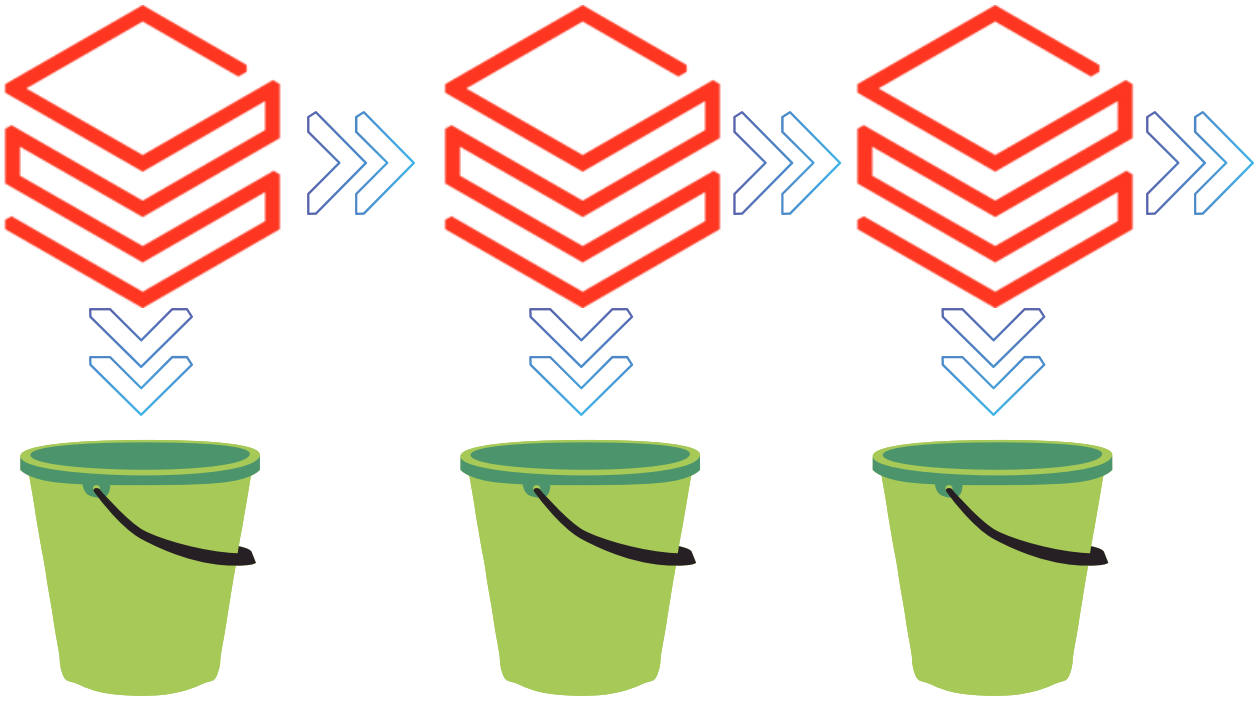
Building a Scalable Data Pipeline for Airbnb Listings Analytics in NYC

Bronze Layer

Silver Layer

Gold Layer

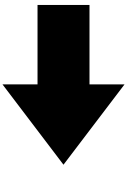


Upload the Raw data in DataBricks and write it in S3 in Delta format

Perform all Transformations and cleaning process and store the data in S3

Perform all Aggregate operations ‘and store the data in S3

AWS Glue



AWS Athena

AWS QuickSight

Using Glue Crawler table is created and visualisation is done using Athena

Objective:

The project addresses key challenges in hospitality analytics by providing data-driven insights into Airbnb listings and host behaviours. The pipeline will support:

- Revenue optimization – Price benchmarking and anomaly detection
- Operational efficiency – Automated host performance tracking
- Market analysis – Neighbourhood-level demand patterns
- Customer experience – Review trends and availability monitoring

The solution must balance scalability, cost-efficiency, and business agility while handling geographic and temporal data. Business Goal: This pipeline delivers tangible business outcomes:

- 15-25% improvement in optimal pricing identification
- 40% reduction in manual market analysis time
- Actionable host performance benchmarks

The solution transforms raw listing data into strategic assets using AWS services, enabling data-driven decision making for hosts, managers, and analysts. Dataset Source: Kaggle Title: Airbnb New York Dataset URL: https://www.kaggle.com/datasets/dgomonov/new-york-city-airbnb-open-data Description: This dataset contains Airbnb New York Dataset listings including host id, host names, id, name prices, minimum_night, reviews ect., Prerequisite:

- Databricks Community version
- AWS Account

Bronze Layer

Step 1: Create S3 Bucket for “Bronze”, ”Silver” and “Gold”
Go to AWS Search Console S3 Create Bucket
Provide Bucket name
Click on “Create Bucket” [Refer Image 1]
S3 Bucket Flow Chart:

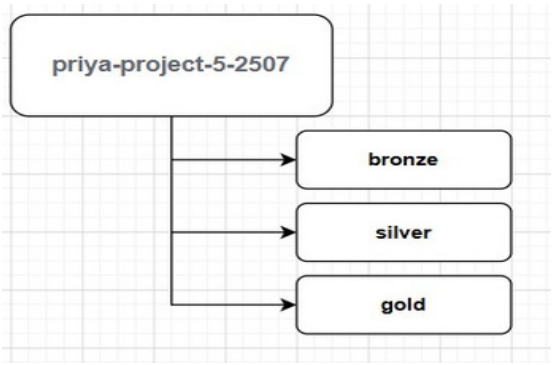
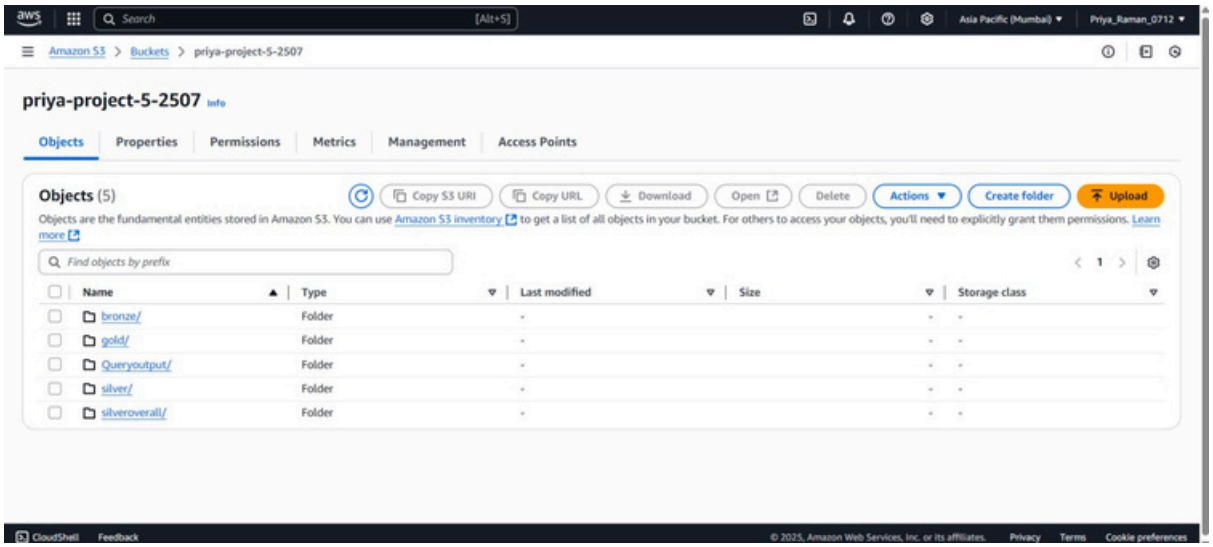


