DWDM LAB PROJECT

ETL using AWS Glue and Redshift

Dataset Used : AMAZON PRODUCT SALES

Team Members

SERIAL NO.	ROLL NO.	NAME
01.	01	Aboli Patne
02.	08	Shamika Aney
03.	23	Akshay Padia
04.	34	Hariom Nabira

Supervised by :-
Prof. Arti Karandikar

Problem Definition

To efficiently process and clean raw e-commerce product data from Amazon using the ETL pipeline on AWS for improved analytics.

Introduction:

In the age of data-driven decision-making, raw datasets often contain inconsistencies, null values, and non-standard formats that hinder analysis. To address this, we leverage AWS's powerful ETL services specifically AWS Glue and Glue Crawler to extract Amazon product listings, transform them through a series of custom cleaning and formatting rules, and load the structured data into a usable format. This ensures high-quality, standardized product data that supports accurate insights and efficient querying for business intelligence applications.

Dataset

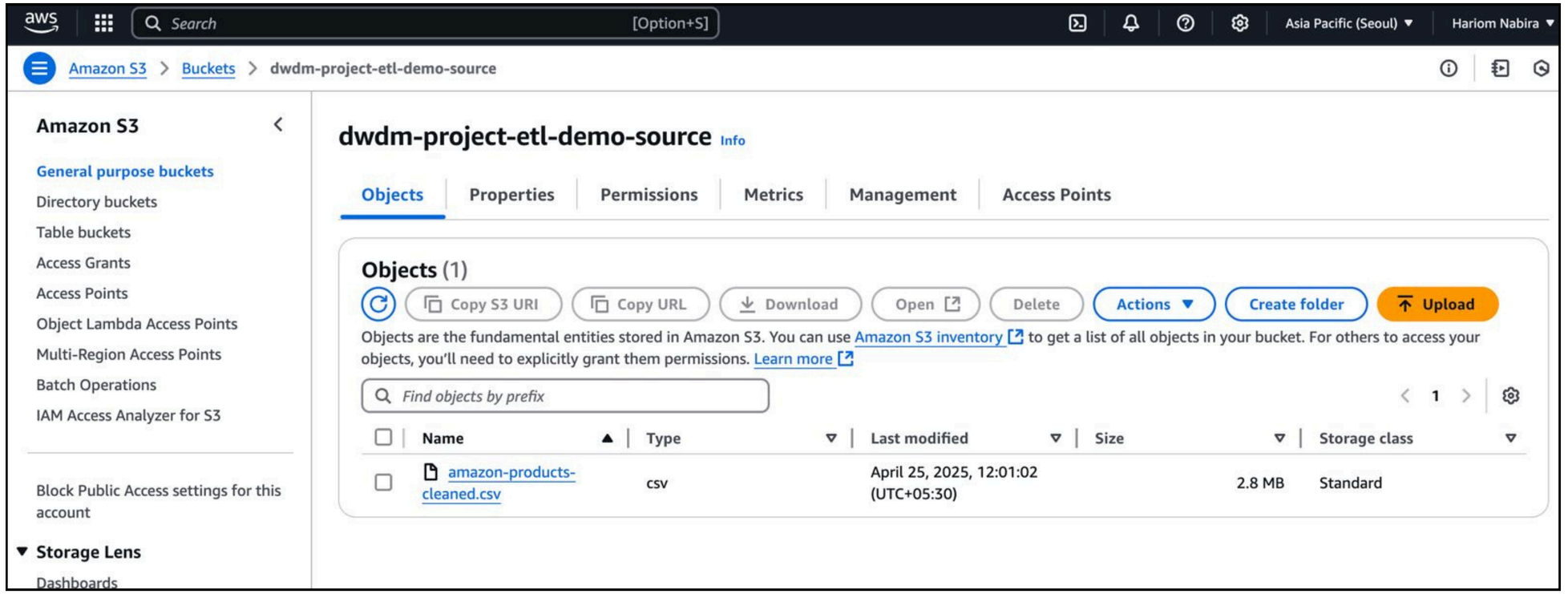
- **Dataset Name:** Amazon Products Dataset
- **Size:** 1000 records × 24 attributes
- **Purpose:** This dataset contains structured product listings scraped from Amazon, including key fields such as product title, brand, price, reviews, ratings, categories, and availability.
- **Key Features:**
  - Product metadata (title, brand, description)
  - Pricing (final_price, currency)
  - Performance metrics (reviews_count, rating, answered_questions)
  - Media info (images_count, video_count, image_url)
  - Additional attributes like asin, seller_id, manufacturer, availability

