000 WHEN MATCHED UPDATE price = price \* 0.5;**

**Explanation**

The UPDATE statement is used to modify the existing records in a table that match the WHERE condition. In this case, we are updating the products where the price is strictly greater than 1000.

Syntax:

1. UPDATE table\_name
2. SET column\_name = expr
3. WHERE condition

Reference:

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

## Question 57

A data engineer wants to create a relational object by pulling data from two tables. The relational object will only be used in the current session. In order to save on storage costs, the date engineer wants to avoid copying and storing physical data.

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

* **External table**
* **Temporary view**
* **Managed table**
* **Global Temporary view**
* **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 Temporary view since it’s tied to a Spark session and dropped when the session ends.

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

## Question 58

A data engineer has a database named **db\_hr**, and they want to know where this database was created in the underlying storage.

Which of the following commands can the data engineer use to complete this task?

* **DESCRIBE db\_hr**
* **DESCRIBE EXTENDED db\_hr**
* **DESCRIBE DATABASE db\_hr**
* **SELECT location FROM db\_hr.db**
* **There is no need for a command since all databases are created under the default hive metastore directory**

**Explanation**

The DESCRIBE DATABASE or DESCRIBE SCHEMA returns the metadata of an existing database (schema). The metadata information includes the database’s name, comment, and location on the filesystem. If the optional EXTENDED option is specified, database properties are also returned.

Syntax:

DESCRIBE DATABASE [ EXTENDED ] database\_name

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

## Question 59

Which of the following commands a data engineer can use to register the table **orders** from an existing SQLite database ?

* CREATE TABLE orders
  1. USING sqlite
  2. OPTIONS (
  3. url "jdbc:sqlite:/bookstore.db",
  4. dbtable "orders"
  5. )
* CREATE TABLE orders

  USING org.apache.spark.sql.jdbc

  OPTIONS (

    url "jdbc:sqlite:/bookstore.db",

    dbtable "orders"

  )

* CREATE TABLE orders

  USING cloudfiles

  OPTIONS (

    url "jdbc:sqlite:/bookstore.db",

    dbtable "orders"

  )

* CREATE TABLE orders
  1. USING EXTERNAL
  2. OPTIONS (
  3. url "jdbc:sqlite:/bookstore.db",
  4. dbtable "orders"
  5. )
* CREATE TABLE orders

USING DATABASE

OPTIONS (

url "jdbc:sqlite:/bookstore.db",

dbtable "orders"

)

**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>

## Question 60:

When dropping a Delta table, which of the following explains why both the table's metadata and the data files will be deleted ?

* **The table is shallow cloned**
* **The table is external**
* **The user running the command has the necessary permissions to delete the data files**
* **The table is managed**
* **The data files are older than the default retention period**

**Explanation**

Managed tables are tables whose metadata and the data are managed by Databricks.

When you run DROP TABLE on a managed table, both the metadata and the underlying data files are deleted.

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

## Question 61

Given the following commands:

CREATE DATABASE db\_hr;

USE db\_hr;

CREATE TABLE employees;

In which of the following locations will the employees table be located?

* **dbfs:/user/hive/warehouse**
* **dbfs:/user/hive/warehouse/db\_hr.db**
* **dbfs:/user/hive/warehouse/db\_hr**
* **dbfs:/user/hive/databases/db\_hr.db**
* **More information is needed to determine the correct answer**

**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**. The database folder have the extension (.db)

And since we are creating the table also without specifying a LOCATION clause, the table becomes a managed table created under the database directory (in **db\_hr.db** folder)

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

## Question 62

## 

Which of the following code blocks can a data engineer use to create a Python function to multiply two integers and return the result?

* def multiply\_numbers(num1, num2):

print(num1 \* num2)

* def fun: multiply\_numbers(num1, num2):

return num1 \* num2

* def multiply\_numbers(num1, num2):

return num1 \* num2

* fun multiply\_numbers(num1, num2):

return num1 \* num2

* fun def multiply\_numbers(num1, num2):
  1. return num1 \* num2

**Explanation**

In Python, a function is defined using the def keyword. Here, we used the return keyword since the question clearly asks to return the result, and not printing the output.

