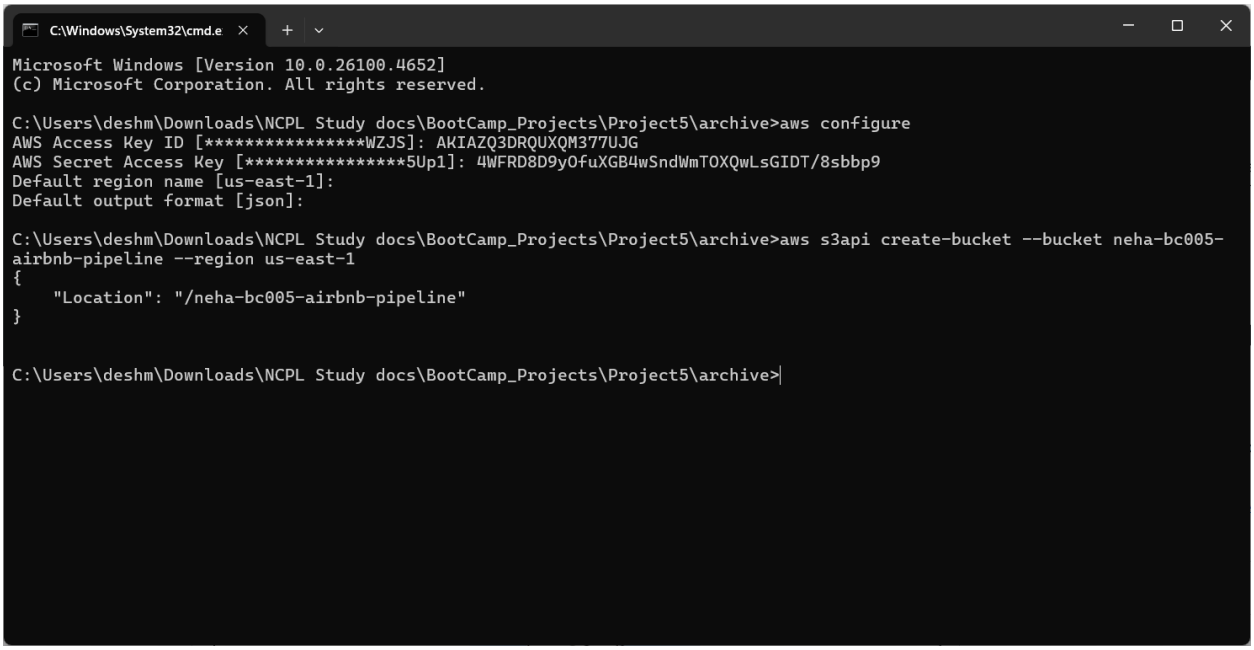
Building a Scalable Data Pipeline for Airbnb Listings Analytics in NYC

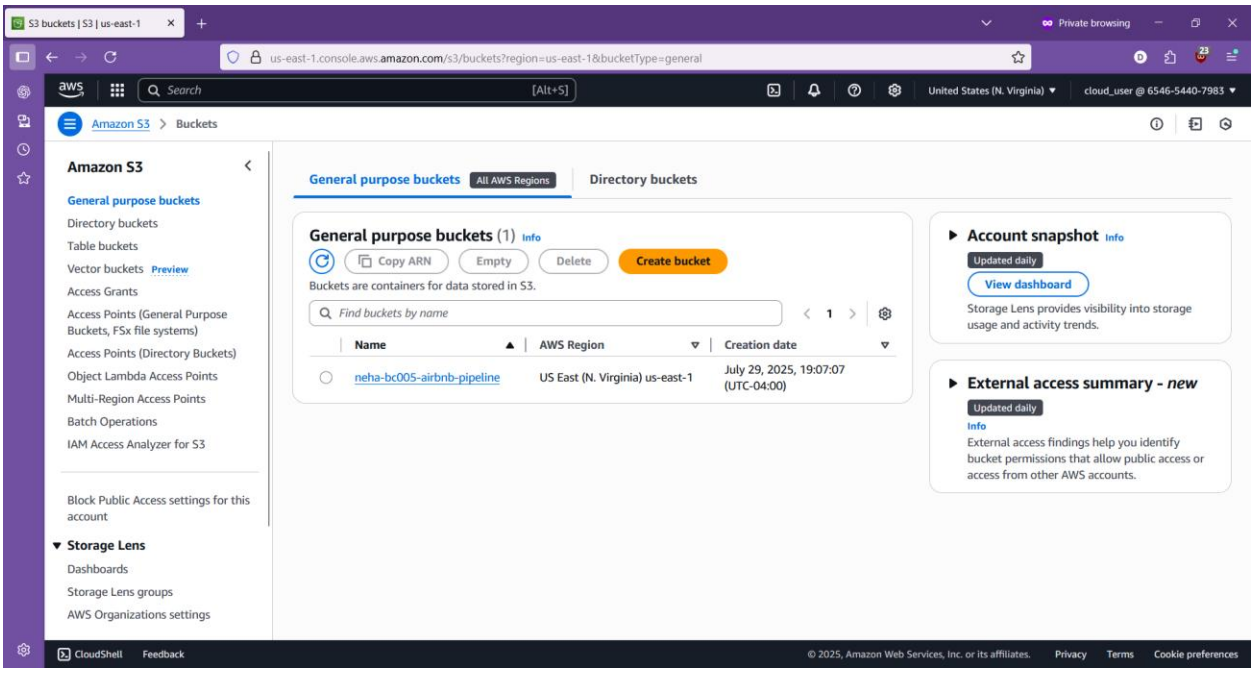
[Airbnb Listings Data Pipeline- Business-Focused Data Pipeline for Hospitality Insights]

[Bootcamp- AWS Data Engineering Project- 5]

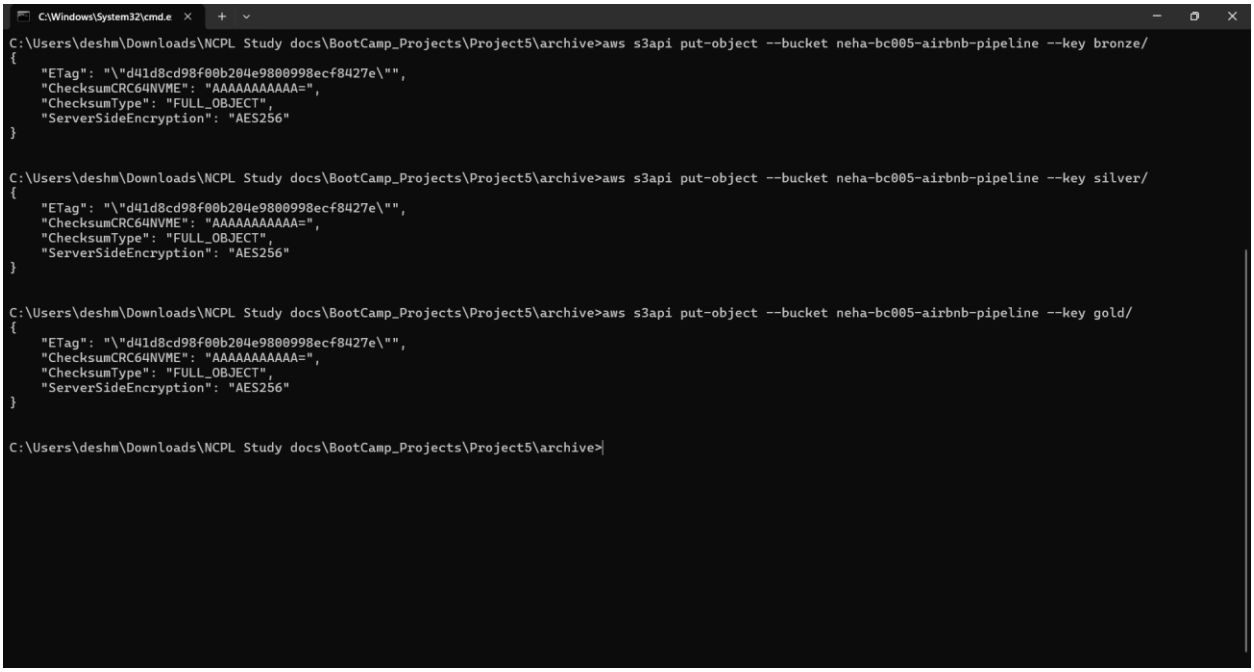
1. Create an S3 Bucket Using CLI
   Command: aws s3api create-bucket --bucket neha-bc005-airbnb-pipeline --region us-east-1



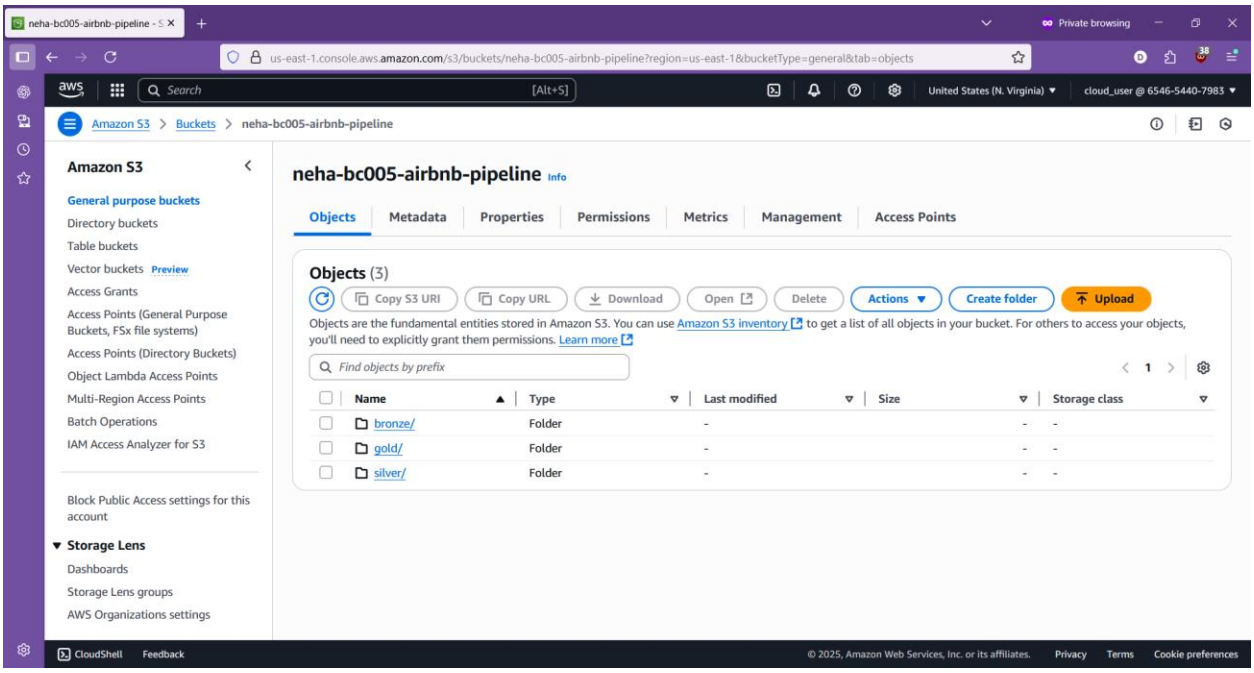
2. Configure S3 in AWS



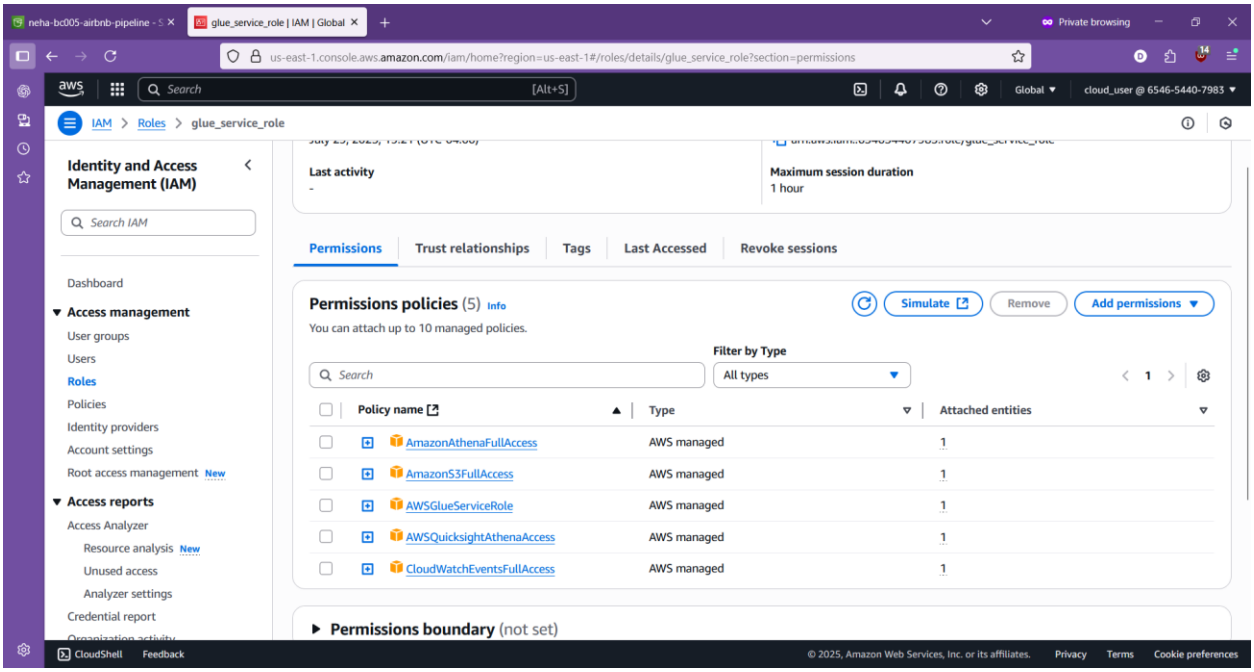
3. Create sub folders in S3
   Commands:
   aws s3api put-object --bucket neha-bc005-airbnb-pipeline --key bronze/
   aws s3api put-object --bucket neha-bc005-airbnb-pipeline --key silver/
   aws s3api put-object --bucket neha-bc005-airbnb-pipeline --key gold/



