**Web Page: Student Data Entry Form**

The screenshot below (not embedded, but described for documentation) shows the web-based student data entry form used in this project. This form allows users to input student details and their subject marks. The fields included are:

- Registration Number (Reg No)

- Name

- Class

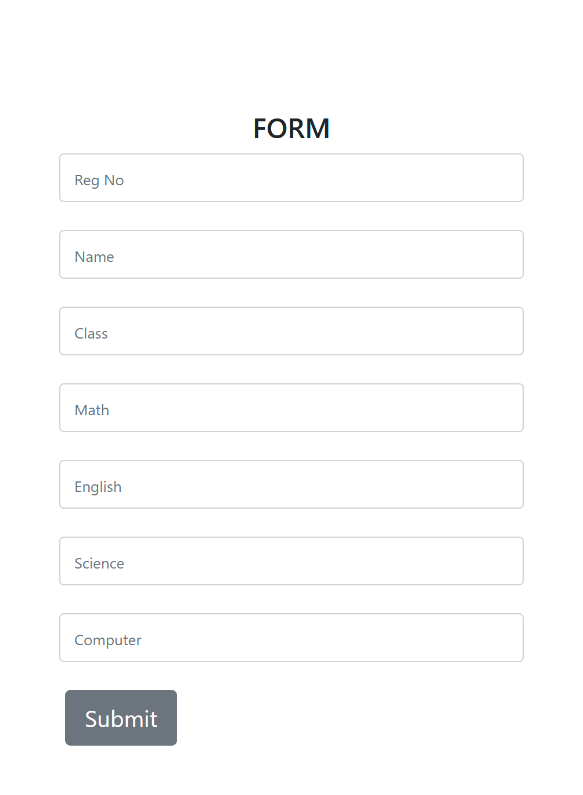
- Math

- English

- Science

- Computer

After filling in the required information, users can submit the form to save the data. This interface ensures that all necessary student and academic information is collected in a structured and user-friendly manner, supporting accurate data entry for further processing in the ETL pipeline.