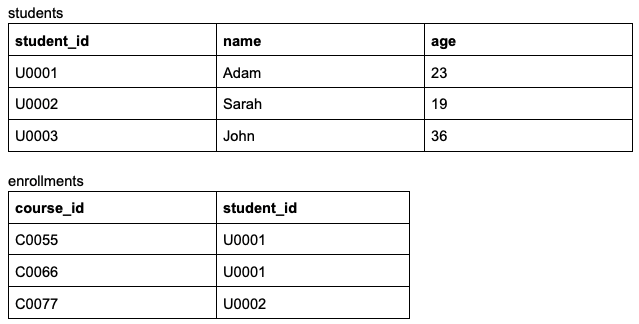
Syntax:

1. def function\_name(params):
2. return params

Reference: <https://www.w3schools.com/python/python_functions.asp>

## Question 63

Given the following 2 tables:

****

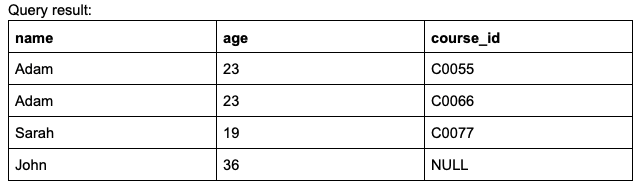
Fill in the blank to make the following query returns the below result:

SELECT students.name, students.age, enrollments.course\_id

FROM students

\_\_\_\_\_\_\_\_\_\_\_\_\_ enrollments

ON students.student\_id = enrollments.student\_id

****

* **RIGHT JOIN**
* **LEFT JOIN**
* **INNER JOIN**
* **ANTI JOIN**
* **CROSS JOIN**

**Explanation**

LEFT JOIN returns all values from the left table and the matched values from the right table, or appends NULL if there is no match. In the above example, we see NULL in the course\_id of John (U0003) since he is not enrolled in any course.

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

Bottom of Form

## Question 64

Which of the following SQL keywords can be used to rotate rows of a table by turning row values into multiple columns ?

* **ROTATE**
* **TRANSFORM**
* **PIVOT**
* **GROUP BY**
* **ZORDER BY**

**Explanation**

PIVOT transforms the rows of a table by rotating unique values of a specified column list into separate columns. In other words, It converts a table from a long format to a wide format.

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

## Question 65

Fill in the below blank to get the number of courses incremented by 1 for each student in array column **students**.

SELECT

faculty\_id,

students,

\_\_\_\_\_\_\_\_\_\_\_ AS new\_totals

FROM faculties

* **TRANSFORM (students, total\_courses + 1)**
* **TRANSFORM (students, i -> i.total\_courses + 1)**
* **FILTER (students, total\_courses + 1)**
* **FILTER (students, i -> i.total\_courses + 1)**
* **CASE WHEN students.total\_courses IS NOT NULL THEN students.total\_courses + 1**

**ELSE NULL**

**END**

**Explanation**

transform(input\_array, lambd\_function) is a higher order function that returns an output array from an input array by transforming each element in the array using a given lambda function.

Example:

SELECT transform(array(1, 2, 3), x -> x + 1);

output: [2, 3, 4]

Reference:

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

## Question 66

Fill in the below blank to successfully create a table using data from CSV files located at **/path/input**

CREATE TABLE my\_table

(col1 STRING, col2 STRING)

\_\_\_\_\_\_\_\_\_\_\_\_

OPTIONS (header = "true",

delimiter = ";")

LOCATION = "/path/input"

* **FROM CSV**
* **USING CSV**
* **USING DELTA**
* **AS**
* **AS CSV**

**Explanation**

CREATE TABLE USING allows to specify an external data source type like CSV format, and with any additional options. This creates an external table pointing to files stored in an external location.

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

## Question 67

Which of the following statements best describes the usage of CREATE SCHEMA command ?

* **It’s used to create a table schema (columns names and datatype)**
* **It’s used to create a Hive catalog**
* **It’s used to infer and store schema in “cloudFiles.schemaLocation”**
* **It’s used to create a database**
* **It’s used to merge the schema when writing data into a target table**

**Explanation**

CREATE SCHEMA is an alias for CREATE DATABASE statement. While usage of SCHEMA and DATABASE is interchangeable, SCHEMA is preferred.

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

## Question 68

Which of the following statements is **Not** true about CTAS statements ?