4. Configure in S3



5. Create IAM Role for Glue

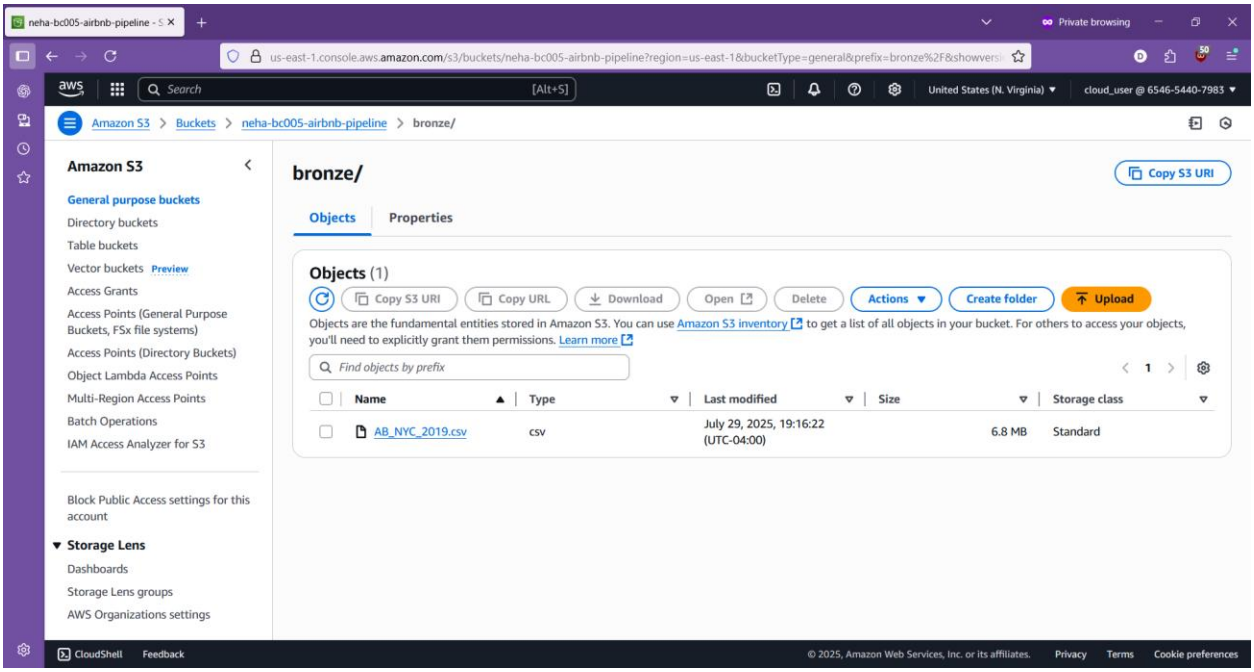


Bronze Layer

1. Upload airbnb_nyc.csv to S3 (Bronze Layer) through CLI
   Command: aws s3 cp AB_NYC_2019.csv s3://neha-bc005-airbnb-pipeline/bronze/

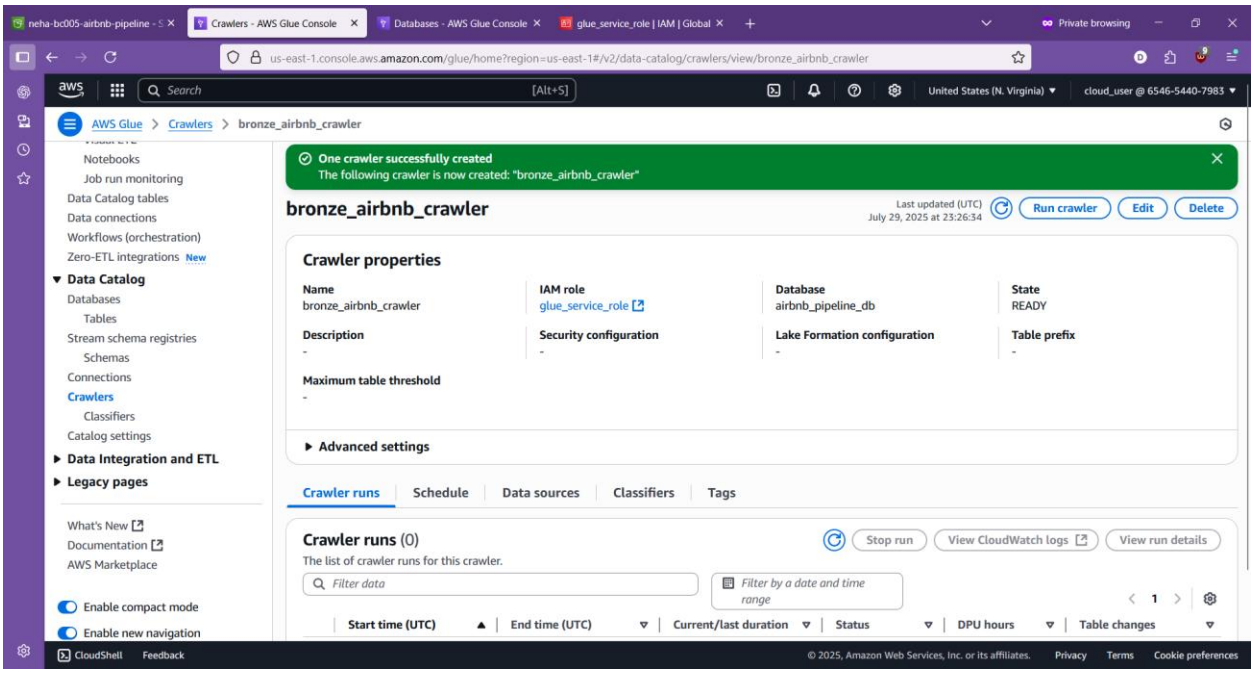
```
C:\Windows\System32\cmd.e
bnb-pipeline --key gold/
{
    "ETag": "\"d41d8cd98f00b204e9800998ecf8427e\"",
    "ChecksumCRC64NVME": "AAAAAAAAAAA=",
    "ChecksumType": "FULL_OBJECT",
    "ServerSideEncryption": "AES256"
}


C:\Users\deshm\Downloads\NCPL Study docs\BootCamp_Projects\Project5\archive>dir
 Volume in drive C is OS
 Volume Serial Number is D681-949E

 Directory of C:\Users\deshm\Downloads\NCPL Study docs\BootCamp_Projects\Project5\archive

2025-07-29  08:39 AM    <DIR>          .
2025-07-29  08:39 AM    <DIR>          ..
2025-07-29  08:39 AM         7,077,973 AB_NYC_2019.csv
2025-07-29  08:39 AM           192,340 New_York_City_.png
               2 File(s)      7,270,313 bytes
               2 Dir(s)  686,856,646,656 bytes free

C:\Users\deshm\Downloads\NCPL Study docs\BootCamp_Projects\Project5\archive>cd
C:\Users\deshm\Downloads\NCPL Study docs\BootCamp_Projects\Project5\archive

C:\Users\deshm\Downloads\NCPL Study docs\BootCamp_Projects\Project5\archive>aws s3 cp AB_NYC_2019.csv s3://neha-bc005-ai
rbnb-pipeline/bronze/
upload: .\AB_NYC_2019.csv to s3://neha-bc005-airbnb-pipeline/bronze/AB_NYC_2019.csv

C:\Users\deshm\Downloads\NCPL Study docs\BootCamp_Projects\Project5\archive>
```