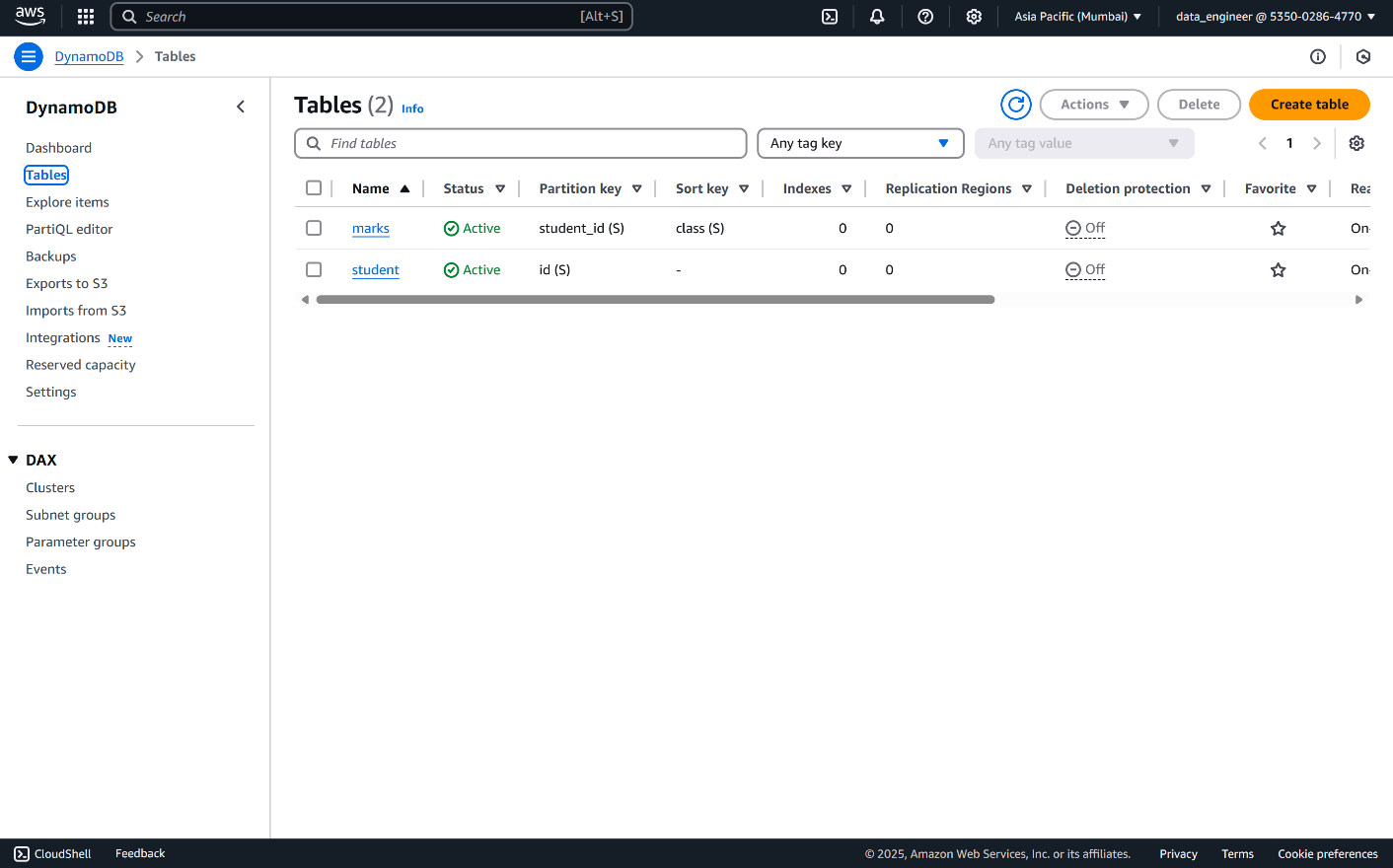
**AWS DynamoDB: Table Overview**

The screenshot below (not embedded, but described for documentation) displays the AWS DynamoDB console, showing two tables used in this project:

- marks: Stores student marks data. The partition key is `student\_id` and the sort key is `class`.

- student: Stores student information. The partition key is `id`.

Both tables are in Active status, ensuring high availability and reliability for data storage. These tables are integral to the backend data architecture, supporting efficient storage and retrieval of student and marks data for the ETL and analytics processes.



## AWS Lambda: Function Overview

The screenshot below (not embedded, but described for documentation) shows the AWS Lambda function named `project-datewithdata`. This Lambda function is integrated with DynamoDB and is responsible for processing data events or automating backend workflows within the project.

