databricks

Databricks Exam Guide

Databricks Certified Data Analyst Associate

Purpose of this Exam Guide

This exam guide gives you an overview of the Databricks Certified Data Analyst Associate exam and what it covers to help you determine your exam readiness. **This version covers the current version as of Sep 30, 2025.**

Audience Description

The Databricks Certified Data Analyst Associate exam evaluates a candidate's proficiency with the Databricks Data Intelligence Platform, assessing their ability to manage data with Unity Catalog - this includes discovering, querying, cleaning, and managing certified datasets, import Data by utilizing various methods such as the UI, S3 ingestion, Delta Sharing for external systems, API-driven intake, Auto Loader, and the Marketplace feature, and executing and optimizing queries for Data Analysis - including creating views, performing aggregate operations, combining tables with joins, filtering, sorting, and analyzing queries using auditing, history, logs, and Liquid clustering features. Additionally, the exam covers the basics of working with Dashboards and Visualisations, understanding the fundamentals of developing, sharing, and maintaining AI/BI Genie spaces within Databricks, data modelling with Databricks SQL, and securing data by adhering to best practices for data storage and management.

About the Exam

- Number of items: 45 scored multiple-choice questions
- Time Limit: 90 minutes
- Delivery method: Online proctored or test center
- Prerequisite: None is required
- Validity: 2 years
- Recertification: Recertification is required every two years to maintain your certified status. To recertify, you must take the full exam that is currently live.

Recommended Training

- Instructor-led: Data Analysis with Databricks
- Self-paced (available in Databricks Academy): Data Analysis with Databricks. This self-paced course will soon be replaced with the following two modules.
    - AI/BI for Data Analysts
    - SQL Analytics on Databricks

Exam Outline

Section 1: Understanding of Databricks Data Intelligence Platform

- Describe the core components of the Databricks Intelligence Platform, including Mosaic AI, DeltaLive tables, Lakeflow Jobs, Data Intelligence Engine, Delta Lake, Unity Catalog, and Databricks SQL
- Understand catalogs, schemas, managed and external tables, access controls, views, certified tables, and lineage within the Catalog Explorer interface.
- Describe the role and features of Databricks Marketplace

Section 2: Managing Data

- Use Unity Catalog to discover, query, and manage certified datasets
- Use the Catalog Explorer to tag a data asset and view its lineage
- Perform data cleaning on Unity Catalog Tables in SQL, including removing invalid data or handling missing values

Section 3: Importing Data

- Explain the approaches for bringing data into Databricks, covering ingestion from S3, data sharing with external systems via Delta Sharing, API-driven data intake, the Auto Loader feature, and Marketplace.
- Use the Databricks Workspace UI to upload a data file to the platform.

Section 4: Executing queries using Databricks SQL and Databricks SQL Warehouses

- Utilize Databricks Assistant within a Notebook or SQL Editor to facilitate query writing and debugging.
- Explain the role a SQL Warehouse plays in query execution.
- Querying cross-system analytics by joining data from a Delta table and a federated data source.
- Create a materialized view, including knowing when to use Streaming Tables and Materialized Views, and differentiate between dynamic and materialized views
- Perform aggregate operations such as count, approximate count distinct, mean, and summary statistics.
- Write queries to combine tables using various join operations (inner, left, right, and so on) with single or multiple keys, as well as set operations like union and union all, including the differences between the joins (inner, left, right, and so on).
- Perform sorting and filtering operations on a table

- Create managed tables and external tables, including creating tables by joining data from multiple sources (e.g., CSV, Parquet, Delta tables) to create unified datasets, including Unity Catalog
- Use Delta Lake's time travel to access and query historical data versions.

Section 5: Analyzing Queries

- Understand the Features, Benefits, and Supported Workloads of Photon
- Identify poorly performing queries in the Databricks Intelligence platform, such as Query Insights, Query Profiler log, etc.
- Utilize Delta Lake to audit and view history, validate results, and compare historical results or trends.
- Utilize query history and caching to reduce development time and query latency
- Apply Liquid Clustering to improve query speed when filtering large tables on specific columns.
- Fix a query to achieve the desired results.