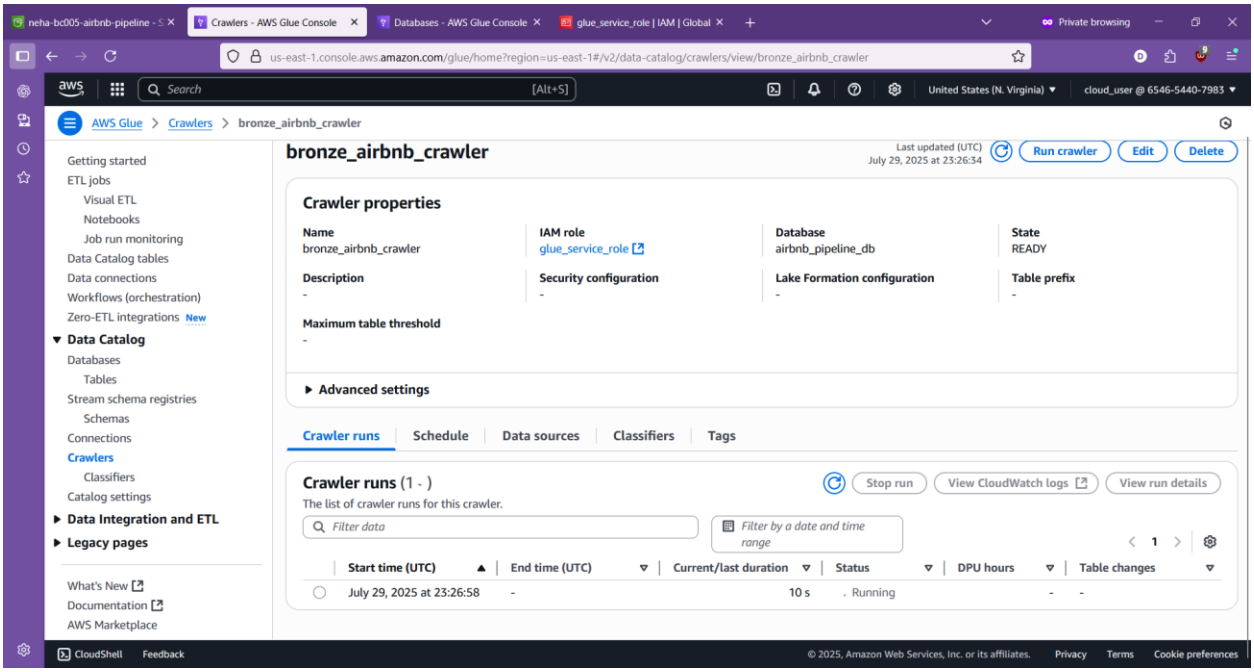
2. Configure in S3



Silver Layer - Clean Airbnb Data with AWS Glue

1. Create a Glue Crawler for Bronze Data





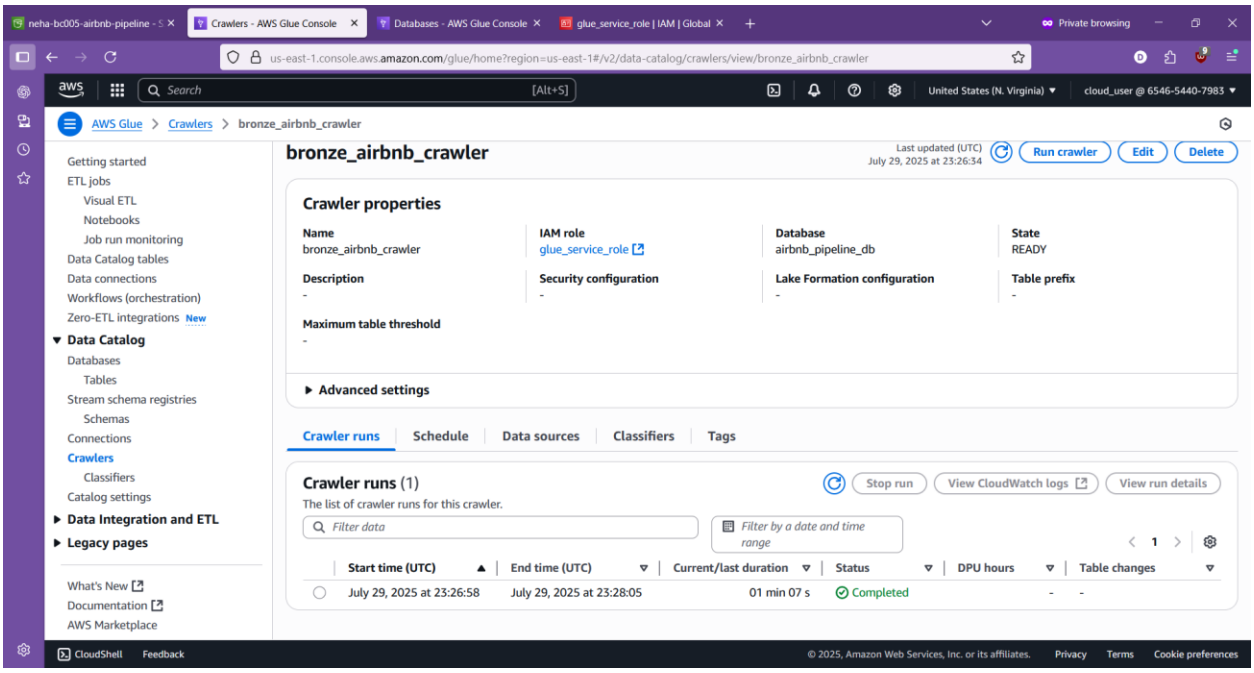
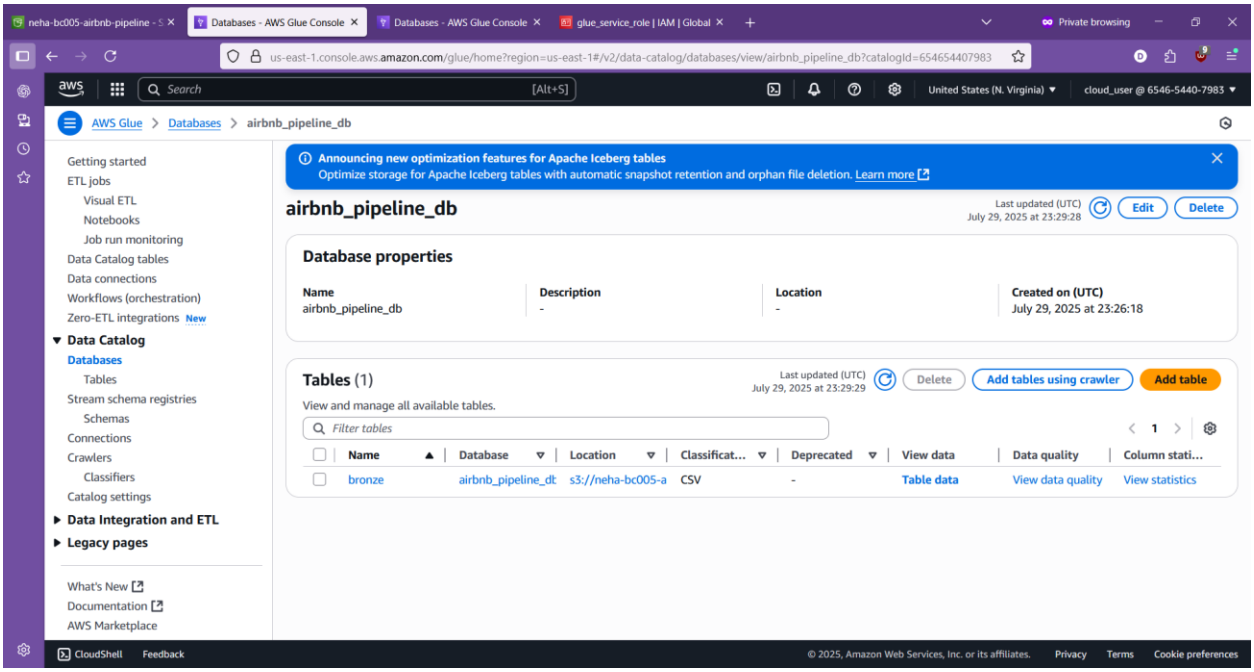
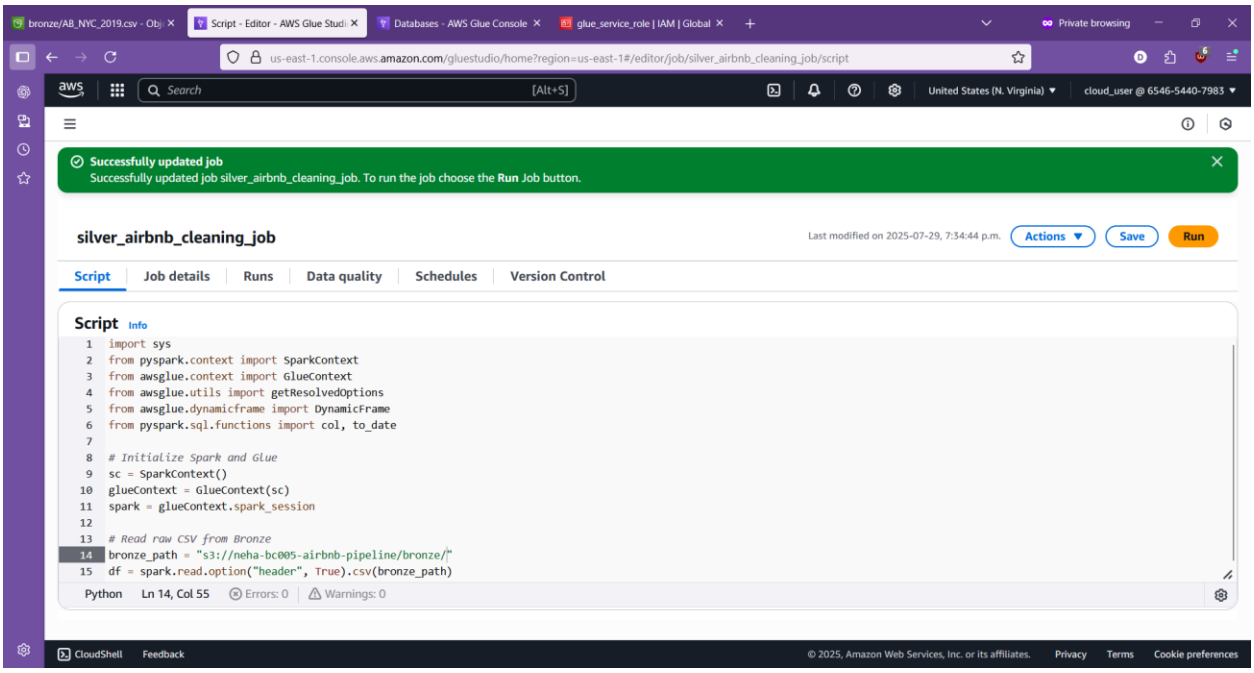
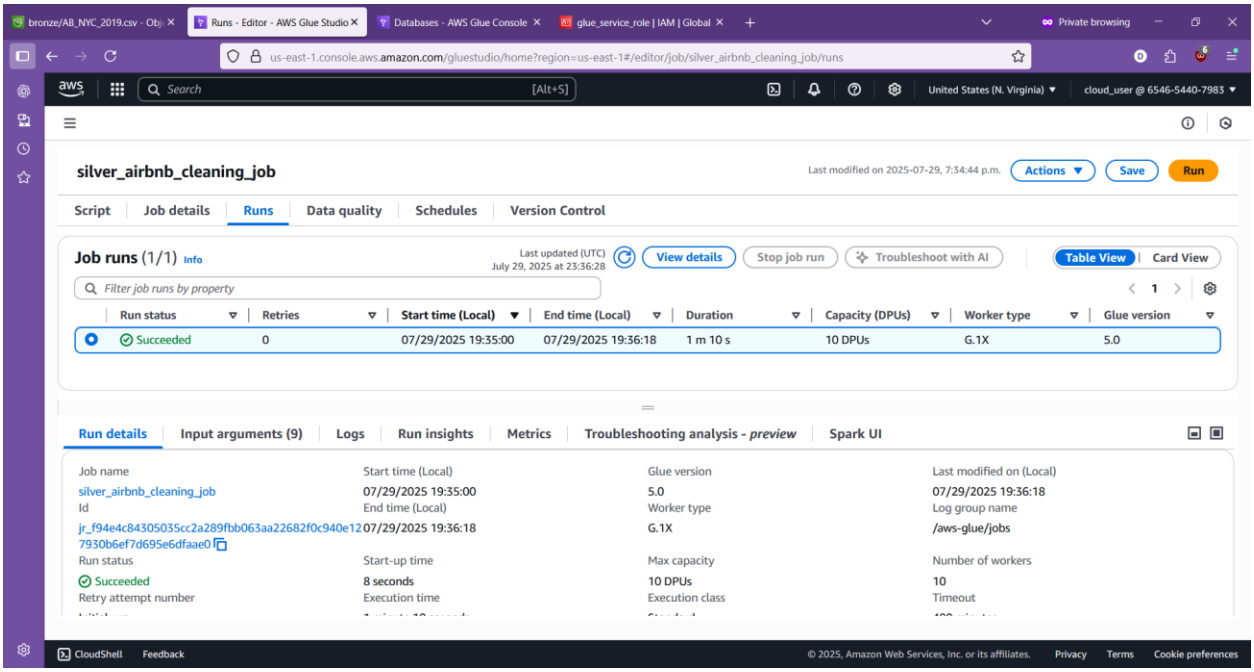


Table created:

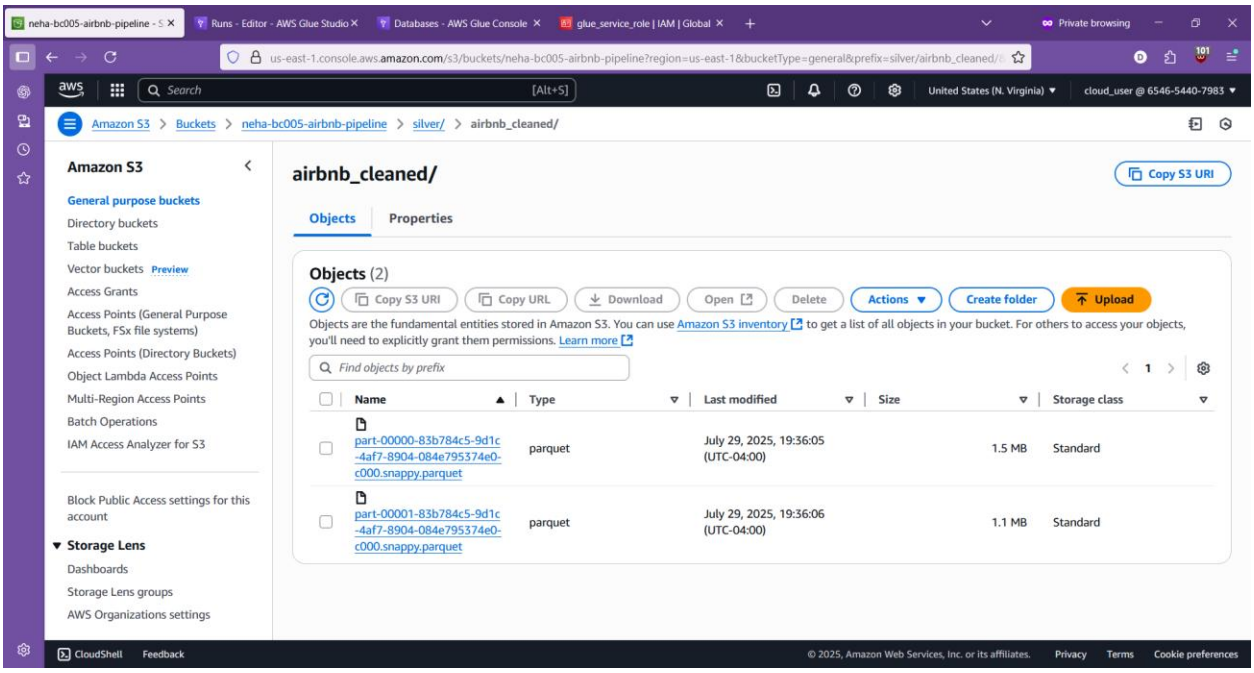


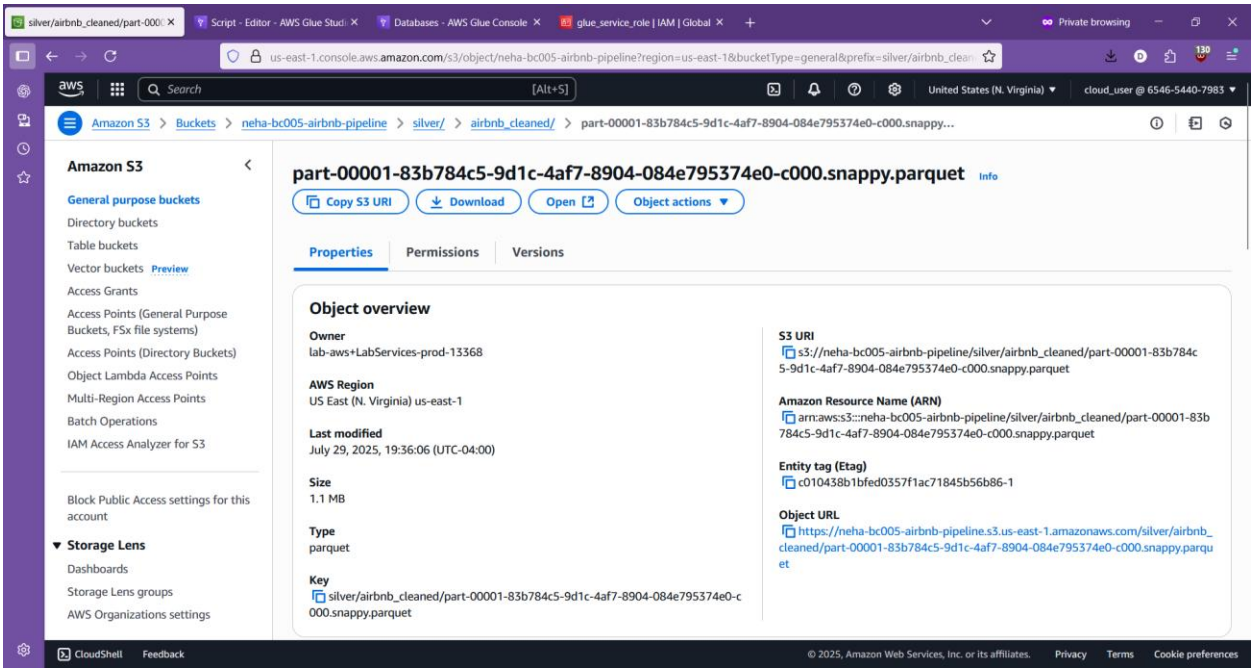
2. Create a Glue Job to Clean and Enrich Data