\*\*Key details:\*\*

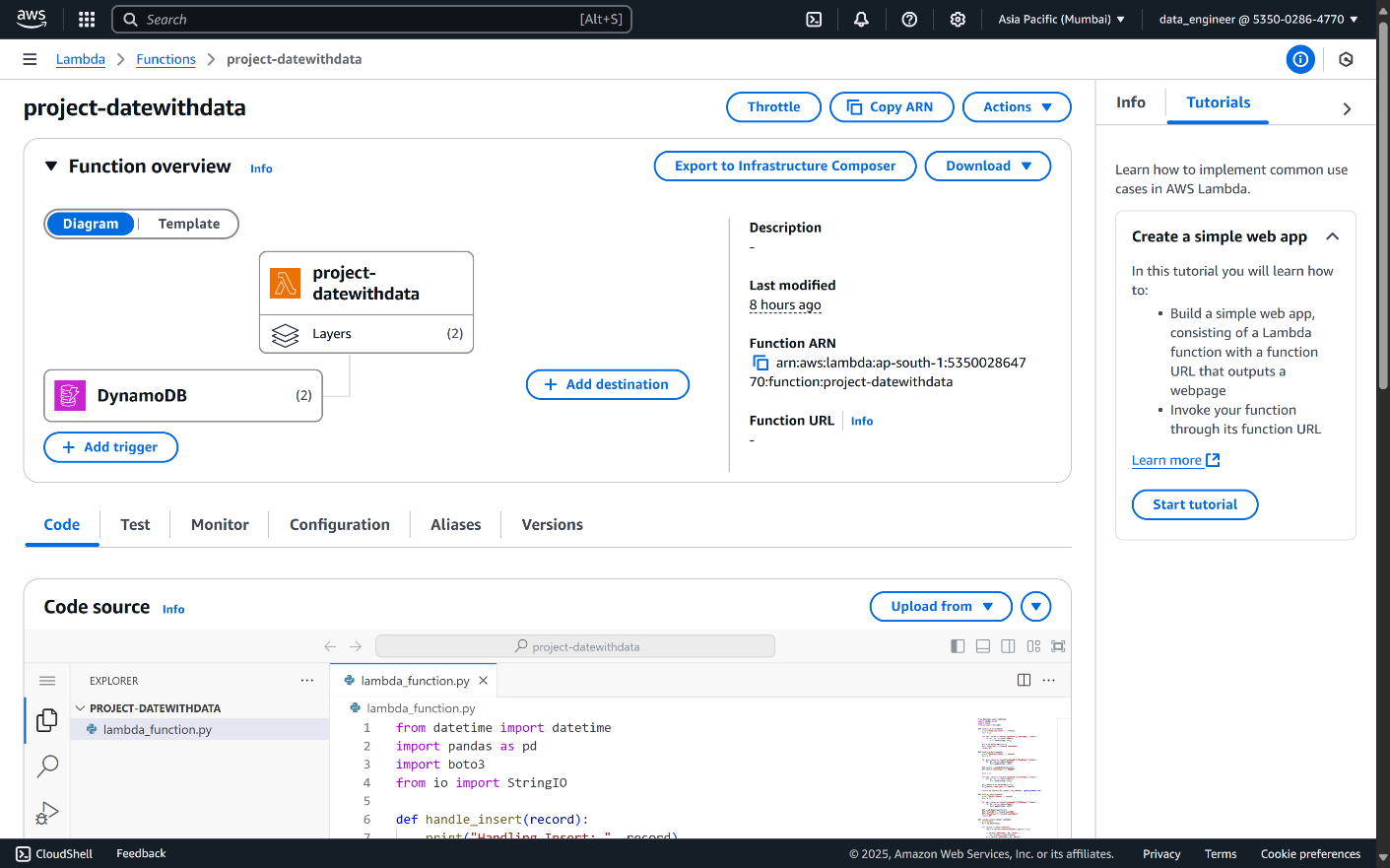
- The function interacts with DynamoDB tables to handle data operations.

- It utilizes additional layers for extended functionality.

- The code section shows the use of Python libraries such as `datetime`, `pandas`, and `boto3` for data processing and AWS service integration.

- The Lambda function is triggered by events (such as data inserts or updates) and can automate ETL or data transformation tasks.

This serverless component ensures scalable, event-driven processing and seamless integration with other AWS services, supporting the automation and reliability of the data pipeline.



## AWS S3: Data Storage and ETL Integration

Amazon S3 (Simple Storage Service) is used as the primary data storage solution in this project. S3 buckets serve as both the staging area for raw data and the data warehouse for processed outputs.

