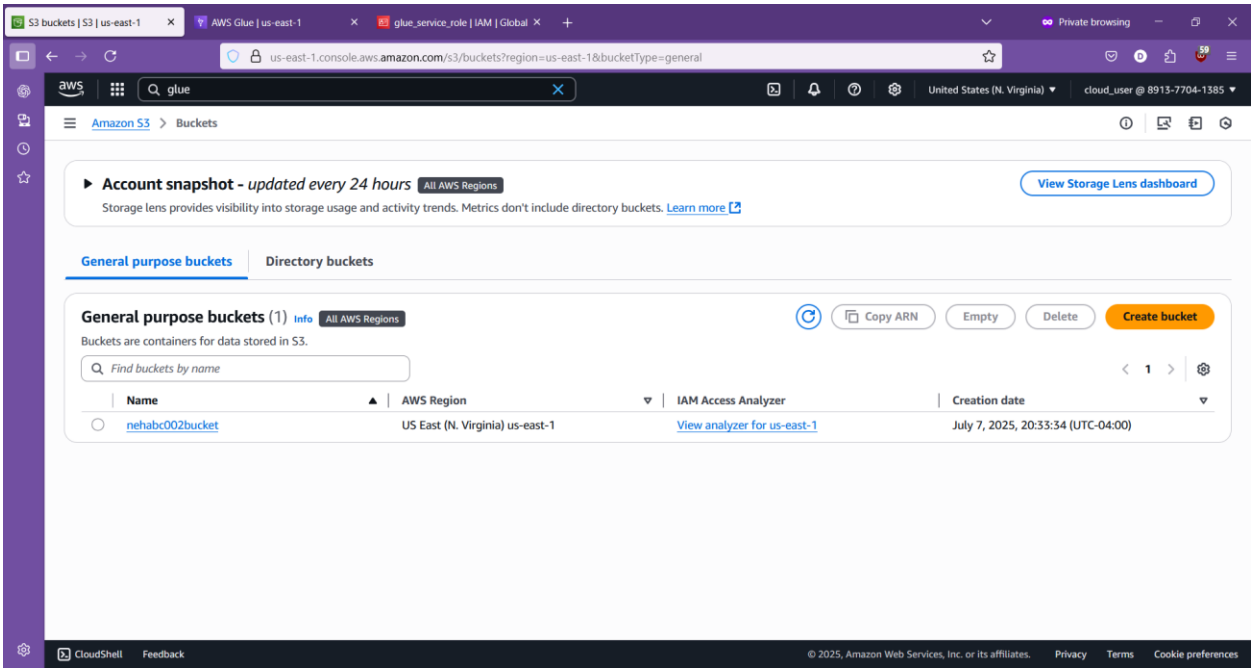
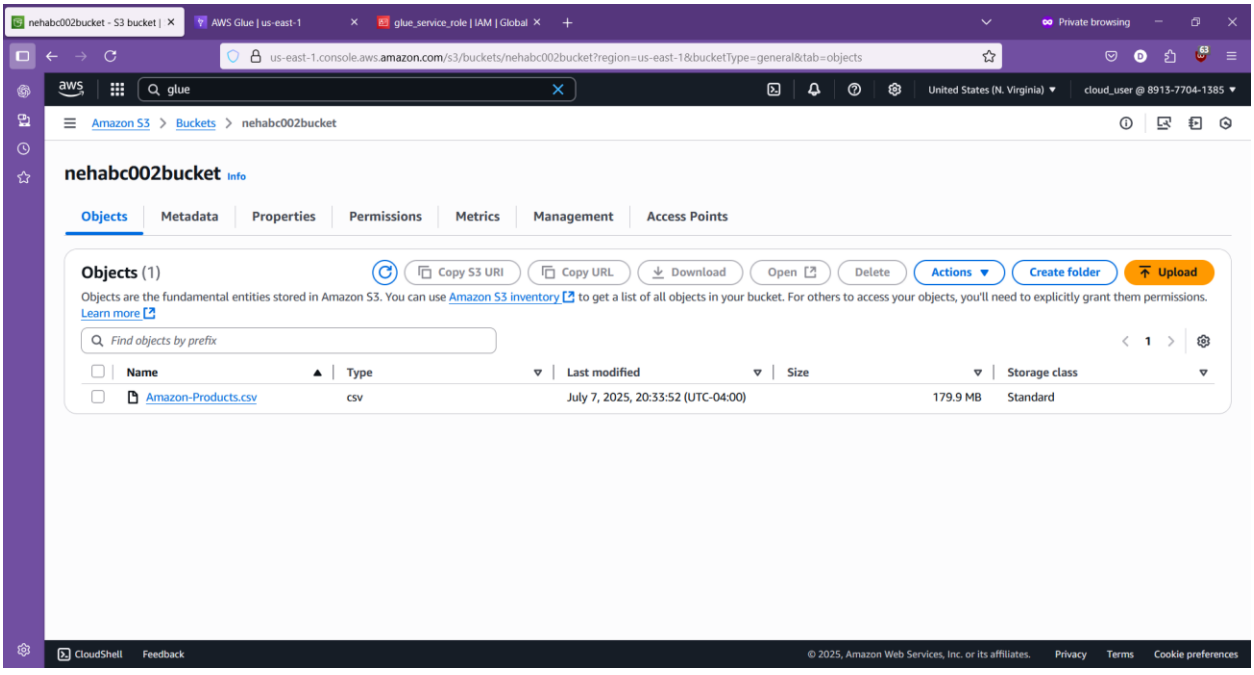
Advanced Product Analytics Platform Using Databricks & AWS

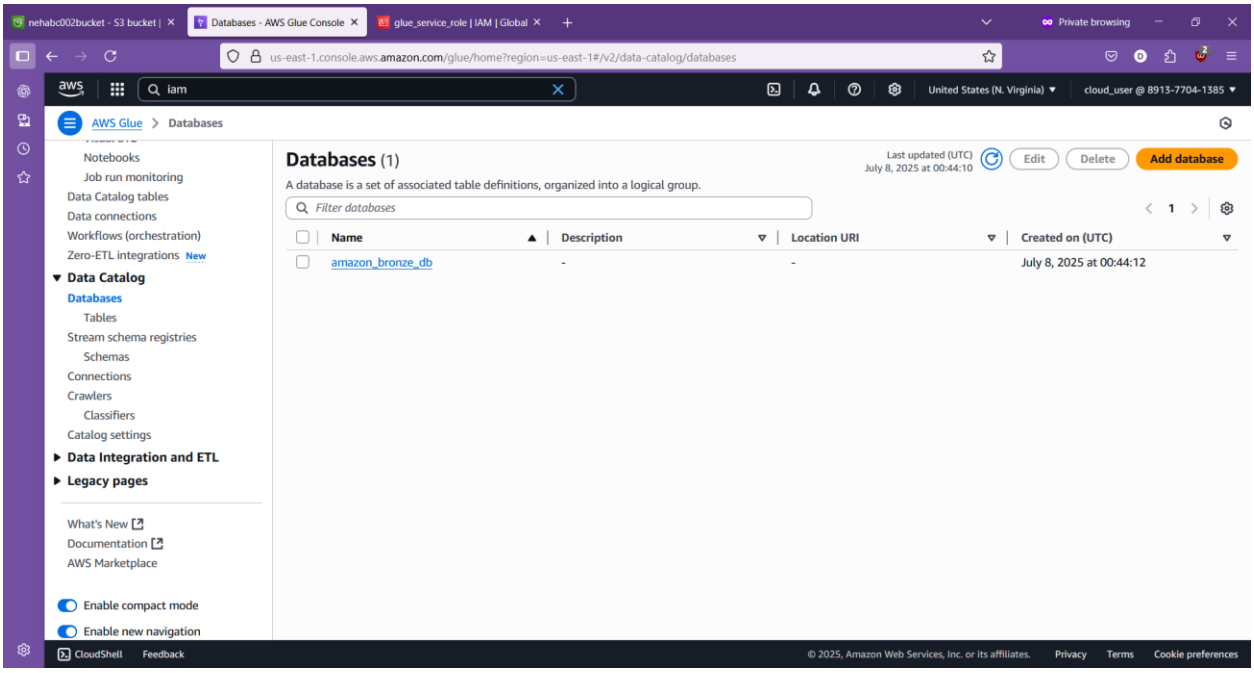
1. Bronze Layer (Raw Ingestion) - S3 Database



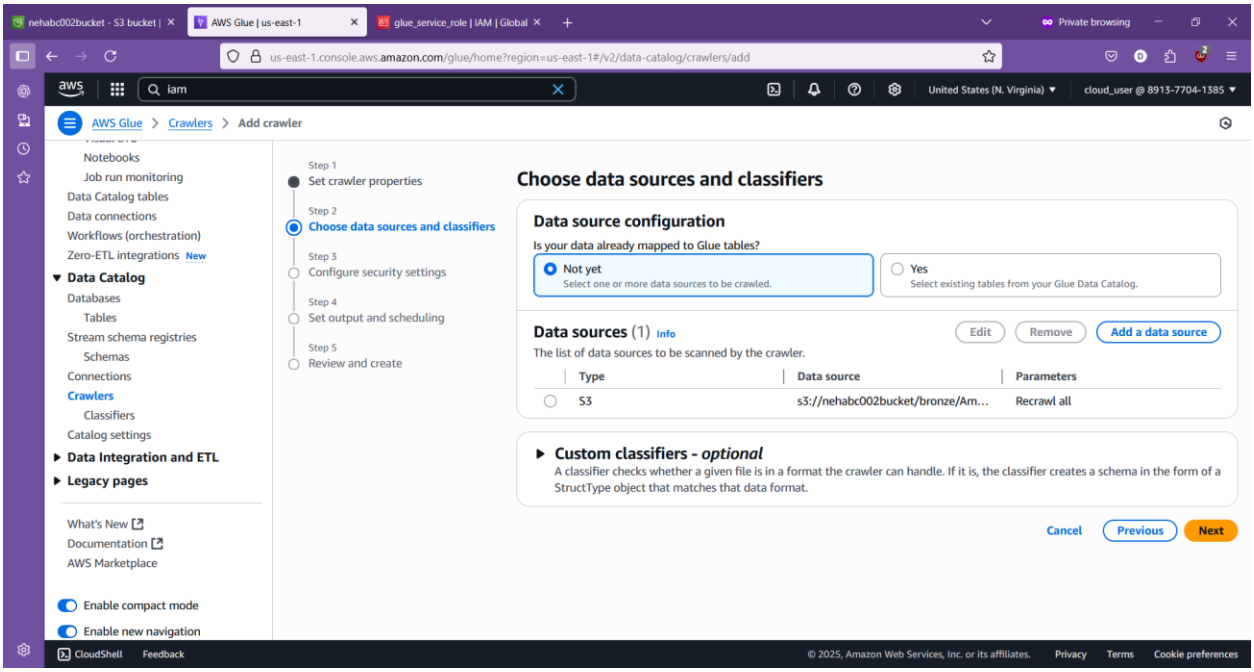


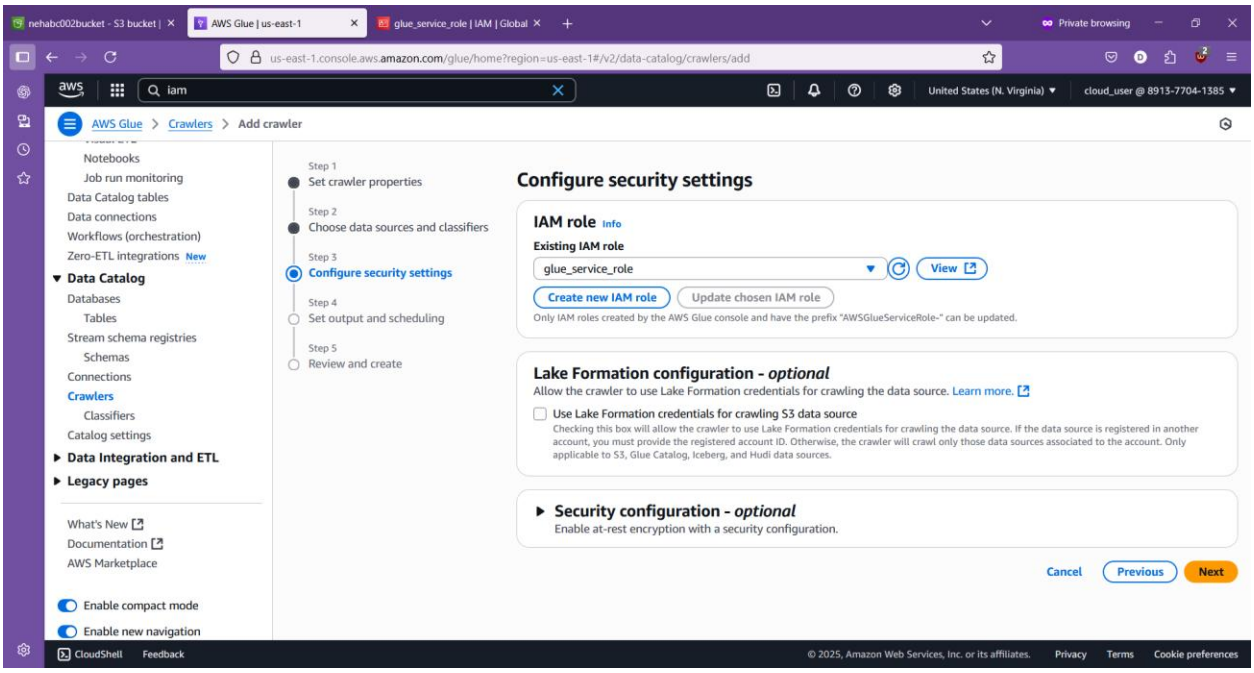
Step 1. Create Glue catalog for schema detection.

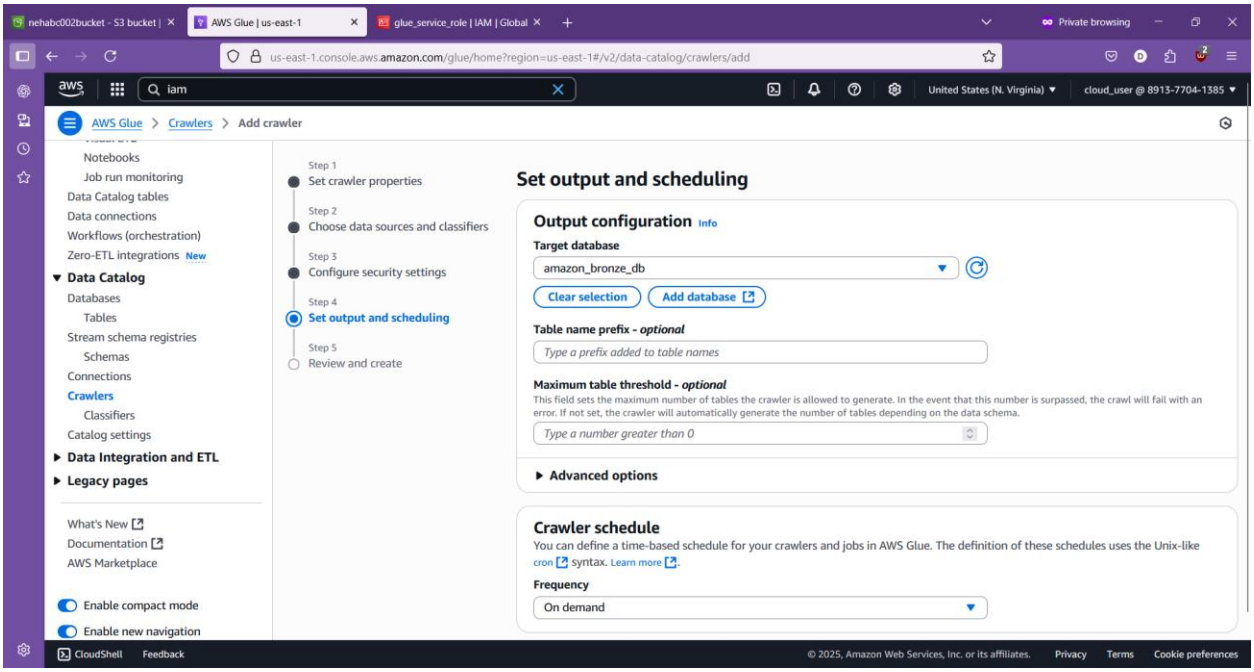
1. Create a Glue Database

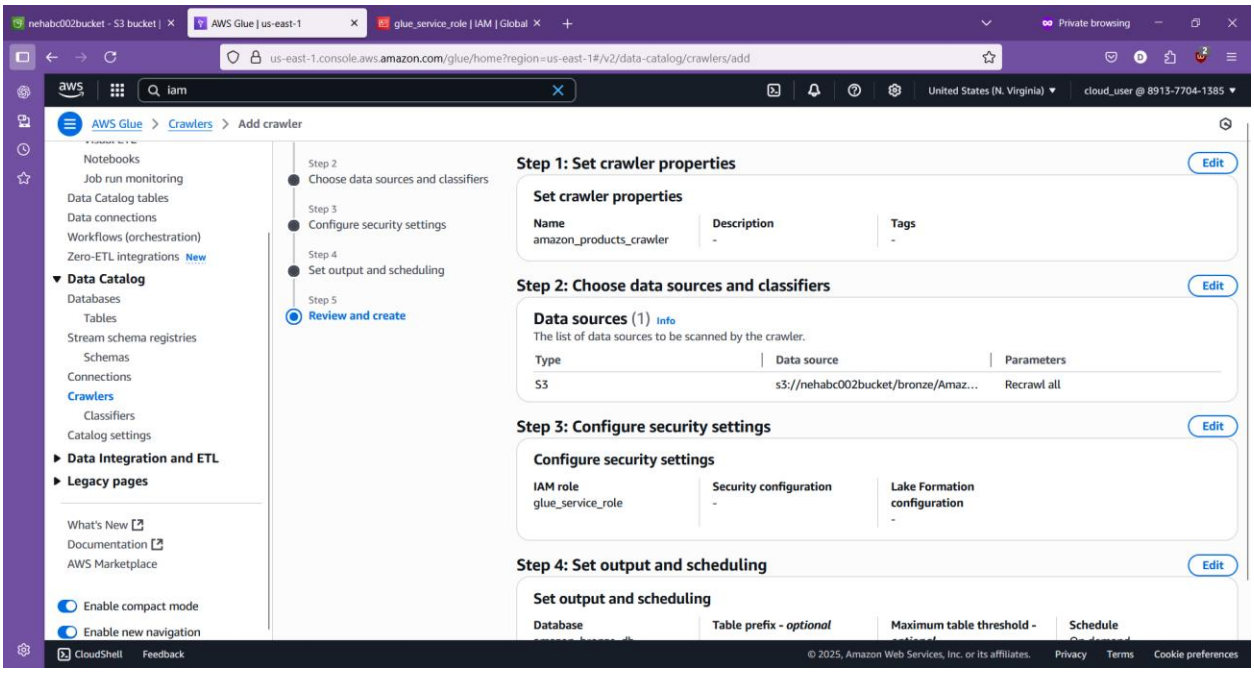


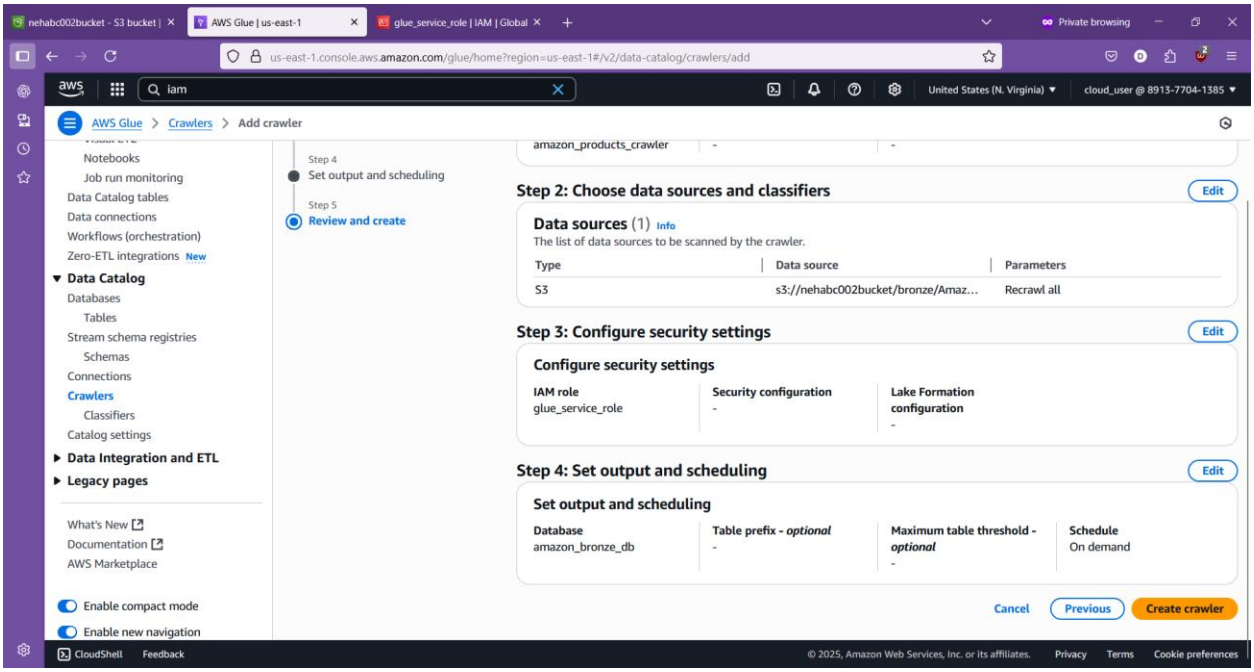
2. Create a Glue Crawler

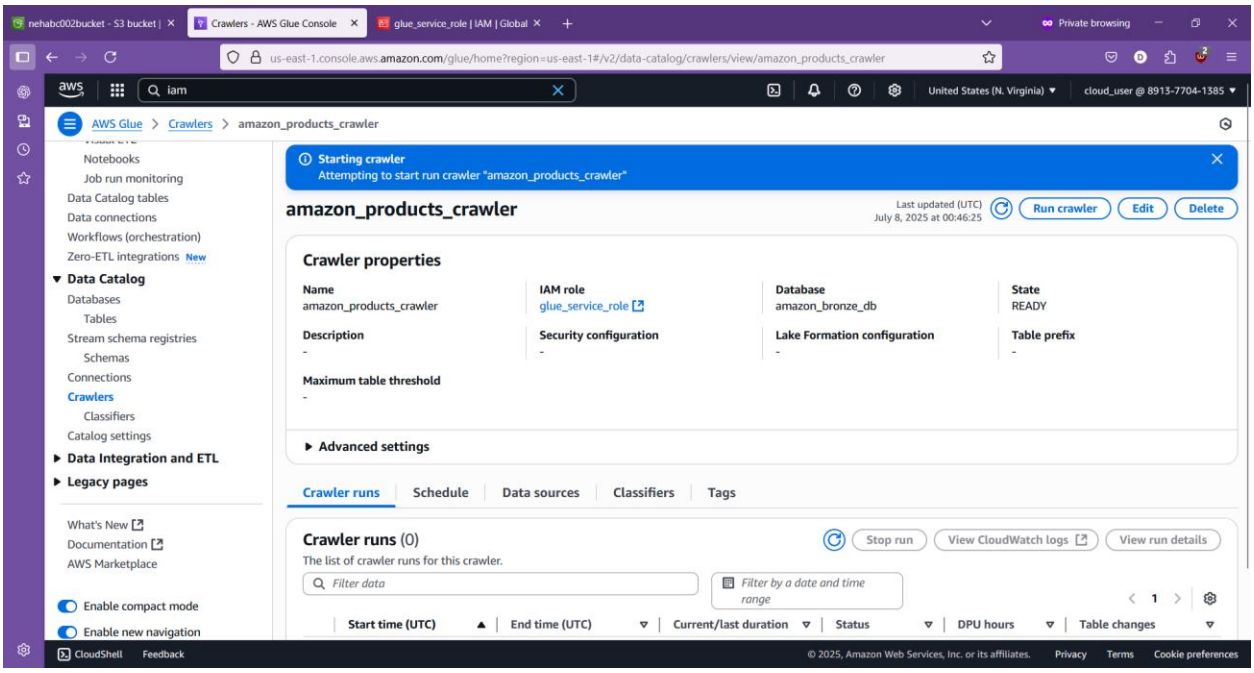


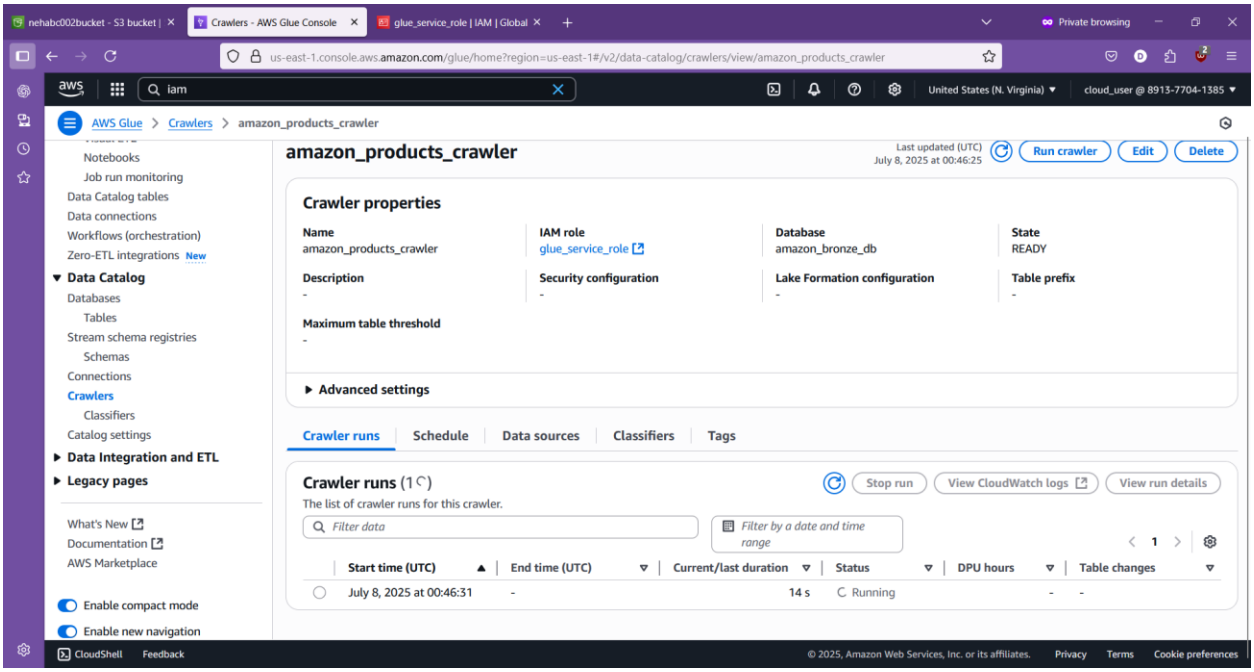

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/add
iam
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > Add crawler
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Step 1
Set crawler properties
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create
Configure security settings
IAM role Info
Existing IAM role
glue_service_role
View
Create new IAM role
Update chosen IAM role
Only IAM roles created by the AWS Glue console and have the prefix "AWSGlueServiceRole-" can be updated.
Lake Formation configuration - optional
Allow the crawler to use Lake Formation credentials for crawling the data source. Learn more.
Use Lake Formation credentials for crawling S3 data source
Checking this box will allow the crawler to use Lake Formation credentials for crawling the data source. If the data source is registered in another account, you must provide the registered account ID. Otherwise, the crawler will crawl only those data sources associated to the account. Only applicable to S3, Glue Catalog, Iceberg, and Hudi data sources.
Security configuration - optional
Enable at-rest encryption with a security configuration.
Cancel
Previous
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