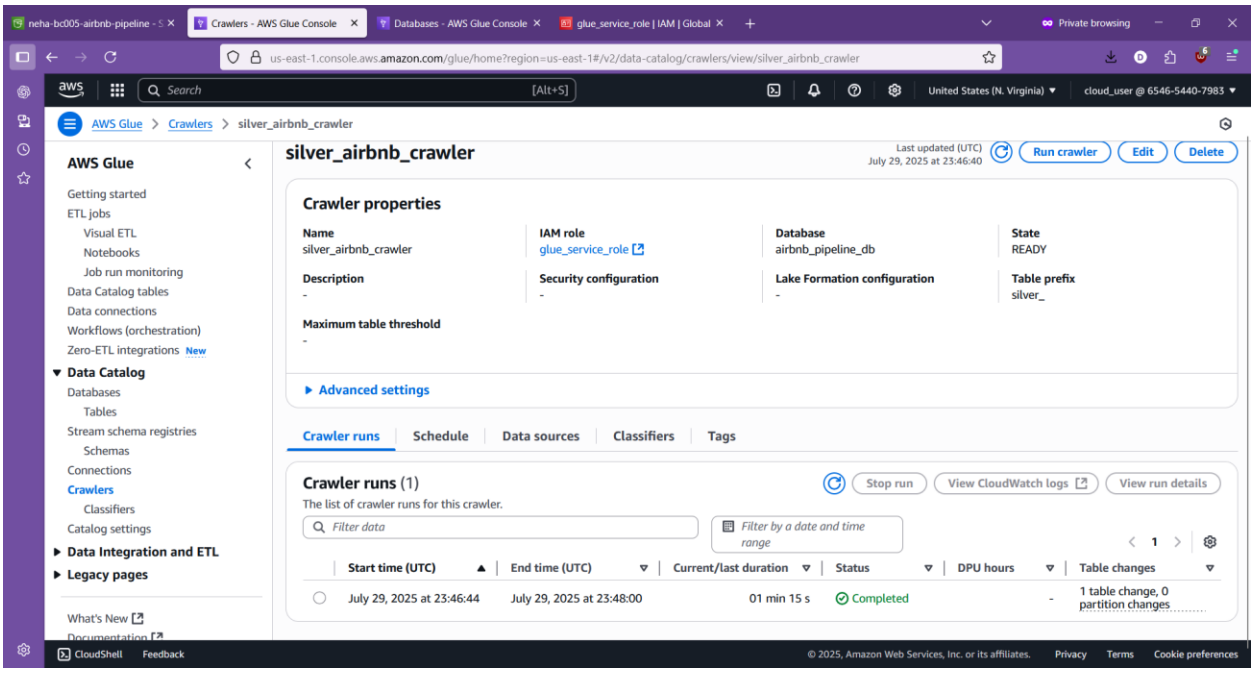


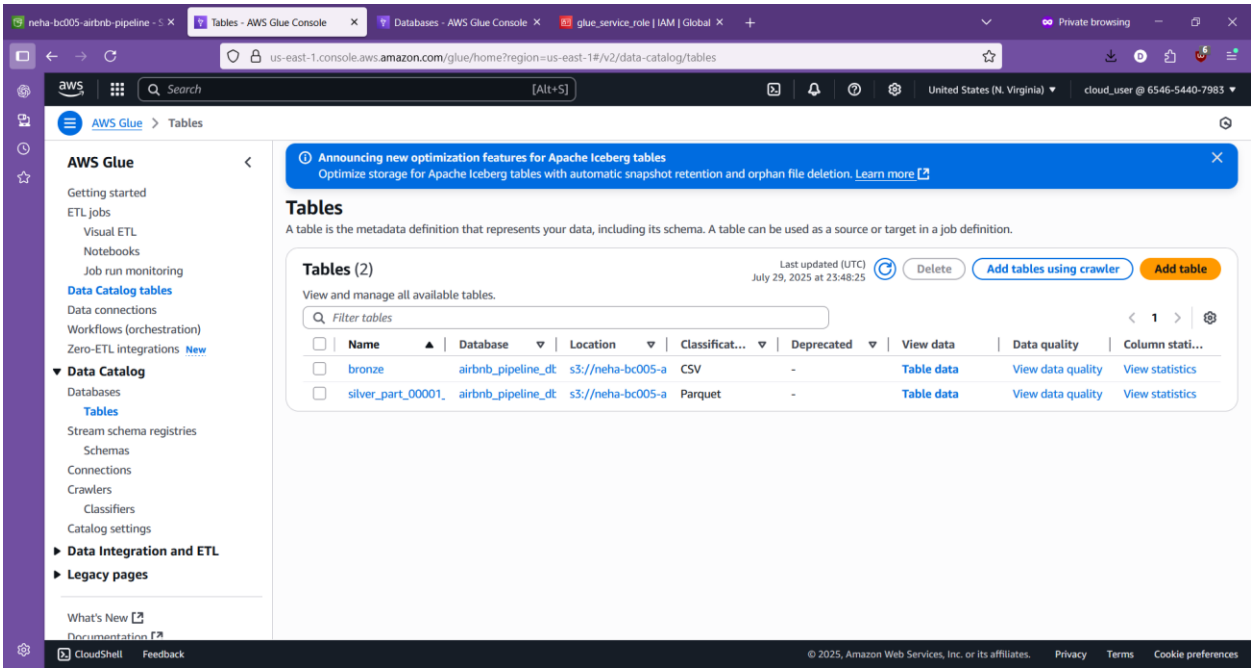
3. Verify output in S3

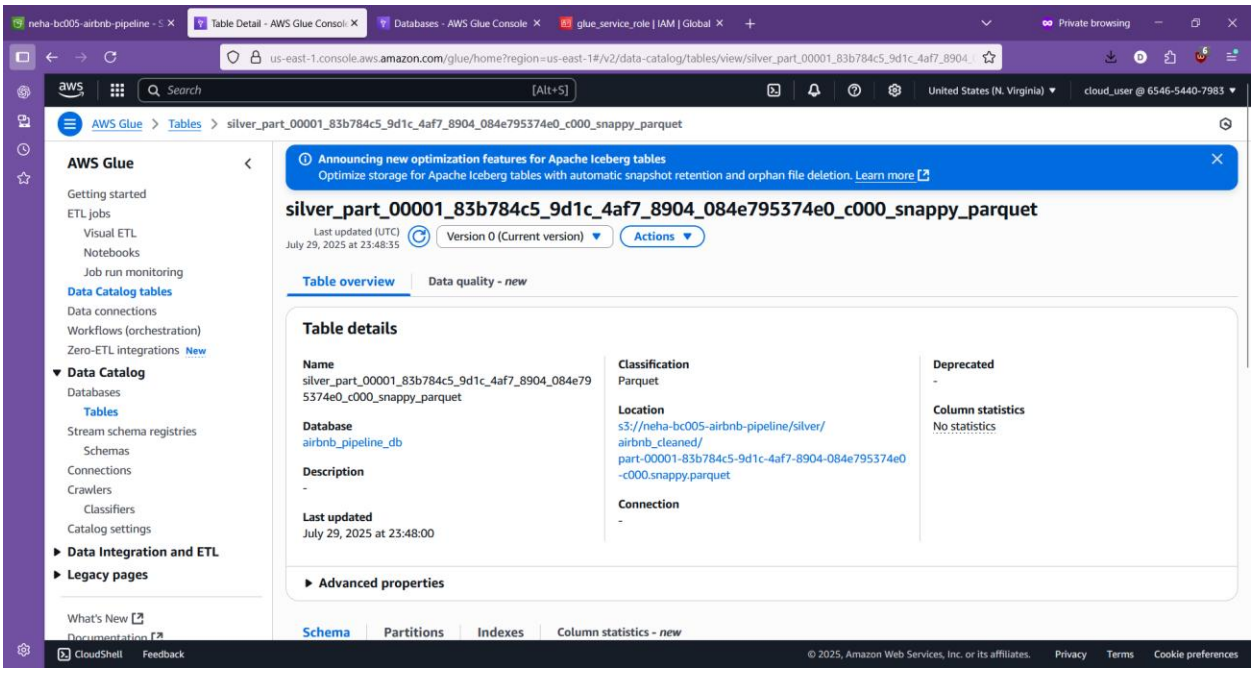


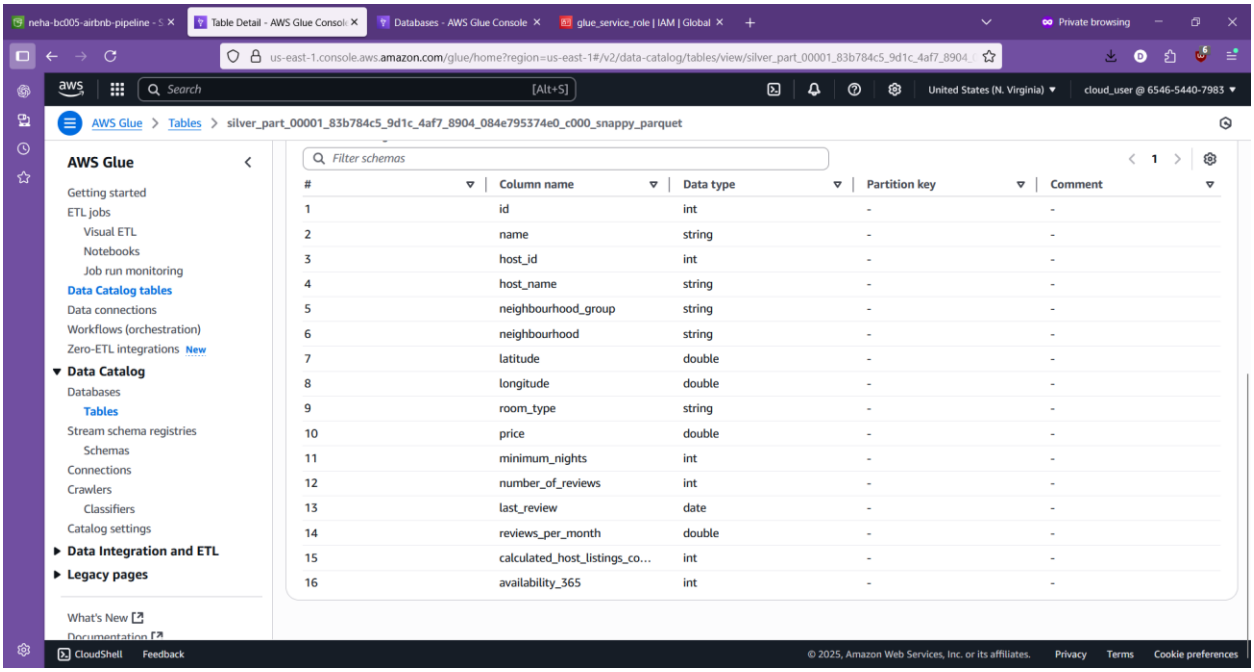


4. Crawler for silver data


Crawlers - AWS Glue Console
Databases - AWS Glue Console
glue_service_role | IAM | Global
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/crawlers/view/silver_airbnb_crawler
AWS Glue > Crawlers > silver_airbnb_crawler
silver_airbnb_crawler
Run crawler
Edit
Delete
Crawler properties
Name
silver_airbnb_crawler
IAM role
glue_service_role
Database
airbnb_pipeline_db
State
READY
Description
Security configuration
Lake Formation configuration
Table prefix
silver_
Maximum table threshold
Advanced settings
Crawler runs
Schedule
Data sources
Classifiers
Tags
Crawler runs (1)
Stop run
View CloudWatch logs
View run details
The list of crawler runs for this crawler.
Start time (UTC)
End time (UTC)
Current/last duration
Status
DPU hours
Table changes
July 29, 2025 at 23:46:44
July 29, 2025 at 23:48:00
01 min 15 s
Completed
1 table change, 0 partition changes


Tables - AWS Glue Console
Databases - AWS Glue Console
glue_service_role | IAM | Global
us-east-1.console.aws.amazon.com/glue/home?region=us-east-1#/v2/data-catalog/tables
AWS Glue > Tables
Announcing new optimization features for Apache Iceberg tables
Optimize storage for Apache Iceberg tables with automatic snapshot retention and orphan file deletion. Learn more
Tables
A table is the metadata definition that represents your data, including its schema. A table can be used as a source or target in a job definition.
Tables (2)
Delete
Add tables using crawler
Add table
View and manage all available tables.
Name
Database
Location
Classificat...
Deprecated
View data
Data quality
Column stati...
bronze
airbnb_pipeline_db
s3://neha-bc005-a
CSV
Table data
View data quality
View statistics
silver_part_00001_
airbnb_pipeline_db
s3://neha-bc005-a
Parquet
Table data
View data quality
View statistics



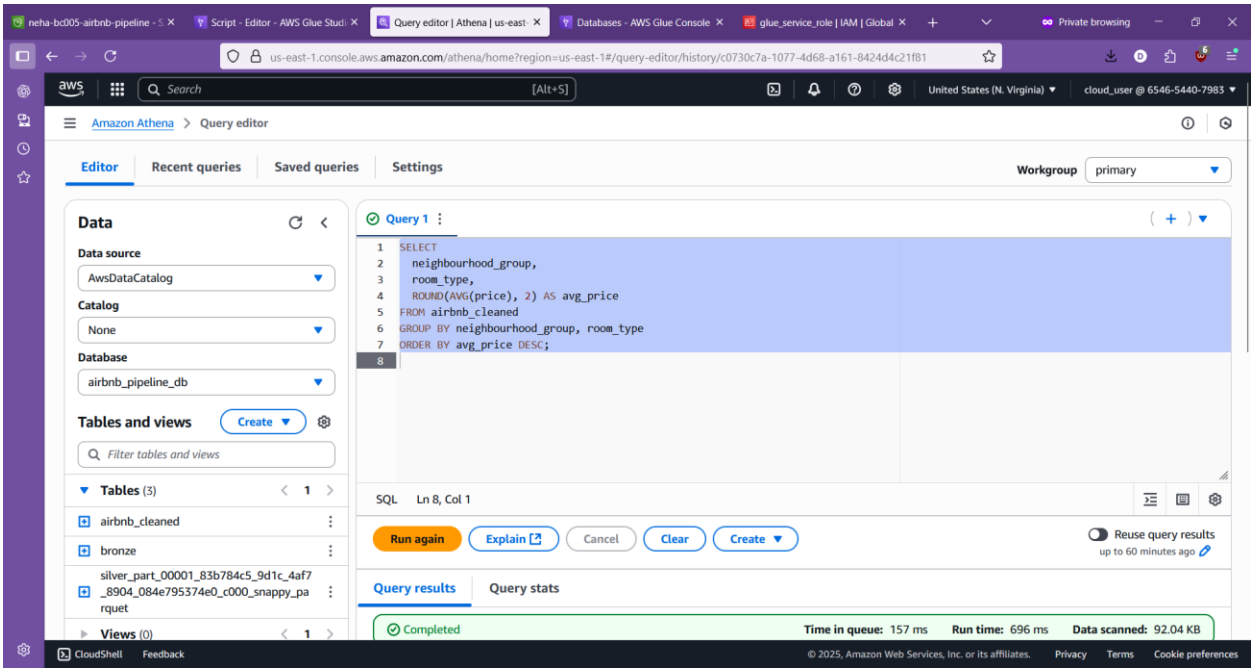


Athena Query: Average Price by Neighborhood and Room Type

Query:

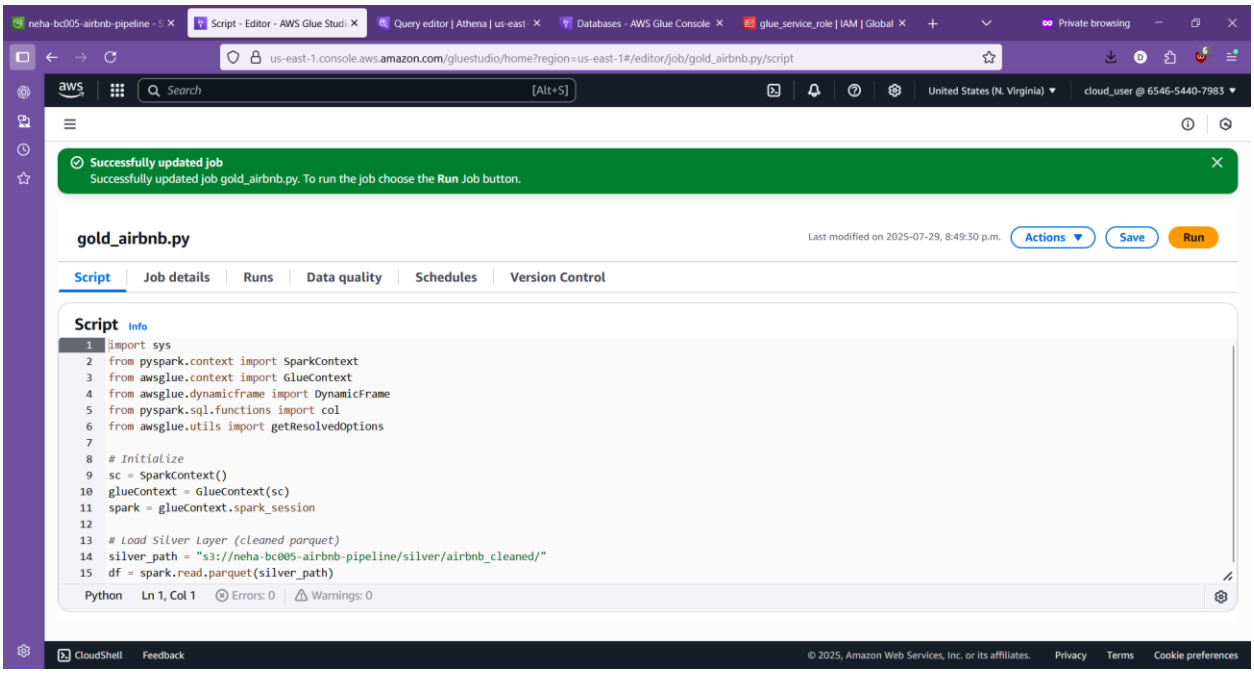
```
SELECT
  neighbourhood_group,
  room_type,
  ROUND(AVG(price), 2) AS avg_price
```