Extraction

In the extraction phase of the ETL (Extract, Transform, Load) process, Amazon product sales data was first uploaded to an **Amazon S3 bucket**, which served as the centralized data lake. Leveraging **AWS Glue**, a metadata crawler was configured to automatically scan the S3 location and detect the structure of the data. The **crawler** parsed the CSV file to infer the schema identifying data types, column names, and formatting and stored this metadata in the **AWS Glue Data Catalog**.

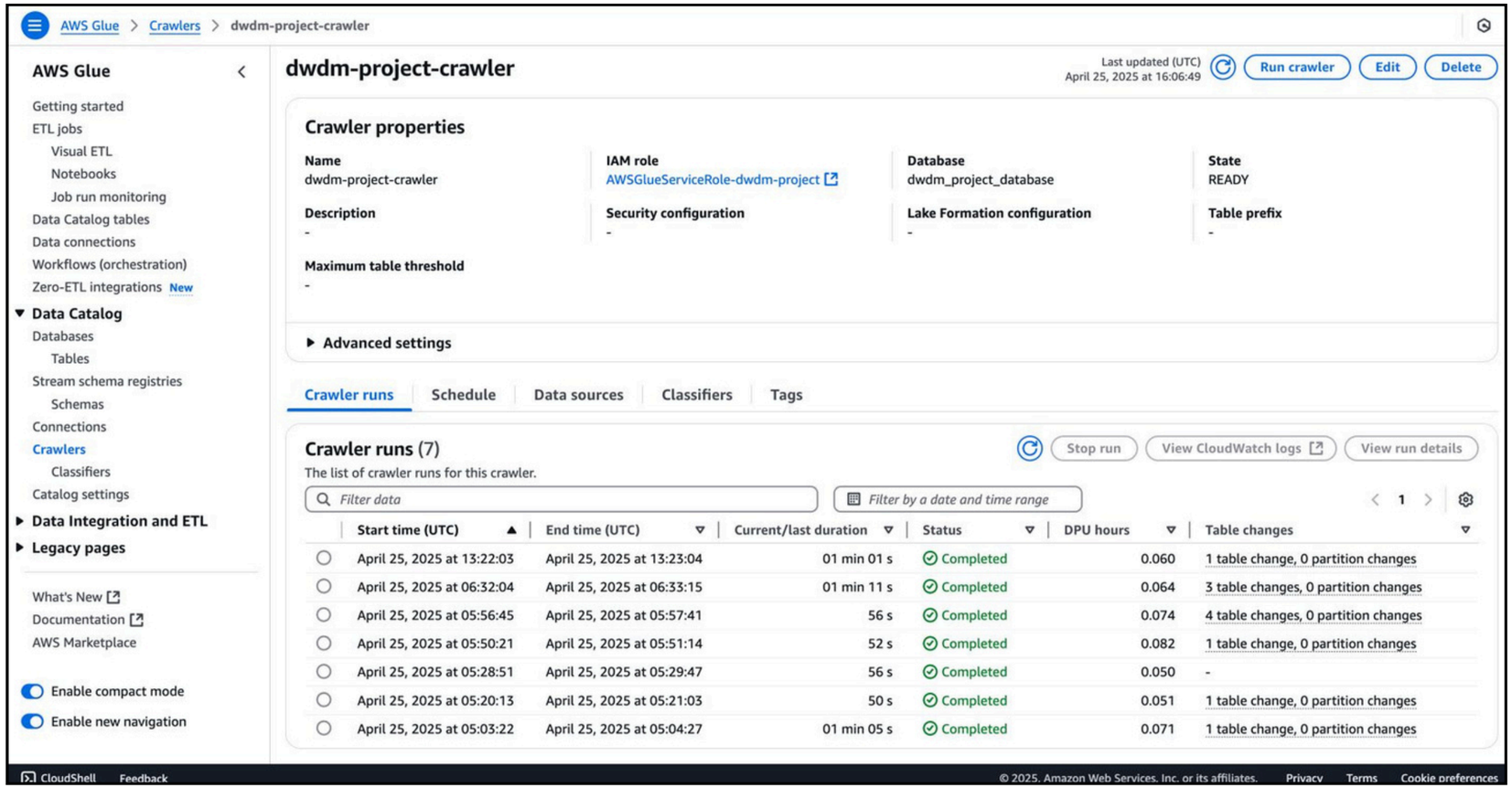
Extraction