Image 1:



Step 2: Update the IAM Role in IAM USER to read and write files to S3 bucket from Databricks and vice versa

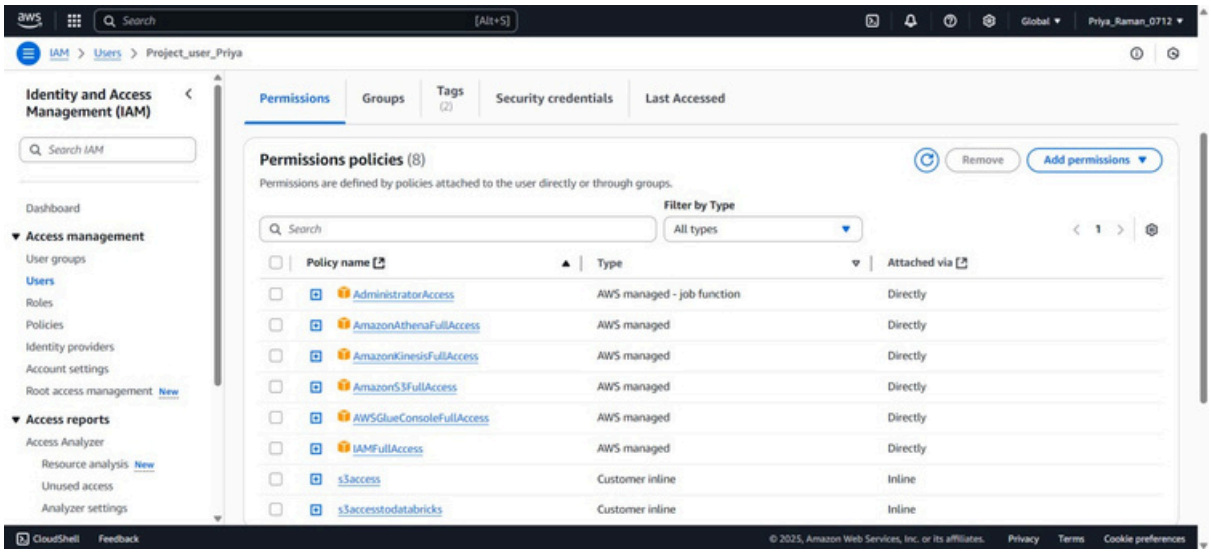
Go to AWS Search Console IAM User Project_user_Priya Add Permissions Create Inline Policy
Add the following code
Create the new permissions[Refer Image 2]

JSON Code:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket",
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::priya-project-5-2507 ",
```