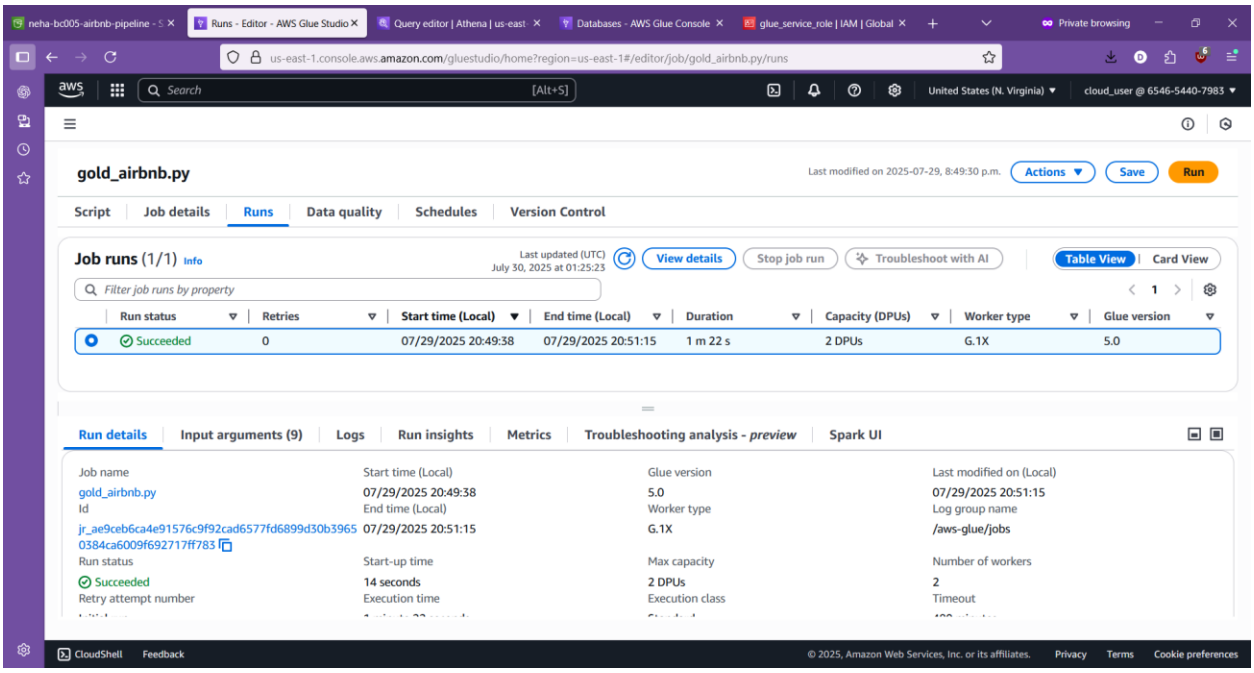
```
FROM airbnb_cleaned
GROUP BY neighbourhood_group, room_type
ORDER BY avg_price DESC;
```



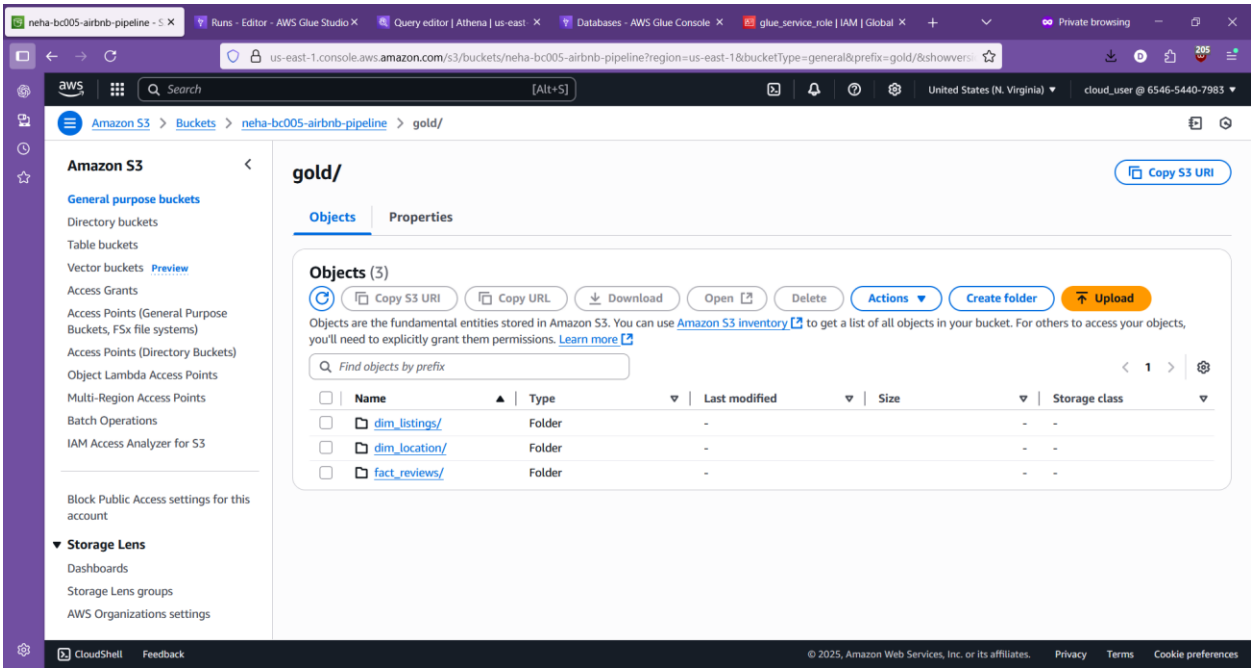
Gold Layer – Business-Ready Data

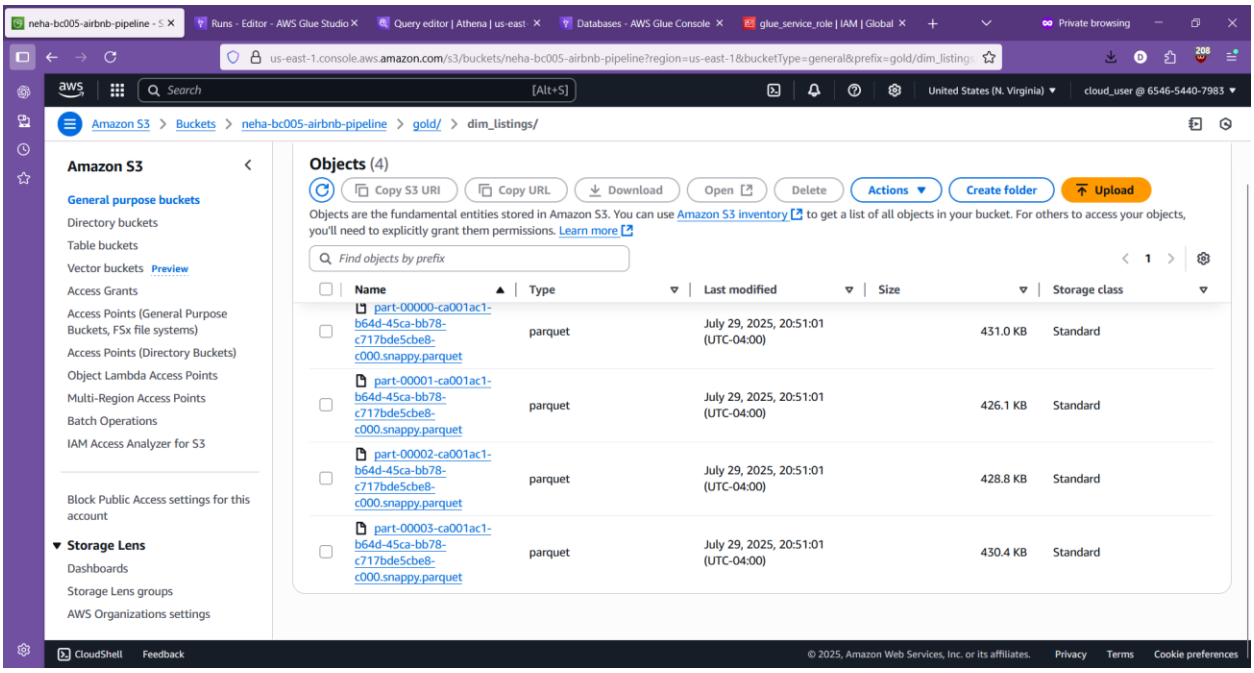
1. Create AWS Glue Job for Gold Layer

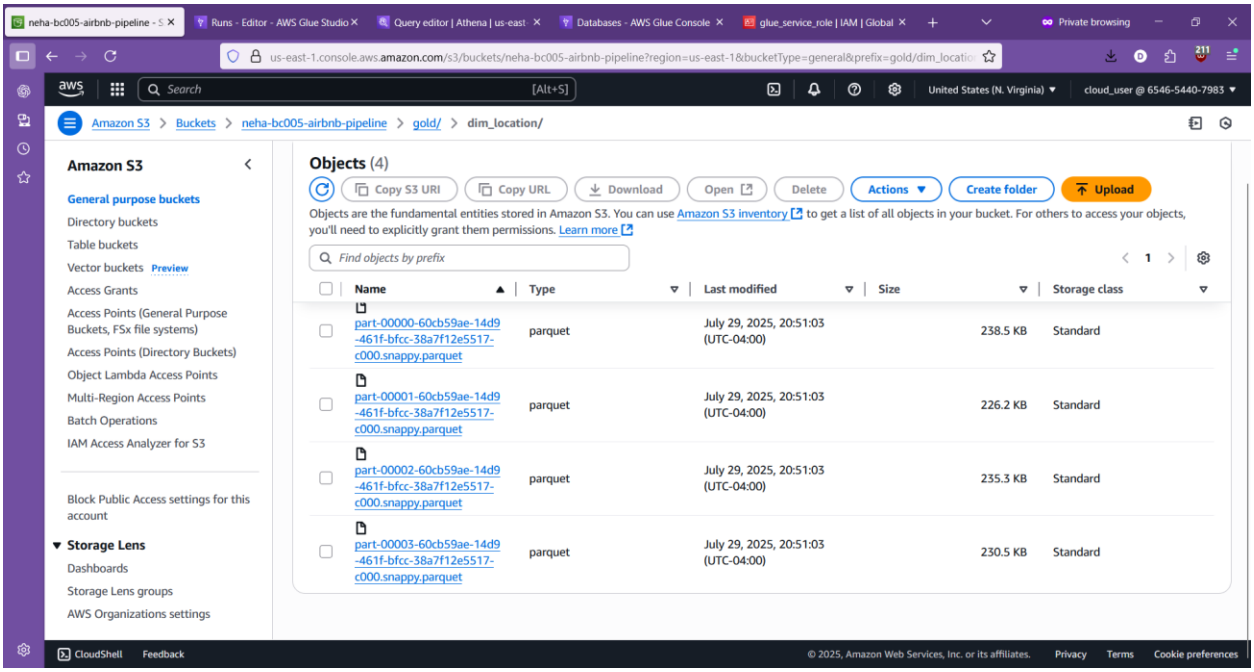


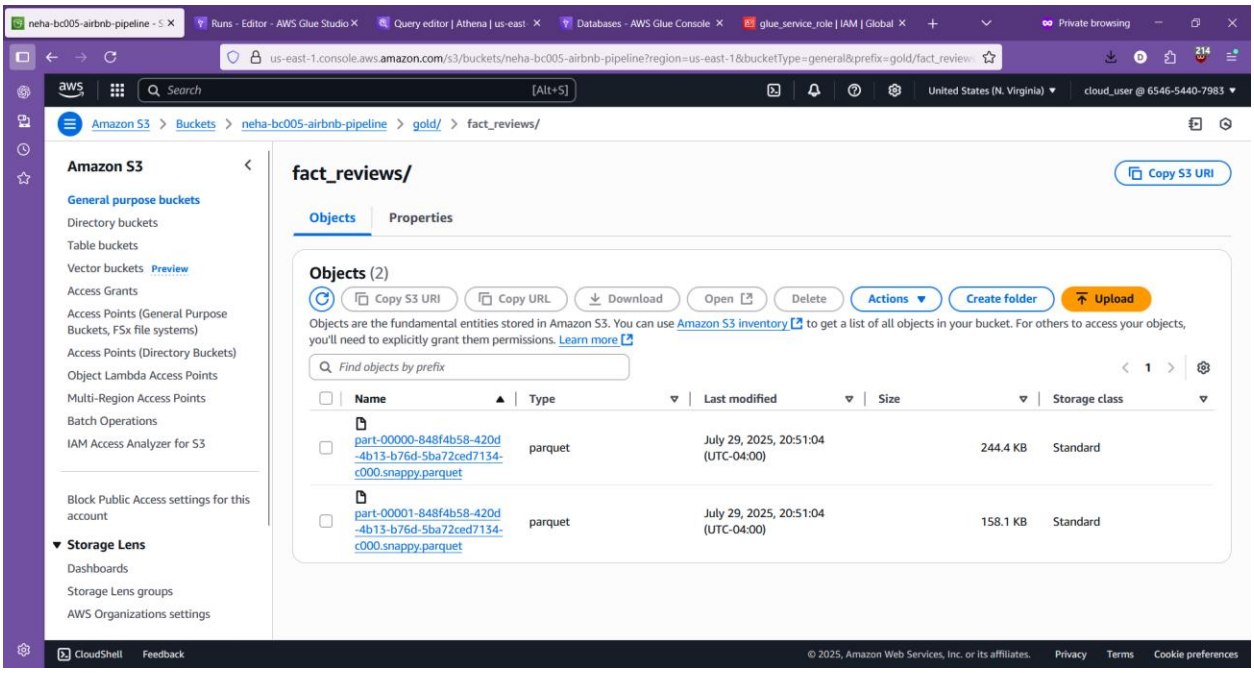


2. Verify the output in S3




Amazon S3 > Buckets > neha-bc005-airbnb-pipeline > gold/ > dim_listings/
Objects (4)
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
part-00000-ca001ac1-b64d-45ca-bb78-c717bde5cbe8-c000.snappy.parquet
parquet
July 29, 2025, 20:51:01 (UTC-04:00)
431.0 KB
Standard
part-00001-ca001ac1-b64d-45ca-bb78-c717bde5cbe8-c000.snappy.parquet
parquet
July 29, 2025, 20:51:01 (UTC-04:00)
426.1 KB
Standard
part-00002-ca001ac1-b64d-45ca-bb78-c717bde5cbe8-c000.snappy.parquet
parquet
July 29, 2025, 20:51:01 (UTC-04:00)
428.8 KB
Standard
part-00003-ca001ac1-b64d-45ca-bb78-c717bde5cbe8-c000.snappy.parquet
parquet
July 29, 2025, 20:51:01 (UTC-04:00)
430.4 KB
Standard