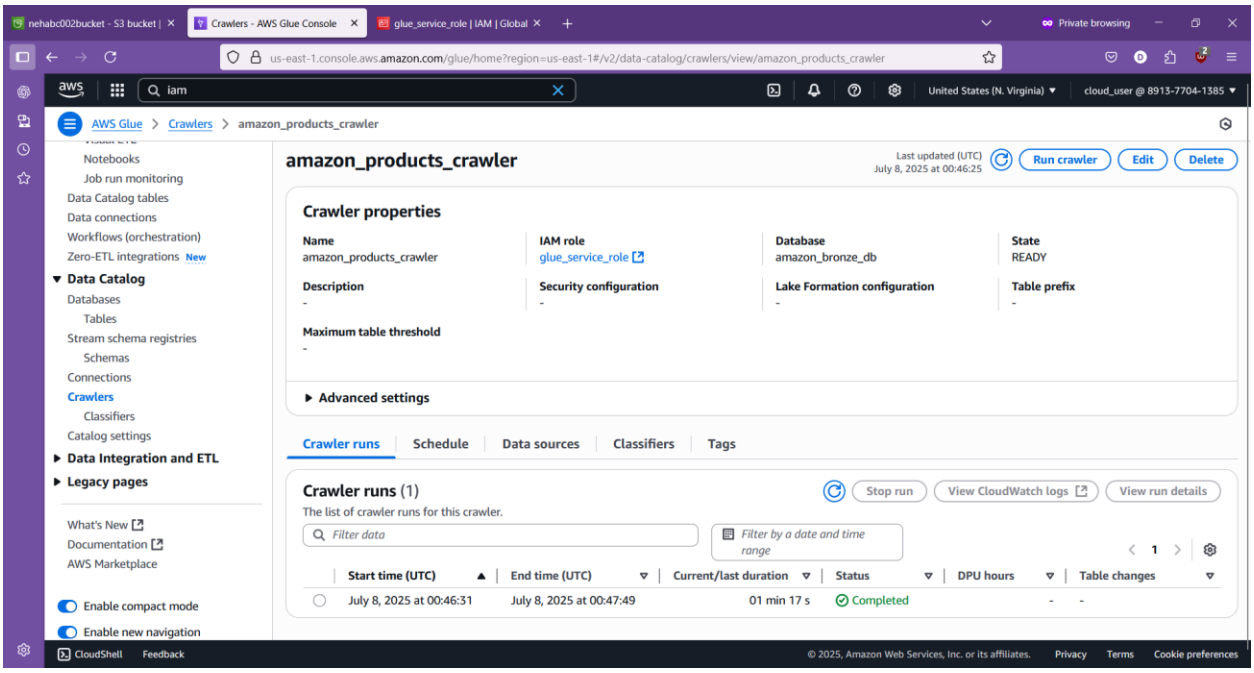

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/add
iam
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > Add crawler
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Step 1
Set crawler properties
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create
Set output and scheduling
Output configuration Info
Target database
amazon_bronze_db
Clear selection
Add database
Table name prefix - optional
Type a prefix added to table names
Maximum table threshold - optional
This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.
Type a number greater than 0
Advanced options
Crawler schedule
You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. Learn more.
Frequency
On demand
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/add
iam
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > Add crawler
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create
Step 1: Set crawler properties
Edit
Set crawler properties
Name
amazon_products_crawler
Description
-
Tags
-
Step 2: Choose data sources and classifiers
Edit
Data sources (1) Info
The list of data sources to be scanned by the crawler.
Type
Data source
Parameters
S3
s3://nehabc002bucket/bronze/Amaz...
Recrawl all
Step 3: Configure security settings
Edit
Configure security settings
IAM role
glue_service_role
Security configuration
-
Lake Formation configuration
-
Step 4: Set output and scheduling
Edit
Set output and scheduling
Database
Table prefix - optional
Maximum table threshold - optional
Schedule
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

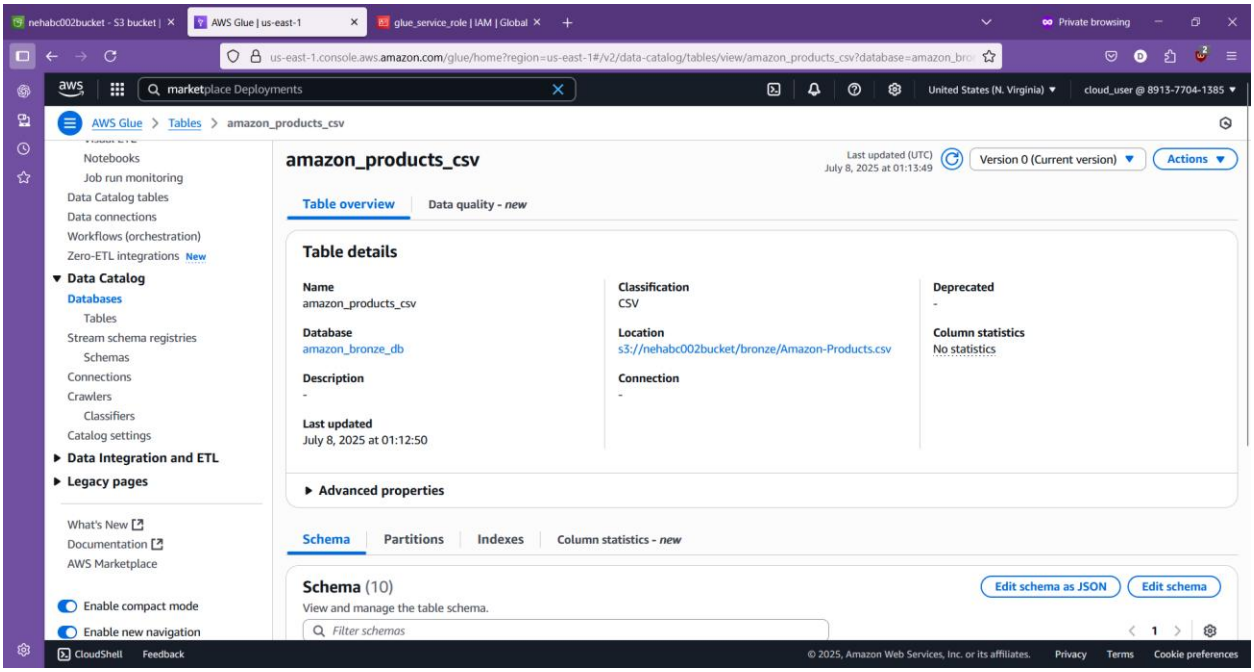

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/add
iam
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > Add crawler
amazon_products_crawler
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Step 4
Set output and scheduling
Step 5
Review and create
Step 2: Choose data sources and classifiers
Edit
Data sources (1) Info
The list of data sources to be scanned by the crawler.
Type
Data source
Parameters
S3
s3://nehabc002bucket/bronze/Amaz...
Recrawl all
Step 3: Configure security settings
Edit
Configure security settings
IAM role
glue_service_role
Security configuration
-
Lake Formation configuration
-
Step 4: Set output and scheduling
Edit
Set output and scheduling
Database
amazon_bronze_db
Table prefix - optional
-
Maximum table threshold - optional
-
Schedule
On demand
Cancel
Previous
Create crawler
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

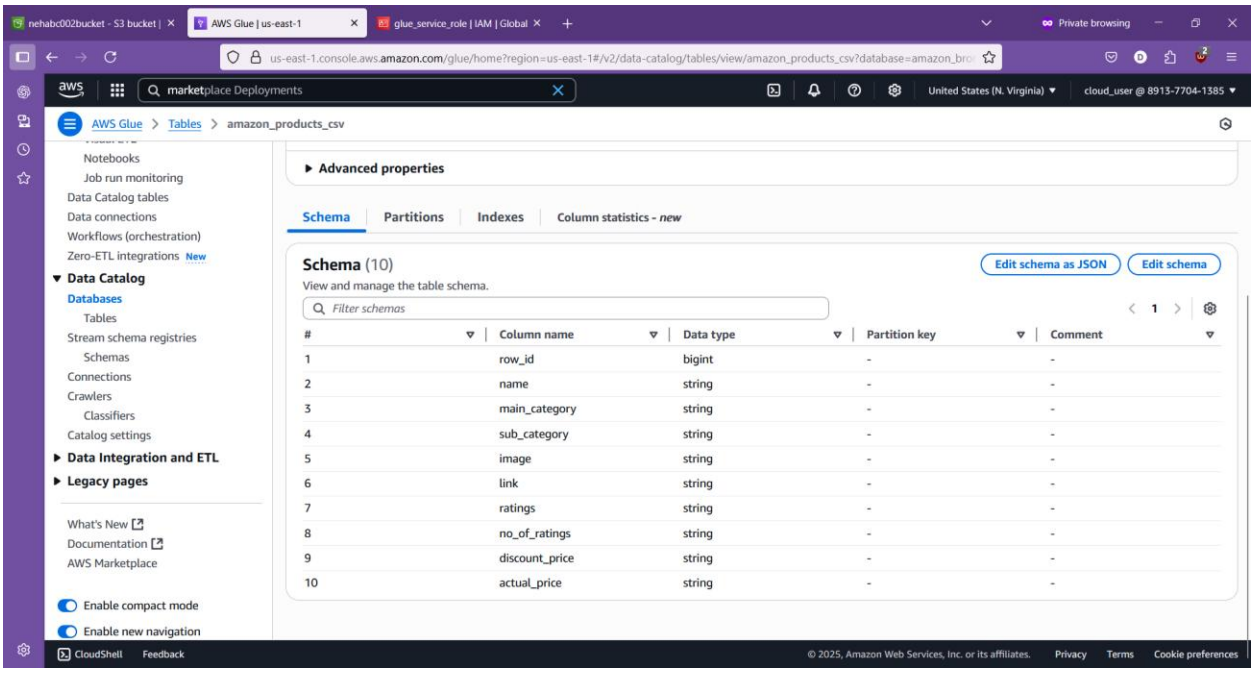

nehabc002bucket - S3 bucket |
Crawlers - AWS Glue Console
glue_service_role | IAM | Global
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/view/amazon_products_crawler
iam
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > amazon_products_crawler
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Starting crawler
Attempting to start run crawler "amazon_products_crawler"
amazon_products_crawler
Last updated (UTC)
July 8, 2025 at 00:46:25
Run crawler
Edit
Delete
Crawler properties
Name
amazon_products_crawler
IAM role
glue_service_role
Database
amazon_bronze_db
State
READY
Description
-
Security configuration
-
Lake Formation configuration
-
Table prefix
-
Maximum table threshold
-
Advanced settings
Crawler runs
Schedule
Data sources
Classifiers
Tags
Crawler runs (0)
The list of crawler runs for this crawler.
Stop run
View CloudWatch logs
View run details
Filter data
Filter by a date and time range
Start time (UTC)
End time (UTC)
Current/last duration
Status
DPU hours
Table changes
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


nehabc002bucket - S3 bucket |
Crawlers - AWS Glue Console
glue_service_role | IAM | Global
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/view/amazon_products_crawler
iam
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > amazon_products_crawler
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
amazon_products_crawler
Last updated (UTC)
July 8, 2025 at 00:46:25
Run crawler
Edit
Delete
Crawler properties
Name
amazon_products_crawler
IAM role
glue_service_role
Database
amazon_bronze_db
State
READY
Description
-
Security configuration
-
Lake Formation configuration
-
Table prefix
-
Maximum table threshold
-
Advanced settings
Crawler runs
Schedule
Data sources
Classifiers
Tags
Crawler runs (1)
The list of crawler runs for this crawler.
Stop run
View CloudWatch logs
View run details
Filter data
Filter by a date and time range
Start time (UTC)
End time (UTC)
Current/last duration
Status
DPU hours
Table changes
July 8, 2025 at 00:46:31
-
14 s
Running
-
-
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


amazon_products_crawler
Crawler properties
Name
amazon_products_crawler
IAM role
glue_service_role
Database
amazon_bronze_db
State
READY
Description
Security configuration
Lake Formation configuration
Table prefix
Maximum table threshold
Advanced settings
Crawler runs
Schedule
Data sources
Classifiers
Tags
Crawler runs (1)
The list of crawler runs for this crawler.
Stop run
View CloudWatch logs
View run details
Start time (UTC)
End time (UTC)
Current/last duration
Status
DPU hours
Table changes
July 8, 2025 at 00:46:31
July 8, 2025 at 00:47:49
01 min 17 s
Completed

3. Table:


amazon_products_csv
Table overview
Data quality - new
Table details
Name
amazon_products_csv
Classification
CSV
Deprecated
Database
amazon_bronze_db
Location
s3://nehabc002bucket/bronze/Amazon-Products.csv
Column statistics
No statistics
Description
Connection
Last updated
July 8, 2025 at 01:12:50
Advanced properties
Schema
Partitions
Indexes
Column statistics - new
Schema (10)
View and manage the table schema.
Edit schema as JSON
Edit schema

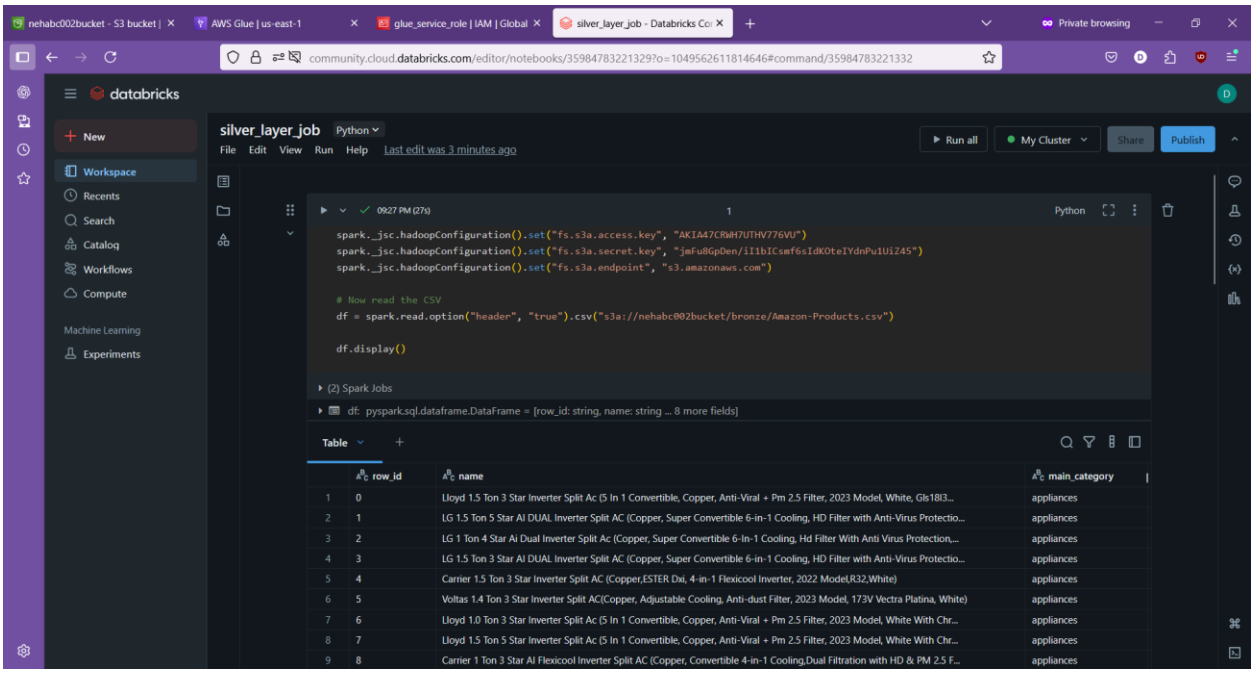


Silver Layer (Data Cleaning & Transformation)

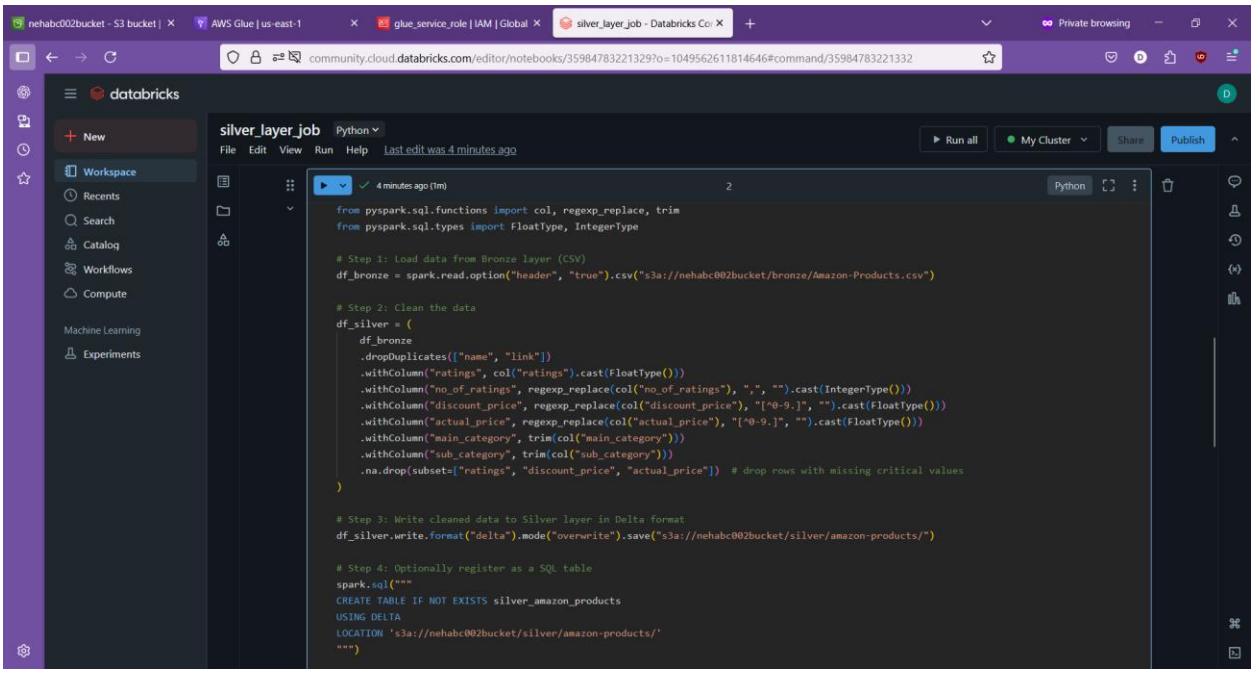
Objectives:

1. Clean and convert price fields (remove ₹, commas, cast to float)
2. Convert ratings and number of ratings to numeric
3. Remove duplicates and handle missing values
4. Trim and standardize category fields

Step 1. Connect the dataset in S3 bucket with Databricks



Step 2. Silver Layer Script in Databricks (PySpark)

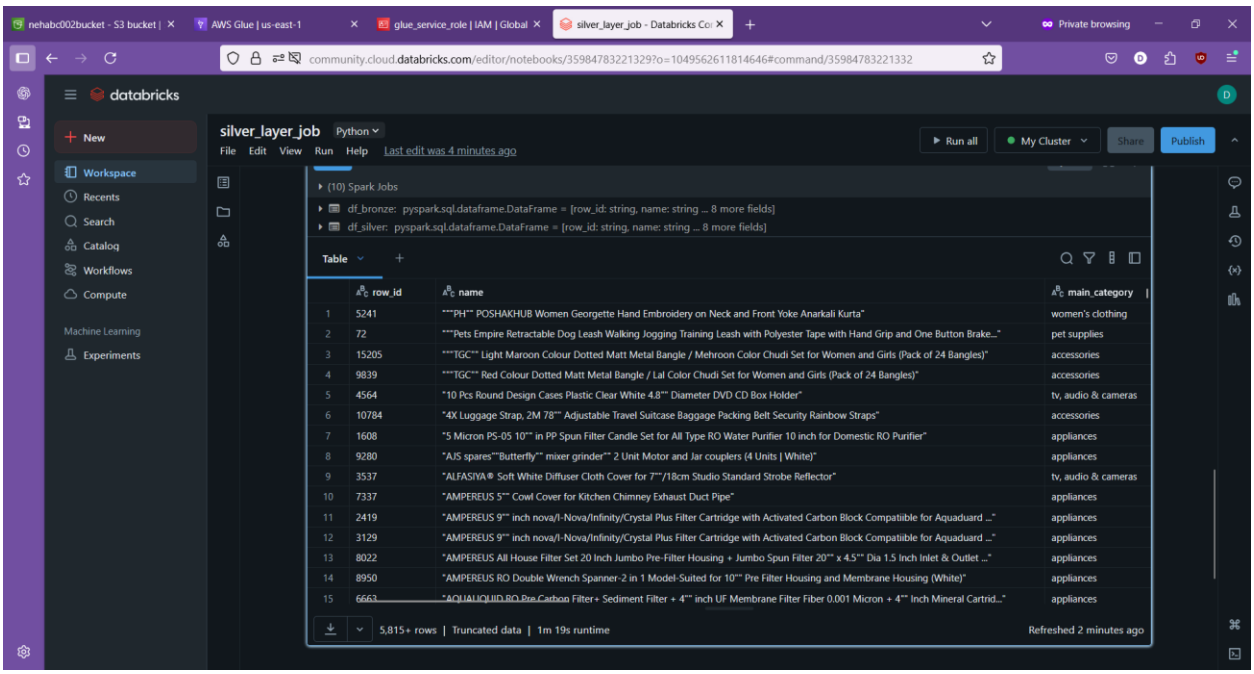


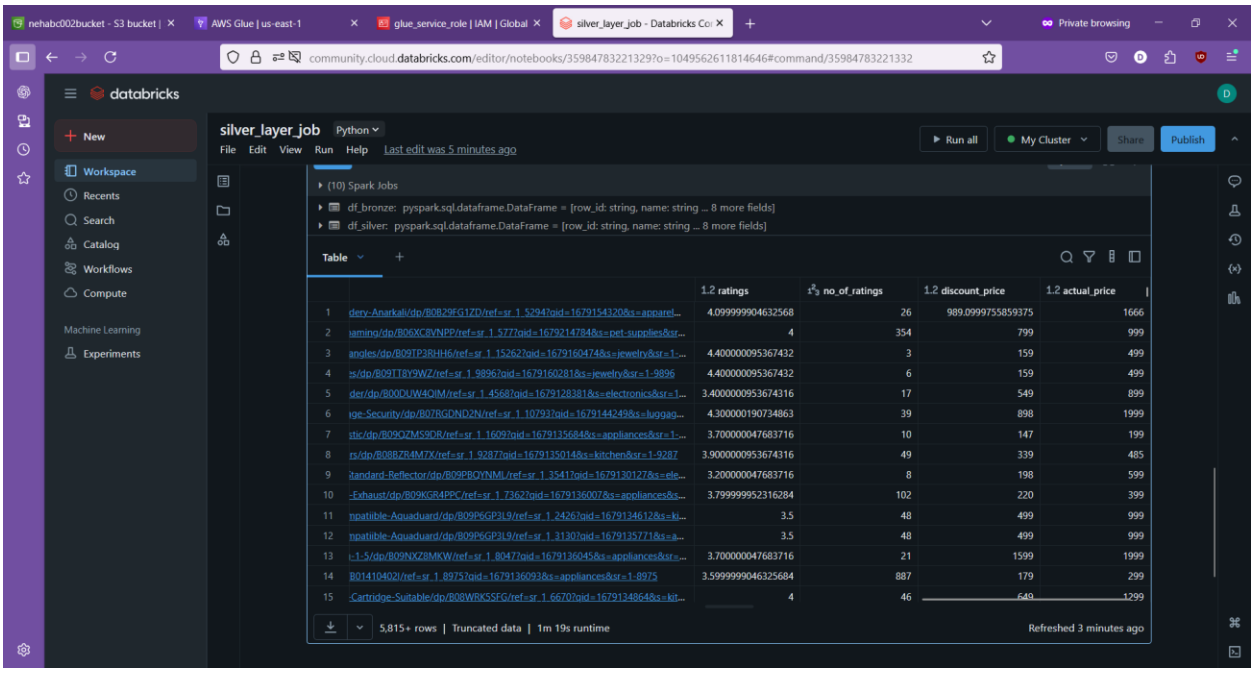
Code:

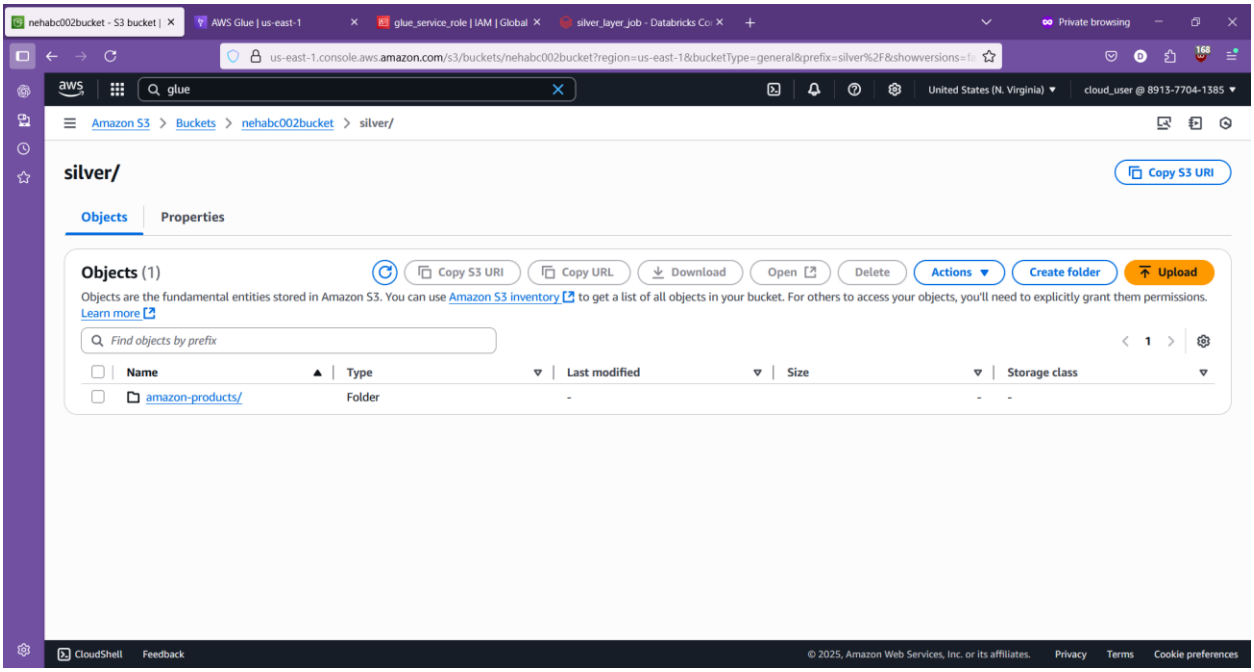
```
from pyspark.sql.functions import col, regexp_replace, trim
from pyspark.sql.types import FloatType, IntegerType

# Step 1: Load data from Bronze layer (CSV)
df_bronze = spark.read.option("header", "true").csv("s3a://nehabc002bucket/bronze/Amazon-Products.csv")

# Step 2: Clean the data
df_silver = (
    df_bronze
    .dropDuplicates(["name", "link"])
    .withColumn("ratings", col("ratings").cast(FloatType()))
    .withColumn("no_of_ratings", regexp_replace(col("no_of_ratings"), ",", "").cast(IntegerType()))
    .withColumn("discount_price", regexp_replace(col("discount_price"), "[^0-9.]", "").cast(FloatType()))
    .withColumn("actual_price", regexp_replace(col("actual_price"), "[^0-9.]", "").cast(FloatType()))
    .withColumn("main_category", trim(col("main_category")))
```

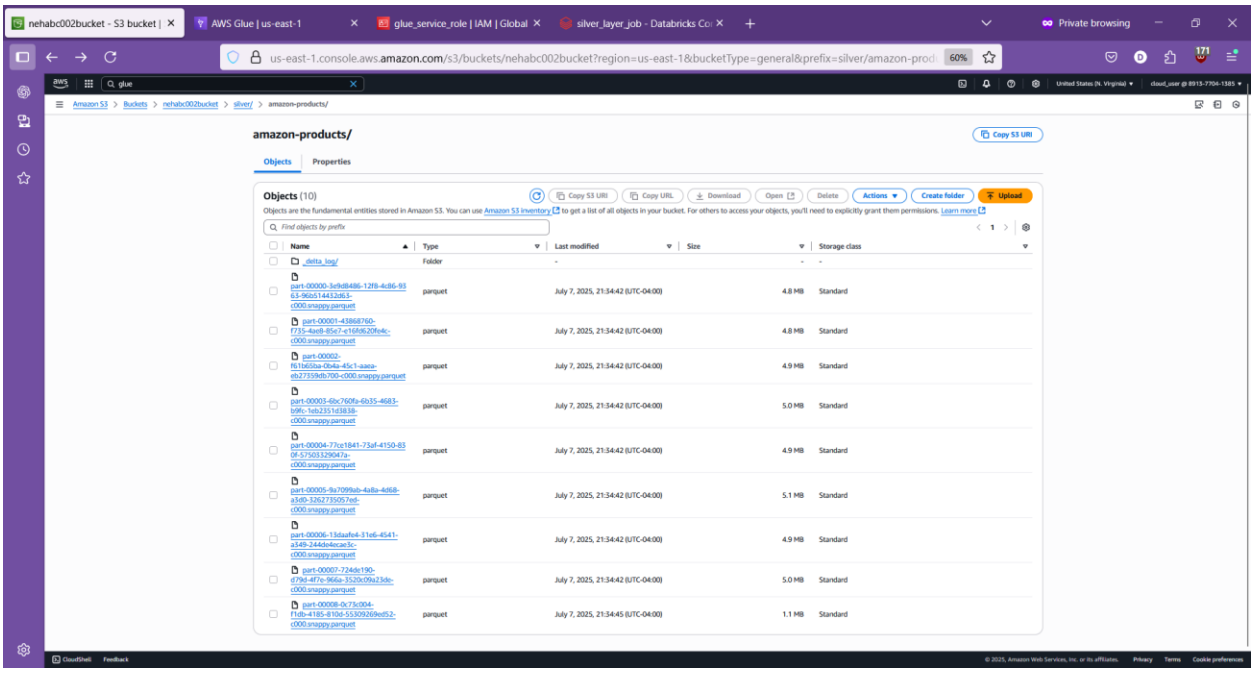
```
    .withColumn("sub_category", trim(col("sub_category")))
    .na.drop(subset=["ratings", "discount_price", "actual_price"])  # drop rows with missing critical values
)

# Step 3: Write cleaned data to Silver layer in Delta format
df_silver.write.format("delta").mode("overwrite").save("s3a://nehabc002bucket/silver/amazon-products/")

# Step 4: Optionally register as a SQL table
spark.sql("""
CREATE TABLE IF NOT EXISTS silver_amazon_products
USING DELTA
LOCATION 's3a://nehabc002bucket/silver/amazon-products/'
""")

# Display sample
df_silver.display()
```

Output:




nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
community.cloud.databricks.com/editor/notebooks/3598478322132 9?o=1049562611814646#command/35984783221332
databricks
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
silver_layer_job
Python
File Edit View Run Help Last edit was 5 minutes ago
Run all
My Cluster
Share
Publish
(10) Spark Jobs
df_bronze: pyspark.sql.dataframe.DataFrame = [row_id: string, name: string ... 8 more fields]
df_silver: pyspark.sql.dataframe.DataFrame = [row_id: string, name: string ... 8 more fields]
Table
1.2 ratings
no_of_ratings
1.2 discount_price
1.2 actual_price
1 dery-Anarkali/dp/B0B29FG1ZD/ref=sr_1_5294?qid=1679154320&s=apparel... 4.099999904632568 26 989.0999755859375 1666
2 aming/dp/B06XC8VNPP/ref=sr_1_577?qid=1679214784&s=pet-supplies&sr... 4 354 799 999
3 angles/dp/B09TP3RHH6/ref=sr_1_15262?qid=1679160474&s=jewelry&sr=1-... 4.400000095367432 3 159 499
4 s/dp/B09TT8Y9WZ/ref=sr_1_9896?qid=1679160281&s=jewelry&sr=1-9896 4.400000095367432 6 159 499
5 der/dp/B00DUW4OIM/ref=sr_1_4568?qid=1679128381&s=electronics&sr=1... 3.4000000953674316 17 549 899
6 ige-Security/dp/B07RGDND2N/ref=sr_1_10793?qid=1679144249&s=luggag... 4.300000190734863 39 898 1999
7 stic/dp/B09QZMS9DR/ref=sr_1_1609?qid=1679135684&s=appliances&sr=1-... 3.700000047683716 10 147 199
8 rs/dp/B088ZR4M7X/ref=sr_1_9287?qid=1679135014&s=kitchen&sr=1-9287 3.9000000953674316 49 339 485
9 itandard-Reflector/dp/B09PBQYNML/ref=sr_1_3541?qid=1679130127&s=ele... 3.200000047683716 8 198 599
10 -Exhaust/dp/B09KGR4PPC/ref=sr_1_7362?qid=1679136007&s=appliances&s... 3.799999952316284 102 220 399
11 npatible-Aquaguard/dp/B09P6GP3L9/ref=sr_1_2426?qid=1679134612&s=ki... 3.5 48 499 999
12 npatible-Aquaguard/dp/B09P6GP3L9/ref=sr_1_3130?qid=1679135771&s=a... 3.5 48 499 999
13 -1-5/dp/B09NXZ8MKW/ref=sr_1_8047?qid=1679136045&s=appliances&sr=... 3.700000047683716 21 1599 1999
14 B01410402I/ref=sr_1_8975?qid=1679136093&s=appliances&sr=1-8975 3.5999999046325684 887 179 299
15 -Cartridge-Suitable/dp/B08WRK5SFG/ref=sr_1_6670?qid=1679134864&s=kit... 4 46 649 1299
5,815+ rows | Truncated data | 1m 19s runtime
Refreshed 3 minutes ago


nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/s3/buckets/nehabc002bucket?region=us-east-1&bucketType=general&prefix=silver%2F&showversions=f
aws
glue
United States (N. Virginia)
cloud_user @ 8913-7704-1385
Amazon S3 > Buckets > nehabc002bucket > silver/
silver/
Copy S3 URI
Objects
Properties
Objects (1)
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions.
Learn more
Find objects by prefix
1
Name
Type
Last modified
Size
Storage class
amazon-products/
Folder
-
-
-
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

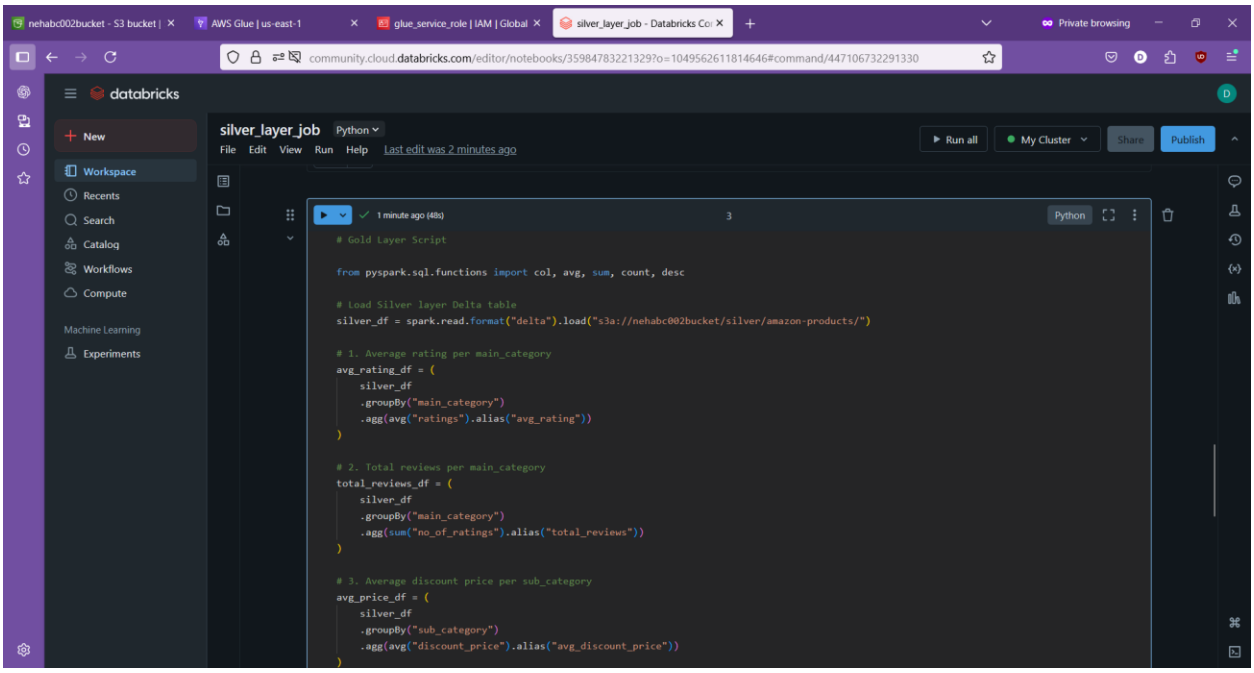


Gold Layer: Business Aggregations with PySpark

Objective:

Metric	Description
Top Categories	Highest-rated product categories
Avg Ratings per Category	Mean rating by main_category
Total Reviews per Category	Sum of no_of_ratings grouped by category
Average Price per Category	Avg of discount_price per sub_category
High-Demand Products	Products with most reviews

Gold Layer Script in Databricks



Code:

```
# Gold Layer Script

from pyspark.sql.functions import col, avg, sum, count, desc

# Load Silver layer Delta table
silver_df = spark.read.format("delta").load("s3a://nehabc002bucket/silver/amazon-products/")

# 1. Average rating per main_category
avg_rating_df = (
    silver_df
    .groupBy("main_category")
    .agg(avg("ratings").alias("avg_rating"))
)

# 2. Total reviews per main_category
total_reviews_df = (
    silver_df
```

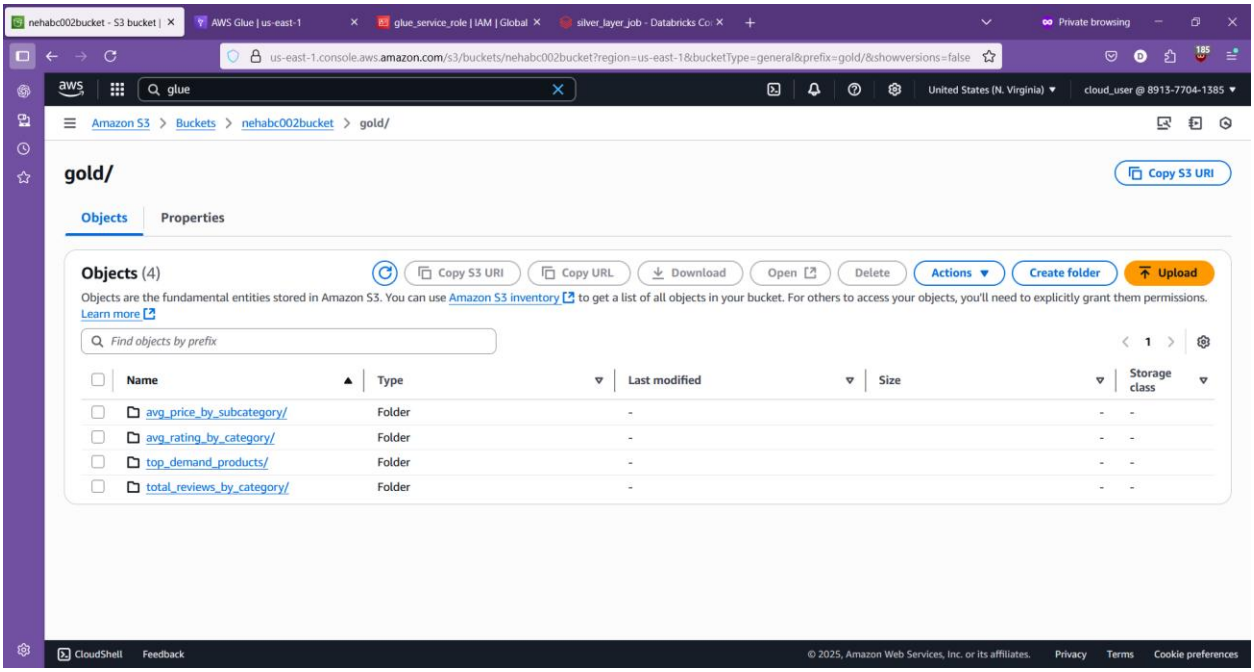
```
    .groupBy("main_category")
    .agg(sum("no_of_ratings").alias("total_reviews"))
)

# 3. Average discount price per sub_category
avg_price_df = (
    silver_df
    .groupBy("sub_category")
    .agg(avg("discount_price").alias("avg_discount_price"))
)

# 4. High-demand products (top 10 by no_of_ratings)
top_products_df = (
    silver_df
    .select("name", "main_category", "sub_category", "ratings", "no_of_ratings")
    .orderBy(desc("no_of_ratings"))
    .limit(10)
)

# Save each result to separate folder inside the Gold layer
avg_rating_df.write.format("delta").mode("overwrite").save("s3a://nehabc002bucket/gold/avg_rating_by_category/")
total_reviews_df.write.format("delta").mode("overwrite").save("s3a://nehabc002bucket/gold/total_reviews_by_category/")
avg_price_df.write.format("delta").mode("overwrite").save("s3a://nehabc002bucket/gold/avg_price_by_subcategory/")
top_products_df.write.format("delta").mode("overwrite").save("s3a://nehabc002bucket/gold/top_demand_products/")

# Optional: Register as SQL Tables
spark.sql("CREATE TABLE IF NOT EXISTS gold_avg_rating_by_category USING DELTA LOCATION 's3a://nehabc002bucket/gold/avg_rating_by_category/'")
```

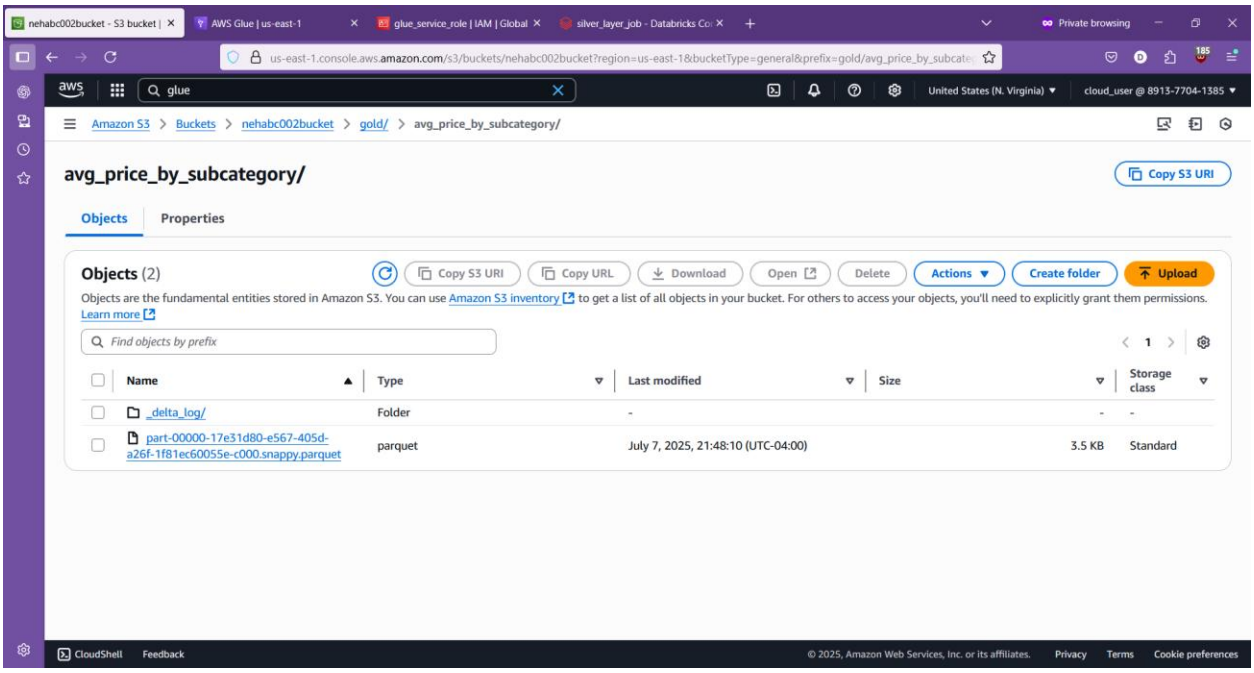
spark.sql("CREATE TABLE IF NOT EXISTS gold_total_reviews_by_category USING DELTA LOCATION 's3a://nehabc002bucket/gold/total_reviews_by_category/'")

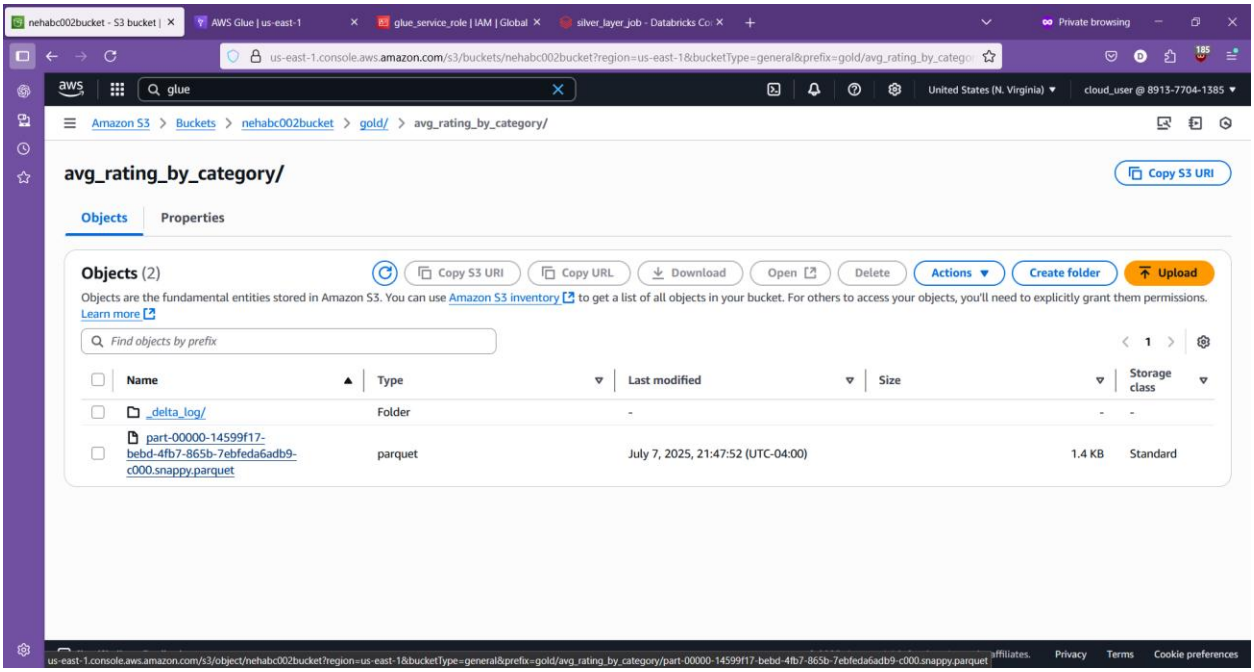
spark.sql("CREATE TABLE IF NOT EXISTS gold_avg_price_by_subcategory USING DELTA LOCATION 's3a://nehabc002bucket/gold/avg_price_by_subcategory/'")