Amazon S3 Bucket



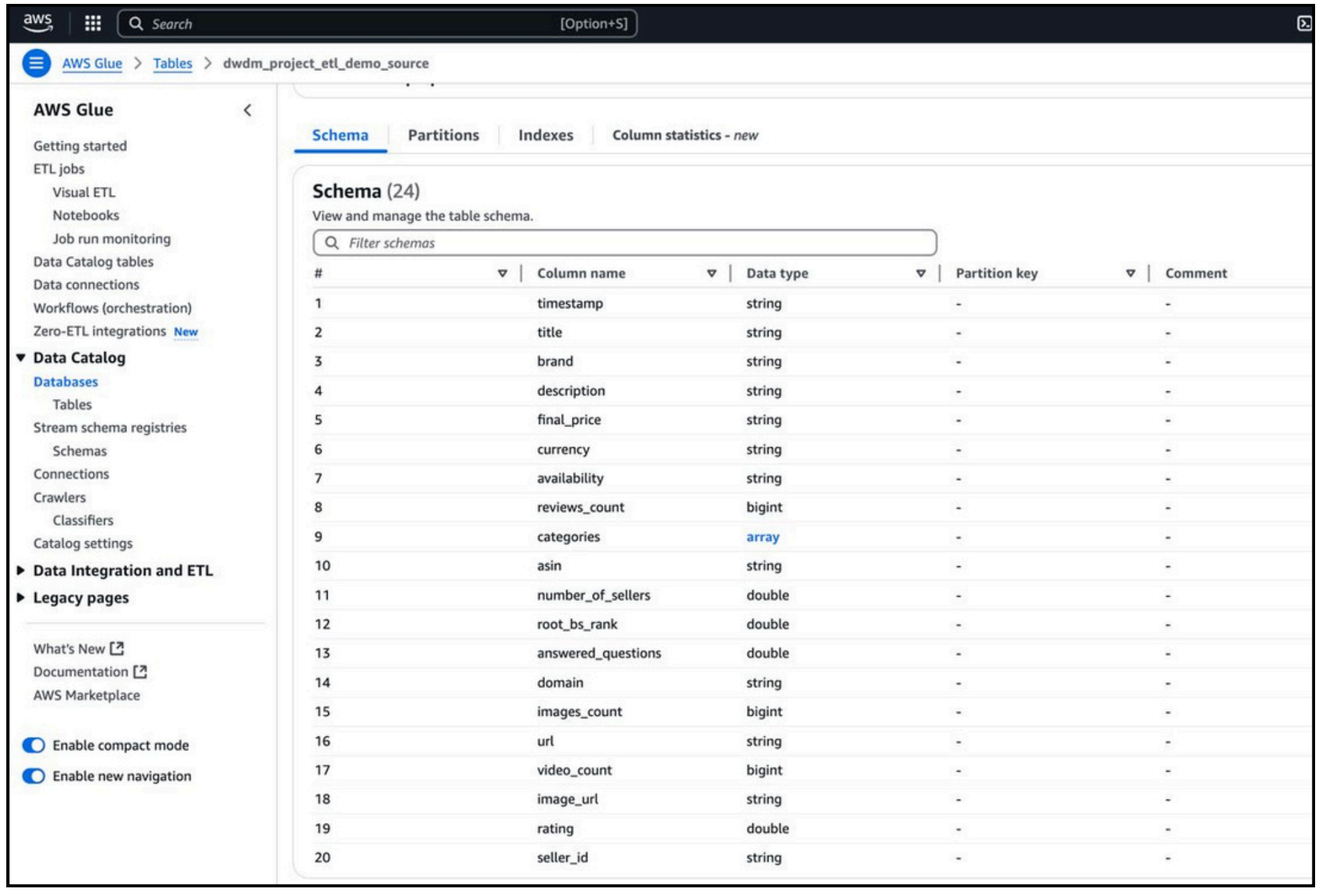
Extraction

Crawler



Extraction

Schema



Transformation

- ETL Job was developed using AWS Glue, where transformation logic was scripted within the Glue job itself.
- Source data was cataloged through an AWS Glue crawler, which automatically classified the schema and made it queryable.
- Custom transformations applied during the process include:
- Availability
  - Values containing the word “left” (e.g., “Only 3 left”) were replaced with "Limited".
  - All variations of “in stock” were standardized to "In Stock".
