



VCE & PDF

Pass4itSure.com

<https://www.pass4itsure.com/databricks-certified-professional-data-engineer>  
2024 Latest pass4itsure DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER PDF and VCE dumps Download

---

# DATABRICKS-CERTIFIED- PR OFESIONAL-DATA-ENGINEER<sup>Q&As</sup>

Databricks Certified Professional Data Engineer Exam

**Pass Databricks DATABRICKS-CERTIFIED-  
PROFESSIONAL-DATA-ENGINEER Exam with 100%  
Guarantee**

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass4itsure.com/databricks-certified-professional-data-engineer.html>

100% Passing Guarantee  
100% Money Back Assurance

Following Questions and Answers are all new published by Databricks  
Official Exam Center



VCE & PDF

Pass4itSure.com

<https://www.pass4itsure.com/databricks-certified-professional-data-engineer>  
2024 Latest pass4itsure DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER PDF and VCE dumps Download

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers





## QUESTION 1

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

user\_id BIGINT, username STRING, user\_utc STRING, user\_region STRING, last\_login BIGINT, auto\_pay BOOLEAN, last\_updated BIGINT

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

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

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

Correct Answer: C

Explanation: This is the correct answer because it efficiently updates the account current table with only the most recent value for each user id. The code filters records in account history using the last updated field and the most recent hour processed, which means it will only process the latest batch of data. It also filters by the max last login by user id, which means it will only keep the most recent record for each user id within that batch. Then, it writes a merge statement to update or insert the most recent value for each user id into account current, which means it will perform an upsert operation based on the user id column. Verified References: [Databricks Certified Data Engineer Professional], under "Delta Lake" section; Databricks Documentation, under "Upsert into a table using merge" section.

## QUESTION 2

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

The workspace administrator at the company is uncertain about where the Databricks workspace used by the contractors should be deployed. Assuming that all data governance considerations are accounted for, which statement accurately informs this decision?

- A. Databricks runs HDFS on cloud volume storage; as such, cloud virtual machines must be deployed in the region



where the data is stored.

B. Databricks workspaces do not rely on any regional infrastructure; as such, the decision should be made based upon what is most convenient for the workspace administrator.

C. Cross-region reads and writes can incur significant costs and latency; whenever possible, compute should be deployed in the same region the data is stored.

D. Databricks leverages user workstations as the driver during interactive development; as such, users should always use a workspace deployed in a region they are physically near.

E. Databricks notebooks send all executable code from the user's browser to virtual machines over the open internet; whenever possible, choosing a workspace region near the end users is the most secure.

Correct Answer: C

Explanation: This is the correct answer because it accurately informs this decision. The decision is about where the Databricks workspace used by the contractors should be deployed. The contractors are based in India, while all the company's data is stored in regional cloud storage in the United States. When choosing a region for deploying a Databricks workspace, one of the important factors to consider is the proximity to the data sources and sinks. Cross-region reads and writes can incur significant costs and latency due to network bandwidth and data transfer fees. Therefore, whenever possible, compute should be deployed in the same region the data is stored to optimize performance and reduce costs. Verified References: [Databricks Certified Data Engineer Professional], under "Databricks Workspace" section; Databricks Documentation, under "Choose a region" section.

### QUESTION 3

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

that appear in the geo\_lookup table.

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

#### Cmd 1

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

#### Cmd 2

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

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



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

Correct Answer: E

Explanation: This is the correct answer because Cmd 1 is written in Python and uses a list comprehension to extract the country names from the geo\_lookup table and store them in a Python variable named countries af. This variable will contain a list of strings, not a PySpark DataFrame or a SQL view. Cmd 2 is written in SQL and tries to create a view named sales af by selecting from the sales table where city is in countries af. However, this command will fail because countries af is not a valid SQL entity and cannot be used in a SQL query. To fix this, a better approach would be to use spark.sql() to execute a SQL query in Python and pass the countries af variable as a parameter. Verified References: [Databricks Certified Data Engineer Professional], under "Language Interoperability" section; Databricks Documentation, under "Mix languages" section.

#### QUESTION 4

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

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

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

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

Correct Answer: C



Explanation: This is the correct answer because the JSON posted to the Databricks REST API endpoint 2.0/jobs/create defines a new job with a name, an existing cluster id, and a notebook task. However, it does not specify any schedule or trigger for the job execution. Therefore, three new jobs with the same name and configuration will be created in the workspace, but none of them will be executed until they are manually triggered or scheduled. Verified References: [Databricks Certified Data Engineer Professional], under "Monitoring and Logging" section; [Databricks Documentation], under "Jobs API - Create" section.

## QUESTION 5

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

- A. Cluster: New Job Cluster; Retries: Unlimited; Maximum Concurrent Runs: Unlimited
- B. Cluster: New Job Cluster; Retries: None; Maximum Concurrent Runs: 1
- C. Cluster: Existing All-Purpose Cluster; Retries: Unlimited; Maximum Concurrent Runs: 1
- D. Cluster: Existing All-Purpose Cluster; Retries: Unlimited; Maximum Concurrent Runs: 1
- E. Cluster: Existing All-Purpose Cluster; Retries: None; Maximum Concurrent Runs: 1

Correct Answer: B

Explanation: This is the best configuration for scheduling Structured Streaming jobs for production, as it automatically recovers from query failures and keeps costs low. A new job cluster is created for each run of the job and terminated when the job completes, which saves costs and avoids resource contention. Retries are not needed for Structured Streaming jobs, as they can automatically recover from failures using checkpointing and write-ahead logs. Maximum concurrent runs should be set to 1 to avoid duplicate output or data loss. Verified References: Databricks Certified Data Engineer Professional, under "Monitoring and Logging" section; Databricks Documentation, under "Schedule streaming jobs" section.

[Latest DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER Dumps](#)

[DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER VCE Dumps](#)

[DATABRICKS-CERTIFIED-PROFESSIONAL-DATA-ENGINEER Braindumps](#)