Section 6: Working with Dashboards and Visualizations in Databricks

- Build dashboards using AI/BI Dashboards, including multi-tabs/page layouts, multiple data sources/datasets, and widgets (visualizations, text, images)
- Create visualizations in notebooks and the SQL editor
- Work with parameters in SQL queries and dashboards, including defining, configuring, and testing parameters
- Configure permissions through the UI to share dashboards with workspace users/groups, external users through shareable links, and embed dashboards in external apps
- Schedule an automatic dashboard refresh.
- Configure an alert with a desired threshold and destination.
- Identify the effective visualization type to communicate insights clearly

Section 7: Developing, Sharing, and Maintaining AI/BI Genie spaces

- Describe the purpose, key features, and components of AI/BI Genie spaces
- Create Genie spaces by defining reasonable sample questions and domain-specific instructions, choosing SQL warehouses, curating Unity Catalog datasets (tables, views…), and vetting queries as Trusted Assets.
- Assign permissions via the UI and distribute Genie spaces using embedded links and external app integrations
- Optimize AI/BI Genie spaces by tracking user questions, response accuracy, and feedback; updating instructions and trusted assets based on stakeholder input; validating accuracy with benchmarks; refreshing Unity Catalog metadata

Section 8: Data Modeling with Databricks SQL

- Apply industry-standard data modeling techniques—such as star, snowflake, and data vault schemas—to analytical workloads.
- Understand how industry-standard models align with the Medallion Architecture.

Section 9: Securing Data

- Use Unity Catalog roles and sharing settings to ensure workspace objects are secure.
- Understand how the 3-level namespace(Catalog / Schema / Tables or Volumes) works in the Unity Catalog
- Apply best practices for storage and management to ensure data security, including table ownership and PII protection.

Sample Questions

Question 1

Objective: Perform aggregate operations such as count, approximate count distinct, mean, and summary statistics.

A data analyst working in a notebook needs to quickly understand the characteristics of a customer dataset. The analyst wants to automatically generate insights such as data distributions, potential quality issues, and summary statistics.

Which insights are automatically provided when using data preview features in a notebook?

A. Row count and column names
B. Summary statistics for numeric, string and date columns, along with histograms showing value distributions for each column.
C. Mean data type information, null counts and standard deviation calculation for numeric columns only
D. Real-time performance metrics and queries execution statistics for the dataset

Question 2

Objective: Utilize Databricks Assistant within a Notebook or SQL Editor to facilitate query writing and debugging.

A data analyst is struggling to understand why a complex SQL query is not returning the expected results.

Which Databricks Assistant command should the analyst use for a step-by-step explanation of the query and potential issues?

A. `/help`
B. `/generate`
C. `/explain`
D. `/optimize`

Question 3

Objective: Create a materialized view, including knowing when to use Streaming Tables and Materialized Views, and differentiate between dynamic and materialized views.

A data analyst is planning a new analytics pipeline and needs to decide between using a Streaming Table and a Materialized View for two different use cases:
Real-time analysis of continuously arriving sensor data.
Frequent, complex queries on static data for business intelligence dashboards.

Which Databricks table type should the analyst use?

A. Use a Materialized View for real-time sensor data, and a Streaming Table for static business intelligence queries.
B. Use a Materialized View for both real-time sensor data and static business intelligence queries to maximize the freshness of data.
C. Use a Streaming Table for both real-time sensor data and static business intelligence queries to simplify pipeline complexity.
D. Use a Streaming Table for real-time analysis of continuously arriving sensor data, and a Materialized View for frequent, complex queries on static data.

Question 4

Objective: Use Delta Lake's time travel to access and query historical data versions.

A data analyst attempts to query a Delta table as it existed 30 days ago, but receives an error.

What is the reason for this error?

A. The table schema was updated after the target date, which prevents time travel to previous versions.
B. The data files and log files needed for that version were deleted by a `VACUUM` operation.
C. The Delta table's permissions were changed, which disables time travel functionality.
D. The table was renamed, which removes access to historical versions.

Question 5

Objective: Configure an alert with a desired threshold and destination.

A data pipeline processes a high volume of incoming sensor readings. To prevent potential equipment damage, the data analyst needs to be notified immediately if the average temperature reported by these sensors within the last 15 minutes exceeds a critical threshold.

Which action should the analyst perform to implement this requirement?

A. Set up a Databricks SQL Alert that runs a query calculating the 15-minute average temperature, triggers when the value exceeds the critical threshold, and sends instant notifications to the desired channels (e.g., email, Slack).
B. Create a Databricks Dashboard with a temperature chart and manually refresh it every 15 minutes to check if the reading exceeds the threshold.
C. Use a Databricks Job to run a notebook periodically and review the output logs manually to detect threshold breaches.
D. Modify Databricks cluster Configuration to automatically scale up resources or restart when temperature metrics are high, with alert notifications every 15 minutes.

Answers

1. B
2. C
3. D
4. B
5. A