- Number of Sellers
  - If missing, the value was set to 1.
  - If already present, the original value was retained.

Transformation

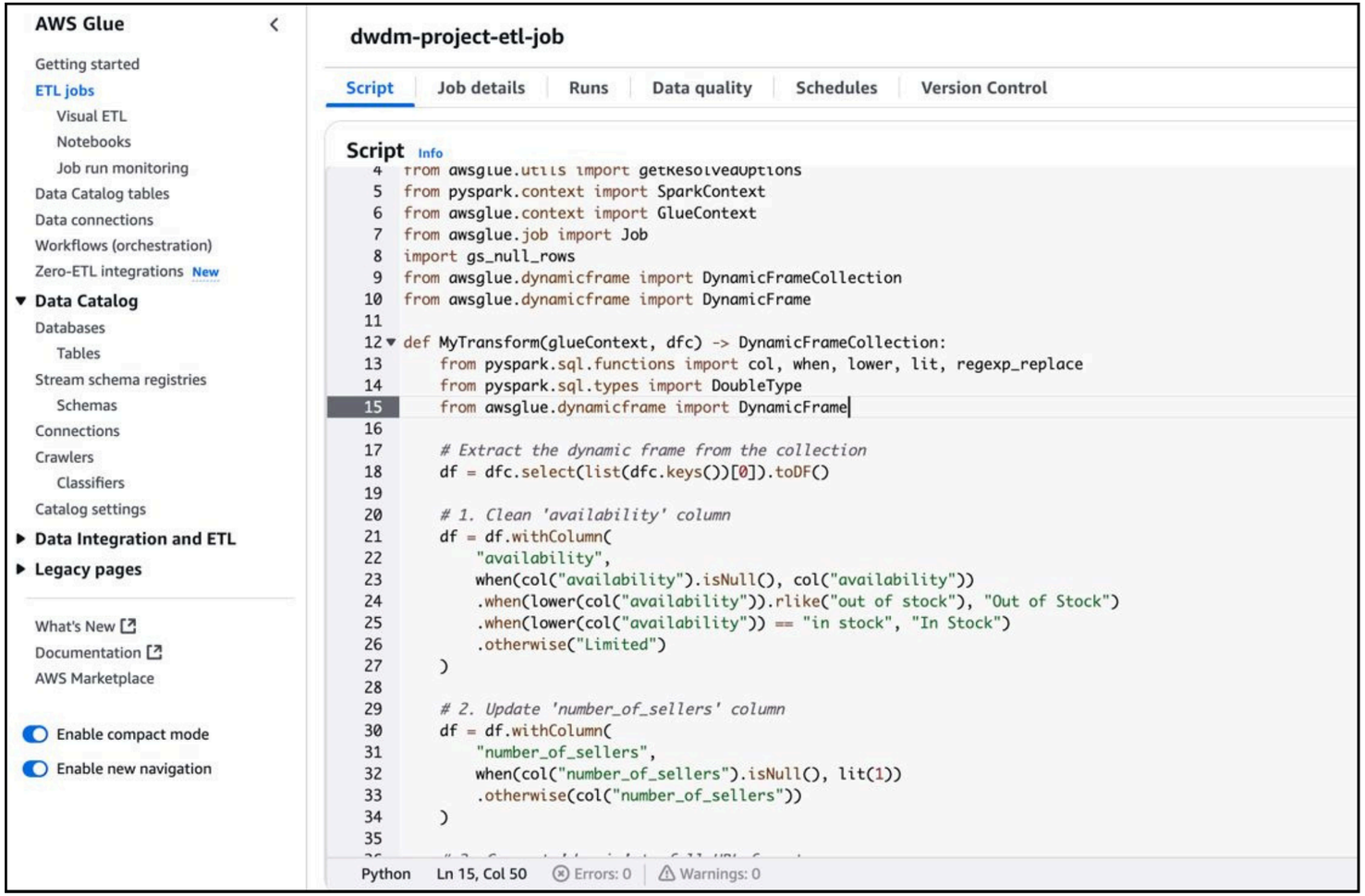
- Domain
  - Converted shortened domain names (e.g., www.amazon.com) to full URLs like "https://www.amazon.com/".
  - If missing, set to default "https://amazon.com/".
- Manufacturer
  - Replaced null or missing values with "unknown".
- Plus Content
  - If the field was null, it was filled with False.
- Top Review
  - If no review was present, replaced with "no review".

