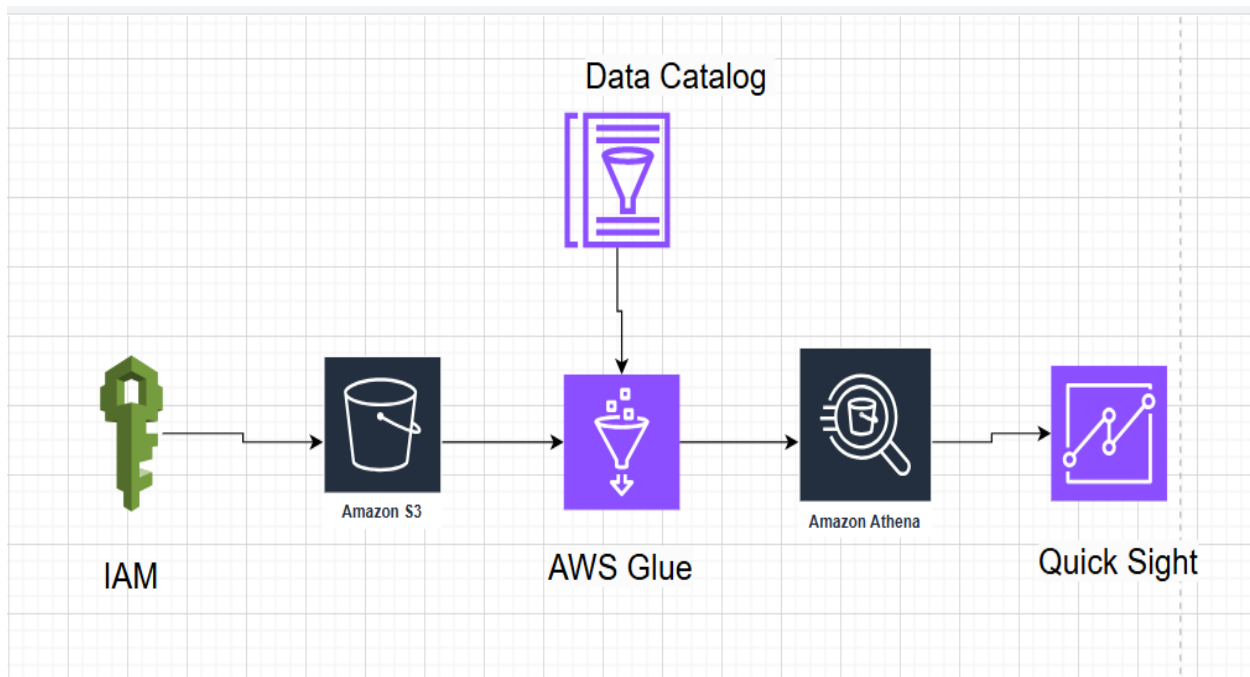


AWS Analytics: Project Documentation

Prepared by: Sagar Naduvinkeri - [LinkedIn](#)

Project Overview

The objective of this project was to demonstrate end-to-end proficiency in AWS Analytics services. This included securely storing data, automating schema discovery, querying datasets without a traditional database, and presenting insights through dynamic dashboards. By leveraging AWS S3, Glue, Athena, and QuickSight, I built a scalable, efficient, and cost-effective analytics pipeline — a typical workflow for modern cloud-based data analysis.



Step 1: Data Storage with IAM and Amazon S3

1.1 IAM Configuration

To maintain AWS best practices for security and access control, I began by creating a dedicated IAM user for this project. The user was granted limited permissions aligned with the principle of least privilege, including access to Amazon S3, Glue, Athena, and QuickSight. This approach allowed secure resource management and better auditability.

1.2 Amazon S3 Bucket Setup

I created an Amazon S3 bucket, which served as the primary data lake for this project. The raw dataset, an Excel file, was uploaded here. Notably, I uploaded the dataset in snapshot form, partitioned by upload timestamps. This snapshotting strategy offers the following advantages:

- Reduces the need to re-scan entire datasets repeatedly.
- Optimizes Athena query performance.
- Simplifies historical comparisons by maintaining versioned datasets.

Amazon S3 proved to be a highly durable, cost-effective, and scalable storage layer, perfect for data lake architecture.

General purpose buckets (1)

Info

All AWS Regions

Copy ARN

Empty

Delete

Create bucket

Buckets are containers for data stored in S3.

Find buckets by name

< 1 >

	Name	AWS Region	IAM Access Analyzer	Creation date
<input type="radio"/>	sagarnaduvinkeri	US East (Ohio) us-east-2	View analyzer for us-east-2	May 24, 2025, 19:34:36 (UTC-04:00)

Objects (3)

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
<input type="checkbox"/>	<div><div></div>Athena_logs/</div>	Folder	-	-	-
<input type="checkbox"/>	<div><div></div>snapshot_day=01-01-2017/</div>	Folder	-	-	-
<input type="checkbox"/>	<div><div></div>snapshot_day=01-02-2017/</div>	Folder	-	-	-

Step 2: Automating Metadata Discovery with AWS Glue

2.1 AWS Glue Overview

AWS Glue is a fully managed **serverless data integration** service that simplifies ETL (Extract, Transform, Load) operations. It helps automate the discovery and cataloging of datasets without manual schema definitions, making it ideal for dynamic or semi-structured data environments.