\*\*Key details:\*\*

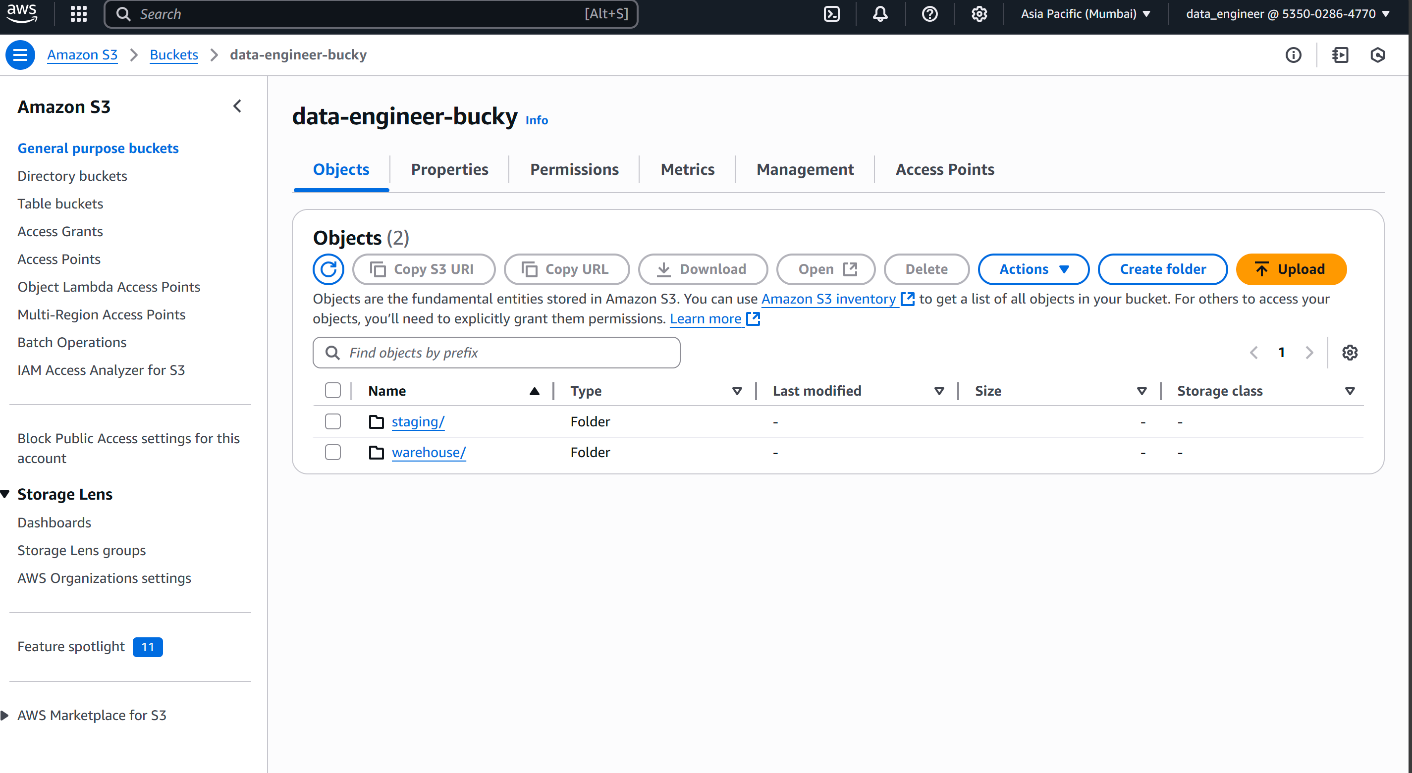
- \*\*Staging Area:\*\* Raw student and marks data files are uploaded to designated S3 bucket paths. These files are typically in CSV format and serve as the input for the ETL process.

- \*\*Data Warehouse:\*\* After ETL processing, the transformed and joined data is stored in S3 in Parquet format, optimized for analytics and reporting.

- \*\*Integration:\*\* S3 is tightly integrated with AWS Glue (for ETL), Lambda (for automation), and other AWS services, enabling seamless data flow and automation.

- \*\*Scalability & Durability:\*\* S3 provides highly scalable and durable storage, ensuring that all data is securely stored and easily accessible for further processing or analysis.

This architecture leverages S3's strengths to support a robust, scalable, and cost-effective data pipeline for the student data management system.



## AWS Glue: ETL Job Execution

The screenshot below (not embedded, but described for documentation) shows the AWS Glue console with a successful ETL job run. AWS Glue is used to automate the extract, transform, and load (ETL) process for student and marks data in this project.

\*\*Key details:\*\*

- The job, named `GlueJob`, completed successfully with a status of "Succeeded."

- The ETL job reads raw data from S3, performs data transformations and joins, and writes the processed data back to S3 in an optimized format.