Transformation

- Null Removal
  - Entire rows or columns with all null values were dropped
- Data Type Conversion
  - Values in string or scientific notation were converted to standard double format.

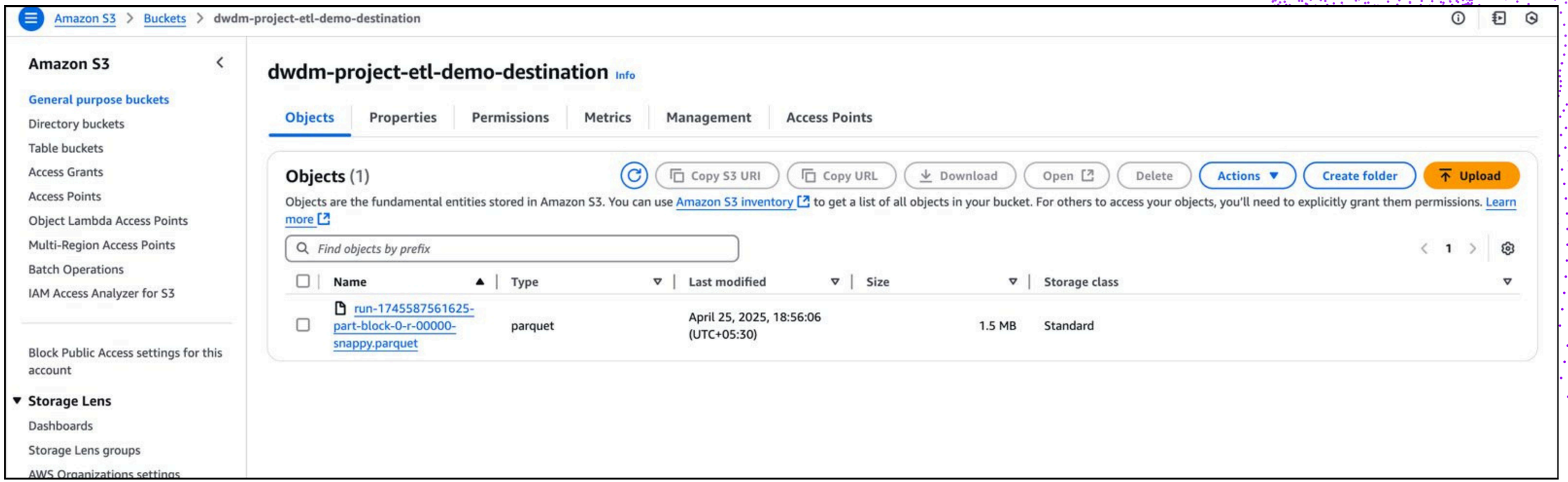
Transformation

Transformation script



Transformation

Transformation destination S3 bucket

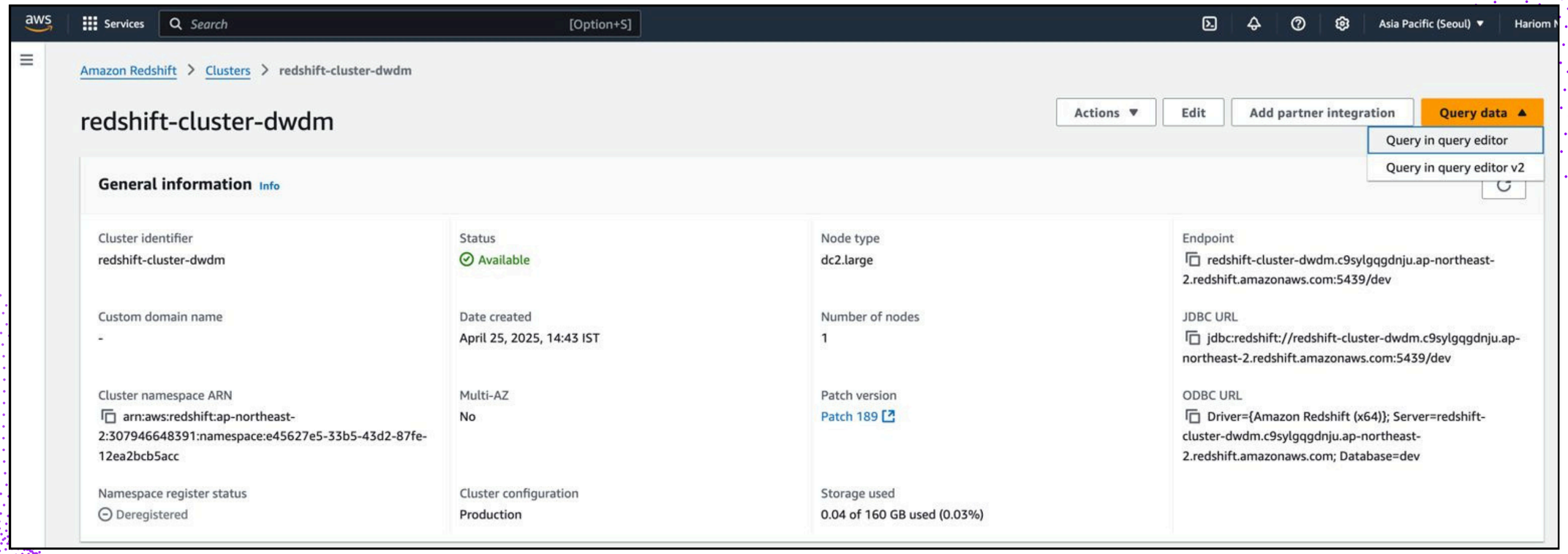


Load

In the final Load phase of the ETL pipeline, the transformed data — now stored in the efficient Parquet file format — was loaded into **Amazon Redshift**, a fully managed, petabyte-scale data warehouse solution. A custom **schema was predefined** in Redshift to match the structure of the transformed dataset. This ensured proper column mapping and optimized query performance. The data was loaded from the Parquet files directly stored in **Amazon S3**, with the bucket path specified as the data source during the Redshift COPY operation.

