

Project Flow



User Creation: I began by creating a new user with Admin access to ensure appropriate permissions for managing resources.

**IAM
USER**

The screenshot shows the AWS IAM console interface. The left sidebar contains navigation links for Identity and Access Management (IAM), including Dashboard, Access management (User groups, Roles, Policies, Identity providers, Account settings, Root access management), and Access reports (Access Analyzer, External access, Unused access, Analyzer settings, Credential report, Organization activity, Service control policies). The main content area displays a notification about Identity Center, a 'Ready to streamline human access to AWS and cloud apps?' message, and a 'Users (1)' section. The 'Users (1)' section includes a search bar and a table listing users. The table has columns for User name, Path, Group, Last activity, MFA, Password age, and Console last sign-in. The user 'analyticsengineer-aditya-google' is listed with a last activity of 4 hours ago and a password age of 10 hours. The bottom of the page shows the CloudShell interface and footer information.

<input type="checkbox"/>	User name	Path	Group	Last activity	MFA	Password age	Console last sign-in
<input type="checkbox"/>	analyticsengineer-aditya-google	/	0	4 hours ago		10 hours	April 19, 2025, 15:16 (...)

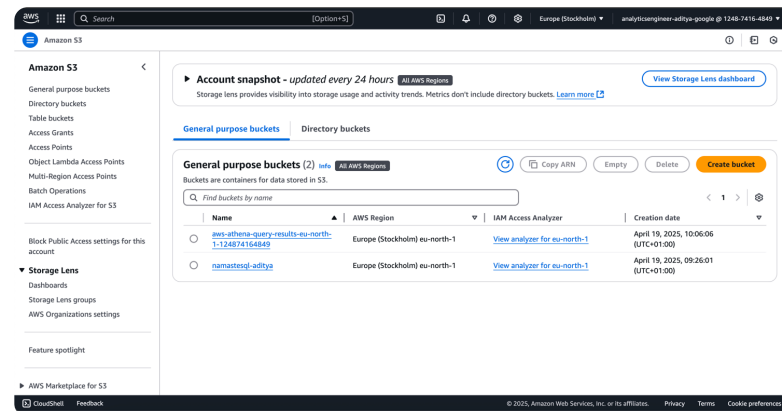


S3 Bucket Setup: I created a new S3 bucket to store the order data. Within this bucket, I organized the data by creating folders specifically for the order records.

Folder and Subfolder Structure: To ensure efficient data management and organization, I set up subfolders within the order data folder. A naming convention was applied to each subfolder, following the format Snapshot=YYYY-MM-DD (e.g., Snapshot=2027-01-01). This structure allowed me to create partitions based on the date.

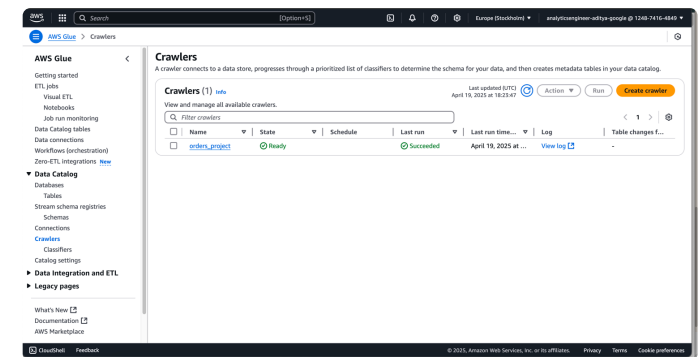
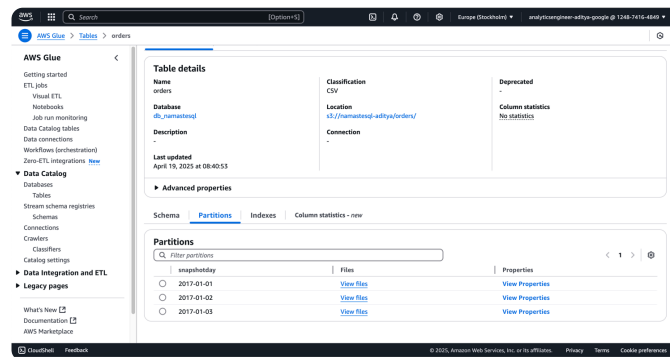
Data Upload: The order data, which was in CSV format, was uploaded to the relevant subfolders in S3, ensuring that each file was properly categorized within its respective partition.