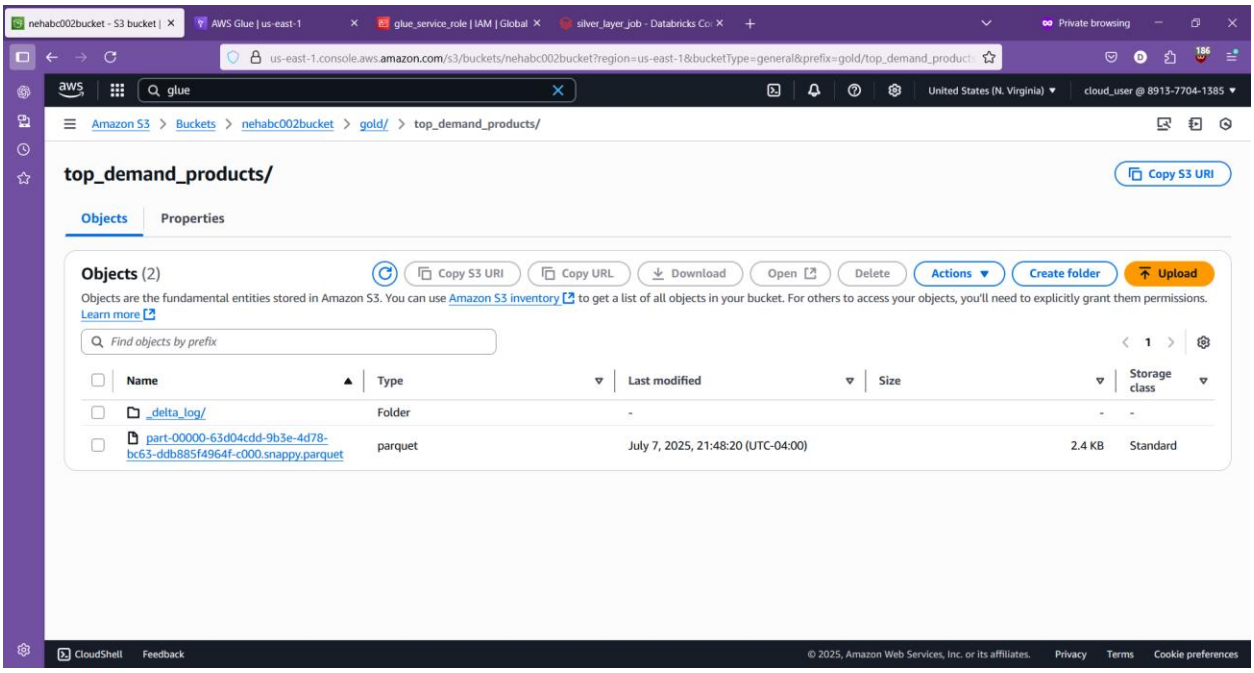
spark.sql("CREATE TABLE IF NOT EXISTS gold_top_demand_products USING DELTA LOCATION 's3a://nehabc002bucket/gold/top_demand_products/'")

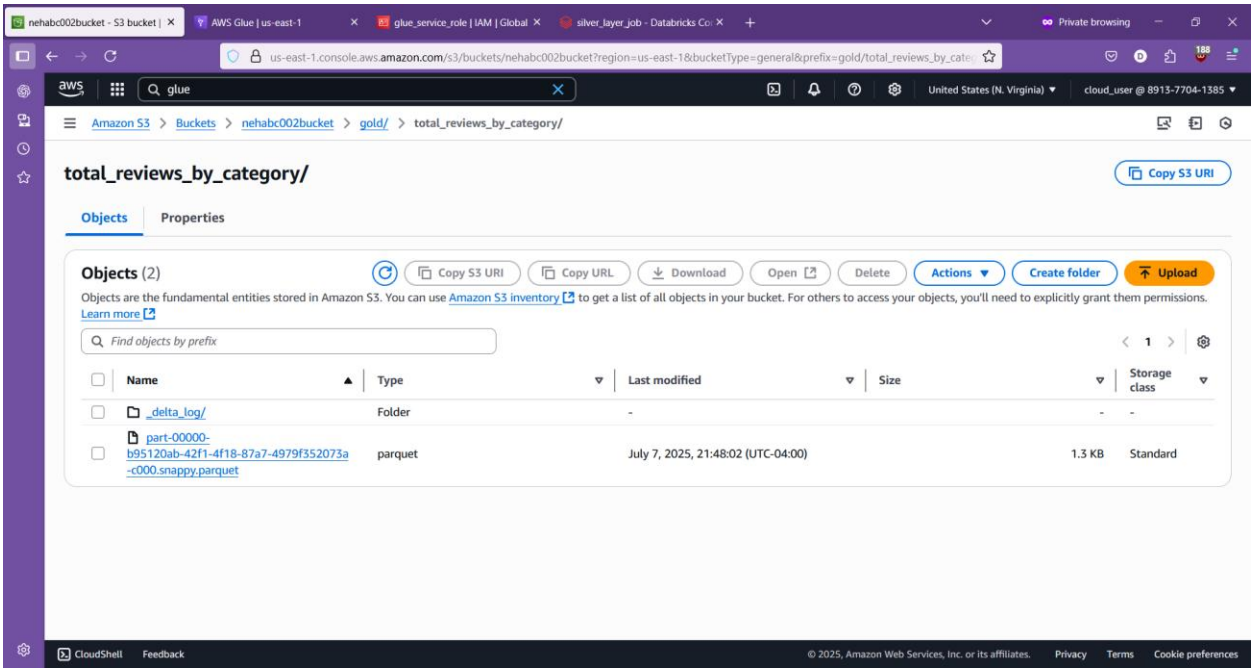
Output Folders in S3:




nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/s3/buckets/nehabc002bucket?region=us-east-1&bucketType=general&prefix=gold/avg_price_by_subcate
glue
United States (N. Virginia)
cloud_user @ 8913-7704-1385
Amazon S3 > Buckets > nehabc002bucket > gold/ > avg_price_by_subcategory/
avg_price_by_subcategory/
Copy S3 URI
Objects
Properties
Objects (2)
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions.
Learn more
Find objects by prefix
Name
Type
Last modified
Size
Storage class
_delta_log/
Folder
part-00000-17e31d80-e567-405d-a26f-1f81ec60055e-c000.snappy.parquet
parquet
July 7, 2025, 21:48:10 (UTC-04:00)
3.5 KB
Standard
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

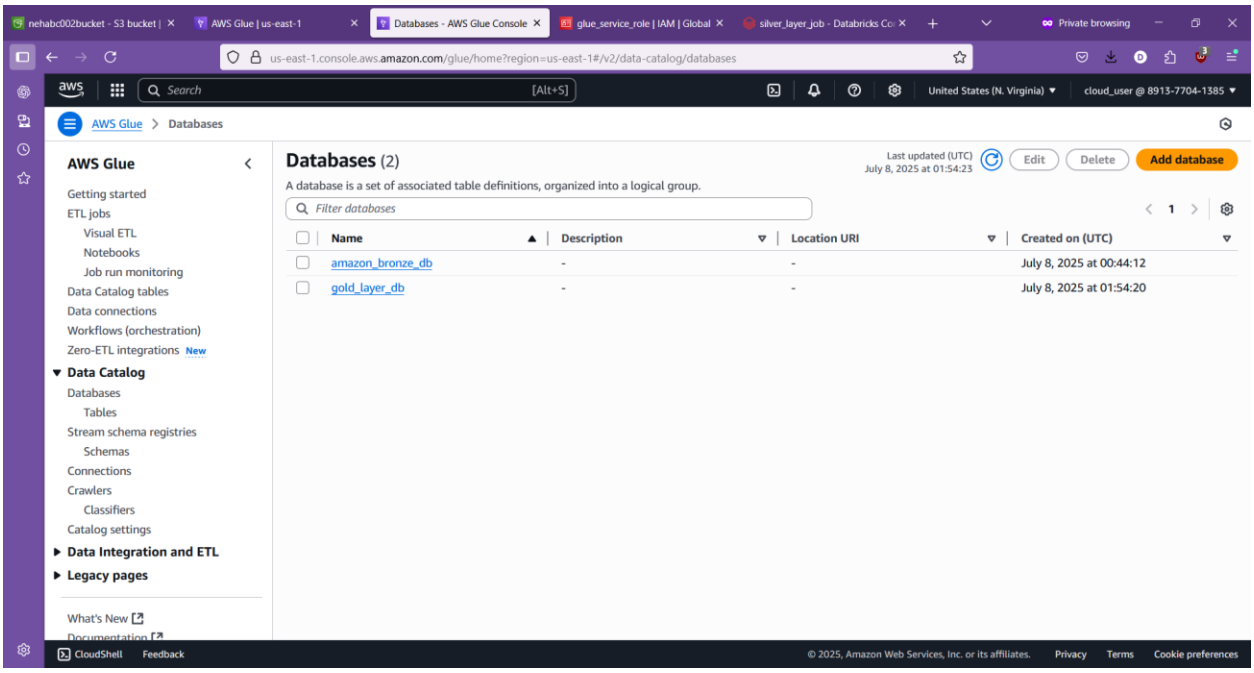

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/s3/buckets/nehabc002bucket?region=us-east-1&bucketType=general&prefix=gold/avg_rating_by_categ
glue
United States (N. Virginia)
cloud_user @ 8913-7704-1385
Amazon S3 > Buckets > nehabc002bucket > gold/ > avg_rating_by_category/
avg_rating_by_category/
Copy S3 URI
Objects
Properties
Objects (2)
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions.
Learn more
Find objects by prefix
Name
Type
Last modified
Size
Storage class
_delta_log/
Folder
part-00000-14599f17-bebd-4fb7-865b-7ebfeda6adb9-c000.snappy.parquet
parquet
July 7, 2025, 21:47:52 (UTC-04:00)
1.4 KB
Standard
us-east-1.console.aws.amazon.com/s3/object/nehabc002bucket?region=us-east-1&bucketType=general&prefix=gold/avg_rating_by_category/part-00000-14599f17-bebd-4fb7-865b-7ebfeda6adb9-c000.snappy.parquet
affiliates.
Privacy
Terms
Cookie preferences



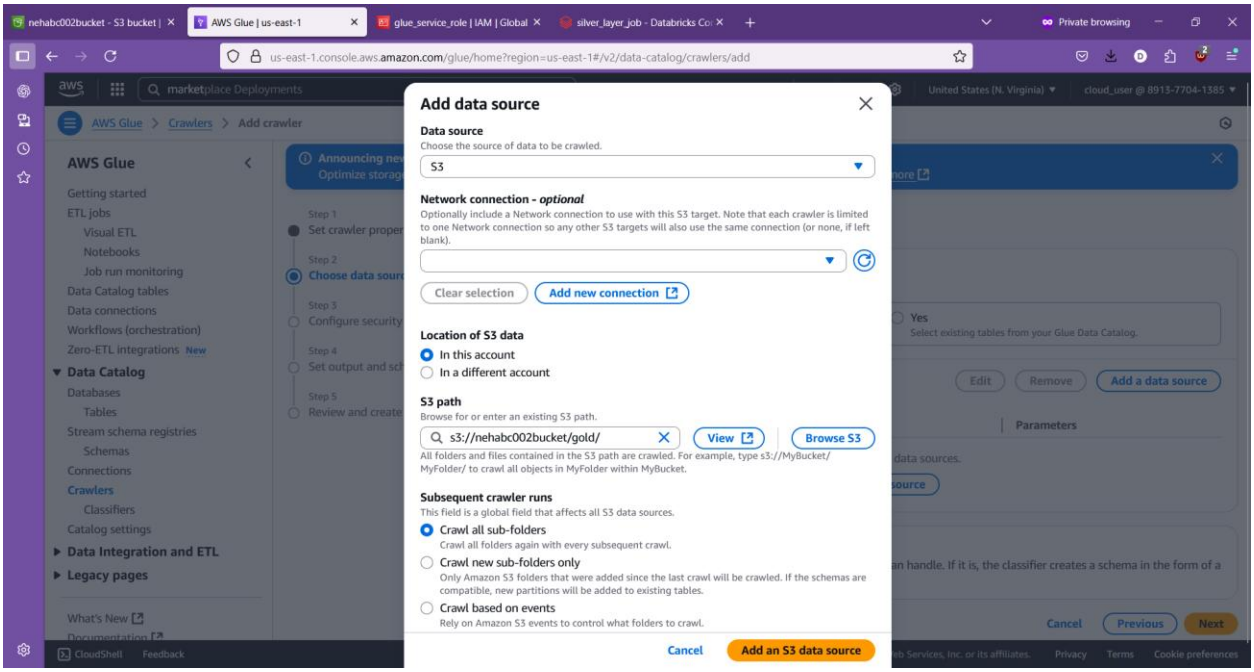


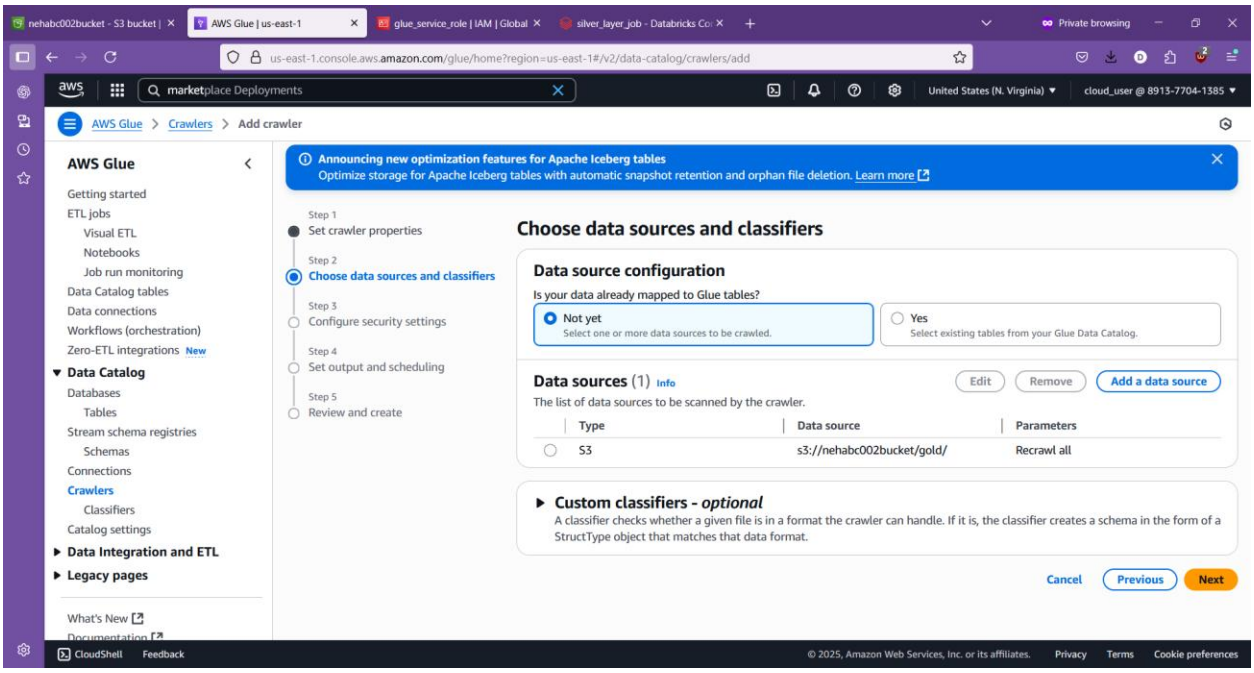
Step 2. Create Glue Crawlers for Gold Tables