- The job run details include:

- Start and end times

- Duration (1 minute 1 second)

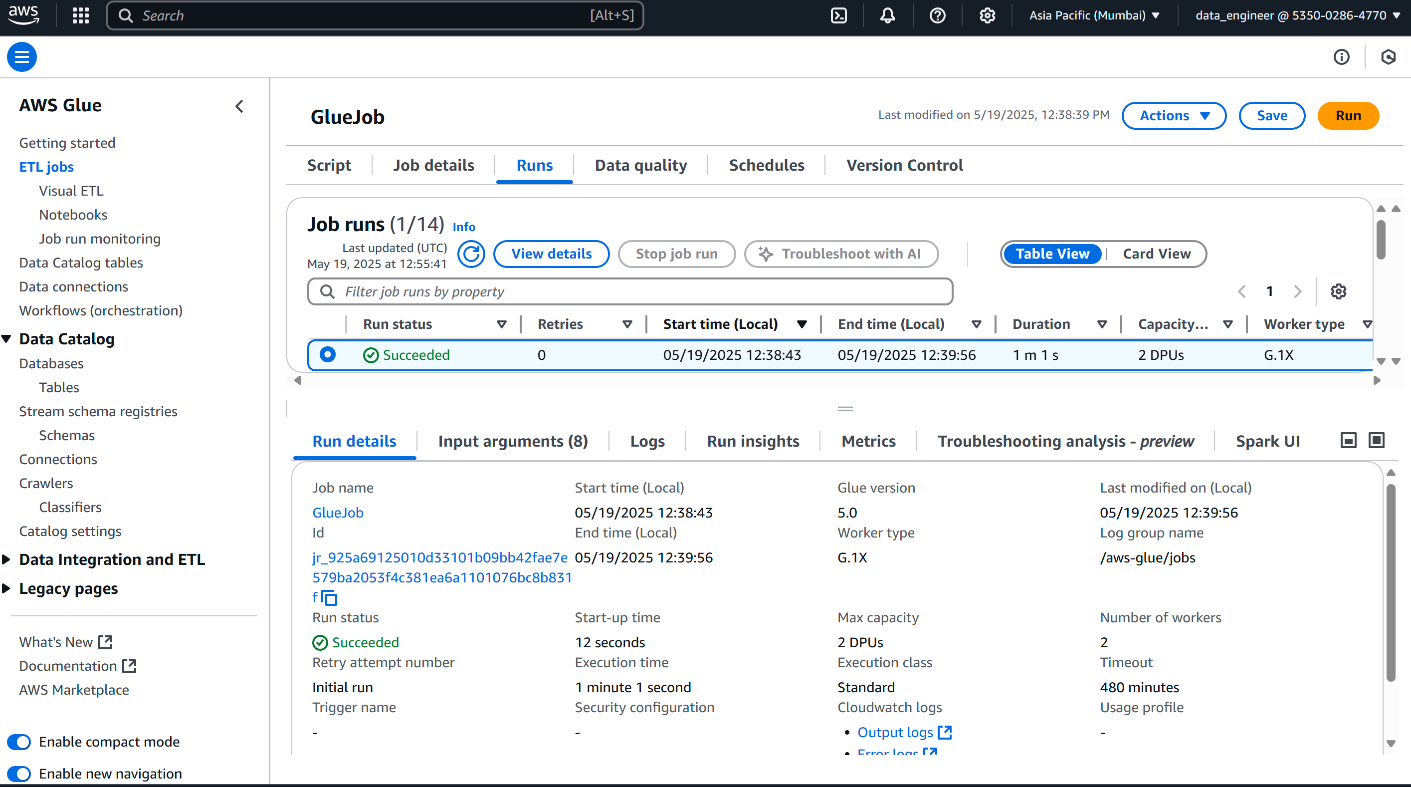
- Worker type and capacity (2 DPUs, G.1X)

- Glue version (5.0)

- Log links for output and error logs

- Glue jobs are highly scalable and can be scheduled or triggered by events, ensuring timely and reliable data processing.

This ETL automation is central to maintaining up-to-date, high-quality data in the student data management system, supporting analytics and reporting needs.