S3 Bucket

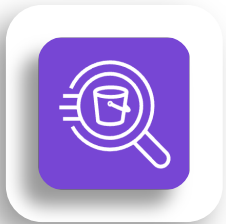




Crawler Setup: I created a new AWS Glue Crawler to automatically detect and catalog the data in S3. This helped in organizing the data in a way that could easily be queried using Athena.



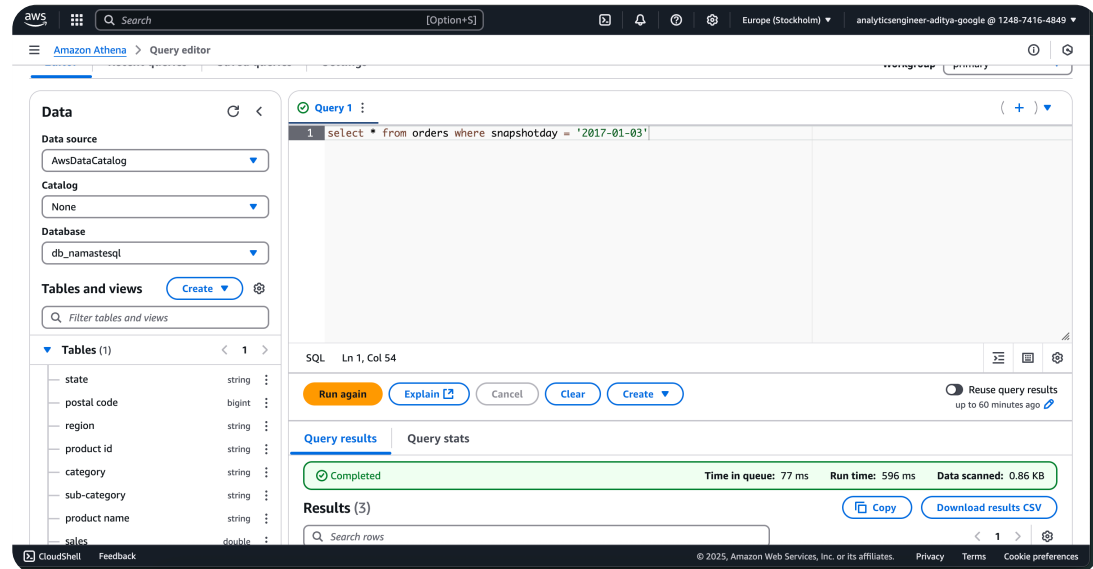
Glue



Data Query in Athena: After setting up the crawler, I used AWS Athena to query the uploaded data. I specified the query result location in S3 for storage and made sure that I could access the results efficiently.

Partition Optimization: One of the key optimizations was partitioning the data. By querying specific partitions (based on the snapshot date), I was able to significantly reduce the query cost. Instead of querying the entire dataset, only the relevant partition was accessed, which improved both performance and cost efficiency.

Athena





QuickSight Setup: To enable quick data visualization, I created a new Amazon QuickSight account. I connected QuickSight to Athena and selected the relevant dataset for analysis.

SPICE Import: To enhance visualization speed, I imported the data into SPICE (Super-fast, Parallel, In-memory Calculation Engine). This allowed for faster and more interactive dashboards.

Data Visualization: Using QuickSight, I created various charts and visualizations to better understand the data. Once finalized, I published these dashboards for stakeholders to access and interact with.

Quick
Sight