Amazon S3 > Buckets > neha-bc005-airbnb-pipeline > gold/ > dim_location/
Objects (4)
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
part-00000-60cb59ae-14d9-461f-bfcc-38a7f12e5517-c000.snappy.parquet
parquet
July 29, 2025, 20:51:03 (UTC-04:00)
238.5 KB
Standard
part-00001-60cb59ae-14d9-461f-bfcc-38a7f12e5517-c000.snappy.parquet
parquet
July 29, 2025, 20:51:03 (UTC-04:00)
226.2 KB
Standard
part-00002-60cb59ae-14d9-461f-bfcc-38a7f12e5517-c000.snappy.parquet
parquet
July 29, 2025, 20:51:03 (UTC-04:00)
235.3 KB
Standard
part-00003-60cb59ae-14d9-461f-bfcc-38a7f12e5517-c000.snappy.parquet
parquet
July 29, 2025, 20:51:03 (UTC-04:00)
230.5 KB
Standard

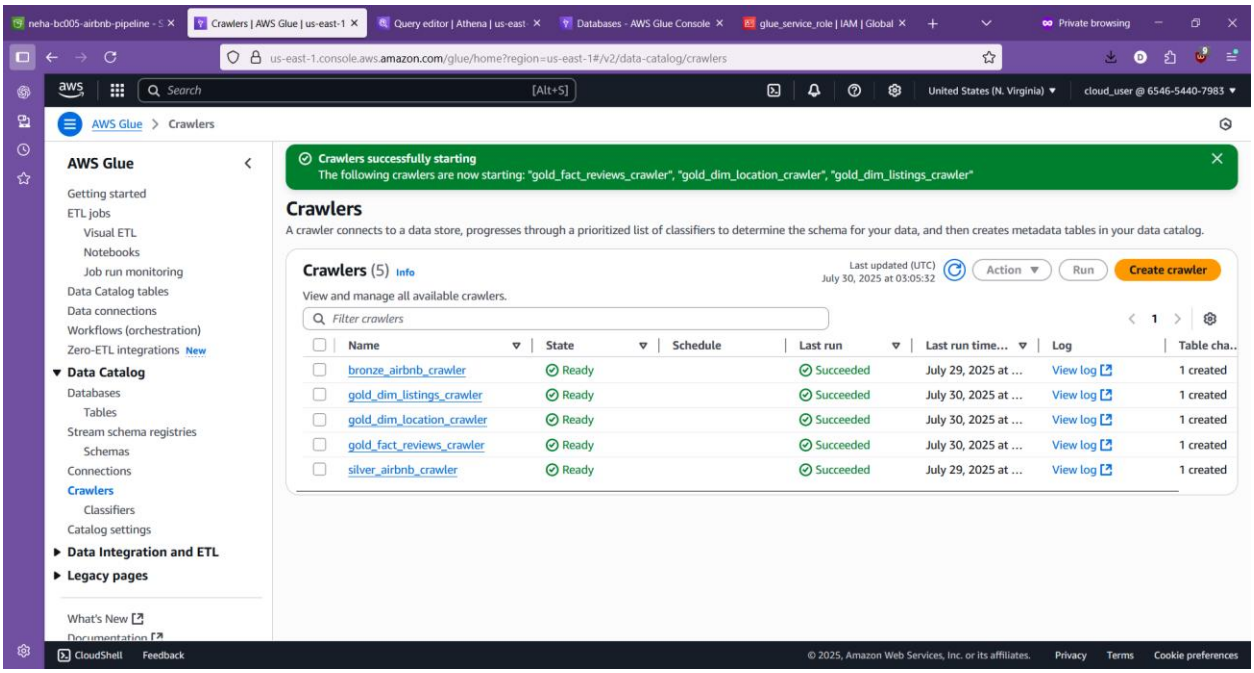


3. Create Crawlers for Gold Tables

There are 3 Gold folders:

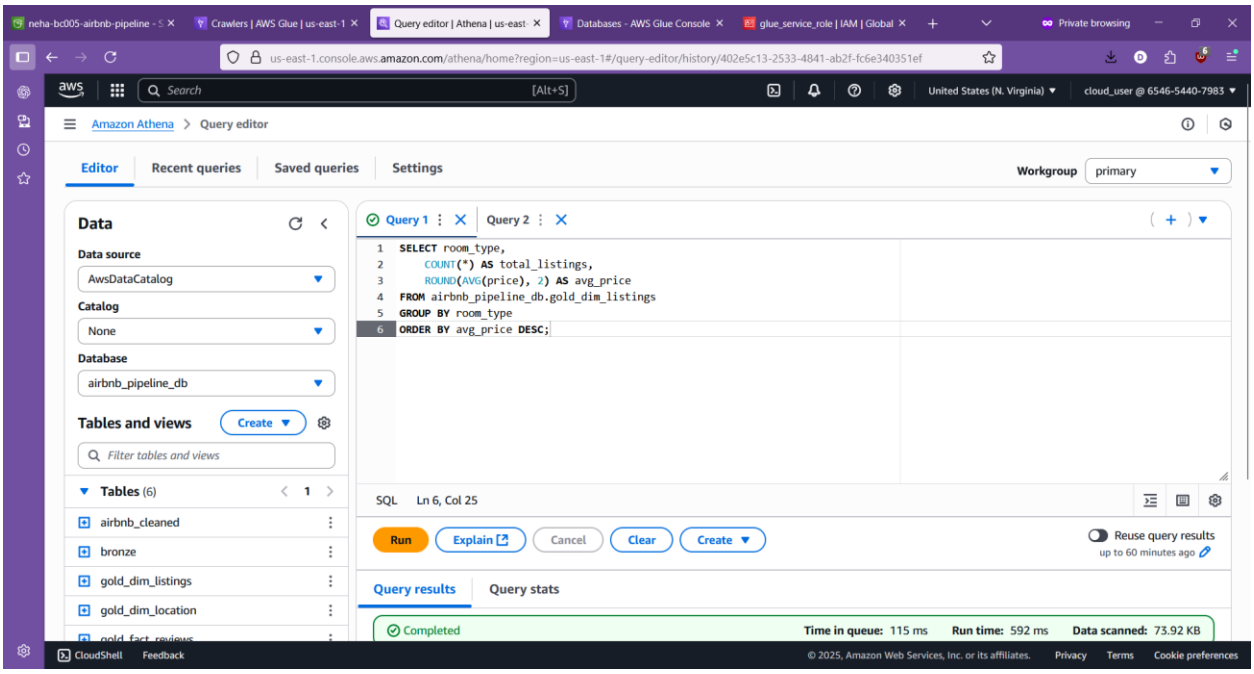
- gold/dim_listings/
- gold/dim_location/
- gold/fact_reviews/

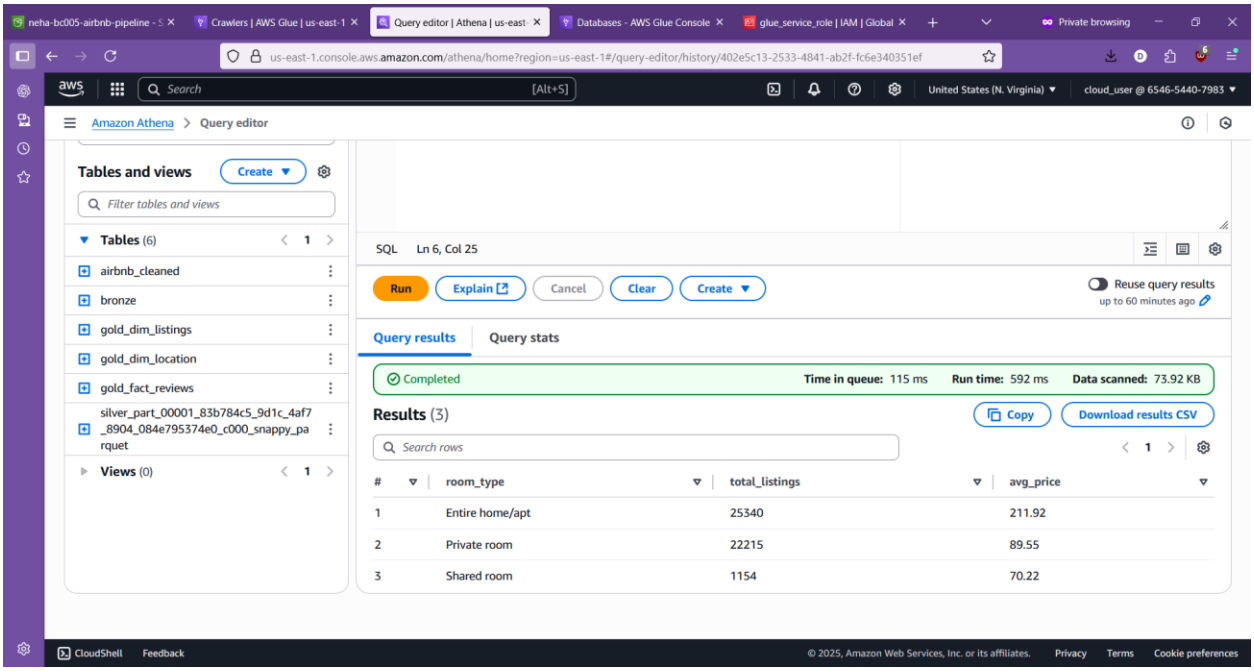
Create one crawler for each.