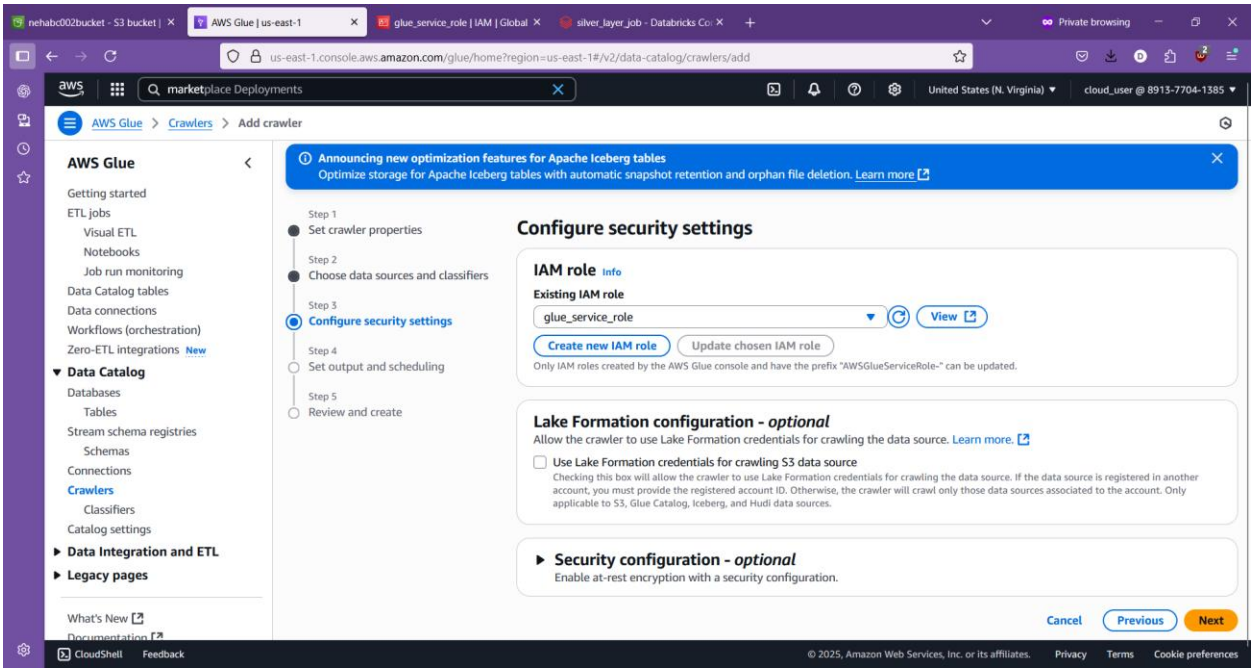
Glue Database

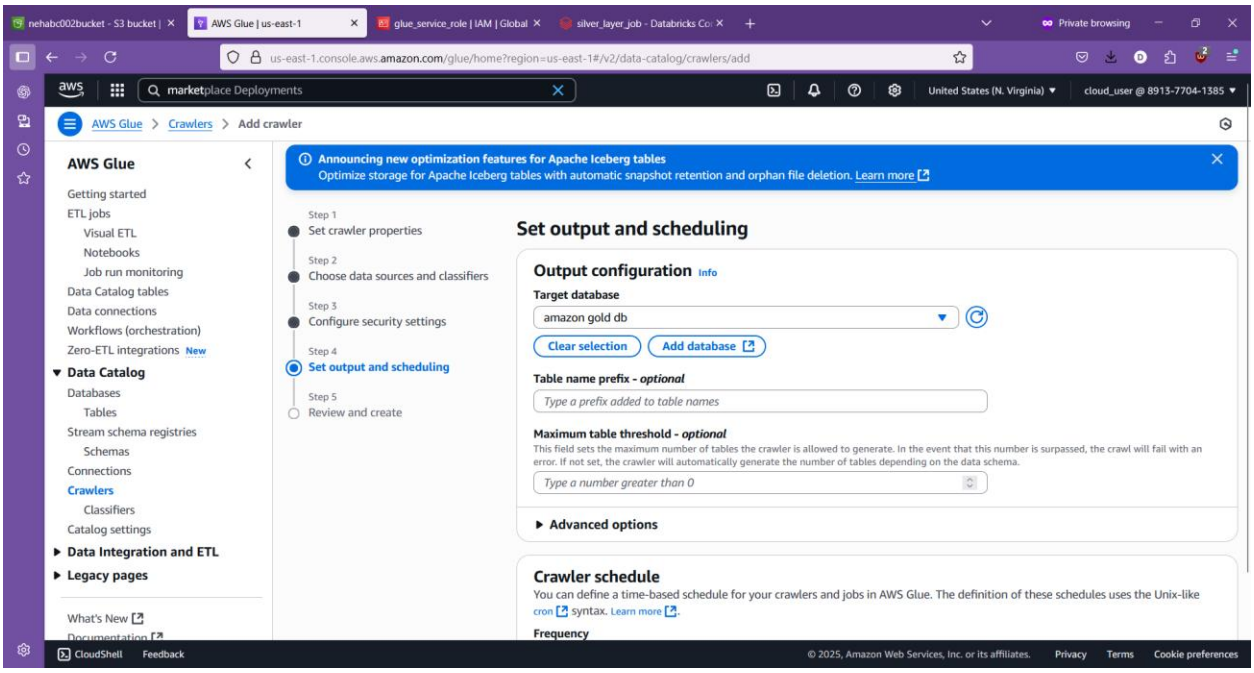


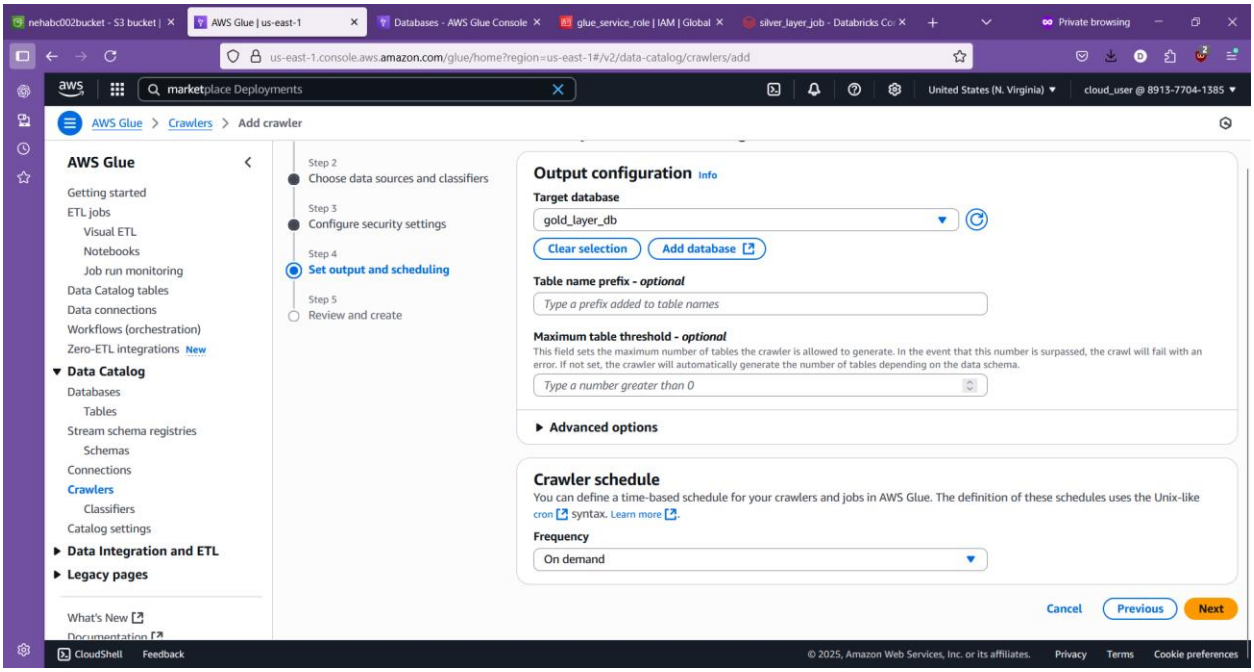
Crawler for Gold Data

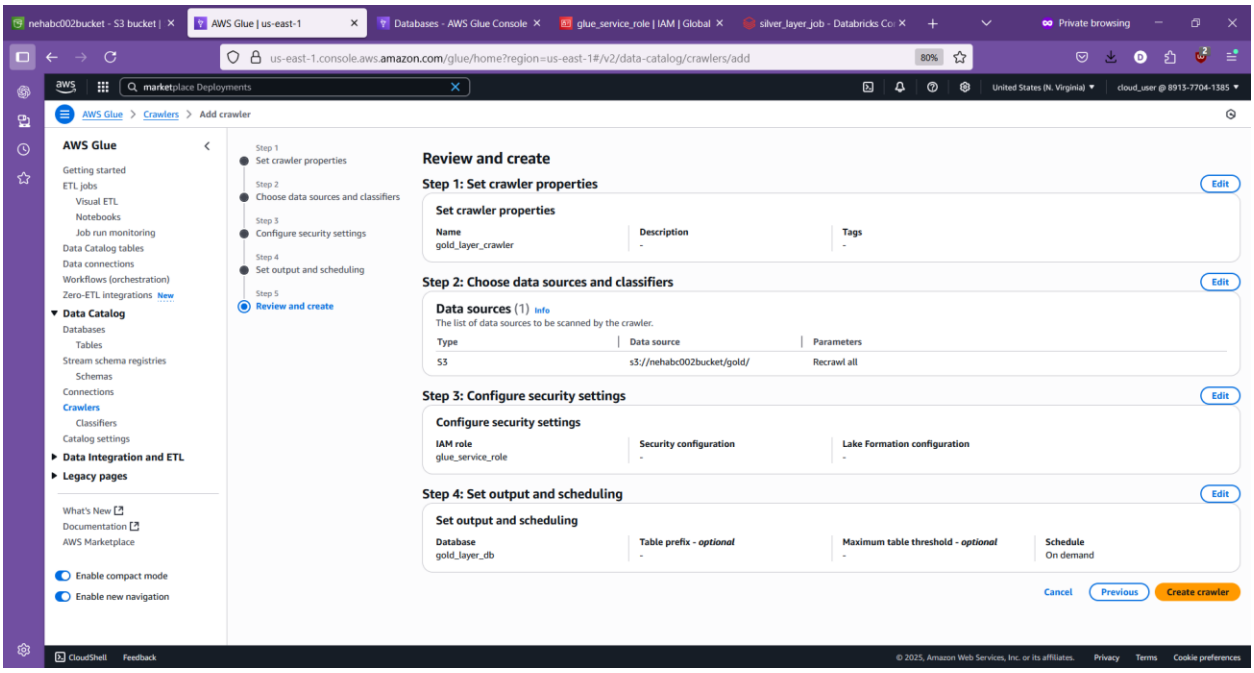


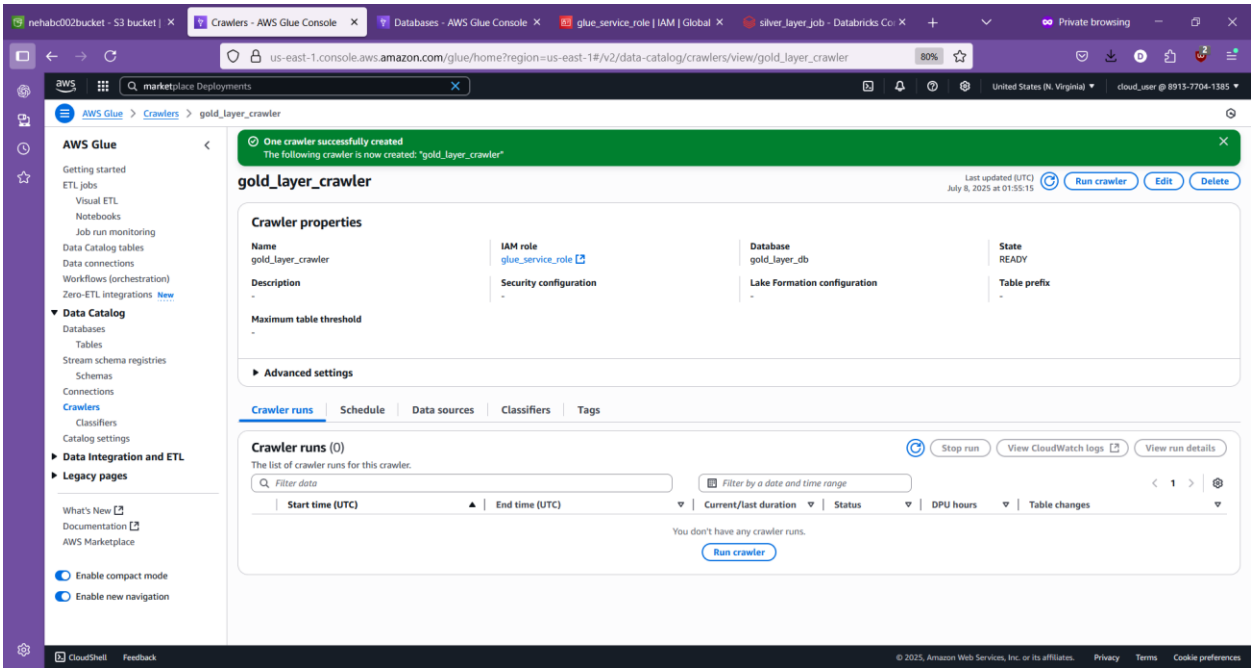

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/add
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > Add crawler
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
Announcing new optimization features for Apache Iceberg tables
Optimize storage for Apache Iceberg tables with automatic snapshot retention and orphan file deletion. Learn more
Step 1
Set crawler properties
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create
Choose data sources and classifiers
Data source configuration
Is your data already mapped to Glue tables?
Not yet
Select one or more data sources to be crawled.
Yes
Select existing tables from your Glue Data Catalog.
Data sources (1) Info
Edit
Remove
Add a data source
The list of data sources to be scanned by the crawler.
Type
Data source
Parameters
S3
s3://nehabc002bucket/gold/
Recrawl all
Custom classifiers - optional
A classifier checks whether a given file is in a format the crawler can handle. If it is, the classifier creates a schema in the form of a StructType object that matches that data format.
Cancel
Previous
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

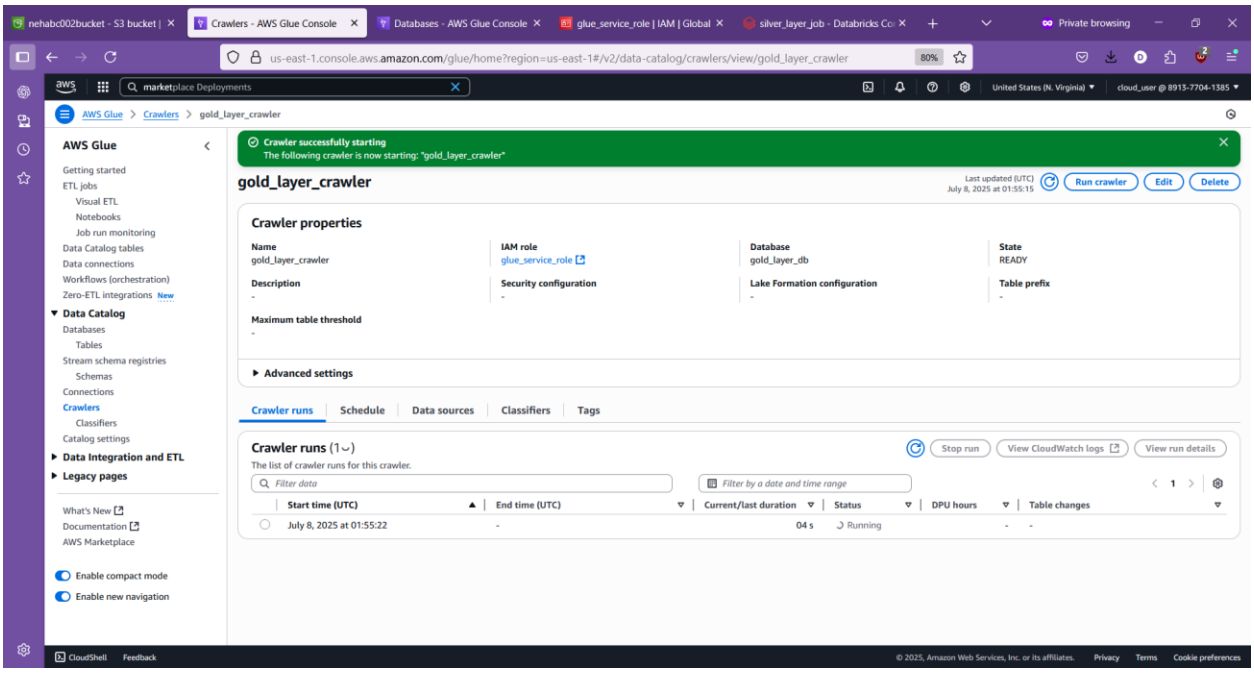

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/add
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > Add crawler
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
Announcing new optimization features for Apache Iceberg tables
Optimize storage for Apache Iceberg tables with automatic snapshot retention and orphan file deletion. Learn more
Step 1
Set crawler properties
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create
Configure security settings
IAM role Info
Existing IAM role
glue_service_role
View
Create new IAM role
Update chosen IAM role
Only IAM roles created by the AWS Glue console and have the prefix "AWSGlueServiceRole-" can be updated.
Lake Formation configuration - optional
Allow the crawler to use Lake Formation credentials for crawling the data source. Learn more.
Use Lake Formation credentials for crawling S3 data source
Checking this box will allow the crawler to use Lake Formation credentials for crawling the data source. If the data source is registered in another account, you must provide the registered account ID. Otherwise, the crawler will crawl only those data sources associated to the account. Only applicable to S3, Glue Catalog, Iceberg, and Hudi data sources.
Security configuration - optional
Enable at-rest encryption with a security configuration.
Cancel
Previous
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

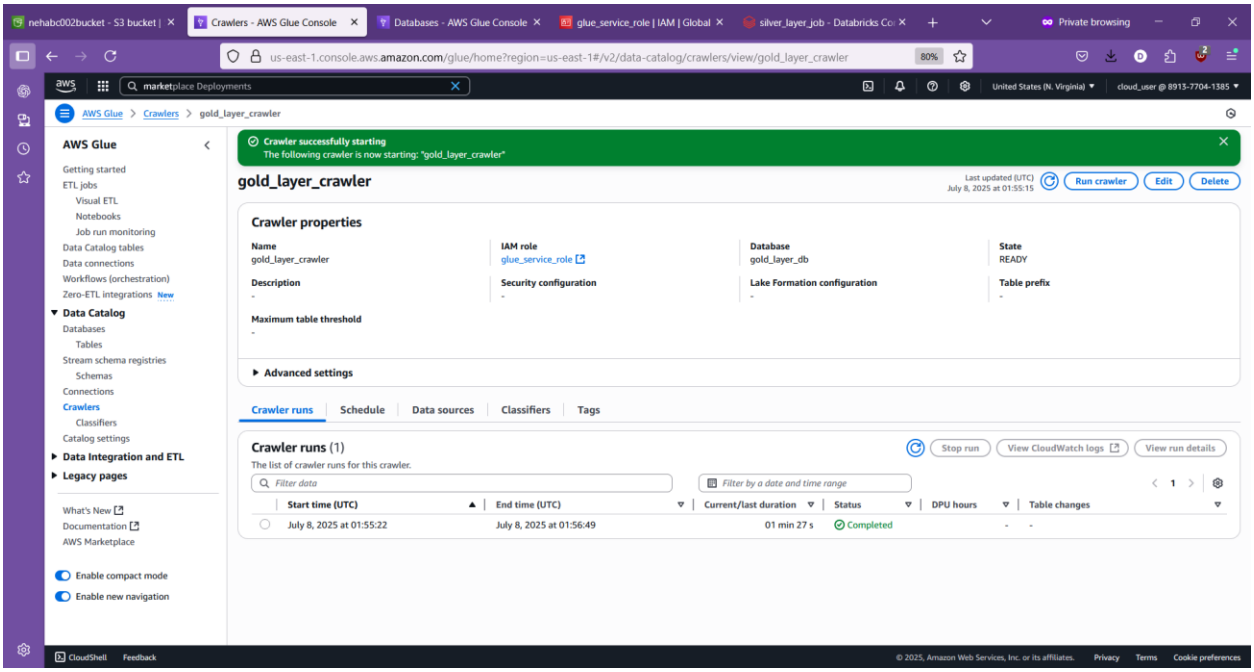

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/add
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > Add crawler
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
Announcing new optimization features for Apache Iceberg tables
Optimize storage for Apache Iceberg tables with automatic snapshot retention and orphan file deletion. Learn more
Step 1
Set crawler properties
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create
Set output and scheduling
Output configuration Info
Target database
amazon gold db
Clear selection
Add database
Table name prefix - optional
Type a prefix added to table names
Maximum table threshold - optional
This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.
Type a number greater than 0
Advanced options
Crawler schedule
You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. Learn more.
Frequency
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

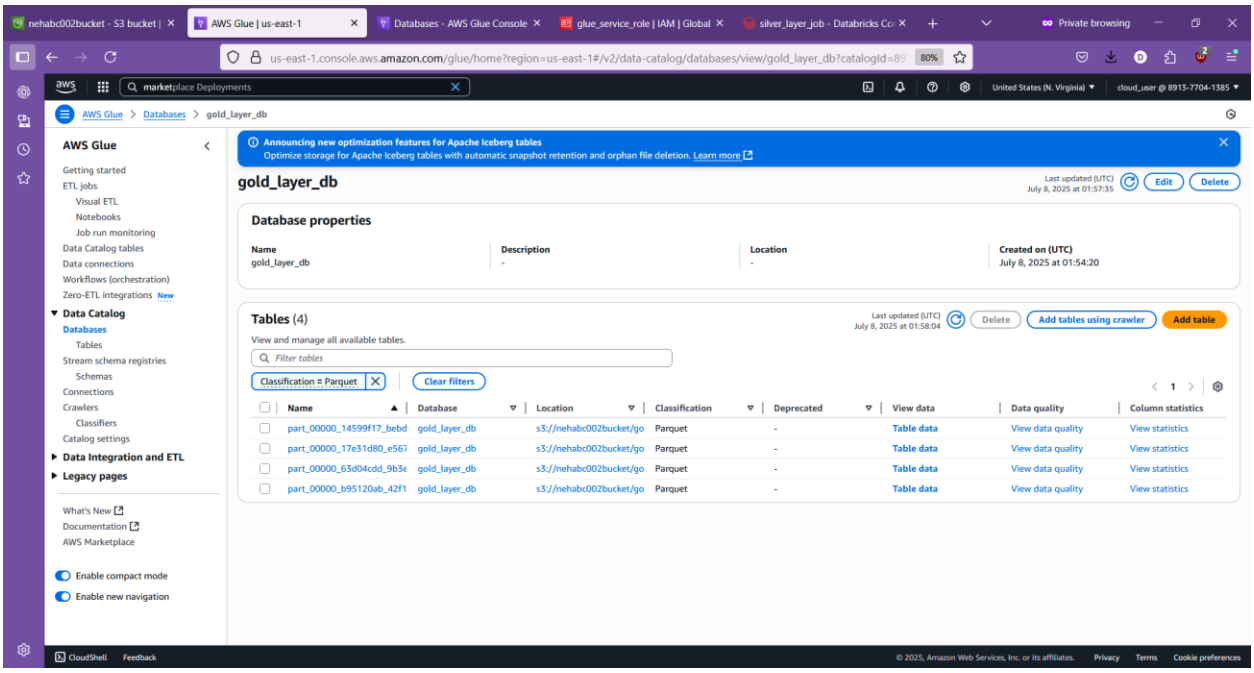

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/add
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > Add crawler
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create
Output configuration Info
Target database
gold_layer_db
Clear selection
Add database
Table name prefix - optional
Type a prefix added to table names
Maximum table threshold - optional
This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.
Type a number greater than 0
Advanced options
Crawler schedule
You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. Learn more.
Frequency
On demand
Cancel
Previous
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

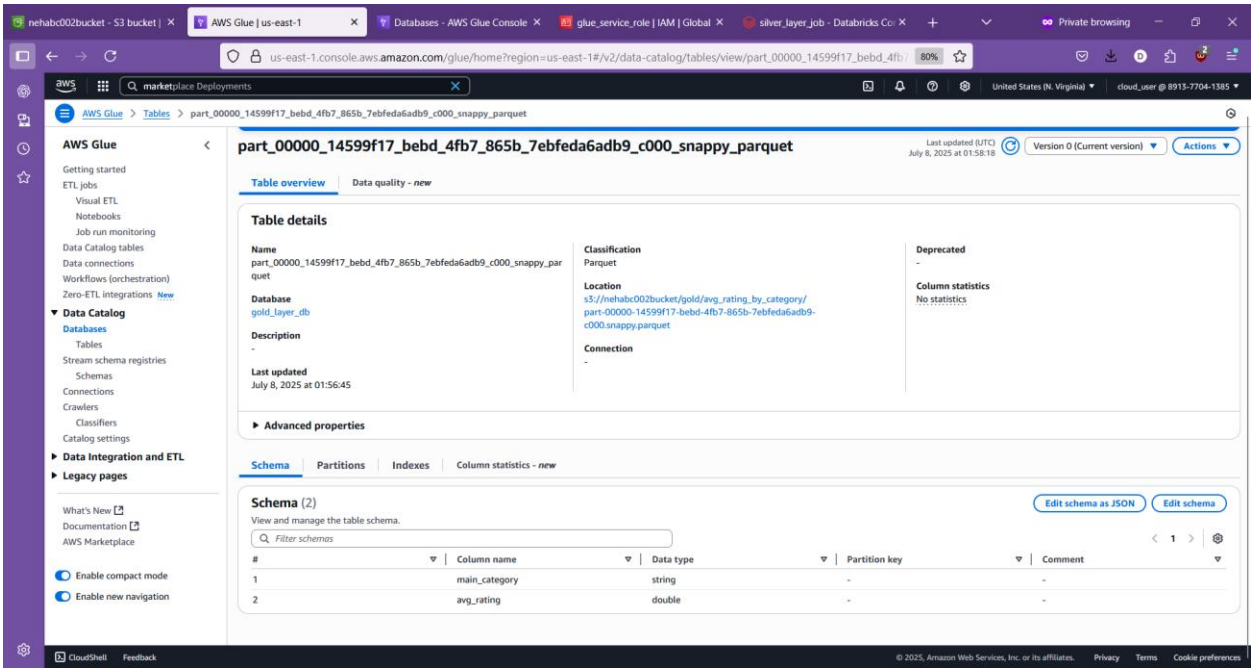

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/add
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > Add crawler
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Step 1 Set crawler properties
Step 2 Choose data sources and classifiers
Step 3 Configure security settings
Step 4 Set output and scheduling
Step 5 Review and create
Review and create
Step 1: Set crawler properties
Edit
Set crawler properties
Name
gold_layer_crawler
Description
-
Tags
-
Step 2: Choose data sources and classifiers
Edit
Data sources (1) Info
The list of data sources to be scanned by the crawler.
Type
Data source
Parameters
S3
s3://nehabc002bucket/gold/
Recrawl all
Step 3: Configure security settings
Edit
Configure security settings
IAM role
glue_service_role
Security configuration
-
Lake Formation configuration
-
Step 4: Set output and scheduling
Edit
Set output and scheduling
Database
gold_layer_db
Table prefix - optional
-
Maximum table threshold - optional
-
Schedule
On demand
Cancel
Previous
Create crawler
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

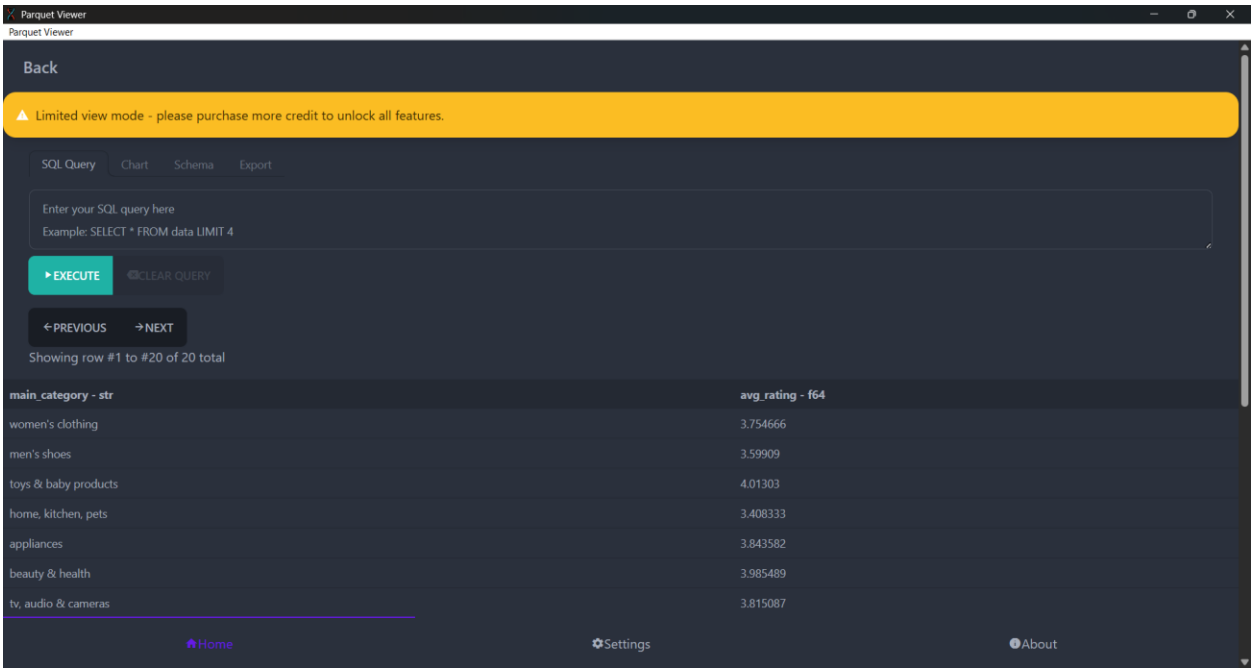

nehabc002bucket - S3 bucket |
Crawlers - AWS Glue Console
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/view/gold_layer_crawler
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > gold_layer_crawler
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
One crawler successfully created
The following crawler is now created: "gold_layer_crawler"
gold_layer_crawler
Last updated (UTC)
July 8, 2025 at 01:55:15
Run crawler
Edit
Delete
Crawler properties
Name
gold_layer_crawler
IAM role
glue_service_role
Database
gold_layer_db
State
READY
Description
-
Security configuration
-
Lake Formation configuration
-
Table prefix
-
Maximum table threshold
-
Advanced settings
Crawler runs
Schedule
Data sources
Classifiers
Tags
Crawler runs (0)
The list of crawler runs for this crawler.
Stop run
View CloudWatch logs
View run details
Filter data
Filter by a date and time range
Start time (UTC)
End time (UTC)
Current/last duration
Status
DPU hours
Table changes
You don't have any crawler runs.
Run crawler
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

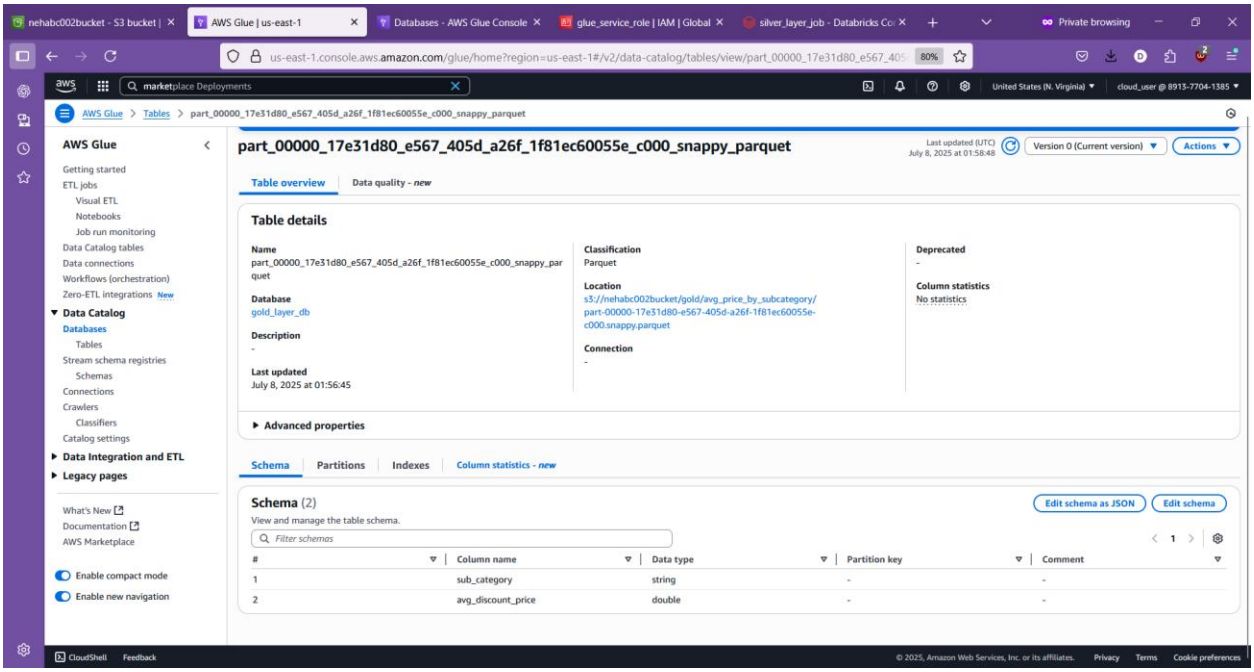

nehabc002bucket - S3 bucket |
Crawlers - AWS Glue Console
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/view/gold_layer_crawler
80%
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > gold_layer_crawler
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Crawler successfully starting
The following crawler is now starting: "gold_layer_crawler"
gold_layer_crawler
Last updated (UTC)
July 8, 2025 at 01:55:15
Run crawler
Edit
Delete
Crawler properties
Name
gold_layer_crawler
IAM role
glue_service_role
Database
gold_layer_db
State
READY
Description
-
Security configuration
-
Lake Formation configuration
-
Table prefix
-
Maximum table threshold
-
Advanced settings
Crawler runs
Schedule
Data sources
Classifiers
Tags
Crawler runs (1)
The list of crawler runs for this crawler.
Stop run
View CloudWatch logs
View run details
Filter data
Filter by a date and time range
Start time (UTC)
End time (UTC)
Current/last duration
Status
DPU hours
Table changes
July 8, 2025 at 01:55:22
-
04 s
Running
-
-
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