* **CTAS statements automatically infer schema information from query results**
* **CTAS statements support manual schema declaration**
* **CTAS statements stand for CREATE TABLE \_ AS SELECT statement**
* **With CTAS statements, data will be inserted during the table creation**
* **All these statements are Not true about CTAS statements**

**Explanation**

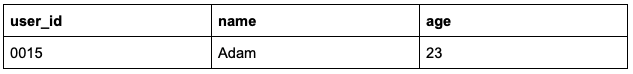
CREATE TABLE AS SELECT statements, or CTAS statements create and populate Delta tables using the output of a SELECT query. CTAS statements automatically infer schema information from query results and do not support manual schema declaration.

Reference: (cf. AS query clause)

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

## Question 69

Which of the following SQL commands will append this new row to the existing Delta table **users**?

****

* **APPEND INTO users VALUES (“0015”, “Adam”, 23)**
* **INSERT VALUES (“0015”, “Adam”, 23)  INTO users**
* **APPEND VALUES (“0015”, “Adam”, 23) INTO users**
* **INSERT INTO users VALUES (“0015”, “Adam”, 23)**
* **UPDATE users VALUES (“0015”, “Adam”, 23)**

**Explanation**

INSERT INTO allows inserting new rows into a Delta table. You specify the inserted rows by value expressions or the result of a query.

Reference: <https://docs.databricks.com/sql/language-manual/sql-ref-syntax-dml-insert-into.html>

## Question 70

Given the following Structured Streaming query:

(spark.table("orders")

.withColumn("total\_after\_tax", col("total")+col("tax"))

.writeStream

.option("checkpointLocation", checkpointPath)

.outputMode("append")

.\_\_\_\_\_\_\_\_\_\_\_

.table("new\_orders") )

Fill in the blank to make the query executes multiple micro-batches to process all available data, then stops the trigger.

* **trigger(“micro-batches”)**
* **trigger(once=True)**
* **trigger(processingTime=”0 seconds")**
* **trigger(micro-batches=True)**
* **trigger(availableNow=True)**

**Explanation**

In Spark Structured Streaming, we use trigger(availableNow=True) to run the stream in batch mode where it processes all available data in multiple micro-batches. The trigger will stop on its own once it finishes processing the available data.

Reference: <https://docs.databricks.com/structured-streaming/triggers.html#configuring-incremental-batch-processing>

## Question 71

Which of the following techniques allows Auto Loader to track the ingestion progress and store metadata of the discovered files ?

* **mergeSchema**
* **COPY INTO**
* **Watermarking**
* **Checkpointing**
* **Z-Ordering**

**Explanation**

Auto Loader keeps track of discovered files using checkpointing in the checkpoint location. Checkpointing allows Auto loader to provide exactly-once ingestion guarantees.

Reference: <https://docs.databricks.com/ingestion/auto-loader/index.html#how-does-auto-loader-track-ingestion-progress>

## Question 72

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 cause the pipeline to fail.

* **ON VIOLATION FAIL**
* **ON VIOLATION FAIL UPDATE**
* **ON VIOLATION DROP ROW**
* **ON VIOLATION FAIL PIPELINE**
* **There is no need to add ON VIOLATION clause. By default, records violating the constraint cause the pipeline to fail.**

**Explanation**

With ON VIOLATION FAIL UPDATE, records that violate the expectation will cause the pipeline to fail. When a pipeline fails because of an expectation violation, you must fix the pipeline code to handle the invalid data correctly before re-running the pipeline.

Reference:

<https://learn.microsoft.com/en-us/azure/databricks/workflows/delta-live-tables/delta-live-tables-expectations#--fail-on-invalid-records>

## Question 73

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

* **They maintain data that powers analytics, machine learning, and production applications**
* **They maintain raw data ingested from various sources**
* **The table structure in this layer resembles that of the source system table structure with any additional metadata columns like the load time, and input file name.**
* **They provide business-level aggregated version of data**
* **They provide a more refined view of raw data, where it’s filtered, cleaned, and enriched.**

**Explanation**

Silver tables provide a more refined view of the raw data. For example, data can be cleaned and filtered at this level. And we can also join fields from various bronze tables to enrich our silver records

Reference:

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

## Question 74

The data engineer team has a DLT pipeline that updates all the tables at defined intervals until manually stopped. 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.**
* **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 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>

## Question 75

Given the following Structured Streaming query:

(spark.readStream

.table("cleanedOrders")

.groupBy("productCategory")

.agg(sum("totalWithTax"))

.writeStream

.option("checkpointLocation", checkpointPath)

.outputMode("complete")

.table("aggregatedOrders")

)

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**
* **The query is performing a hop from a Bronze table to a Silver table**
* **The query is performing a hop from Silver layer 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 above Structured Streaming query creates business-level aggregates from clean orders data in the silver table cleanedOrders, and loads them in the gold table aggregatedOrders.