```
                "arn:aws:s3:::priya-project-5-2507 /*"
            ]
        }
    ]
}
```

Image 2:



Step 3: Create IAM Role

Go to AWS Search Console IAM Roles Create Role

Select “AWS Service” in “Trusted entity type”

Select “Glue” in “Use case”

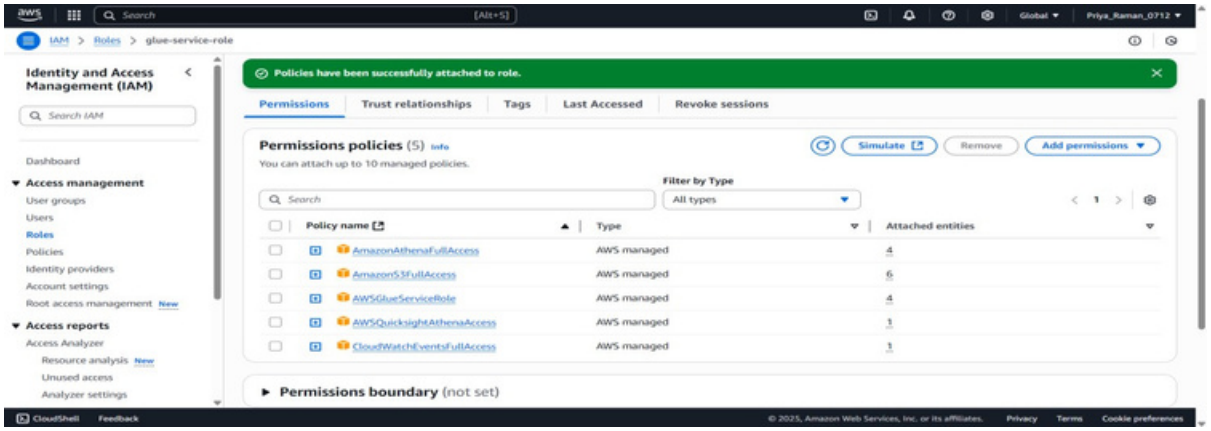
Click Next

Add the following Permissions

- AmazonS3FullAccess
- AWSGlueServiceRole
- CloudWatchEventsFullAccess
- AmazonAthenaFullAccess
- AWSQuicksightAthenaAccess

Provide Role Name as “glue_service_role” [Refer Image 3]

Image 3:



Step 4: Upload the data into Bronze layer

Upload the data in Databricks platform

Write the raw data in AWS S3 bucket in delta format using Databricks platform[Refer Image 4,5,6]

Image 4:

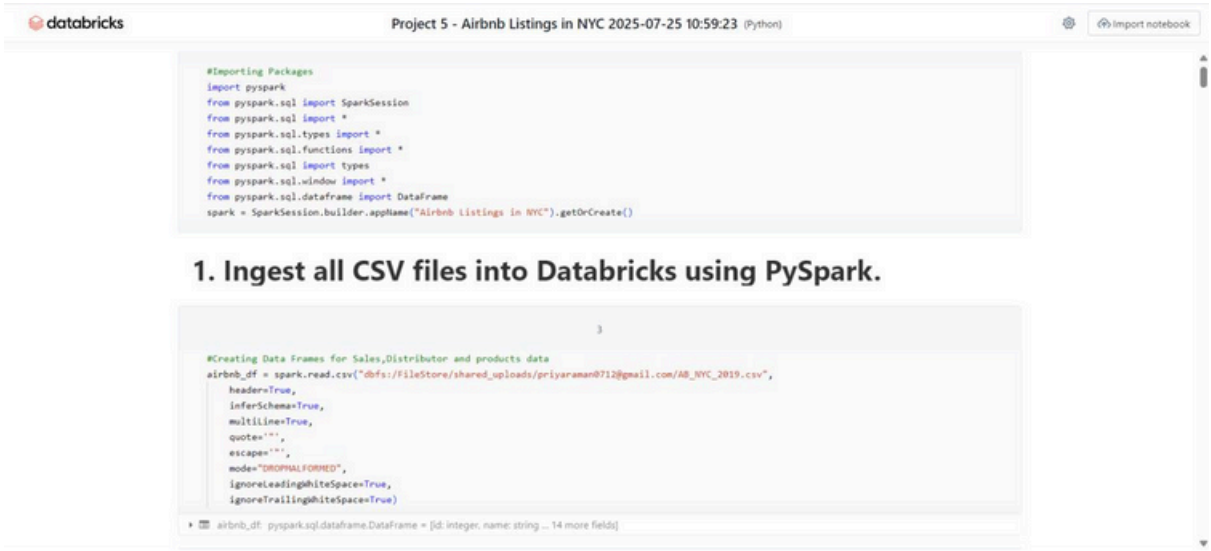


Image 5:

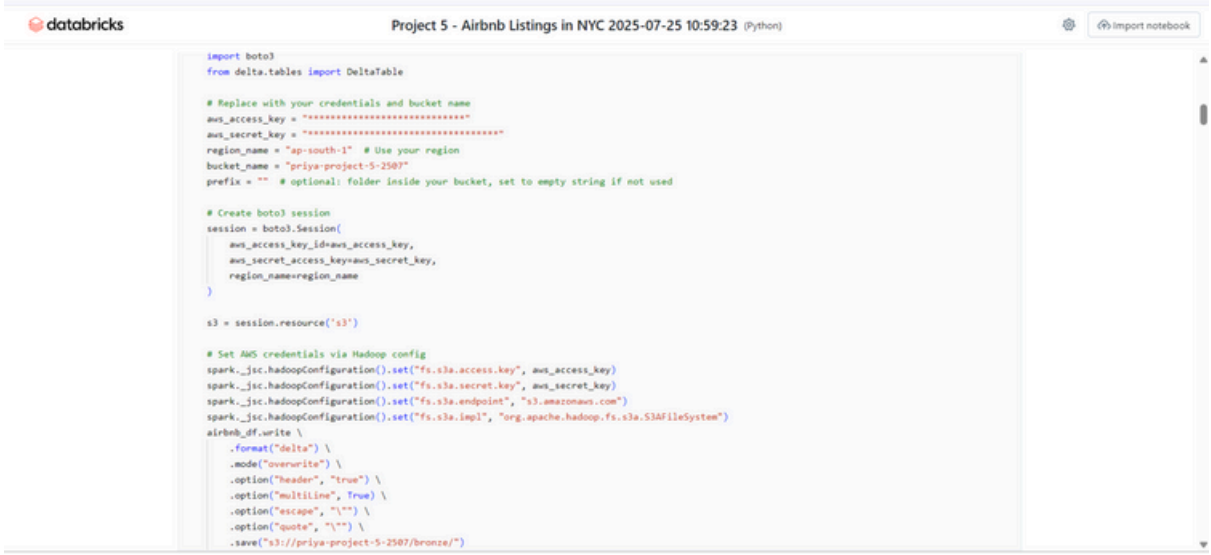
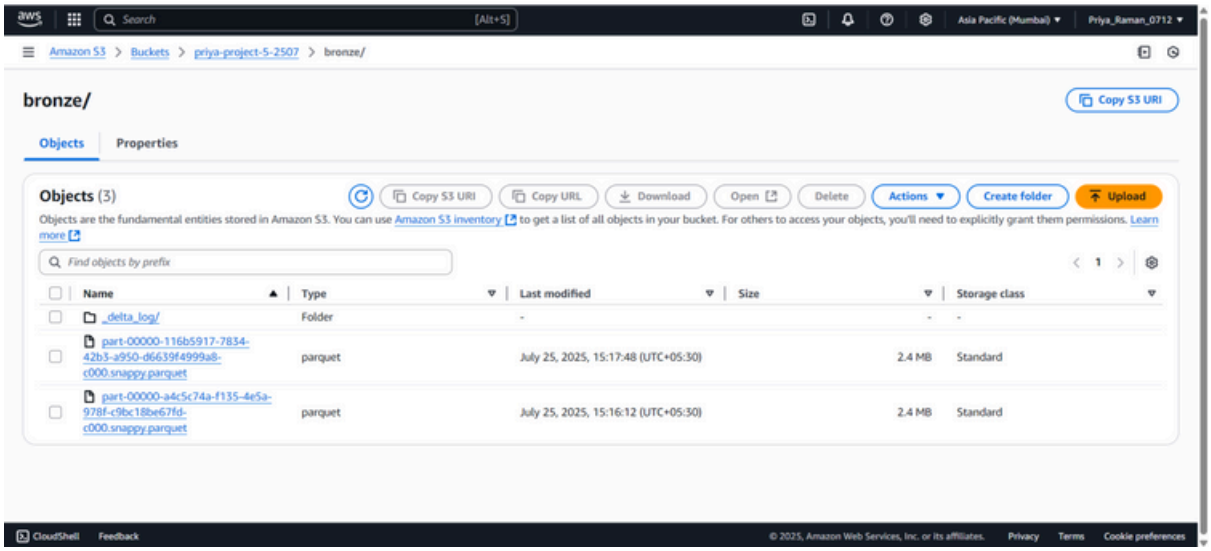


Image 6:



Silver Layer

Step 5: Perform the Transformation task in the Silver layer

1. Clean and standardize the pricing field(remove 0 or negative values). [Refer Image 9]
2. Remove duplicate product entries and fix missing values. [Refer Image 7 & 8]