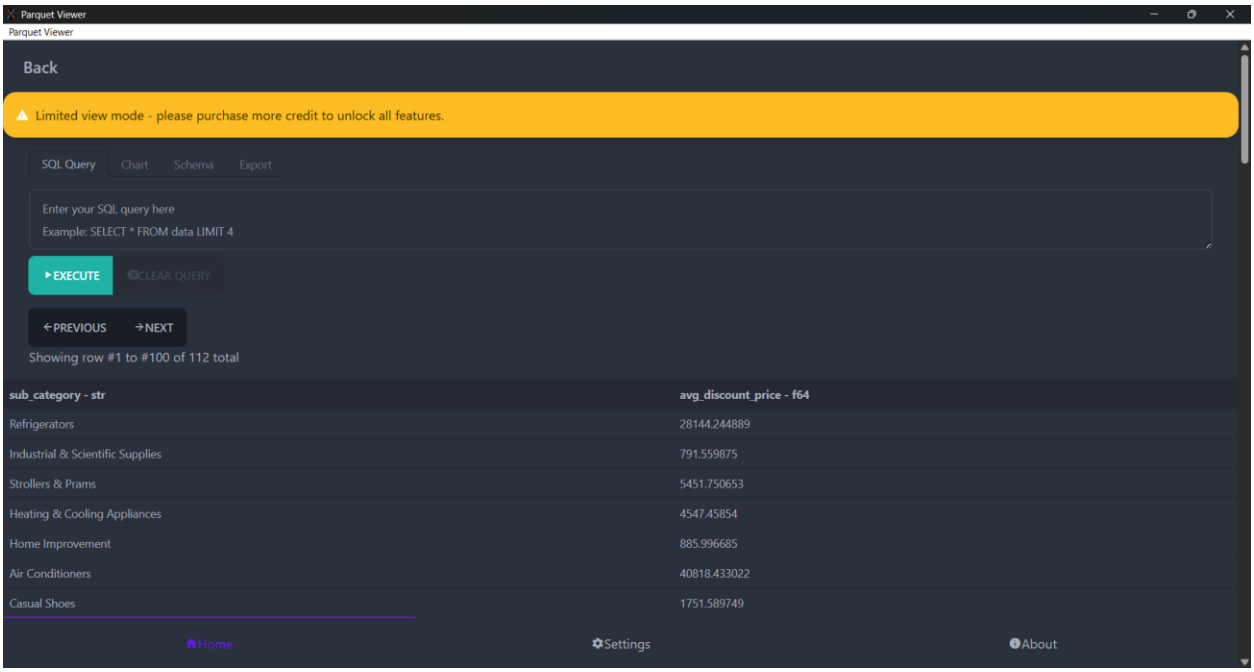

nehabc002bucket - S3 bucket |
Crawlers - AWS Glue Console
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/view/gold_layer_crawler
80%
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Crawlers > gold_layer_crawler
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Crawler successfully starting
The following crawler is now starting: "gold_layer_crawler"
gold_layer_crawler
Last updated (UTC)
July 8, 2025 at 01:55:15
Run crawler
Edit
Delete
Crawler properties
Name
gold_layer_crawler
IAM role
glue_service_role
Database
gold_layer_db
State
READY
Description
-
Security configuration
-
Lake Formation configuration
-
Table prefix
-
Maximum table threshold
-
Advanced settings
Crawler runs
Schedule
Data sources
Classifiers
Tags
Crawler runs (1)
The list of crawler runs for this crawler.
Stop run
View CloudWatch logs
View run details
Filter data
Filter by a date and time range
Start time (UTC)
End time (UTC)
Current/last duration
Status
DPU hours
Table changes
July 8, 2025 at 01:55:22
July 8, 2025 at 01:56:49
01 min 27 s
Completed
-
-
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

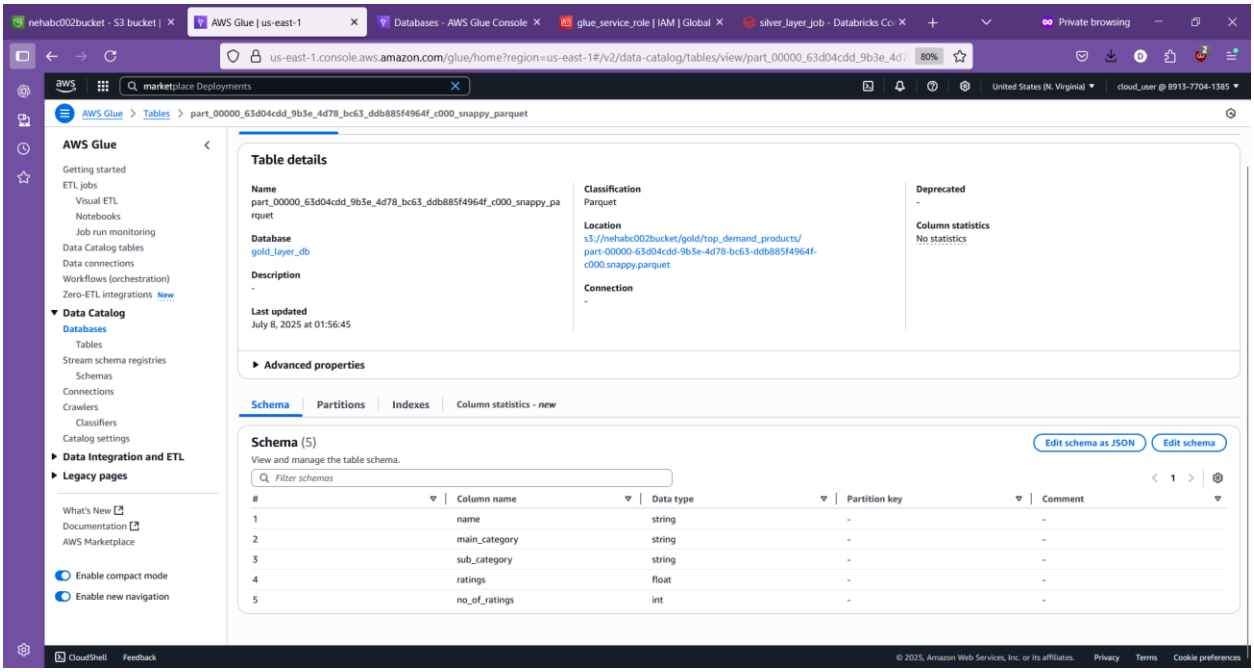

AWS Glue | us-east-1
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/databases/view/gold_layer_db?catalogId=89
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Databases > gold_layer_db
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Announcing new optimization features for Apache Iceberg tables
Optimize storage for Apache Iceberg tables with automatic snapshot retention and orphan file deletion. Learn more
gold_layer_db
Last updated (UTC) July 8, 2025 at 01:57:35
Edit
Delete
Database properties
Name
gold_layer_db
Description
-
Location
-
Created on (UTC)
July 8, 2025 at 01:54:20
Tables (4)
Last updated (UTC) July 8, 2025 at 01:58:04
Delete
Add tables using crawler
Add table
View and manage all available tables.
Filter tables
Classification = Parquet
Clear filters
Name
Database
Location
Classification
Deprecated
View data
Data quality
Column statistics
part_00000_14599f17_bebd
gold_layer_db
s3://nehabc002bucket/go
Parquet
-
Table data
View data quality
View statistics
part_00000_17e31d80_e567
gold_layer_db
s3://nehabc002bucket/go
Parquet
-
Table data
View data quality
View statistics
part_00000_63d04cdd_9b3e
gold_layer_db
s3://nehabc002bucket/go
Parquet
-
Table data
View data quality
View statistics
part_00000_b95120ab_42f1
gold_layer_db
s3://nehabc002bucket/go
Parquet
-
Table data
View data quality
View statistics
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

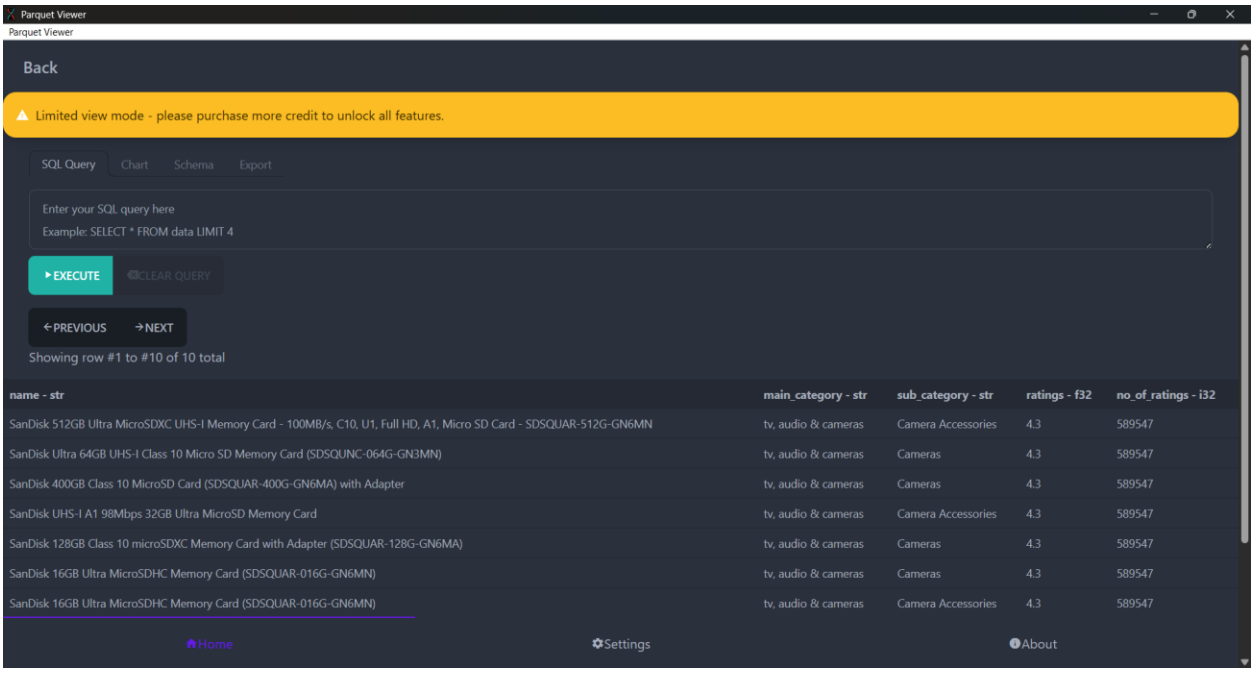

AWS Glue | us-east-1
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/tables/view/part_00000_14599f17_bebd_4fb
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Tables > part_00000_14599f17_bebd_4fb7_865b_7ebfeda6adb9_c000_snappy_parquet
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
part_00000_14599f17_bebd_4fb7_865b_7ebfeda6adb9_c000_snappy_parquet
Last updated (UTC) July 8, 2025 at 01:58:18
Version 0 (Current version)
Actions
Table overview
Data quality - new
Table details
Name
part_00000_14599f17_bebd_4fb7_865b_7ebfeda6adb9_c000_snappy_parquet
Database
gold_layer_db
Description
-
Last updated
July 8, 2025 at 01:56:45
Classification
Parquet
Location
s3://nehabc002bucket/gold/avg_rating_by_category/part-00000-14599f17-bebd-4fb7-865b-7ebfeda6adb9-c000.snappy.parquet
Connection
-
Deprecated
-
Column statistics
No statistics
Advanced properties
Schema
Partitions
Indexes
Column statistics - new
Schema (2)
Edit schema as JSON
Edit schema
View and manage the table schema.
Filter schemas
#
Column name
Data type
Partition key
Comment
1
main_category
string
-
-
2
avg_rating
double
-
-
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

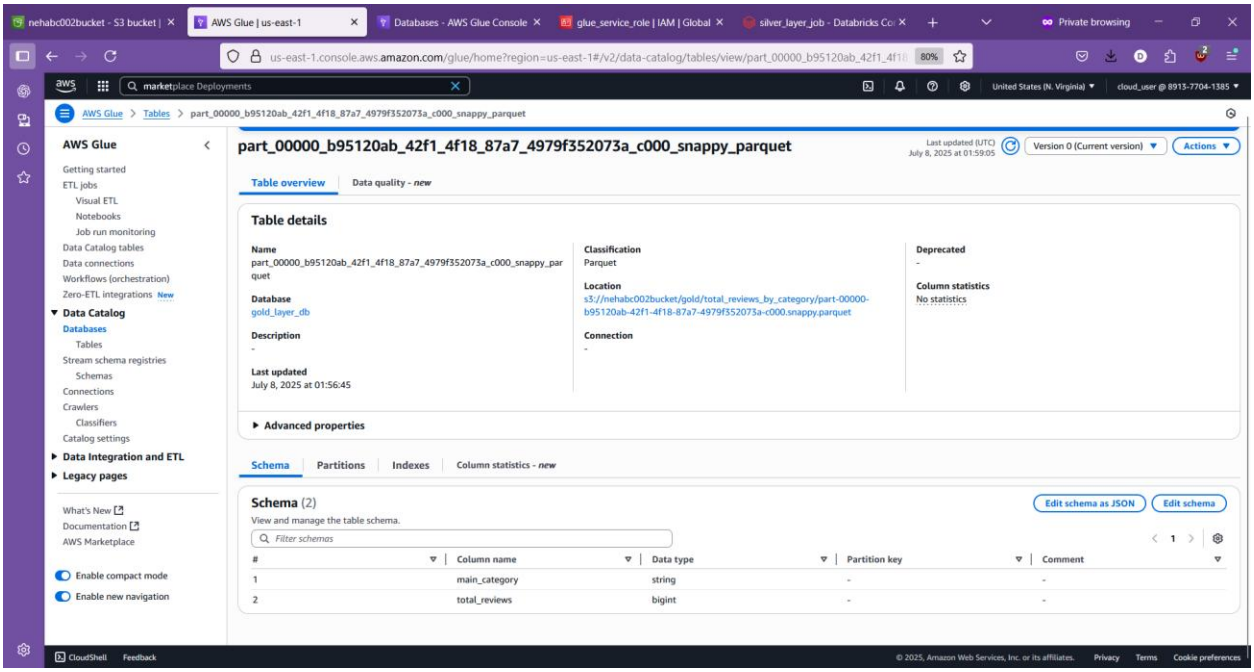

Parquet Viewer
Parquet Viewer
Back
Limited view mode - please purchase more credit to unlock all features.
SQL Query
Chart
Schema
Export
Enter your SQL query here
Example: SELECT * FROM data LIMIT 4
EXECUTE
CLEAR QUERY
←PREVIOUS
→NEXT
Showing row #1 to #20 of 20 total
main_category - str
avg_rating - f64
women's clothing
3.754666
men's shoes
3.59909
toys & baby products
4.01303
home, kitchen, pets
3.408333
appliances
3.843582
beauty & health
3.985489
tv, audio & cameras
3.815087
Home
Settings
About

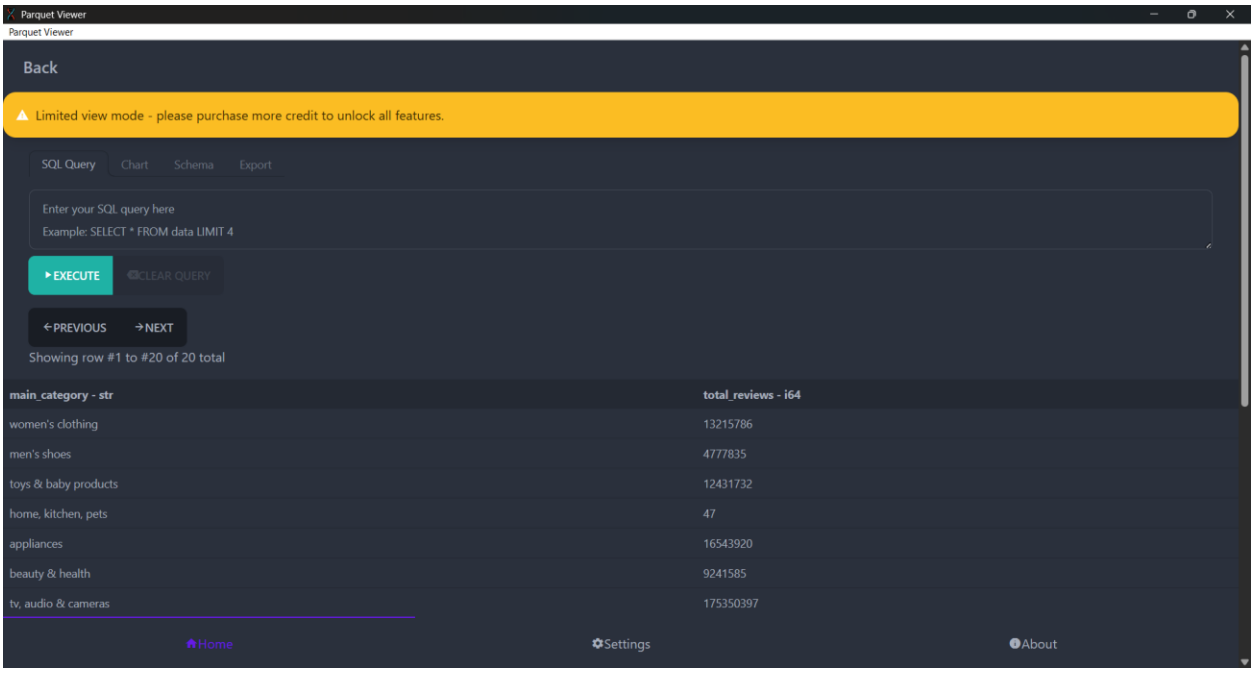

nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/tables/view/part_00000_17e31d80_e567_405
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Tables > part_00000_17e31d80_e567_405d_a26f_1f81ec60055e_c000_snappy_parquet
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
part_00000_17e31d80_e567_405d_a26f_1f81ec60055e_c000_snappy_parquet
Last updated (UTC)
July 8, 2025 at 01:58:48
Version 0 (Current version)
Actions
Table overview
Data quality - new
Table details
Name
part_00000_17e31d80_e567_405d_a26f_1f81ec60055e_c000_snappy_parquet
Database
gold_layer_db
Description
-
Last updated
July 8, 2025 at 01:56:45
Classification
Parquet
Location
s3://nehabc002bucket/gold/avg_price_by_subcategory/part-00000-17e31d80-e567-405d-a26f-1f81ec60055e-c000.snappy.parquet
Connection
-
Deprecated
-
Column statistics
No statistics
Advanced properties
Schema
Partitions
Indexes
Column statistics - new
Schema (2)
View and manage the table schema.
Edit schema as JSON
Edit schema
Filter schemas
| Column name | Data type | Partition key | Comment
1 | sub_category | string | - | -
2 | avg_discount_price | double | - | -
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Parquet Viewer
Parquet Viewer
Back
Limited view mode - please purchase more credit to unlock all features.
SQL Query
Chart
Schema
Export
Enter your SQL query here
Example: SELECT * FROM data LIMIT 4
EXECUTE
CLEAR QUERY
←PREVIOUS
→NEXT
Showing row #1 to #100 of 112 total
sub_category - str
avg_discount_price - f64
Refrigerators
28144.244889
Industrial & Scientific Supplies
791.559875
Strollers & Prams
5451.750653
Heating & Cooling Appliances
4547.45854
Home Improvement
885.996685
Air Conditioners
40818.433022
Casual Shoes
1751.589749
Home
Settings
About


nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/tables/view/part_00000_63d04cdd_9b3e_4d7
80%
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Tables > part_00000_63d04cdd_9b3e_4d78_bc63_ddb885f4964f_c000_snappy_parquet
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
Table details
Name
part_00000_63d04cdd_9b3e_4d78_bc63_ddb885f4964f_c000_snappy_parquet
Database
gold_layer_db
Description
-
Last updated
July 8, 2025 at 01:56:45
Classification
Parquet
Location
s3://nehabc002bucket/gold/top_demand_products/part-00000-63d04cdd-9b3e-4d78-bc63-ddb885f4964f-c000.snappy.parquet
Connection
-
Deprecated
-
Column statistics
No statistics
Advanced properties
Schema
Partitions
Indexes
Column statistics - new
Schema (5)
View and manage the table schema.
Edit schema as JSON
Edit schema
Filter schemas
| Column name | Data type | Partition key | Comment
1 | name | string | - | -
2 | main_category | string | - | -
3 | sub_category | string | - | -
4 | ratings | float | - | -
5 | no_of_ratings | int | - | -
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Parquet Viewer
Parquet Viewer
Back
Limited view mode - please purchase more credit to unlock all features.
SQL Query
Chart
Schema
Export
Enter your SQL query here
Example: SELECT * FROM data LIMIT 4
EXECUTE
CLEAR QUERY
←PREVIOUS
→NEXT
Showing row #1 to #10 of 10 total
name - str
main_category - str
sub_category - str
ratings - f32
no_of_ratings - i32
SanDisk 512GB Ultra MicroSDXC UHS-I Memory Card - 100MB/s, C10, U1, Full HD, A1, Micro SD Card - SDSQUAR-512G-GN6MN
tv, audio & cameras
Camera Accessories
4.3
589547
SanDisk Ultra 64GB UHS-I Class 10 Micro SD Memory Card (SDSQUNC-064G-GN3MN)
tv, audio & cameras
Cameras
4.3
589547
SanDisk 400GB Class 10 MicroSD Card (SDSQUAR-400G-GN6MA) with Adapter
tv, audio & cameras
Cameras
4.3
589547
SanDisk UHS-I A1 98Mbps 32GB Ultra MicroSD Memory Card
tv, audio & cameras
Camera Accessories
4.3
589547
SanDisk 128GB Class 10 microSDXC Memory Card with Adapter (SDSQUAR-128G-GN6MA)
tv, audio & cameras
Cameras
4.3
589547
SanDisk 16GB Ultra MicroSDHC Memory Card (SDSQUAR-016G-GN6MN)
tv, audio & cameras
Cameras
4.3
589547
SanDisk 16GB Ultra MicroSDHC Memory Card (SDSQUAR-016G-GN6MN)
tv, audio & cameras
Camera Accessories
4.3
589547
Home
Settings
About


nehabc002bucket - S3 bucket |
AWS Glue | us-east-1
Databases - AWS Glue Console
glue_service_role | IAM | Global
silver_layer_job - Databricks
Private browsing
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/tables/view/part_00000_b95120ab_42f1_4f18_87a7_4979f352073a_c000_snappy_parquet
marketplace Deployments
United States (N. Virginia)
cloud_user @ 8913-7704-1385
AWS Glue > Tables > part_00000_b95120ab_42f1_4f18_87a7_4979f352073a_c000_snappy_parquet
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations New
Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings
Data Integration and ETL
Legacy pages
What's New
Documentation
AWS Marketplace
Enable compact mode
Enable new navigation
part_00000_b95120ab_42f1_4f18_87a7_4979f352073a_c000_snappy_parquet
Last updated (UTC)
July 8, 2025 at 01:59:05
Version 0 (Current version)
Actions
Table overview
Data quality - new
Table details
Name
part_00000_b95120ab_42f1_4f18_87a7_4979f352073a_c000_snappy_parquet
Database
gold_layer_db
Description
-
Last updated
July 8, 2025 at 01:56:45
Classification
Parquet
Location
s3://nehabc002bucket/gold/total_reviews_by_category/part-00000-b95120ab-42f1-4f18-87a7-4979f352073a-c000.snappy.parquet
Connection
-
Deprecated
-
Column statistics
No statistics
Advanced properties
Schema
Partitions
Indexes
Column statistics - new
Schema (2)
View and manage the table schema.
Edit schema as JSON
Edit schema
Filter schemas
| Column name | Data type | Partition key | Comment
1 | main_category | string | - | -
2 | total_reviews | bigint | - | -
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

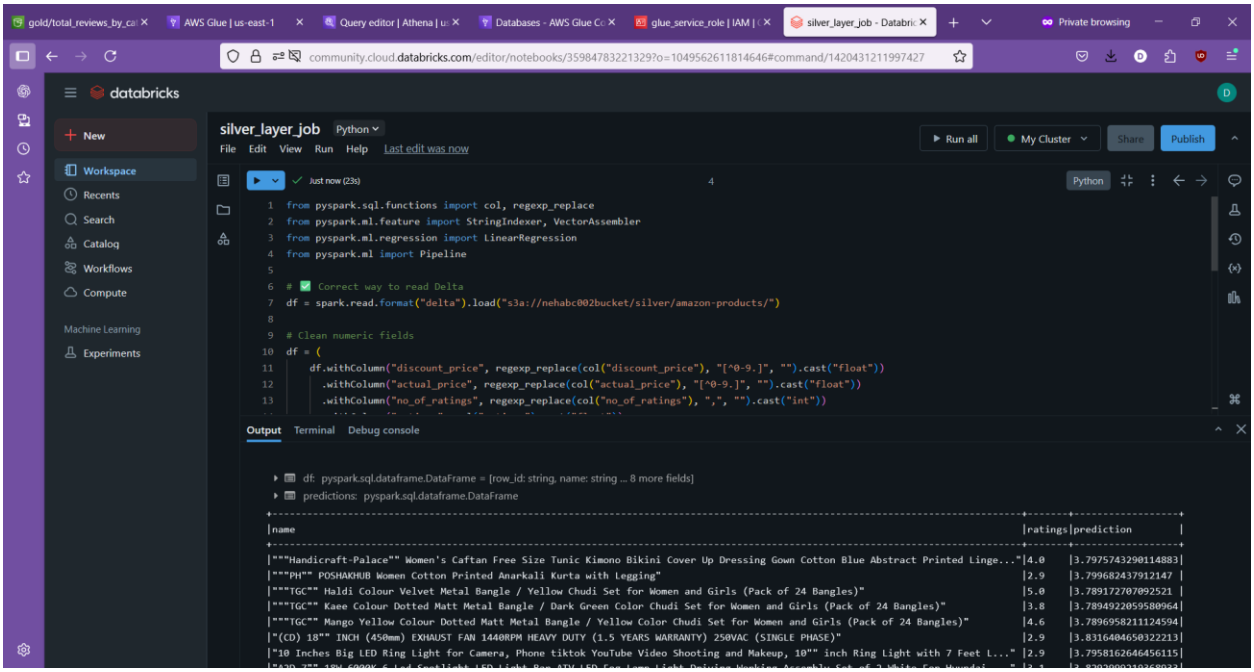


ML Layer

1: Predict Ratings using Regression

Goal:

Predict a product's ratings based on features like discount_price, actual_price, no_of_ratings, main_category / sub_category.