Reference:

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

## Question 76

Given the following Structured Streaming query:

(spark.readStream

.table("orders")

.writeStream

.option("checkpointLocation", checkpointPath)

.table("Output\_Table")

)

Which of the following is the trigger Interval for this query ?

* **Every half second**
* **Every half min**
* **Every half hour**
* **The query will run in batch mode to process all available data at once, then the trigger stops.**
* **More information is needed to determine the correct response**

**Explanation**

By default, if you don’t provide any trigger interval, the data will be processed every half second. This is equivalent to trigger(processingTime=”500ms")

Reference: <https://docs.databricks.com/structured-streaming/triggers.html#what-is-the-default-trigger-interval>

## Question 77

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

CREATE STREAMING LIVE TABLE sales\_silver

AS

SELECT store\_id, total + tax AS total\_after\_tax

FROM LIVE.sales\_bronze

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 ?

* CREATE LIVE TABLE sales\_silver

AS

  SELECT store\_id, total + tax AS total\_after\_tax

  FROM STREAMING(LIVE.sales\_bronze)

* CREATE STREAMING TABLE sales\_silver

AS

  SELECT store\_id, total + tax AS total\_after\_tax

  FROM STREAM(LIVE.sales\_bronze)

* CREATE STREAMING LIVE TABLE sales\_silver

AS

  SELECT store\_id, total + tax AS total\_after\_tax

  FROM STREAMING(sales\_bronze)

* CREATE STREAMING LIVE TABLE sales\_silver

AS

  SELECT store\_id, total + tax AS total\_after\_tax

  FROM STREAMING(LIVE.sales\_bronze)

* CREATE STREAMING LIVE TABLE sales\_silver

AS

  SELECT store\_id, total + tax AS total\_after\_tax

  FROM STREAM(LIVE.sales\_bronze)

**(Correct)**

**Explanation**

In DLT pipelines, You can stream data from other tables in the same pipeline by using the STREAM() function. In this case, you must define a streaming live table using CREATE STREAMING LIVE TABLE syntax.

Remember: to query another live table, prepend always the LIVE. keyword to the table name.

1. CREATE STREAMING LIVE TABLE table\_name
2. AS
3. SELECT \*
4. FROM STREAM(LIVE.another\_table)

Reference: <https://docs.databricks.com/workflows/delta-live-tables/delta-live-tables-incremental-data.html#streaming-from-other-datasets-within-a-pipeline&language-sql>

## Question 78

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

* **They provide a more refined view of the data**
* **They maintain raw data ingested from various sources**
* **The table structure in this layer resembles that of the source system table structure with any additional metadata columns like the load time, and input file name.**
* **They provide business-level aggregations that power analytics, machine learning, and production applications**
* **They represent a filtered, cleaned, and enriched version of data**

**Explanation**

Gold layer is the final layer in the multi-hop architecture, where tables provide business level aggregates often used for reporting and dashboarding, or even for Machine learning.

Reference:

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

## Question 79

The data engineer team has a DLT pipeline that updates all the tables once and then stops. The compute resources of the pipeline terminate when the pipeline is stopped.

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.**
* **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 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>

## Question 80

A data engineer needs to determine whether to use Auto Loader or COPY INTO command in order to load input data files incrementally.

In which of the following scenarios should the data engineer use Auto Loader over COPY INTO command ?

* **If they are going to ingest files in the order of millions or more over time**
* **If they are going to ingest few number of files in the order of thousands**
* **If they are going to load a subset of re-uploaded files**
* **If the data schema is not going to evolve frequently**
* **There is no difference between using Auto Loader and Copy Into command**

**Explanation**

Here are a few things to consider when choosing between Auto Loader and COPY INTO command:

* If you’re going to ingest files in the order of thousands, you can use COPY INTO. If you are expecting files in the order of millions or more over time, use Auto Loader.
* If your data schema is going to evolve frequently, Auto Loader provides better primitives around schema inference and evolution.