3. Change the Datatypes as per data[Refer Image 10]
4. Dropping null values in “name” column
5. Write the file in S3 in parquet format[Refer Image 11]

Image 7:

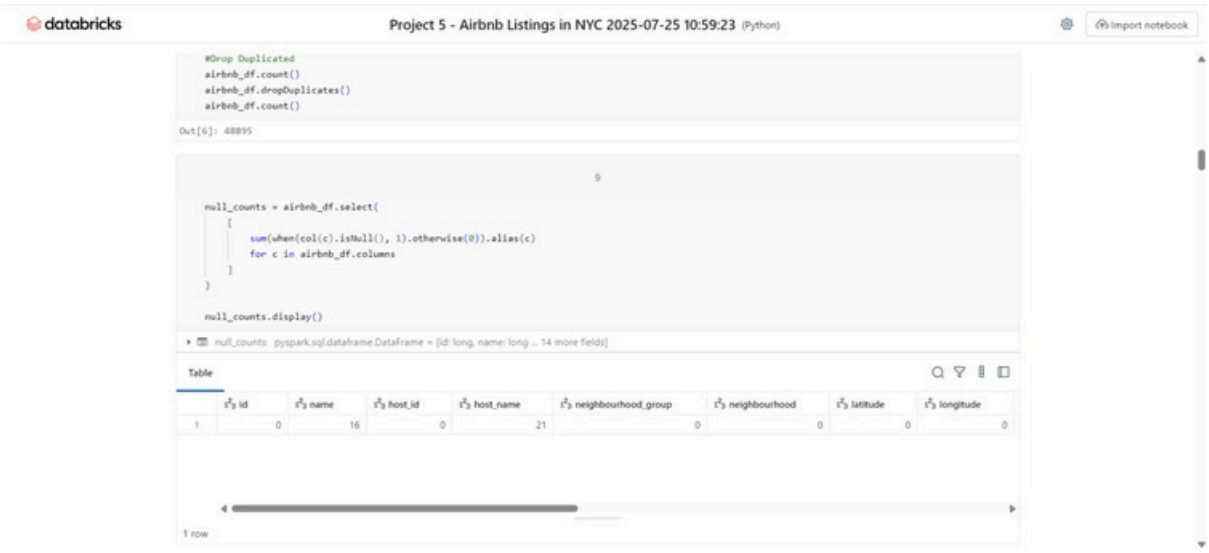


Image 8:

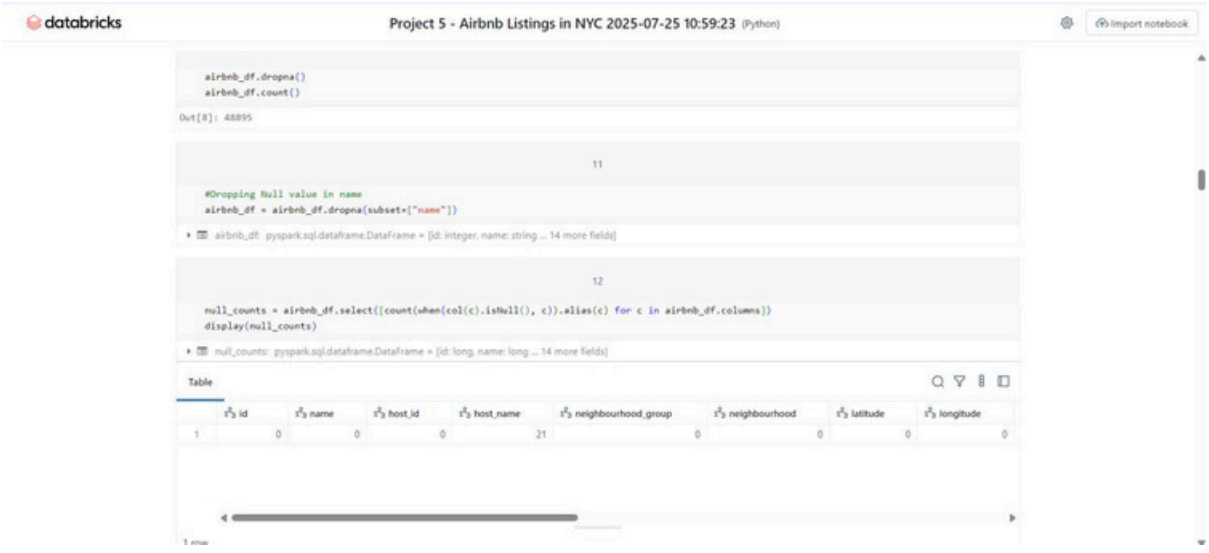


Image 9:

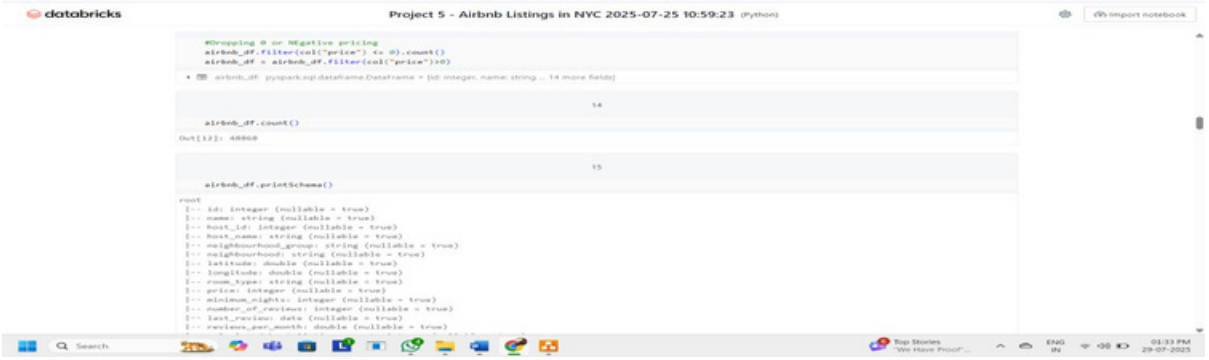


Image 10:

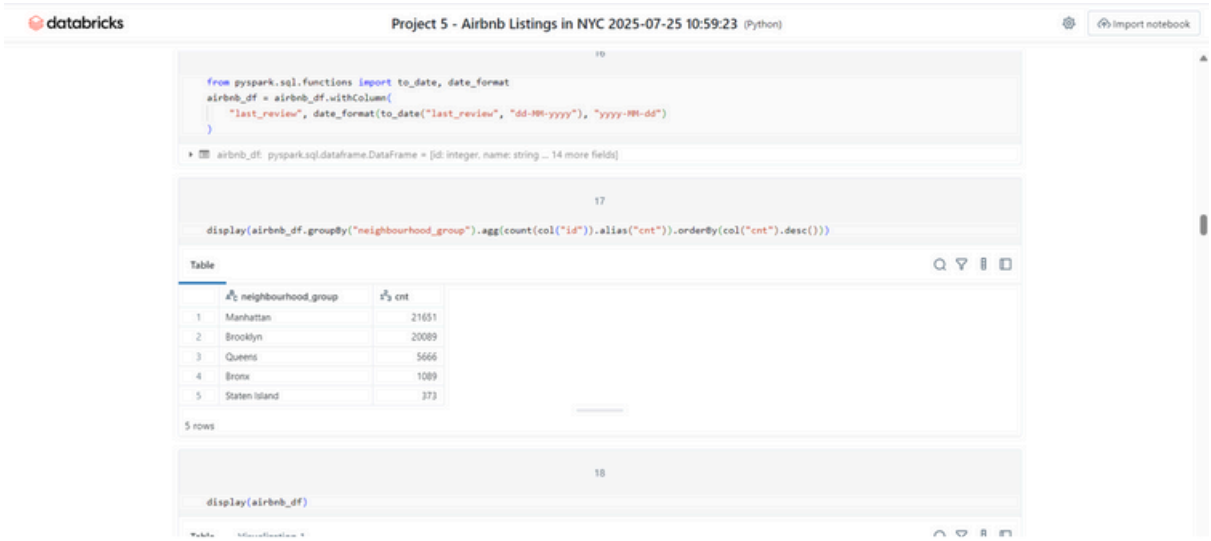
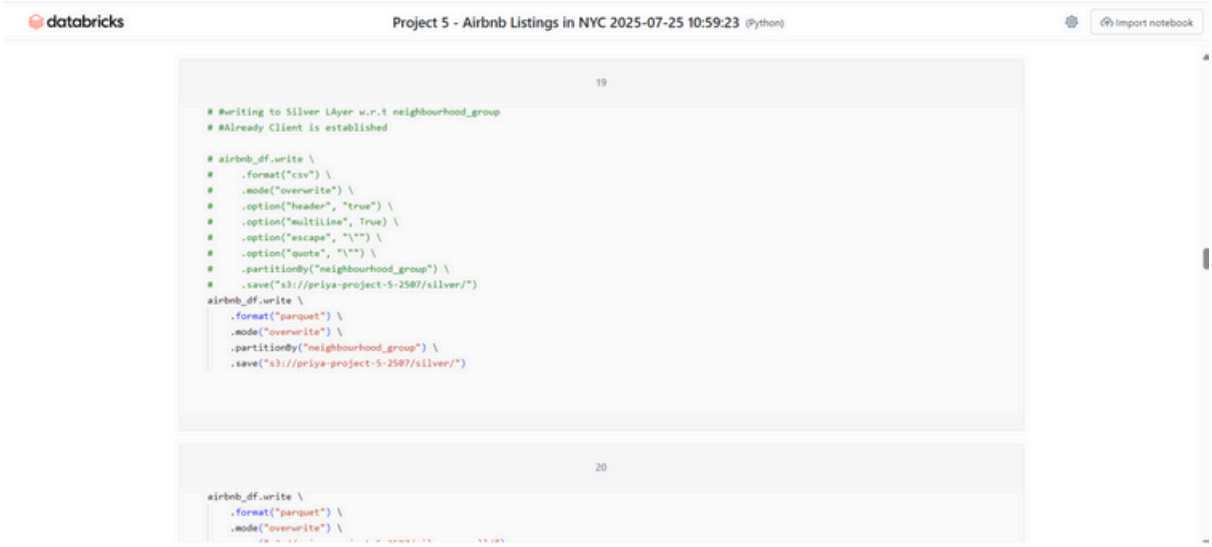
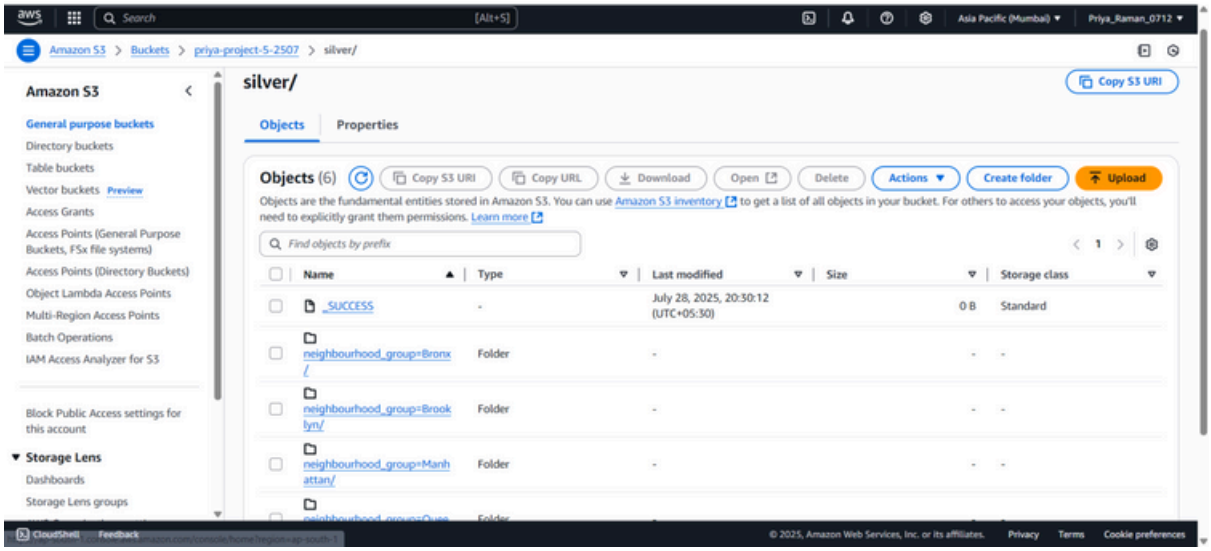


Image 11:



Output:



Gold Layer

Step 6: Perform aggregate function