**AWS CloudWatch: Monitoring and Logging**

The screenshot below (not embedded, but described for documentation) shows the AWS CloudWatch console, specifically the log group for the Lambda function `project-datewithdata`.

Key details:

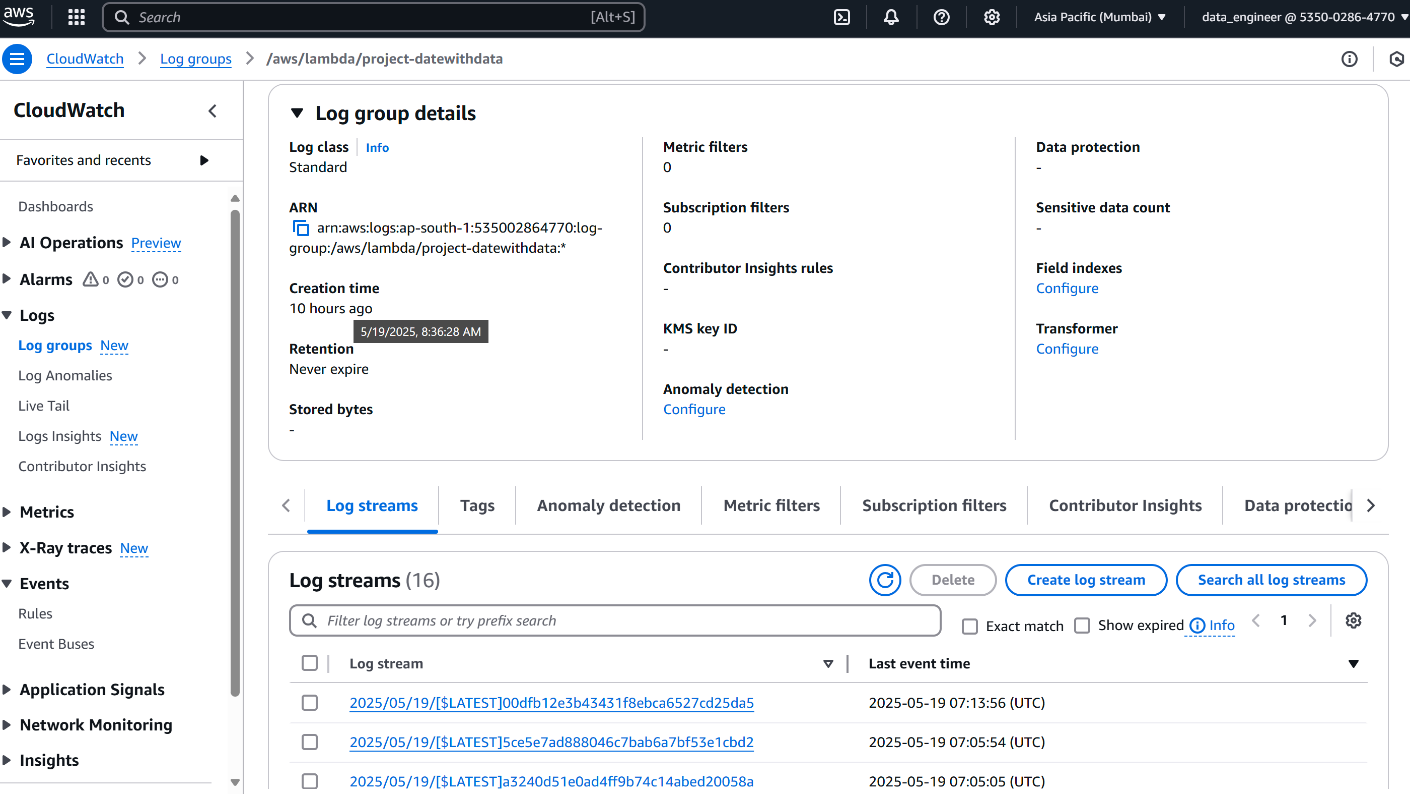
- CloudWatch collects and stores logs generated by AWS Lambda, Glue, and other services in the project.