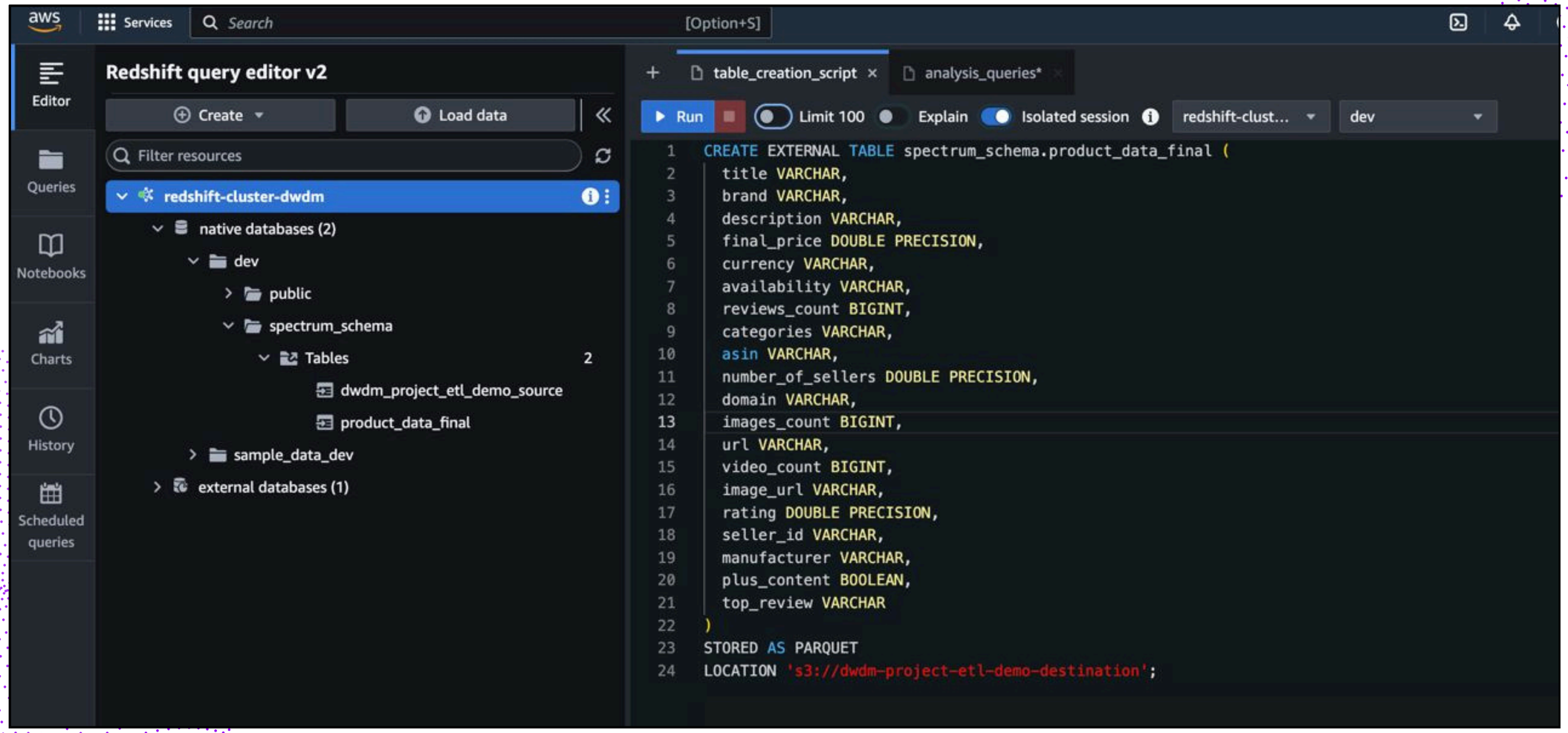
Load

Amazon Redshift Cluster



Load

Table Creation Script



SQL Queries Doc Link

click here

THANKYOU