2.2 Glue Crawler Configuration

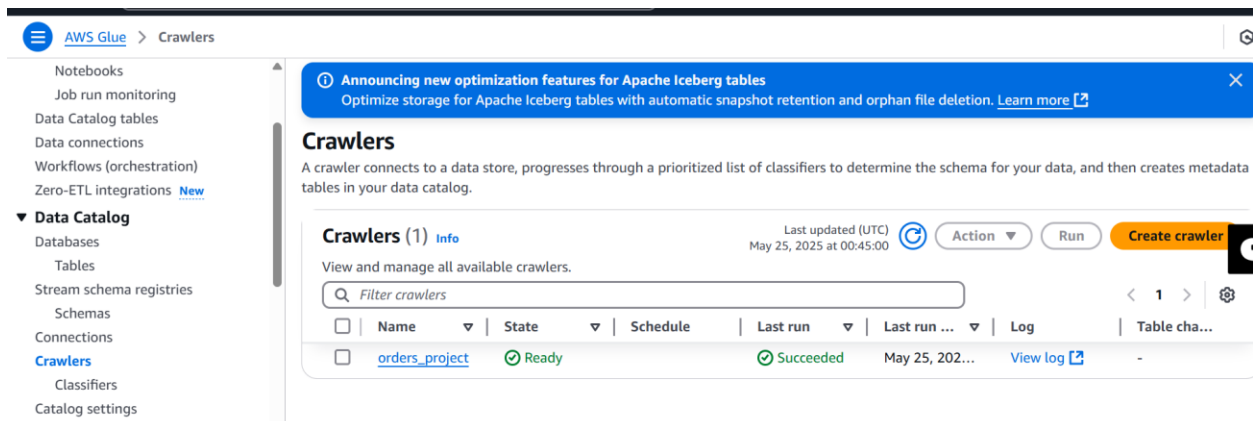
I configured an **AWS Glue Crawler** to scan the S3 bucket. The crawler automatically:

- Detected the uploaded Excel files.
- Inferred the schema from the content.
- Created tables in the AWS Glue **Data Catalog**.

The crawler scheduled runs enabled automatic updates to the schema when new snapshots are added, ensuring up-to-date metadata. Glue's ability to work with multiple data formats (e.g., CSV, Parquet, JSON, Excel) makes it extremely versatile for data analysts.

Benefits in This Scenario:

- Eliminates manual schema mapping.
- Keeps metadata updated as new data arrives.
- Lays the foundation for seamless querying via Athena.



The screenshot displays the AWS Glue console interface. On the left, a navigation pane lists various services including Notebooks, Data Catalog tables, and Crawlers. The main content area shows the 'Crawlers' section with a blue notification banner at the top. Below the banner, a description of a crawler is provided. A table lists the available crawlers, with one crawler named 'orders_project' shown in a 'Ready' state, having successfully completed its last run on May 25, 2025. The table includes columns for Name, State, Schedule, Last run, Last run status, Log, and Table changes.

Name	State	Schedule	Last run	Last run ...	Log	Table cha...
orders_project	Ready		✓ Succeeded	May 25, 202...	View log	-

Crawler runs	Schedule	Data sources	Classifiers	Tags
---------------------	----------	--------------	-------------	------

Crawler runs (4)

Stop run

View CloudWatch logs

View run detail

The list of crawler runs for this crawler.

Filter by a date and time range

<

1

>

	Start time (UTC) ▲	End time (UTC) ▼	Current/last duration ▼	Status ▼	DPU hours
<input type="radio"/>	May 25, 2025 at 00:19:29	May 25, 2025 at 00:20:12	43 s	✔ Completed	
<input type="radio"/>	May 25, 2025 at 00:17:19	May 25, 2025 at 00:18:02	43 s	✔ Completed	
<input type="radio"/>	May 24, 2025 at 23:57:35	May 24, 2025 at 23:58:18	43 s	✔ Completed	
<input type="radio"/>	May 24, 2025 at 23:52:39	May 24, 2025 at 23:53:21	42 s	✔ Completed	

Step 3: Serverless Querying with Amazon Athena

3.1 Athena Integration

With the tables created in the Glue Data Catalog, I used **Amazon Athena** to query the datasets directly from S3 — without moving the data. Athena supports SQL syntax, which made querying intuitive and powerful.