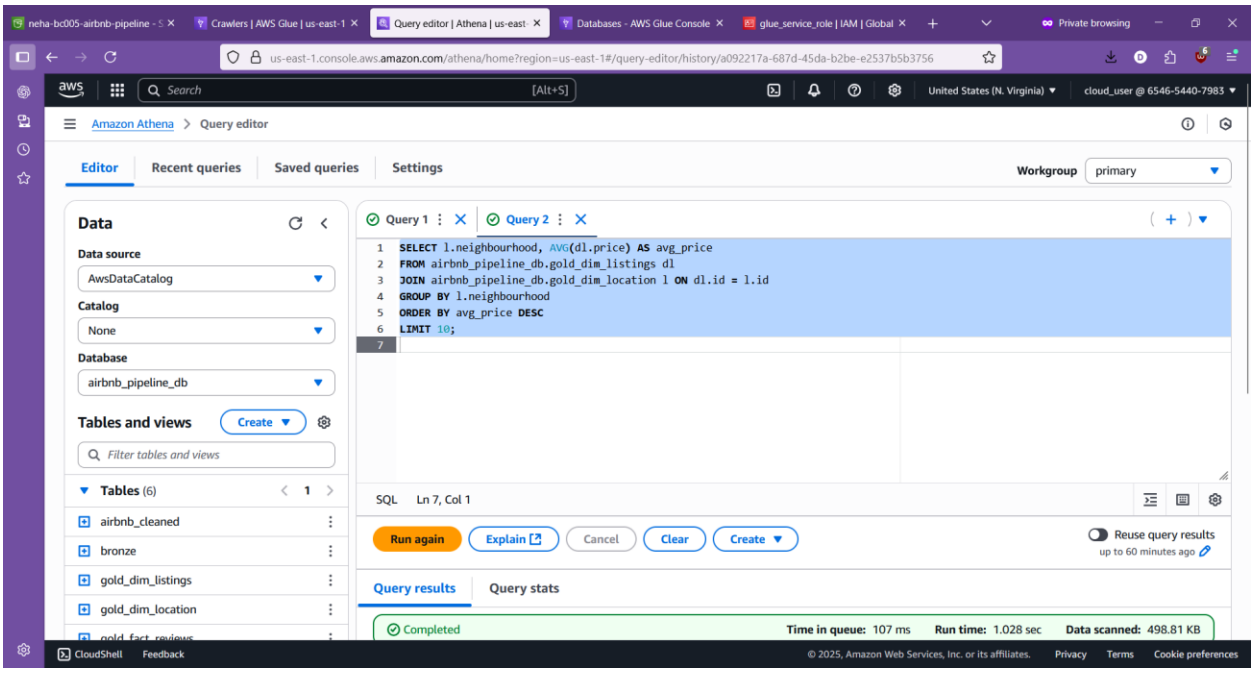
4. Query Gold Tables in Athena

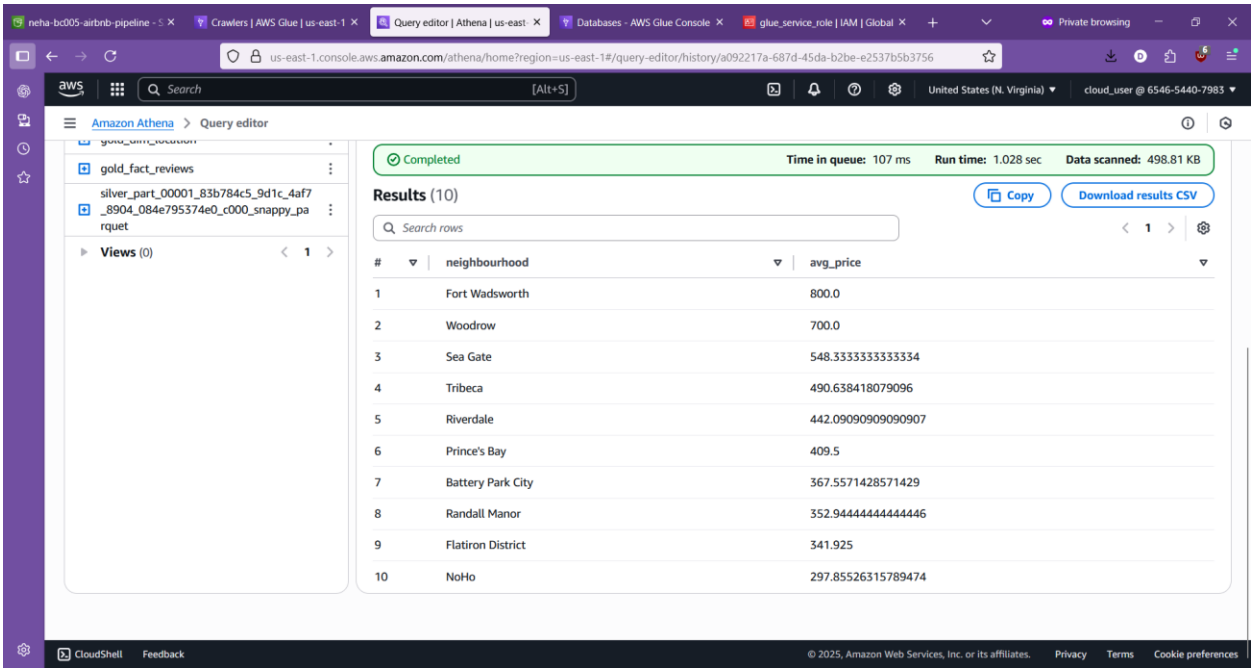
4.1 Listings Overview



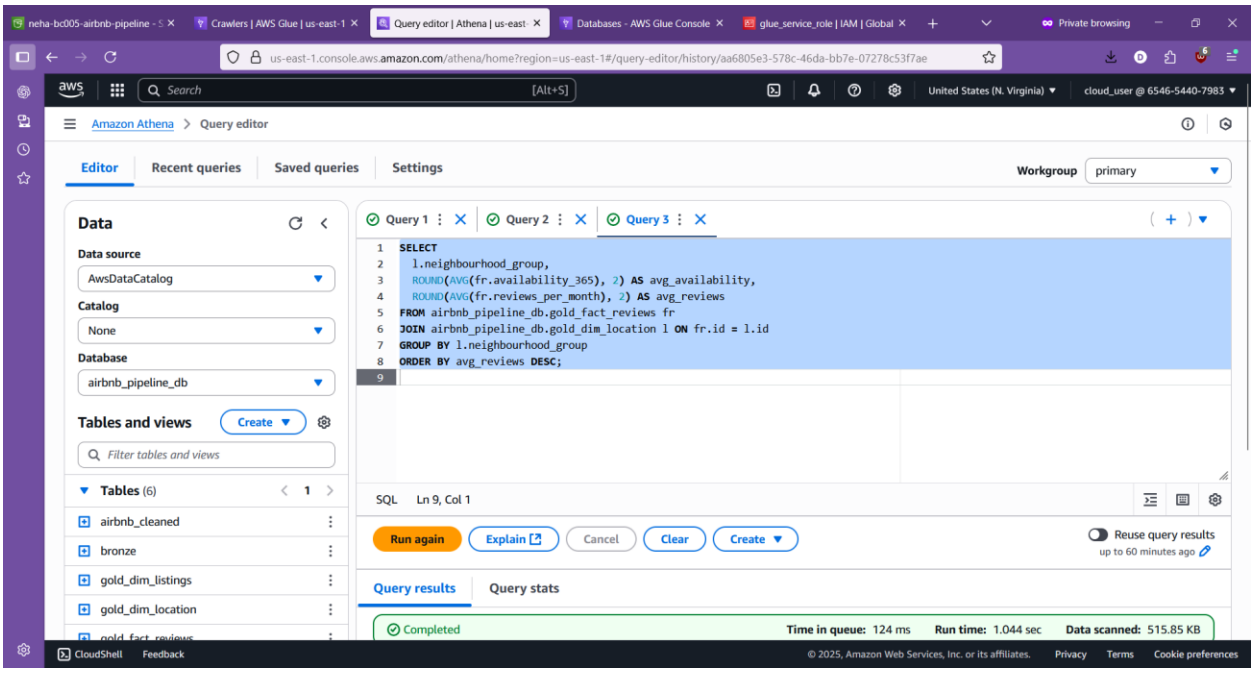


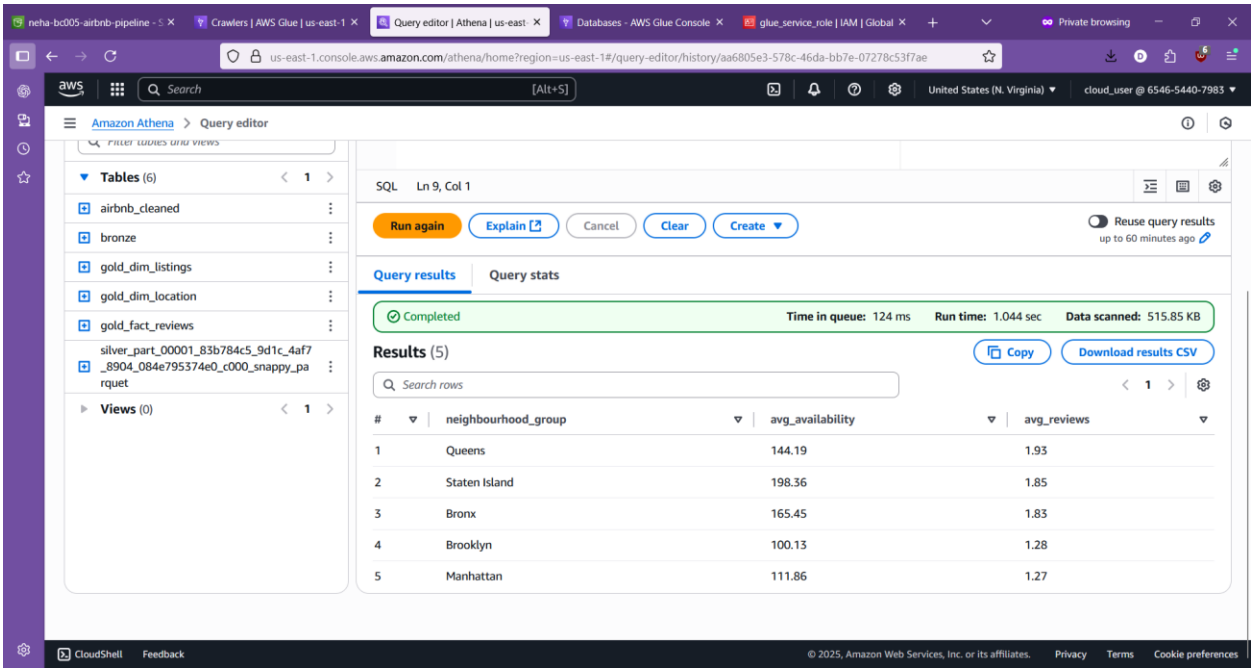
4.2 Most Expensive Neighborhoods





4.3 Availability and Engagement





ML Use Case: Price Prediction Model

Objective:

Predict a listing's price based on its attributes like:

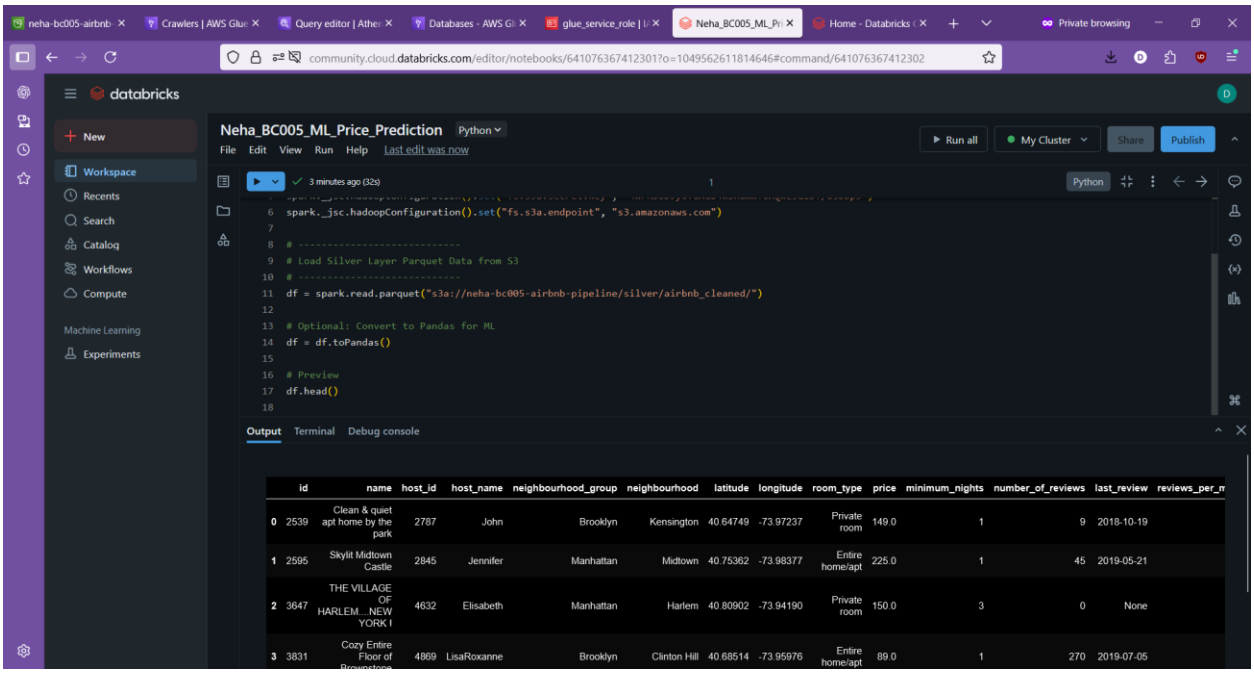
- room type, location, number of reviews, availability, etc.

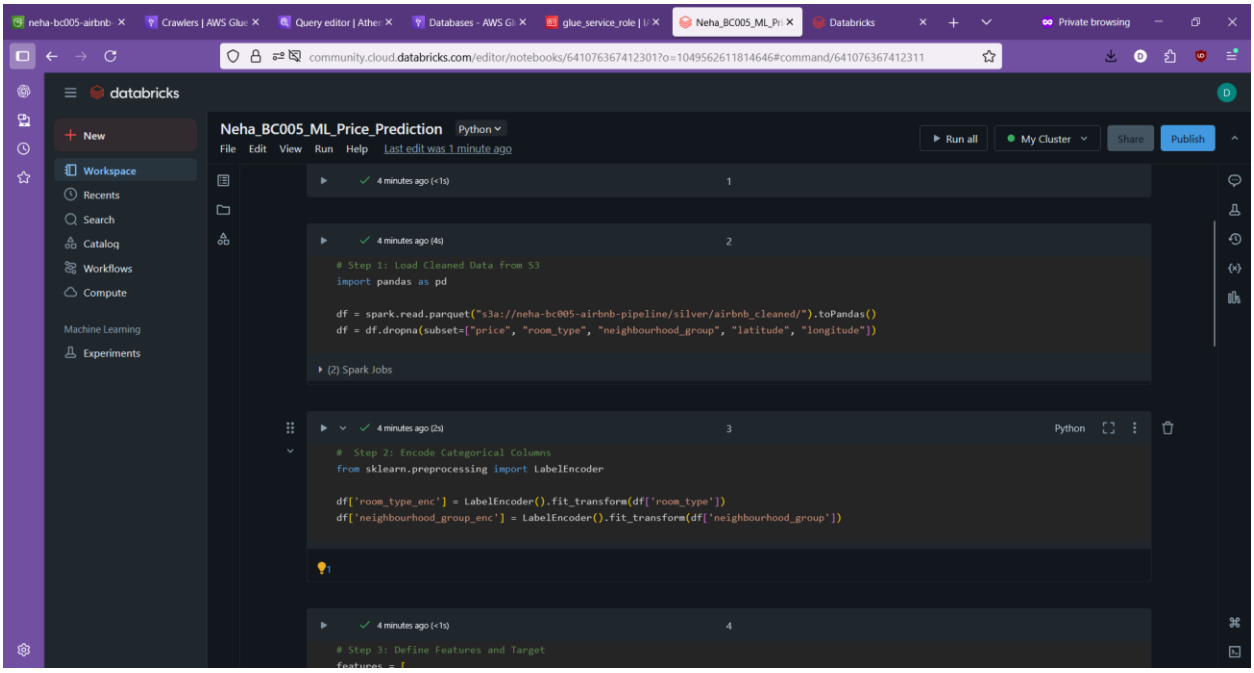
This will help with:

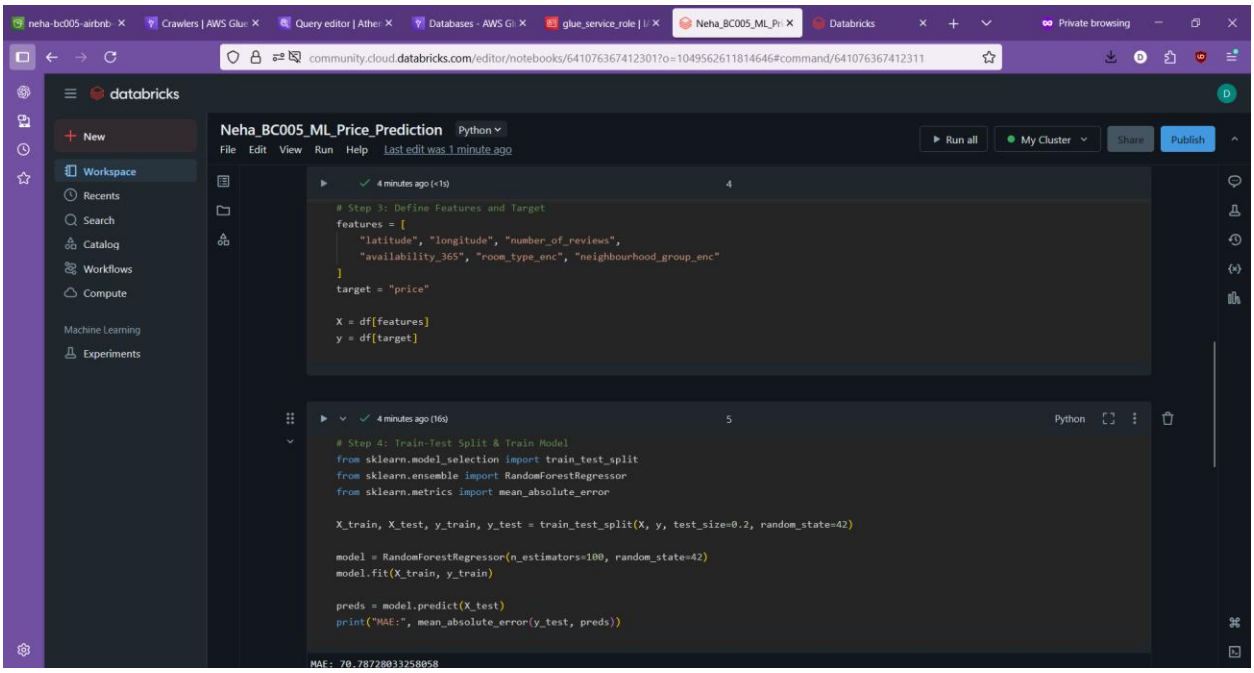
- Price benchmarking, Anomaly detection, Host pricing suggestions