Code:

```
from pyspark.sql.functions import col, regexp_replace
from pyspark.ml.feature import StringIndexer, VectorAssembler
from pyspark.ml.regression import LinearRegression
from pyspark.ml import Pipeline

# Correct way to read Delta
df = spark.read.format("delta").load("s3a://nehabc002bucket/silver/amazon-products/")

# Clean numeric fields
df = (
    df.withColumn("discount_price", regexp_replace(col("discount_price"), "[^0-9.]", "").cast("float"))
      .withColumn("actual_price", regexp_replace(col("actual_price"), "[^0-9.]", "").cast("float"))
      .withColumn("no_of_ratings", regexp_replace(col("no_of_ratings"), ",", "").cast("int"))
      .withColumn("ratings", col("ratings").cast("float"))
)

# Drop nulls
df = df.dropna(subset=["ratings", "discount_price", "actual_price", "no_of_ratings", "main_category"])

# Encode categorical features
category_indexer = StringIndexer(inputCol="main_category", outputCol="main_category_index")

# Assemble features into a vector
assembler = VectorAssembler(
    inputCols=["discount_price", "actual_price", "no_of_ratings", "main_category_index"],
    outputCol="features"
)

# Create model
lr = LinearRegression(featuresCol="features", labelCol="ratings")
```

Build and run pipeline

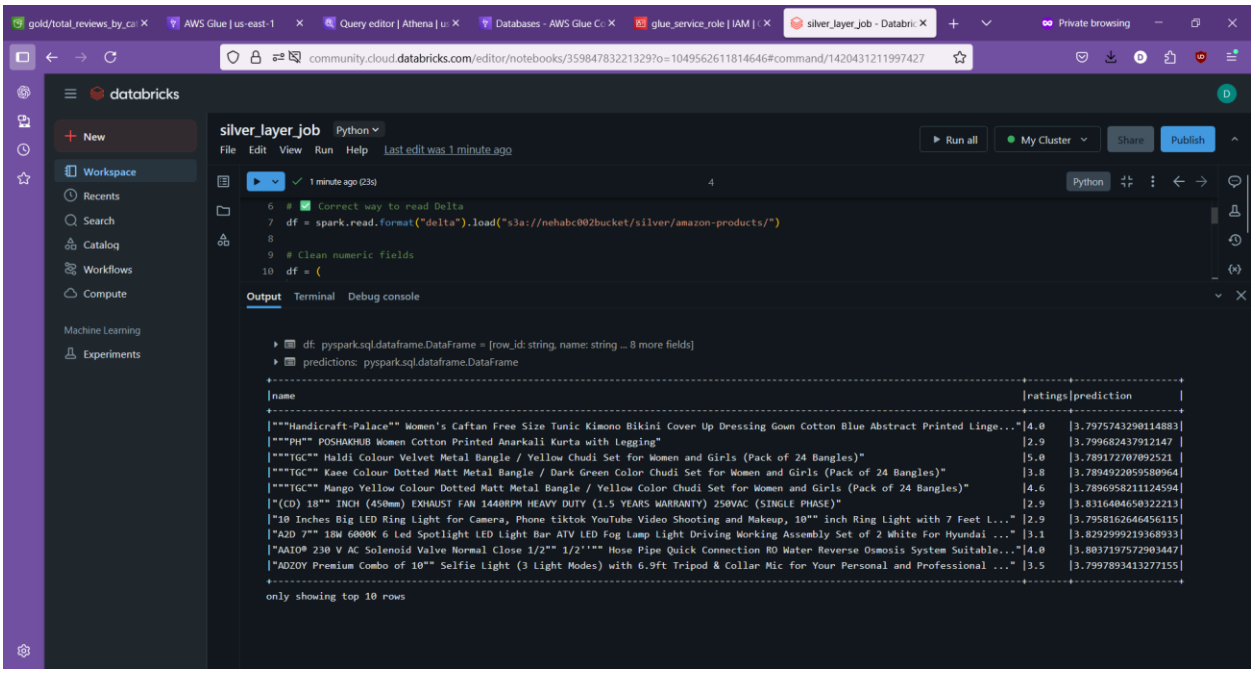
pipeline = Pipeline(stages=[category_indexer, assembler, lr])

model = pipeline.fit(df)

Make predictions

predictions = model.transform(df)

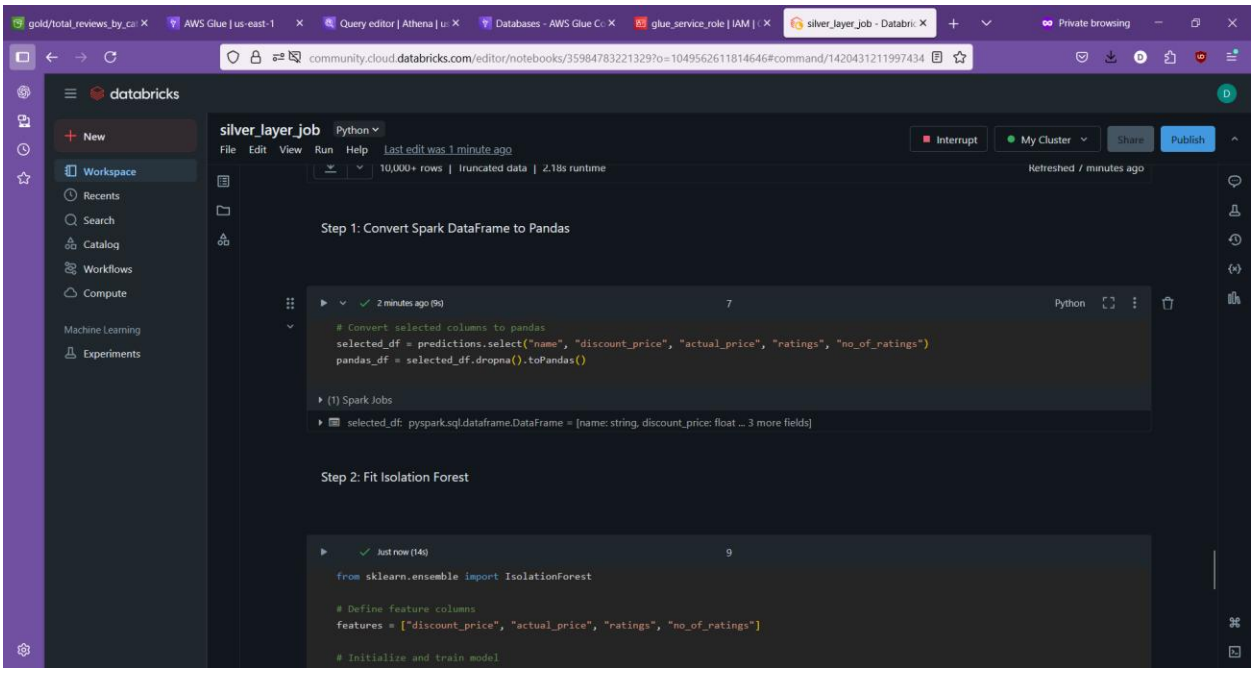
predictions.select("name", "ratings", "prediction").show(10, truncate=False)

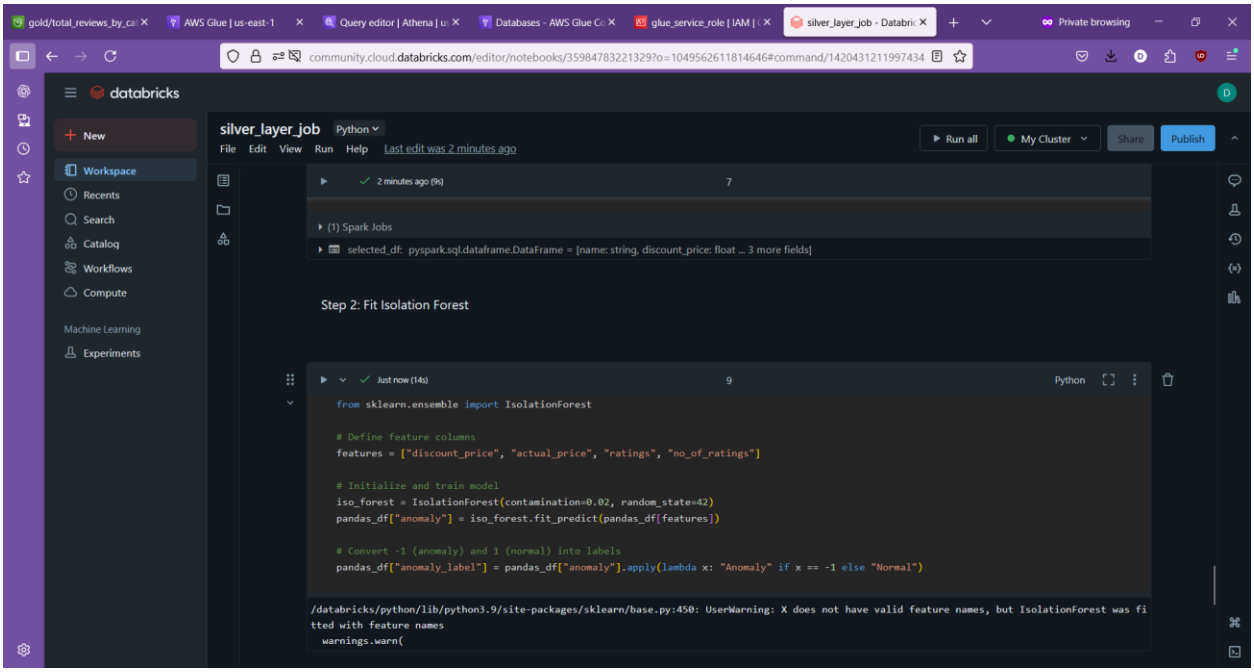


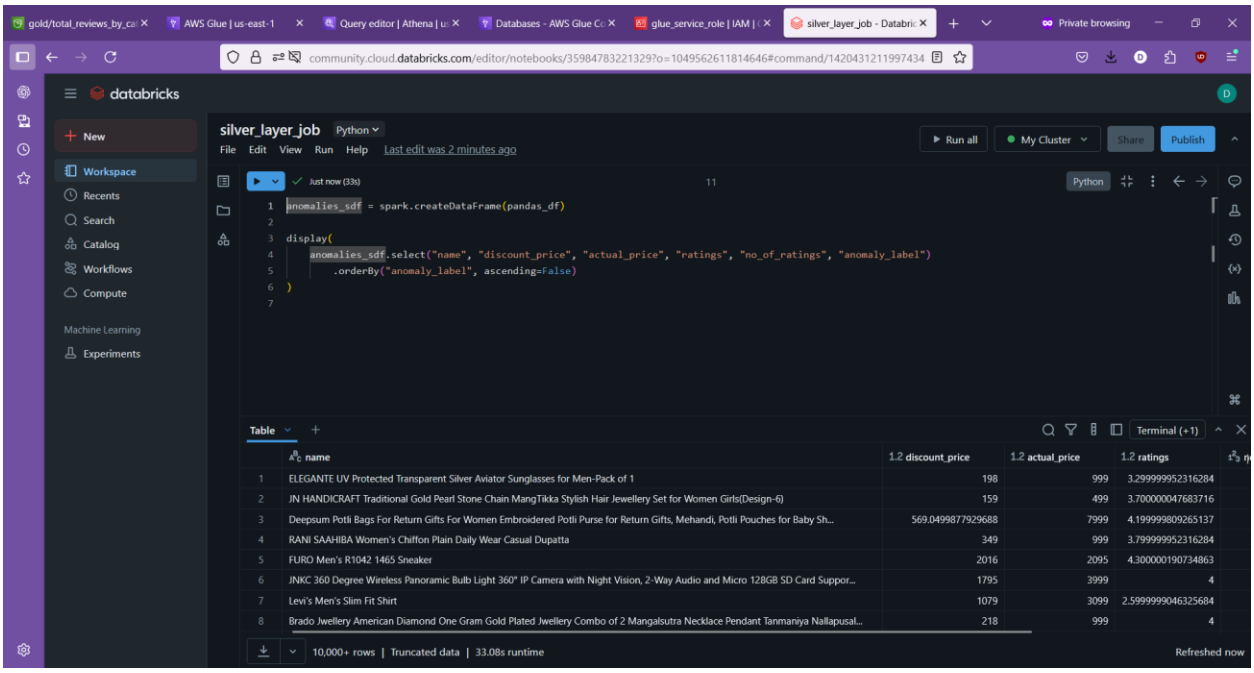
2. Isolation Forest in Databricks (via scikit-learn)

Goal:

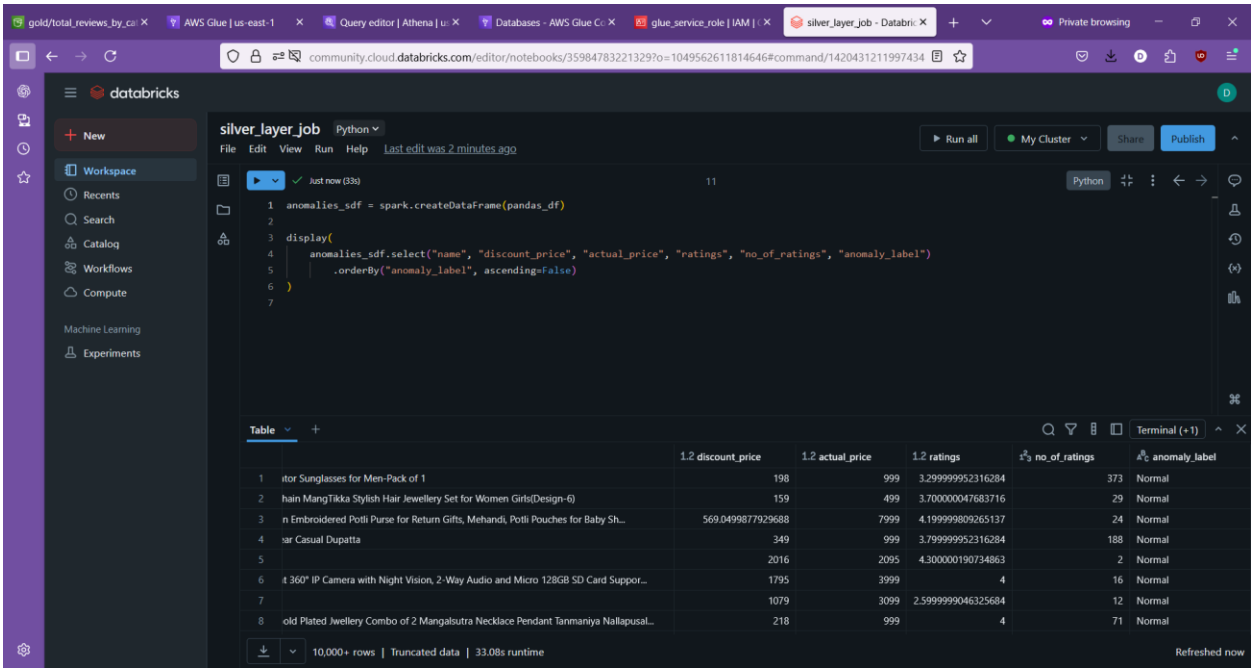
Use IsolationForest to detect anomalous products based on discount_price, actual_price, ratings, no_of_ratings


silver_layer_job
Python
File Edit View Run Help Last edit was 1 minute ago
Interrupt
My Cluster
Share
Publish
10,000+ rows | Truncated data | 2.18s runtime
Refreshed 7 minutes ago
Step 1: Convert Spark DataFrame to Pandas
2 minutes ago (9s)
7
Python
# Convert selected columns to pandas
selected_df = predictions.select("name", "discount_price", "actual_price", "ratings", "no_of_ratings")
pandas_df = selected_df.dropna().toPandas()
(1) Spark Jobs
selected_df: pyspark.sql.dataframe.DataFrame = [name: string, discount_price: float ... 3 more fields]
Step 2: Fit Isolation Forest
Just now (14s)
9
from sklearn.ensemble import IsolationForest
# Define feature columns
features = ["discount_price", "actual_price", "ratings", "no_of_ratings"]
# Initialize and train model


silver_layer_job
Python
File Edit View Run Help Last edit was 2 minutes ago
Run all
My Cluster
Share
Publish
2 minutes ago (9s)
7
(1) Spark Jobs
selected_df: pyspark.sql.dataframe.DataFrame = [name: string, discount_price: float ... 3 more fields]
Step 2: Fit Isolation Forest
Just now (14s)
9
Python
from sklearn.ensemble import IsolationForest
# Define feature columns
features = ["discount_price", "actual_price", "ratings", "no_of_ratings"]
# Initialize and train model
iso_forest = IsolationForest(contamination=0.02, random_state=42)
pandas_df["anomaly"] = iso_forest.fit_predict(pandas_df[features])
Convert -1 (anomaly) and 1 (normal) into labels
pandas_df["anomaly_label"] = pandas_df["anomaly"].apply(lambda x: "Anomaly" if x == -1 else "Normal")
/databricks/python/lib/python3.9/site-packages/sklearn/base.py:450: UserWarning: X does not have valid feature names, but IsolationForest was fitted with feature names
warnings.warn(

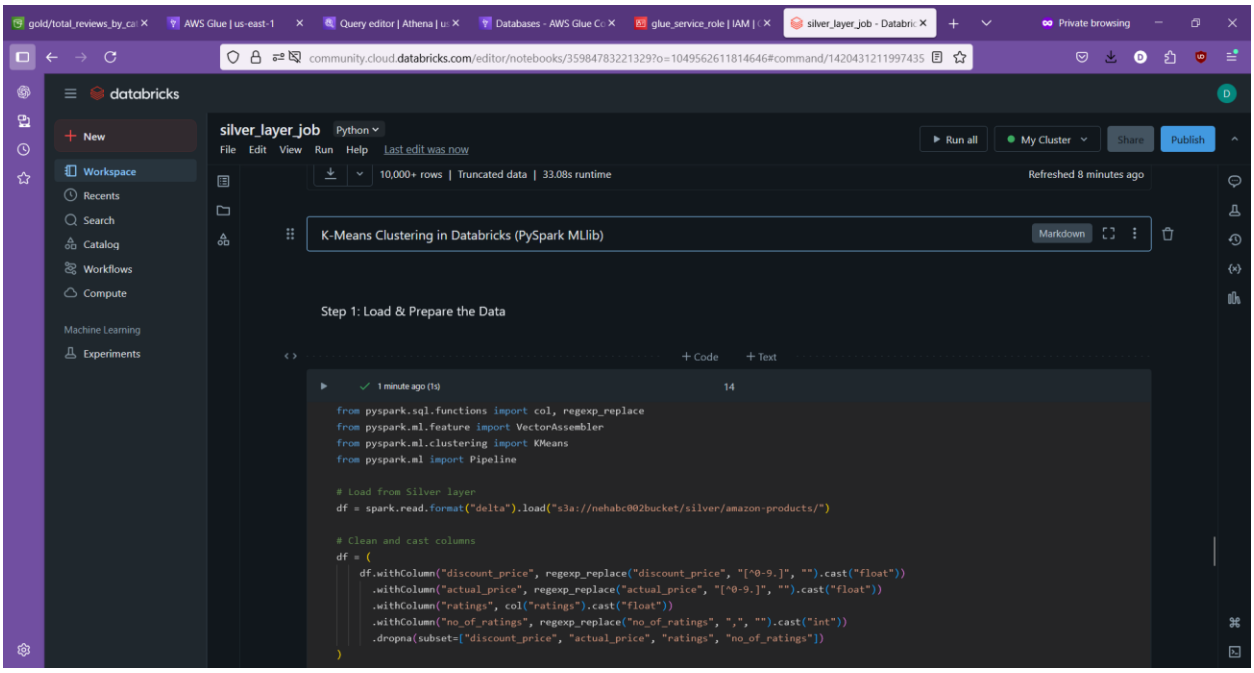


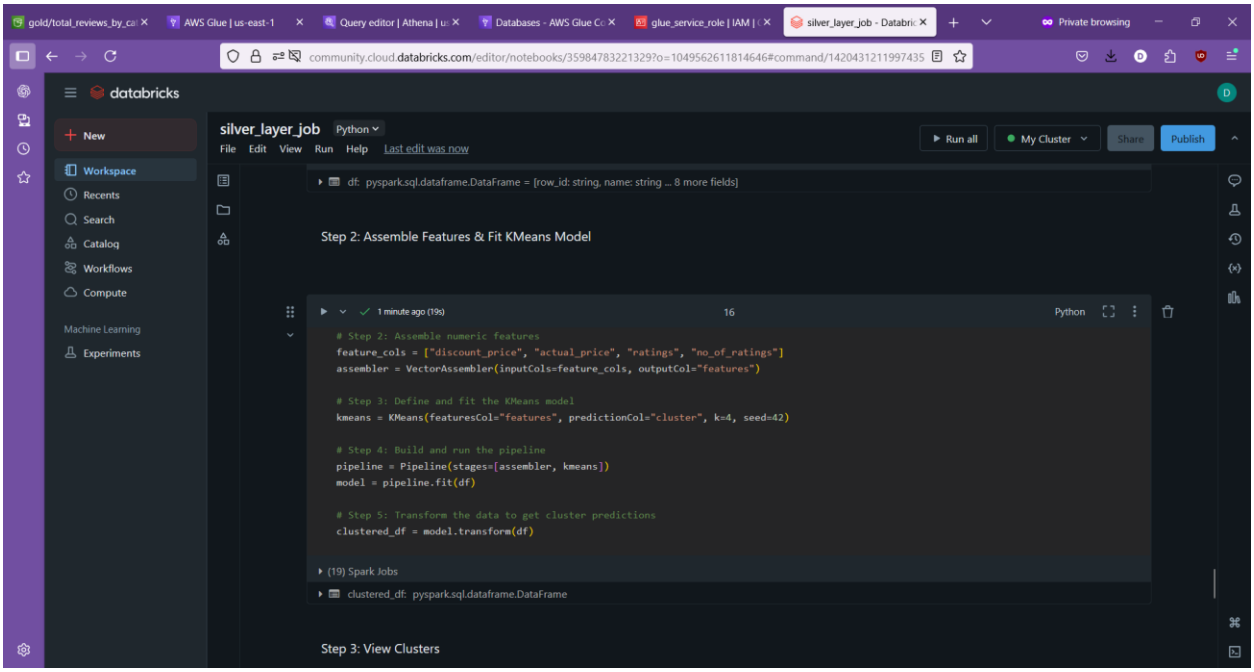
Convert Back to Spark for Visualization:

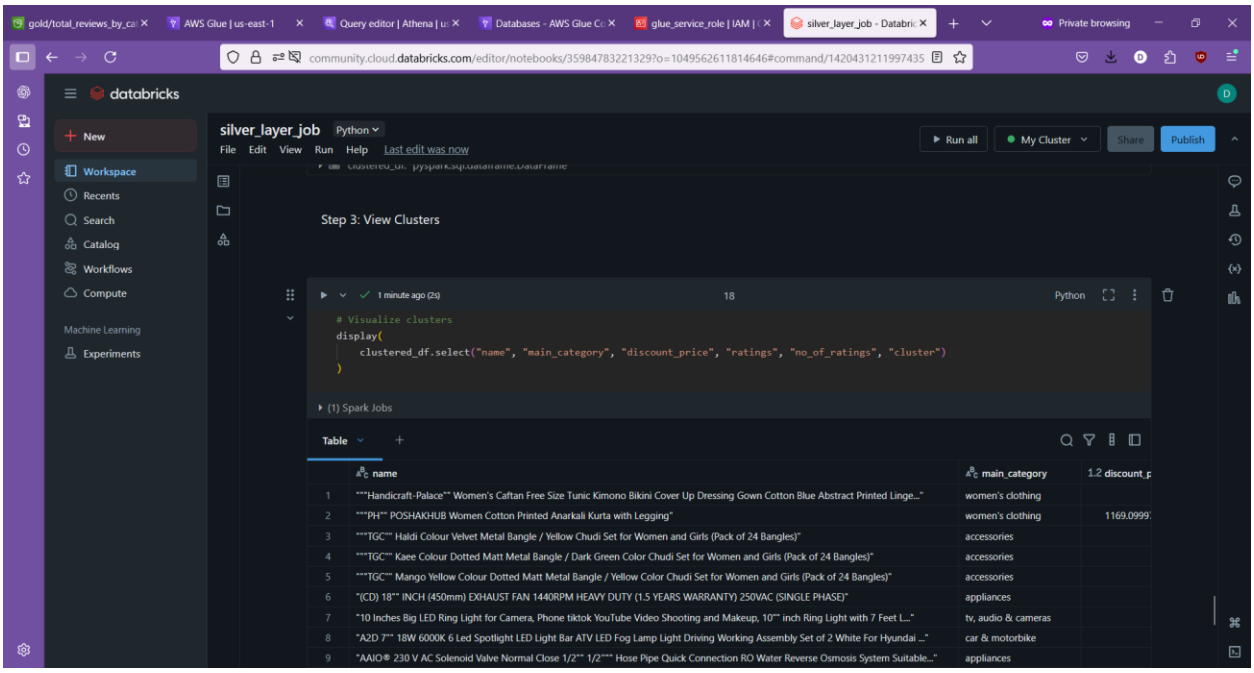


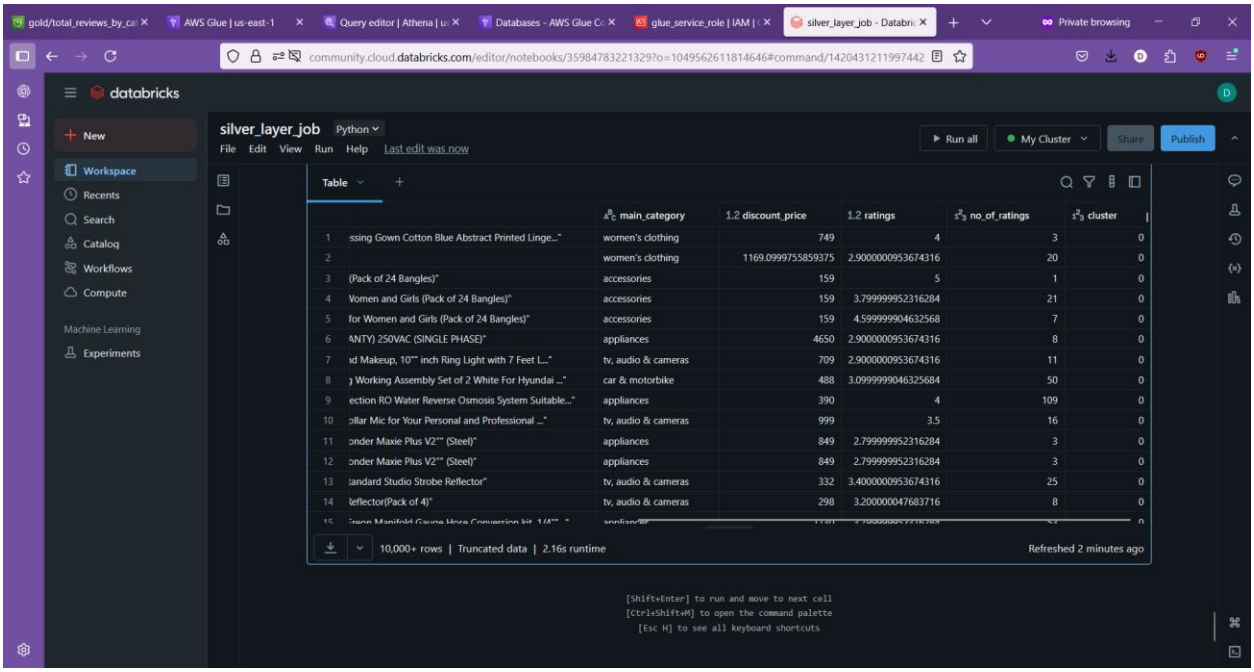
3. K-Means Clustering in Databricks (PySpark MLlib)

Step 1: Prepare the Data


silver_layer_job
Python
File Edit View Run Help Last edit was now
Run all
My Cluster
Share
Publish
10,000+ rows | Truncated data | 33.08s runtime
Refreshed 8 minutes ago
K-Means Clustering in Databricks (PySpark MLlib)
Markdown
Step 1: Load & Prepare the Data
+ Code + Text
1 minute ago (1s)
14
from pyspark.sql.functions import col, regexp_replace
from pyspark.ml.feature import VectorAssembler
from pyspark.ml.clustering import KMeans
from pyspark.ml import Pipeline
# Load from Silver layer
df = spark.read.format("delta").load("s3a://nehabc002bucket/silver/amazon-products/")
# Clean and cast columns
df = (
df.withColumn("discount_price", regexp_replace("discount_price", "[^0-9.]", "").cast("float"))
.withColumn("actual_price", regexp_replace("actual_price", "[^0-9.]", "").cast("float"))
.withColumn("ratings", col("ratings").cast("float"))
.withColumn("no_of_ratings", regexp_replace("no_of_ratings", ",", "").cast("int"))
.dropna(subset=["discount_price", "actual_price", "ratings", "no_of_ratings"])
)


silver_layer_job
Python
File Edit View Run Help Last edit was now
Run all
My Cluster
Share
Publish
df: pyspark.sql.dataframe.DataFrame = [row_id: string, name: string ... 8 more fields]
Step 2: Assemble Features & Fit KMeans Model
1 minute ago (19s)
16
Python
# Step 2: Assemble numeric features
feature_cols = ["discount_price", "actual_price", "ratings", "no_of_ratings"]
assembler = VectorAssembler(inputCols=feature_cols, outputCol="features")
# Step 3: Define and fit the KMeans model
kmeans = KMeans(featuresCol="features", predictionCol="cluster", k=4, seed=42)
# Step 4: Build and run the pipeline
pipeline = Pipeline(stages=[assembler, kmeans])
model = pipeline.fit(df)
# Step 5: Transform the data to get cluster predictions
clustered_df = model.transform(df)
(19) Spark Jobs
clustered_df: pyspark.sql.dataframe.DataFrame
Step 3: View Clusters





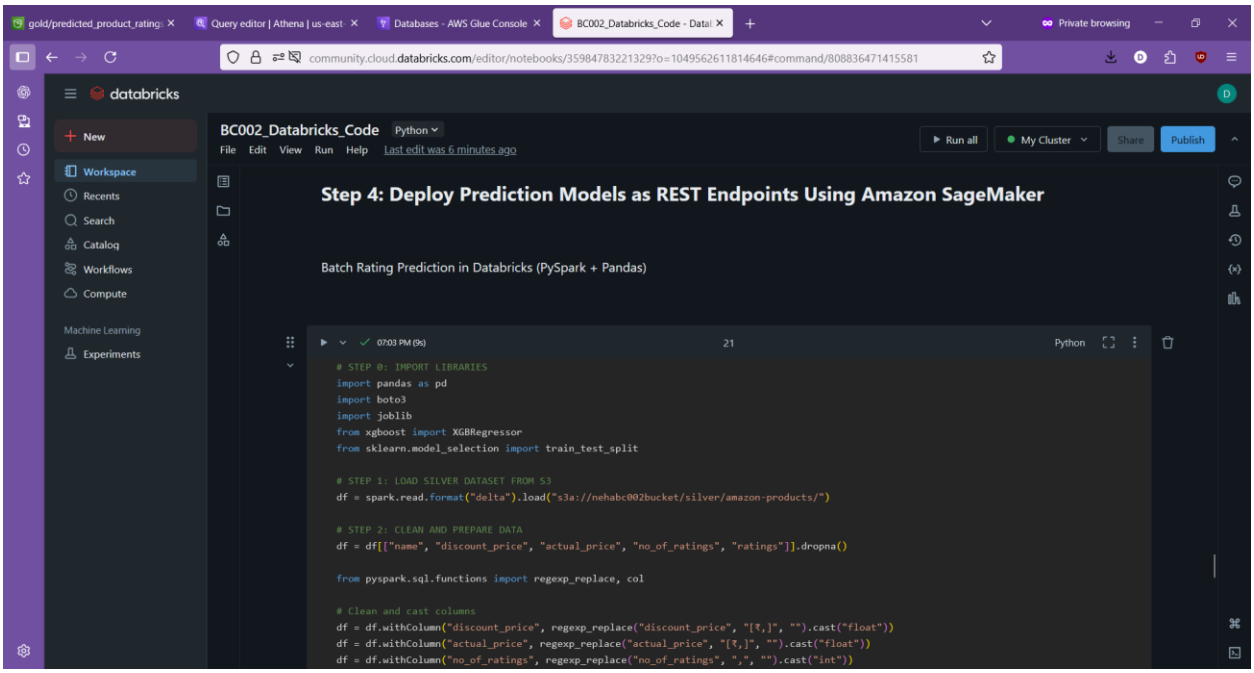
Databricks Code:

https://community.cloud.databricks.com/editor/notebooks/35984783221329?o=1049562611814646#notebook/35984783221329/command/35984783221330

4. Batch Inference in Databricks

Goal:

Predict Ratings for Products (Batch Inference)



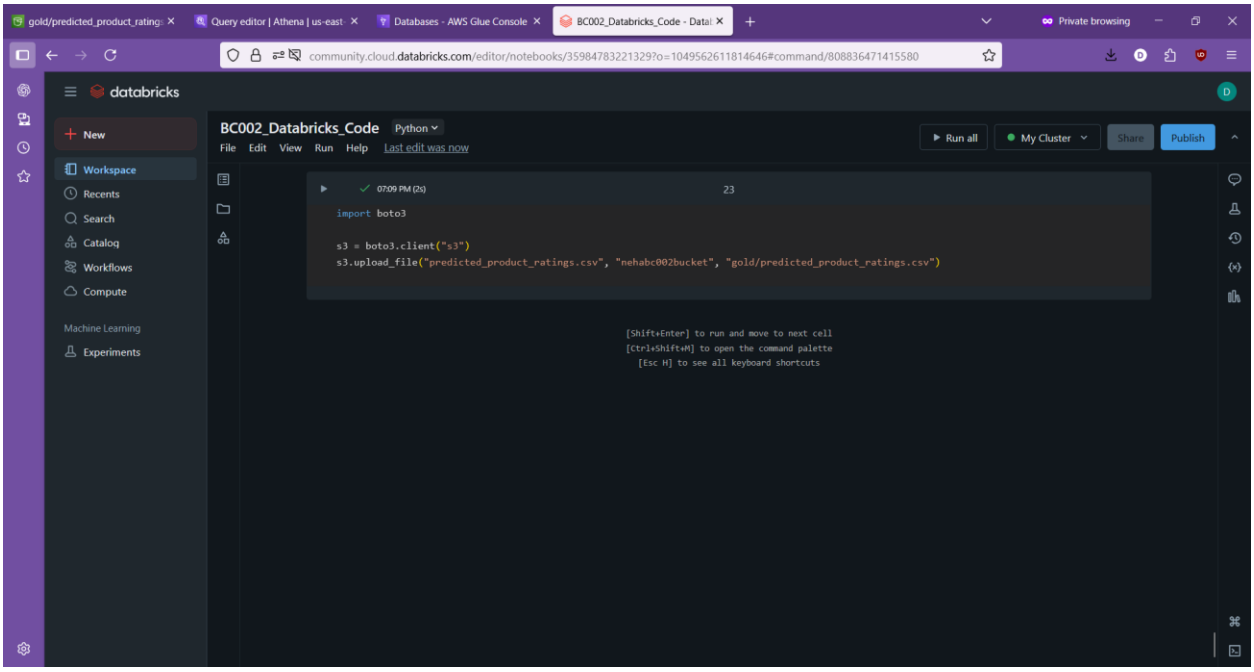
Code:

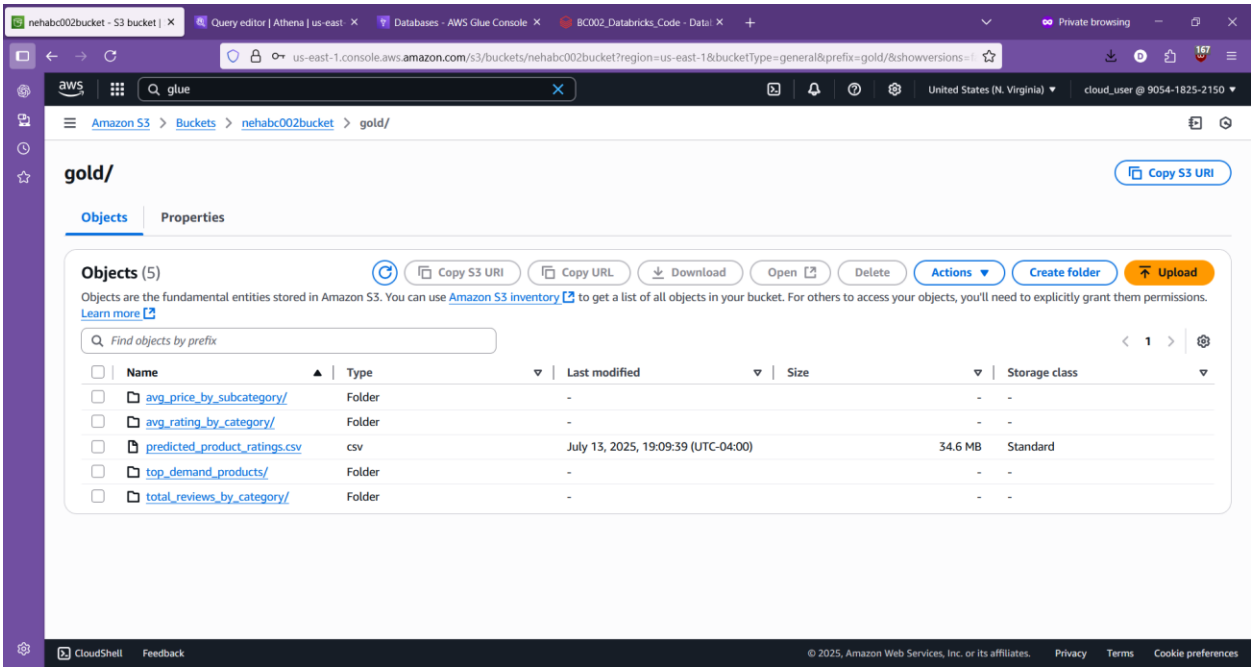
```
# STEP 0: IMPORT LIBRARIES
import pandas as pd
import boto3
import joblib
from xgboost import XGBRegressor
from sklearn.model_selection import train_test_split

# STEP 1: LOAD SILVER DATASET FROM S3
df = spark.read.format("delta").load("s3a://nehabc002bucket/silver/amazon-products/")

# STEP 2: CLEAN AND PREPARE DATA
df = df[["name", "discount_price", "actual_price", "no_of_ratings", "ratings"]].dropna()
```

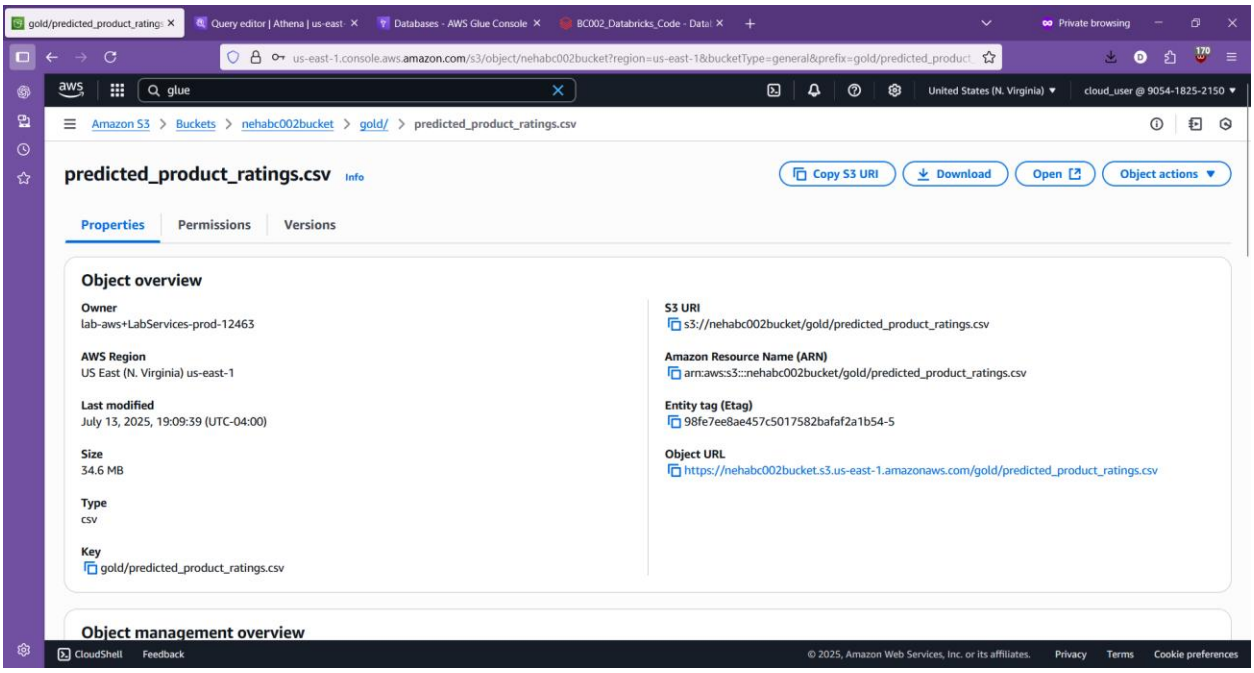
```
from pyspark.sql.functions import regexp_replace, col

# Clean and cast columns
df = df.withColumn("discount_price", regexp_replace("discount_price", "[₹,]", "").cast("float"))
df = df.withColumn("actual_price", regexp_replace("actual_price", "[₹,]", "").cast("float"))
df = df.withColumn("no_of_ratings", regexp_replace("no_of_ratings", ",", "").cast("int"))
df = df.withColumn("ratings", col("ratings").cast("float"))



# Convert to Pandas
df_pd = df.select("name", "discount_price", "actual_price", "no_of_ratings",
"ratings").dropna().toPandas()

# Prepare features and labels from Pandas
X = df_pd[["discount_price", "actual_price", "no_of_ratings"]]
y = df_pd["ratings"]

# Train/test split and model training
from sklearn.model_selection import train_test_split
from xgboost import XGBRegressor

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = XGBRegressor()
model.fit(X_train, y_train)

# Predict
df_pd["predicted_rating"] = model.predict(X)
```

Save result

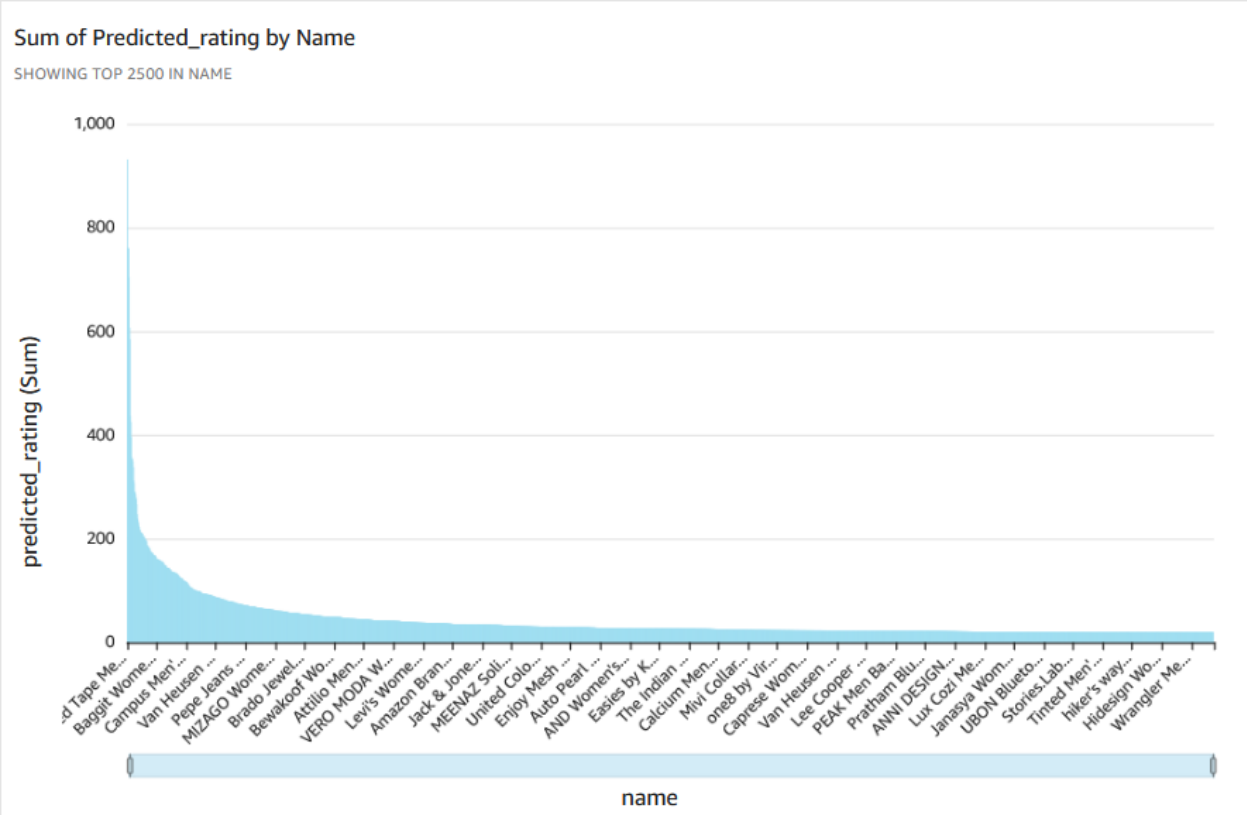
df_pd[["name", "discount_price", "actual_price", "no_of_ratings", "predicted_rating"]].to_csv("predicted_product_ratings.csv", index=False)






predicted_product_ratings.csv
Info
Copy S3 URI
Download
Open
Object actions
Properties
Permissions
Versions
Object overview
Owner
lab-aws+LabServices-prod-12463
AWS Region
US East (N. Virginia) us-east-1
Last modified
July 13, 2025, 19:09:39 (UTC-04:00)
Size
34.6 MB
Type
csv
Key
gold/predicted_product_ratings.csv
S3 URI
s3://nehabc002bucket/gold/predicted_product_ratings.csv
Amazon Resource Name (ARN)
arn:aws:s3:::nehabc002bucket/gold/predicted_product_ratings.csv
Entity tag (Etag)
98fe7ee8ae457c5017582bafaf2a1b54-5
Object URL
https://nehabc002bucket.s3.us-east-1.amazonaws.com/gold/predicted_product_ratings.csv
Object management overview

Quicksight


Sum of Predicted_rating by Name
SHOWING TOP 2500 IN NAME
predicted_rating (Sum)
1,000
800
600
400
200
0
name