Reference: <https://docs.databricks.com/ingestion/index.html#when-to-use-copy-into-and-when-to-use-auto-loader>

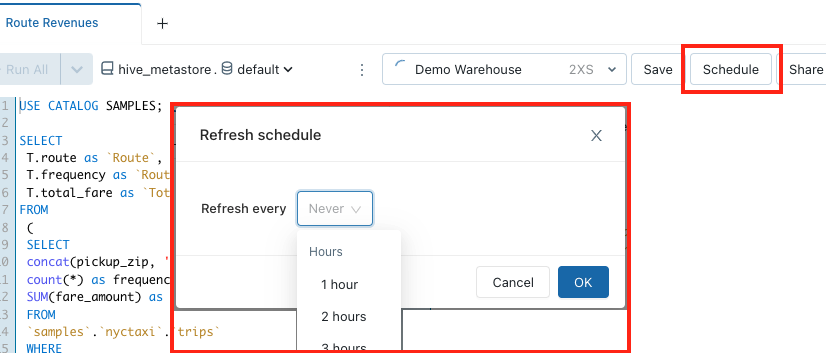
## Question 81

From which of the following locations can a data engineer set a schedule to automatically refresh a Databricks SQL query ?

* **From the jobs Ul**
* **From the SQL warehouses page in Databricks SQL**
* **From the Alerts page in Databricks SQL**
* **From the query's page in Databricks SQL**
* **There is no way to automatically refresh a query in Databricks SQL. Schedules can be set only for dashboards to refresh their underlying queries.**

**Explanation**

In Databricks SQL, you can set a schedule to automatically refresh a query from the query's page.



Reference: <https://docs.databricks.com/sql/user/queries/schedule-query.html>

## Question 82

Databricks provides a declarative ETL framework for building reliable and maintainable data processing pipelines, while maintaining table dependencies and data quality.

Which of the following technologies is being described above?

* **Delta Live Tables**
* **Delta Lake**
* **Databricks Jobs**
* **Unity Catalog Linage**
* **Databricks SQL**

**Explanation**

Delta Live Tables is a framework for building reliable, maintainable, and testable data processing pipelines. You define the transformations to perform on your data, and Delta Live Tables manages task orchestration, cluster management, monitoring, data quality, and error handling.

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

## Question 83

Which of the following services can a data engineer use for orchestration purposes in Databricks platform ?

* **Delta Live Tables**
* **Cluster Pools**
* **Databricks Jobs**
* **Data Explorer**
* **Unity Catalog Linage**

**Explanation**

Databricks Jobs allow to orchestrate data processing tasks. This means the ability to run and manage multiple tasks as a directed acyclic graph (DAG) in a job.

Reference: <https://docs.databricks.com/workflows/jobs/jobs.html>

## Question 84

A data engineer has a Job with multiple tasks that takes more than 2 hours to complete. In the last run, the final task unexpectedly failed.

Which of the following actions can the data engineer perform to complete this Job Run while minimizing the execution time ?

* **They can rerun this Job Run to execute all the tasks**
* **They can repair this Job Run so only the failed tasks will be re-executed**
* **They need to delete the failed Run, and start a new Run for the Job**
* **They can keep the failed Run, and simply start a new Run for the Job**
* **They can run the Job in Production mode which automatically retries execution in case of errors**

**Explanation**

You can repair failed multi-task jobs by running only the subset of unsuccessful tasks and any dependent tasks. Because successful tasks are not re-run, this feature reduces the time and resources required to recover from unsuccessful job runs.

Reference: <https://docs.databricks.com/workflows/jobs/repair-job-failures.html>

## Question 85

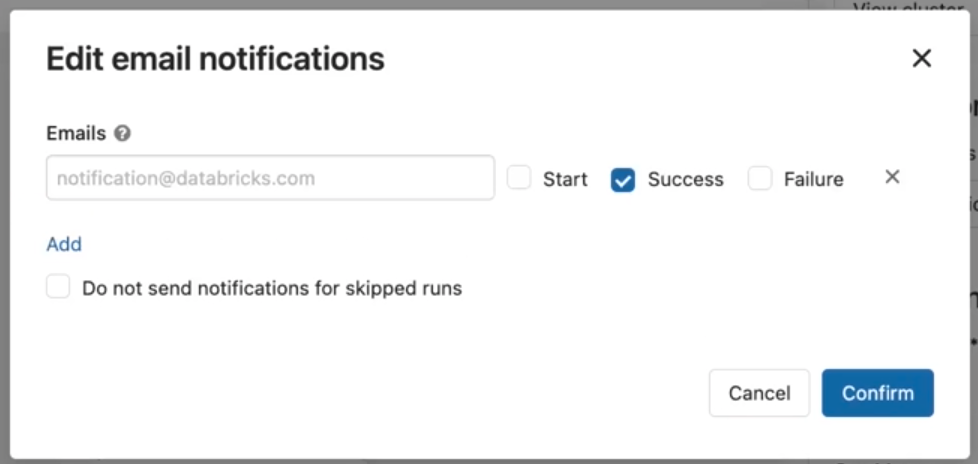
A data engineering team has a multi-tasks Job in production. The team members need to be notified in the case of job failure.