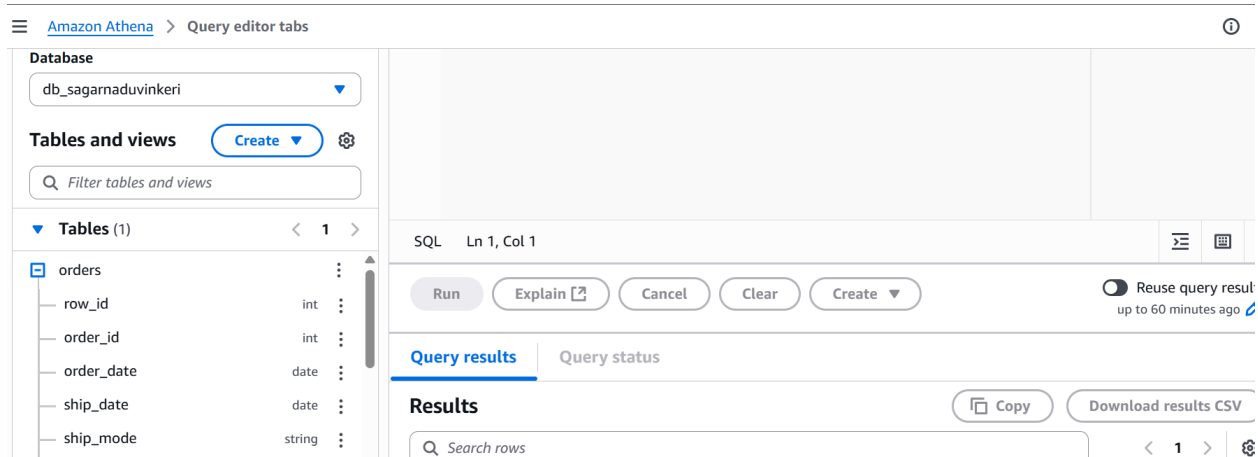
3.2 Use Cases Demonstrated:

- Filtering and aggregating sales data.
- Trend analysis using timestamped snapshots.
- Joins and transformations across multiple partitions (if needed).

Why Athena is Ideal:

- Serverless: No infrastructure provisioning required.
- Pay-per-query: Cost-effective since you only pay for scanned data.
- Speed: Combined with snapshot design and partitioning, queries were faster and efficient.

This step highlighted the power of AWS's **serverless data lake architecture**, reducing overhead and accelerating insights.



Step 4: Business Intelligence with Amazon QuickSight

4.1 QuickSight Setup

Finally, I visualized the processed data using **Amazon QuickSight** — AWS's native BI tool. I connected QuickSight directly to Athena, allowing real-time querying of the underlying S3 datasets.

4.2 Dashboard Creation

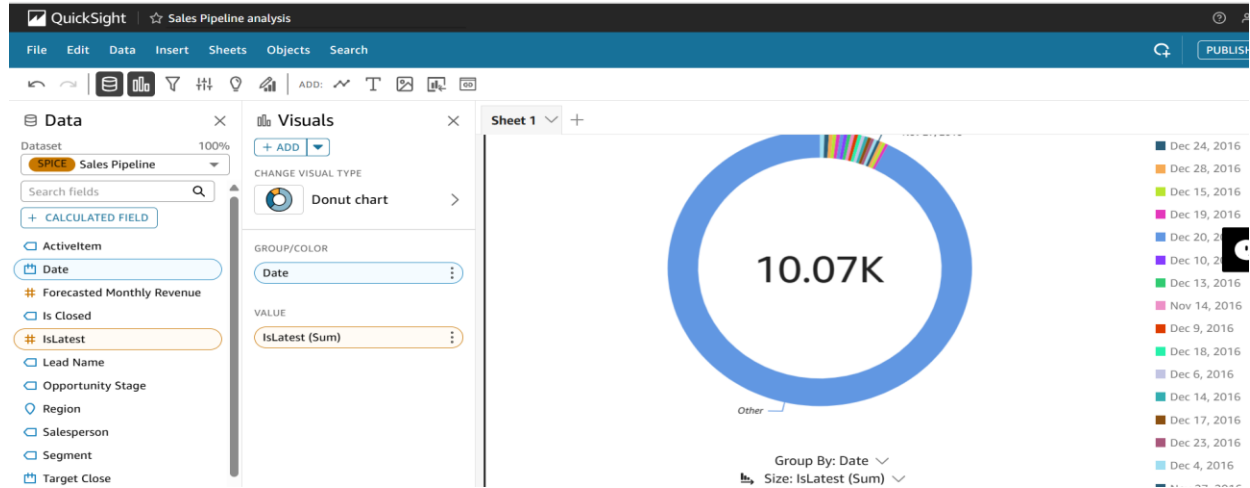
I created an interactive dashboard with:

- Time-series charts tracking trends.
- Pie charts to visualize categorical distributions.
- KPIs (Key Performance Indicators) to highlight metrics like totals, averages, and growth.

QuickSight allowed easy sharing of dashboards with stakeholders and decision-makers, making the project results both **actionable and visually impactful**.

Key Features Utilized:

- Direct integration with Athena.
- Drag-and-drop dashboard building.
- Scheduled refresh for live data monitoring.



Conclusion: This project effectively demonstrates the **power, simplicity, and integration** of AWS Analytics tools (IAM, S3, Glue, Crawler, Athena and Quickshight). Each component worked seamlessly to form an efficient **end-to-end data analytics pipeline**.