Identify the average price for room type [Refer Image 12]
Identify the average review by area[Refer Image 13]
Identify the average review by neighbourhood[Refer Image 14]
Identify the average price by area[Refer Image 15]
Identify the room_type by area[Refer Image 16]
Identify the top_host[Refer Image 17]
Identify the average min_night_requirement[Refer Image 18]
Identify the average high_availability[Refer Image 19]
Identify the average host_performance_summary[Refer Image 20]
Write the data into S3 Buckert in Parquet format[Refer Image 21,22,23]

Image 12:

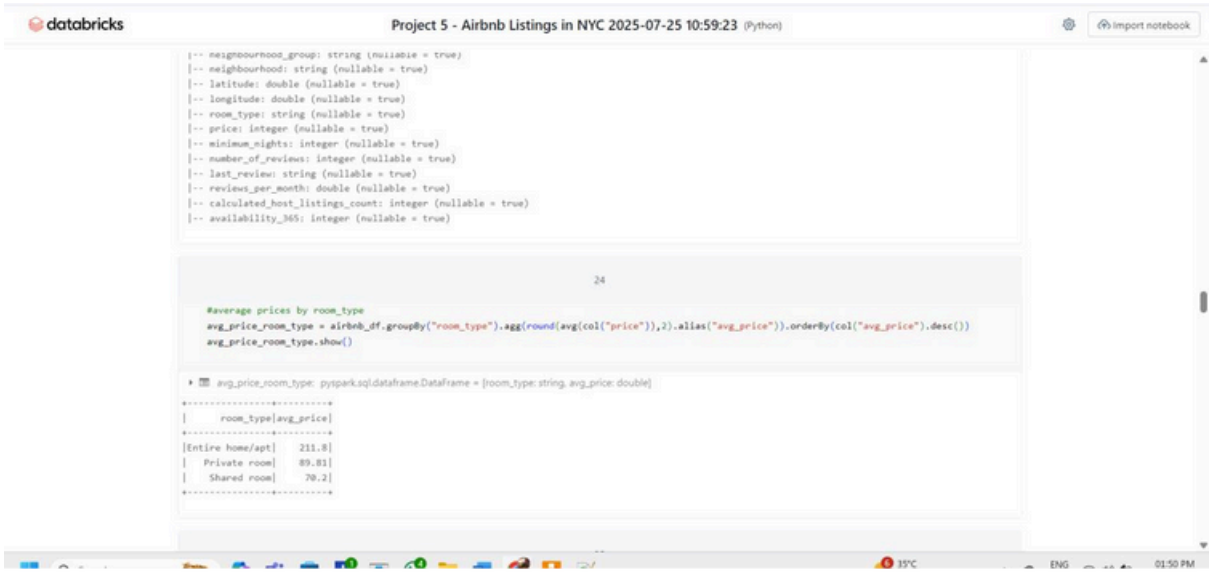


Image 13:

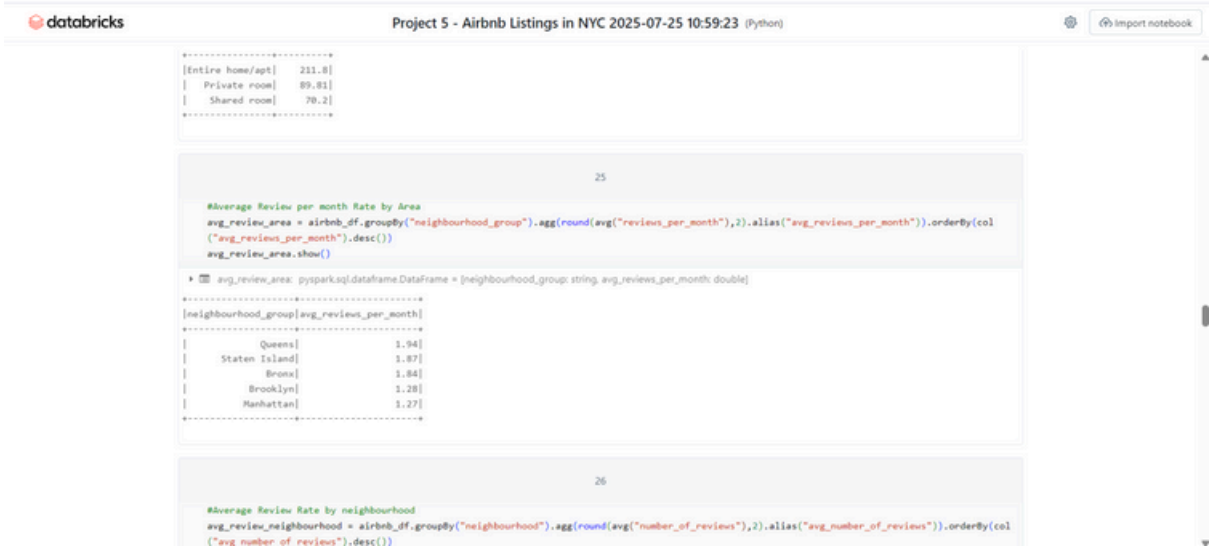


Image 14:

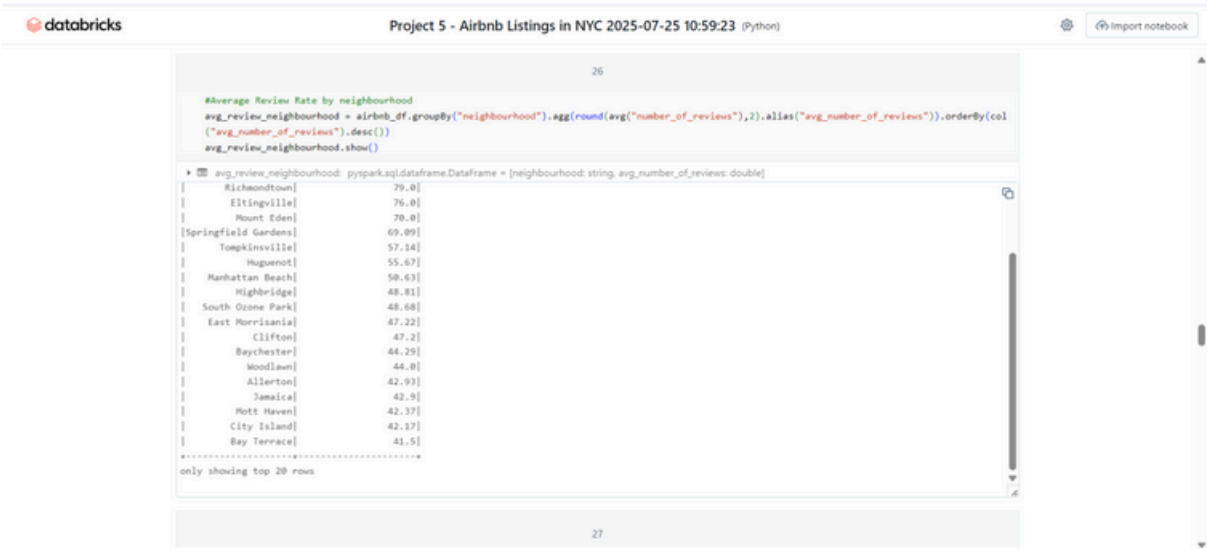


Image 15:

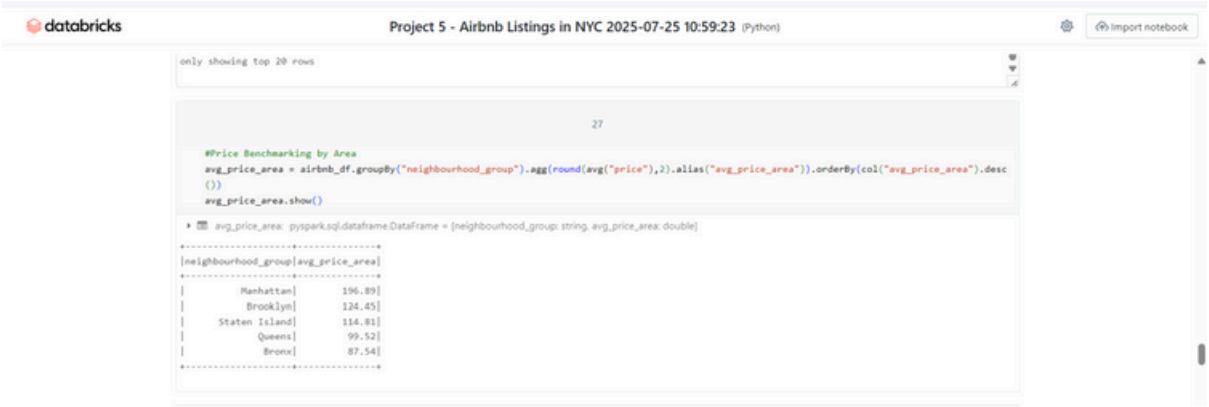


Image 16:

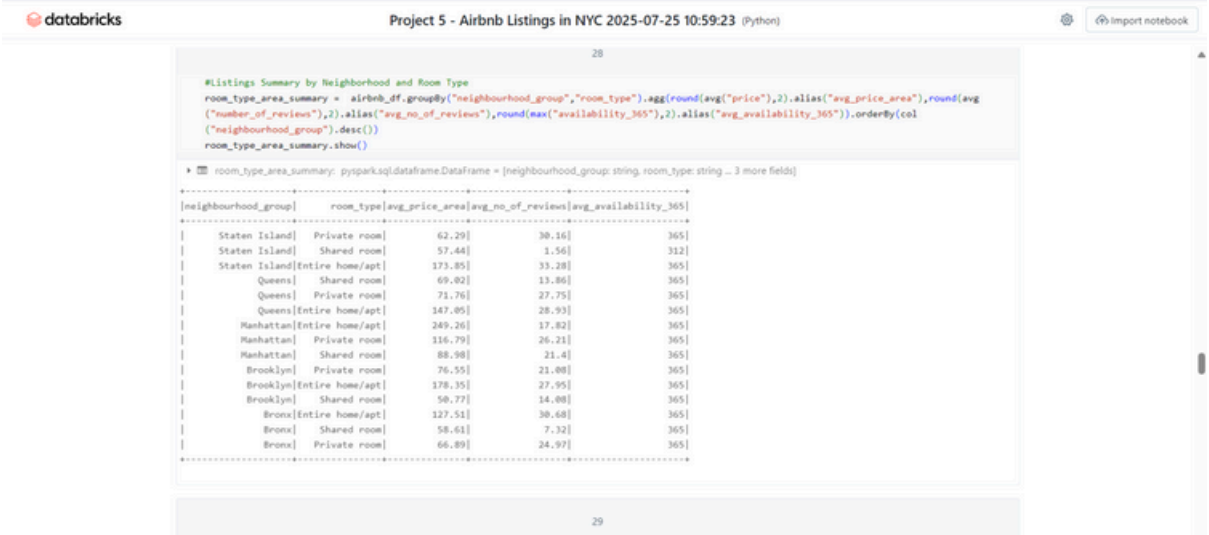


Image 17:

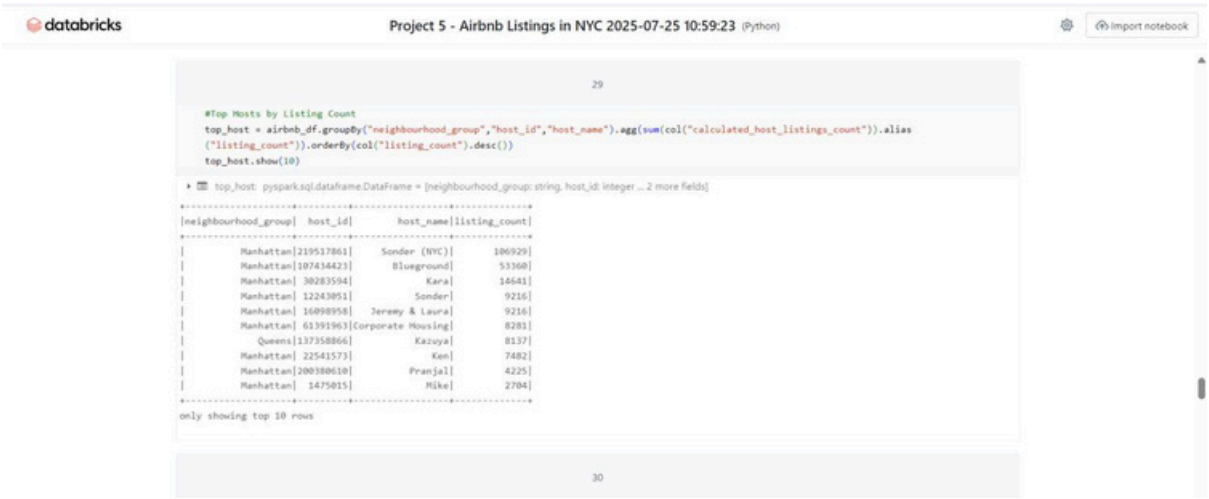


Image 18:

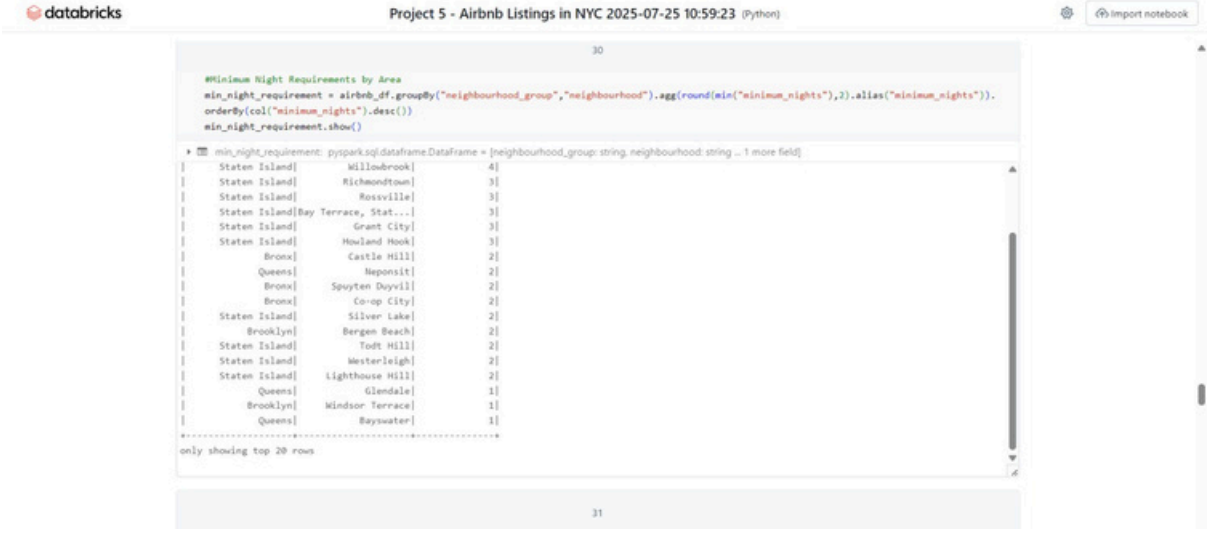


Image 19:

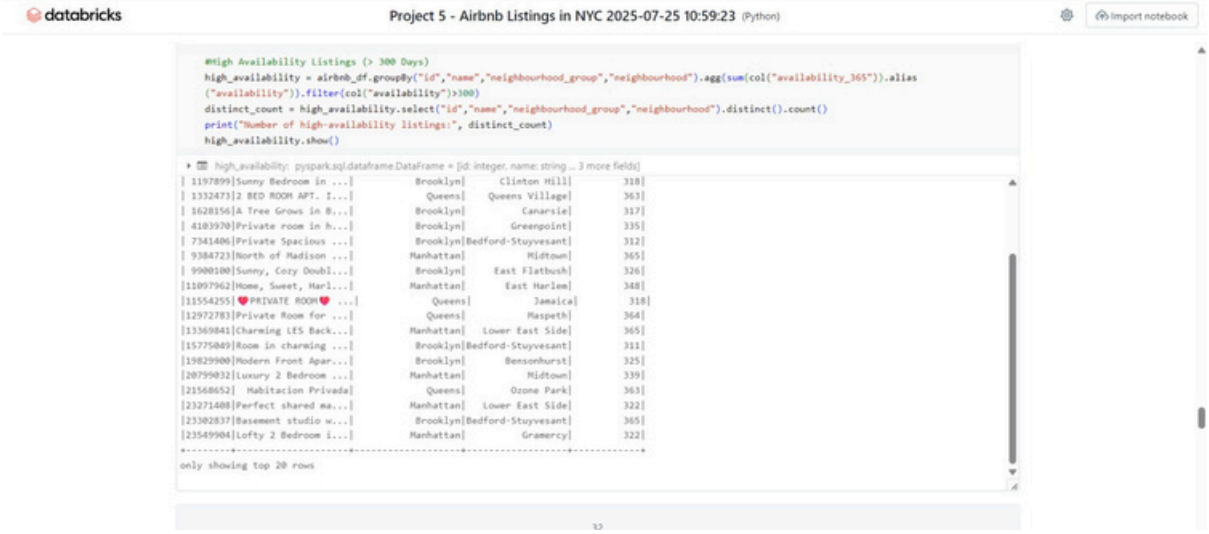


Image 20:

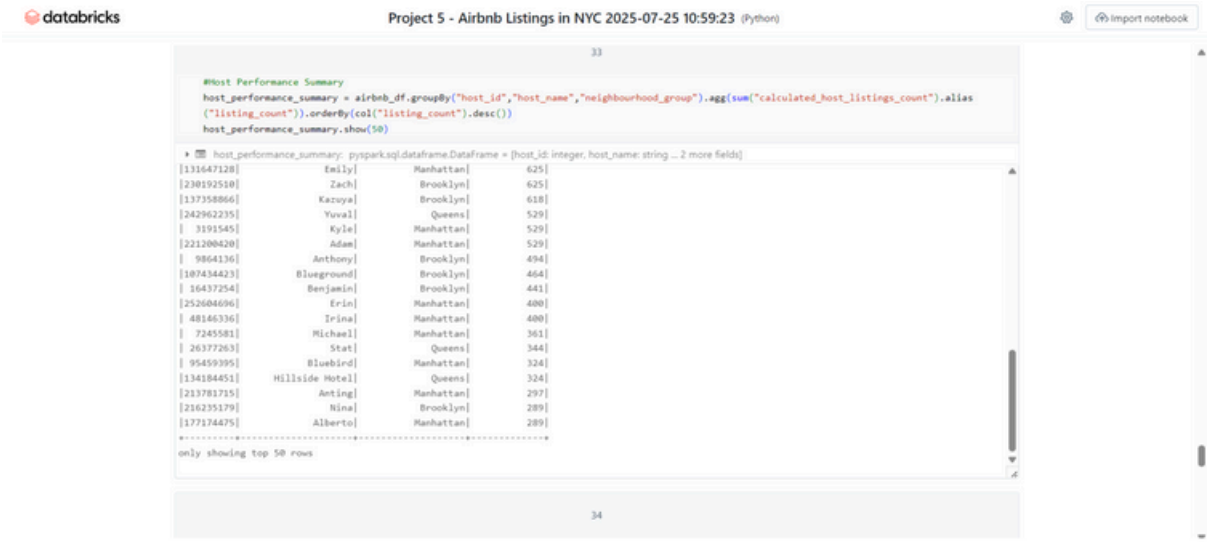


Image 21:

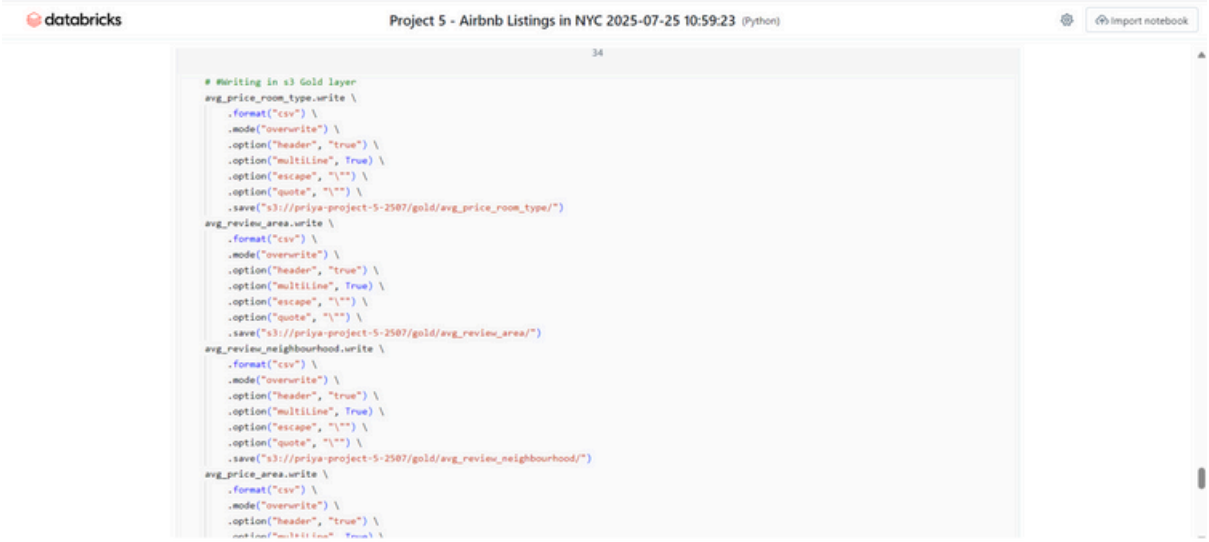
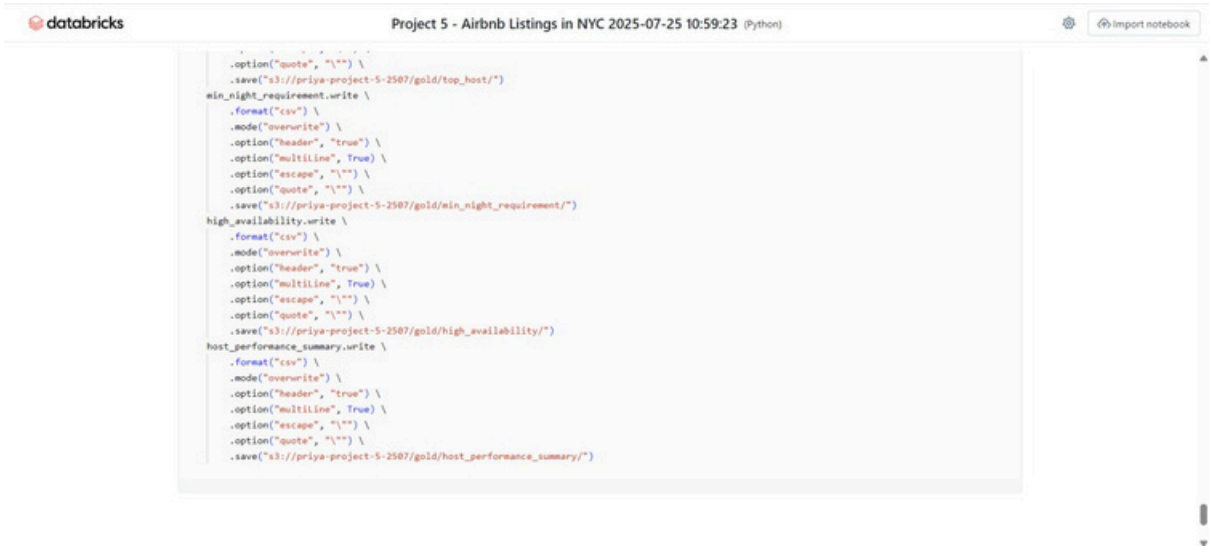
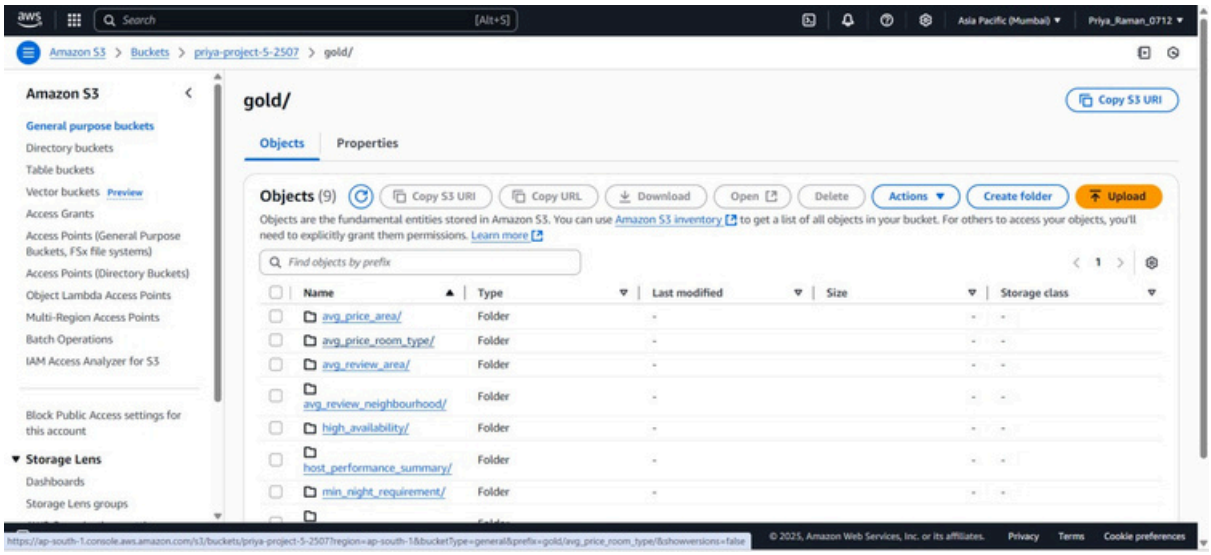


Image 22:

Image 23:

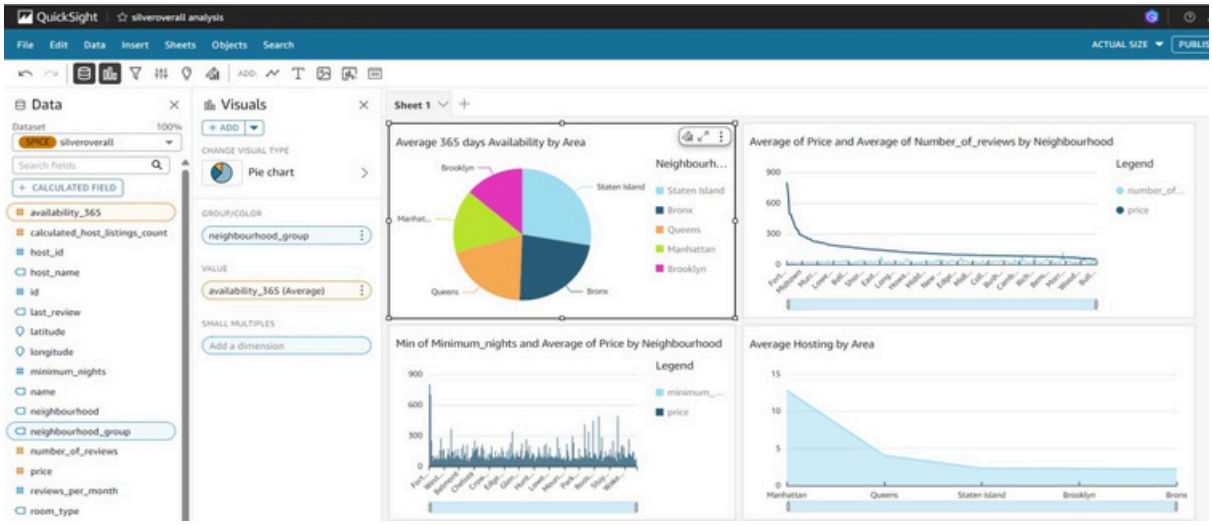


Output:



Visualization

Visualization is done in QuickSight



Appendix:

Databricks link: Tested the Transformation and Cleaning process for Raw is done using PySpark language activity in Databricks platform.

https://databricks-prod-cloudfront.cloud.databricks.com/public/4027ec902e239c93eaaa8714f173bcfc/1230577161393971/3859233799733772/6414225150628508/latest.html