Which of the following approaches can be used to send emails to the team members in the case of job failure ?

* **They can use Job API to programmatically send emails according to each task status**
* **They can configure email notifications settings in the job page**
* **There is no way to notify users in the case of job failure**
* **Only Job owner can be configured to be notified in the case of job failure**
* **They can configure email notifications settings per notebook in the task page**

**Explanation**

Databricks Jobs support 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.



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

## Question 86

For production jobs, which of the following cluster types is recommended to use?

* **All-purpose clusters**
* **Production clusters**
* **Job clusters**
* **On-premises clusters**
* **Serverless clusters**

**Explanation**

Job Clusters are dedicated clusters for a job or task run. A job cluster auto terminates once the job is completed, which saves cost compared to all-purpose clusters.

In addition, Databricks recommends using job clusters in production so that each job runs in a fully isolated environment.

Reference: <https://docs.databricks.com/workflows/jobs/jobs.html#choose-the-correct-cluster-type-for-your-job>

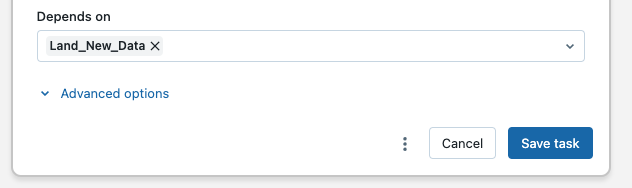
## Question 87

In Databricks Jobs, which of the following approaches can a data engineer use to configure a linear dependency between **Task A** and **Task B** ?

* **They can select the Task A in the Depends On field of the Task B configuration**
* **They can assign Task A an Order number of 1, and assign Task B an Order number of 2**
* **They can visually drag and drop an arrow from Task A to Task B in the Job canvas**
* **They can configure the dependency at the notebook level using the dbutils.jobs utility**
* **Databricks Jobs do not support linear dependency between tasks. This can only be achieved in Delta Live Tables pipelines**

**Explanation**

You can define the order of execution of tasks in a job using the **Depends on** dropdown menu. You can set this field to one or more tasks in the job.



Reference: <https://docs.databricks.com/workflows/jobs/jobs.html#task-dependencies>

## Question 88

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

* **Data Explorer**
* **Cluster event log**
* **Workspace Admin Console**
* **DBFS**
* **There is no way to revoke permissions in Databricks platform. The data engineer needs to clone the table with the updated permissions**

**Explanation**

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

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

## Question 89

A data engineer uses the following SQL query:

GRANT USAGE ON DATABASE sales\_db TO finance\_team

Which of the following is the benefit of the **USAGE**  privilege ?

* **Gives read access on the database**
* **Gives full permissions on the entire database**
* **Gives the ability to view database objects and their metadata**
* No effect! but it's required to perform any action on the database
* **USAGE privilege is not part of the Databricks governance model**

**Explanation**

The USAGE does not give any abilities, but it's an additional requirement to perform any action on a schema (database) object.

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

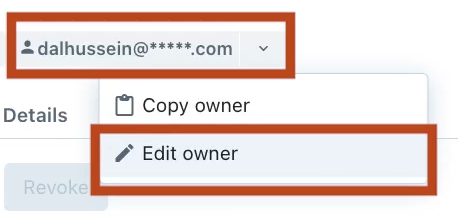
## Question 90

In which of the following locations can a data engineer change the owner of a table?

* **In DBFS, from the properties tab of the table’s data files**
* **In Data Explorer, under the Permissions tab of the table's page**
* **In Data Explorer, from the Owner field in the table's page**
* **In Data Explorer, under the Permissions tab of the database's page, since owners are set at database-level**
* **In Data Explorer, from the Owner field in the database's page, since owners are set at database-level**

**Explanation**

From Data Explorer in Databricks SQL, you can navigate to the table's page to review and change the owner of the table. Simply, click on the Owner field, then **Edit owner** to set the new owner.



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