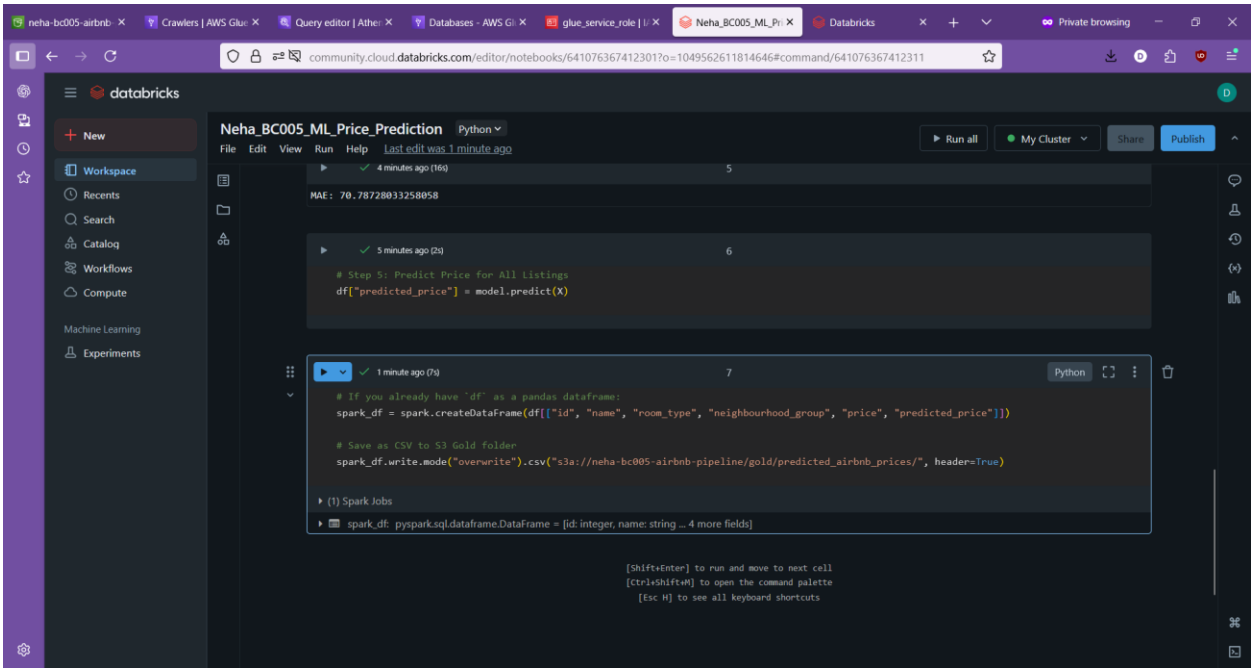
Build Price Prediction in Databricks (with scikit-learn)

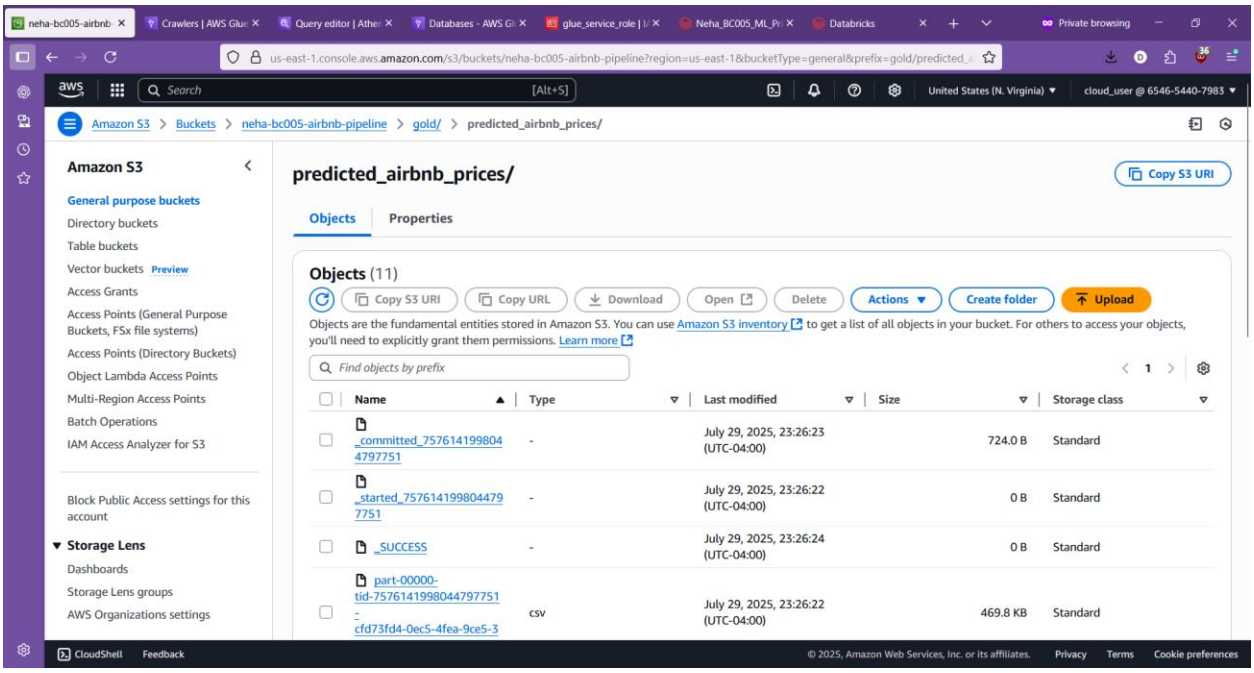
Databricks script:
https://community.cloud.databricks.com/editor/notebooks/641076367412301?o=1049562611814646#command/641076367412311

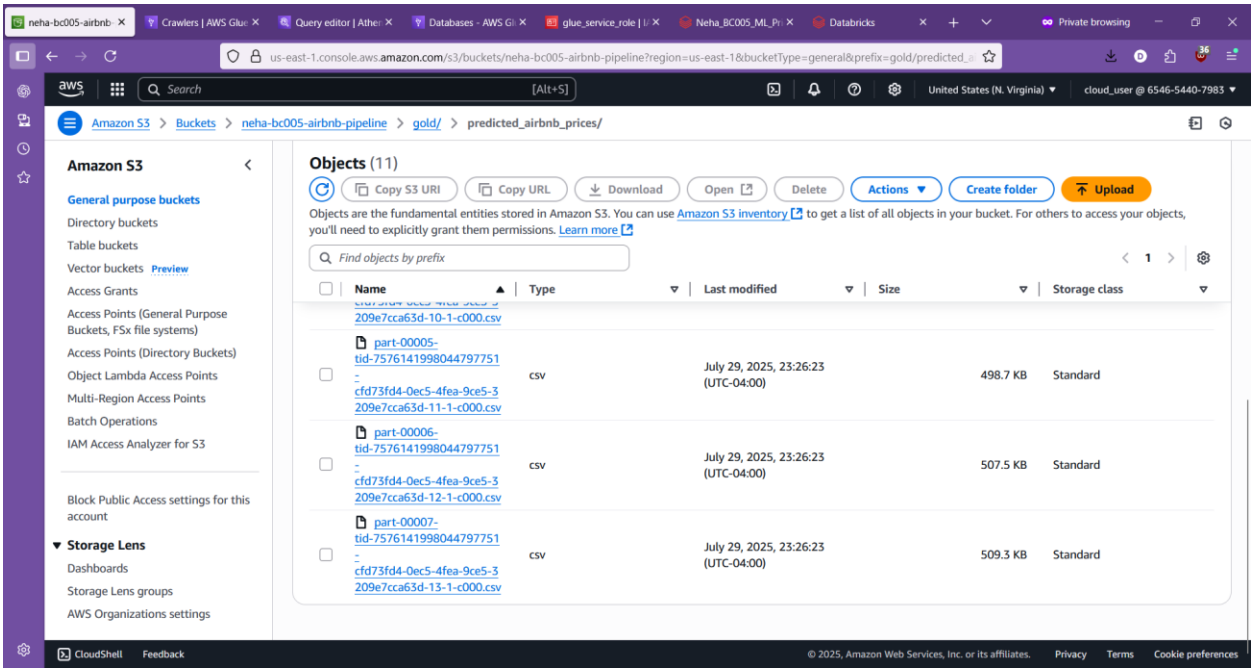





databricks
Neha_BC005_ML_Price_Prediction
Python
File Edit View Run Help Last edit was 1 minute ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
4 minutes ago (<1s)
4
# Step 3: Define Features and Target
features = [
"latitude", "longitude", "number_of_reviews",
"availability_365", "room_type_enc", "neighbourhood_group_enc"
]
target = "price"
X = df[features]
y = df[target]
4 minutes ago (16s)
5
Python
# Step 4: Train-Test Split & Train Model
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
preds = model.predict(X_test)
print("MAE:", mean_absolute_error(y_test, preds))
MAE: 70.78728033258058

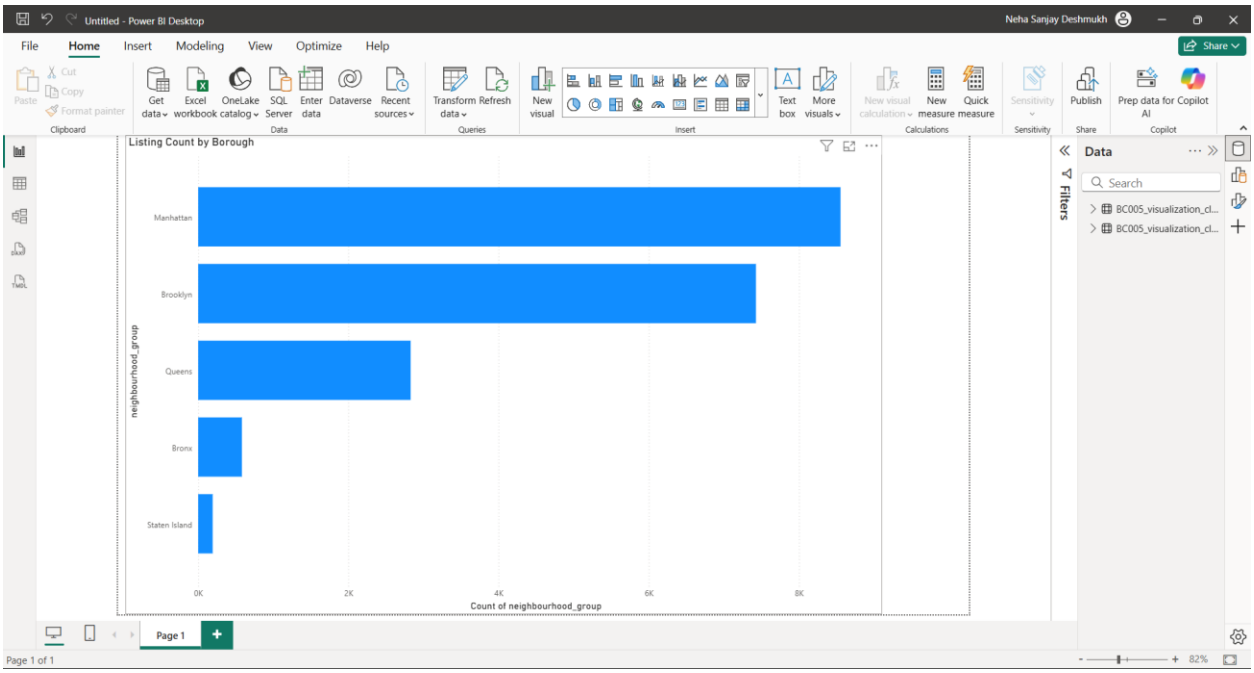

databricks
Neha_BC005_ML_Price_Prediction
Python
File Edit View Run Help Last edit was 1 minute ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
4 minutes ago (16s)
5
MAE: 70.78728033258058
5 minutes ago (2s)
6
Step 5: Predict Price for All Listings
df["predicted_price"] = model.predict(X)
1 minute ago (7s)
7
Python
# If you already have `df` as a pandas dataframe:
spark_df = spark.createDataFrame(df[["id", "name", "room_type", "neighbourhood_group", "price", "predicted_price"]])
# Save as CSV to S3 Gold folder
spark_df.write.mode("overwrite").csv("s3a://neha-bc005-airbnb-pipeline/gold/predicted_airbnb_prices/", header=True)
(1) Spark Jobs
spark_df: pyspark.sql.dataframe.DataFrame = [id: integer, name: string ... 4 more fields]
[Shift+Enter] to run and move to next cell
[Ctrl+Shift+M] to open the command palette
[Esc H] to see all keyboard shortcuts


Amazon S3 > Buckets > neha-bc005-airbnb-pipeline > gold/ > predicted_airbnb_prices/
predicted_airbnb_prices/
Copy S3 URI
Objects
Properties
Objects (11)
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
_committed_7576141998044797751
July 29, 2025, 23:26:23 (UTC-04:00)
724.0 B
Standard
_started_7576141998044797751
July 29, 2025, 23:26:22 (UTC-04:00)
0 B
Standard
_SUCCESS
July 29, 2025, 23:26:24 (UTC-04:00)
0 B
Standard
part-00000-tid-7576141998044797751-cfd73fd4-0ec5-4fea-9ce5-3
csv
July 29, 2025, 23:26:22 (UTC-04:00)
469.8 KB
Standard
Amazon S3
General purpose buckets
Directory buckets
Table buckets
Vector buckets Preview
Access Grants
Access Points (General Purpose Buckets, FSx file systems)
Access Points (Directory Buckets)
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Amazon S3 > Buckets > neha-bc005-airbnb-pipeline > gold/ > predicted_airbnb_prices/
Objects (11)
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
209e7cca63d-10-1-c000.csv
part-00005-tid-7576141998044797751-cfd73fd4-0ec5-4fea-9ce5-3209e7cca63d-11-1-c000.csv
csv
July 29, 2025, 23:26:23 (UTC-04:00)
498.7 KB
Standard
part-00006-tid-7576141998044797751-cfd73fd4-0ec5-4fea-9ce5-3209e7cca63d-12-1-c000.csv
csv
July 29, 2025, 23:26:23 (UTC-04:00)
507.5 KB
Standard
part-00007-tid-7576141998044797751-cfd73fd4-0ec5-4fea-9ce5-3209e7cca63d-13-1-c000.csv
csv
July 29, 2025, 23:26:23 (UTC-04:00)
509.3 KB
Standard
Amazon S3
General purpose buckets
Directory buckets
Table buckets
Vector buckets Preview
Access Grants
Access Points (General Purpose Buckets, FSx file systems)
Access Points (Directory Buckets)
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Visualization

1. Listing Count by Borough - Shows how listings are distributed across NYC.



2. Average Price by Room Type - Pricing differences among Entire home, Private room, Shared room.