- The log group `/aws/lambda/project-datewithdata` contains multiple log streams, each representing a sequence of events or executions.

- Log details include creation time, retention policy, and event timestamps, supporting traceability and troubleshooting.

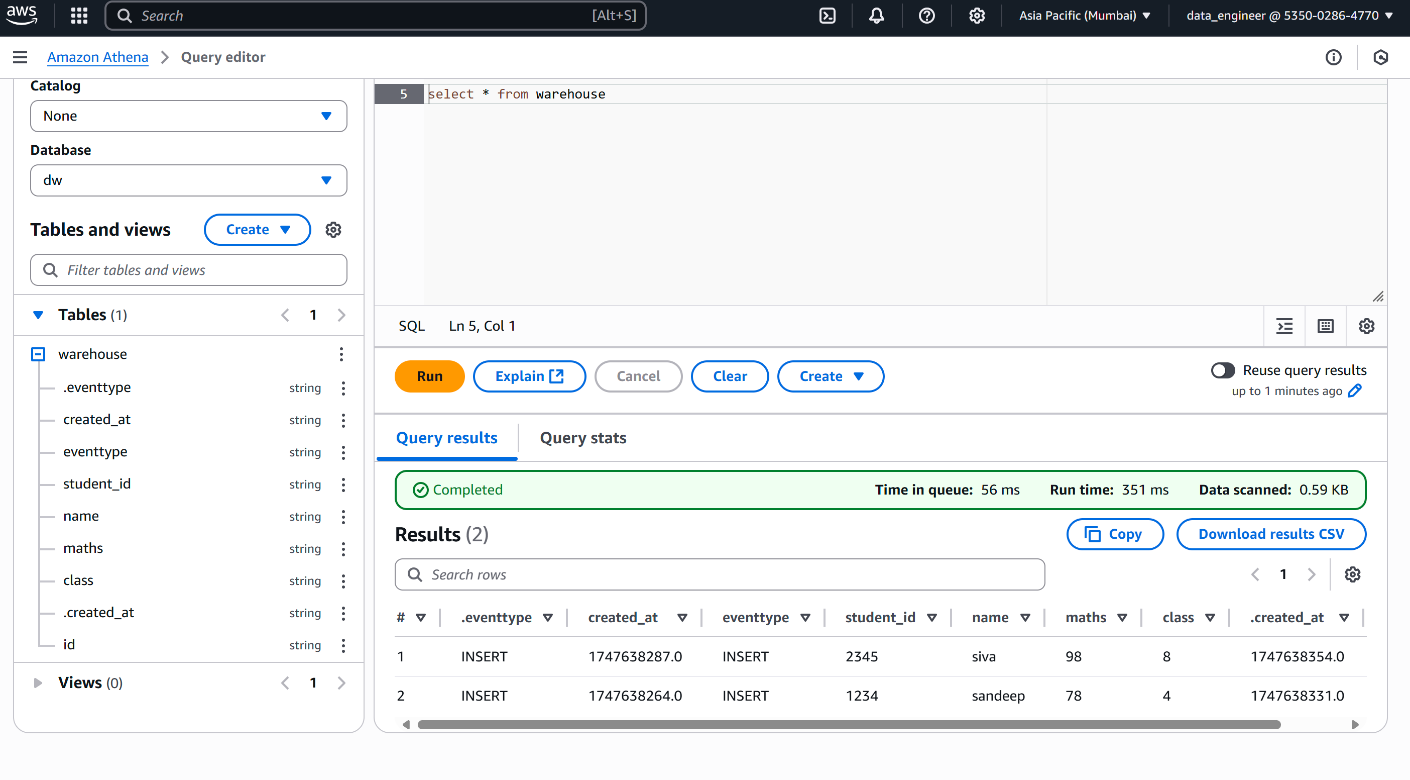
- CloudWatch enables real-time monitoring, alerting, and analysis of application and infrastructure logs.

This logging and monitoring setup is essential for maintaining operational visibility, diagnosing issues, and ensuring the reliability and performance of the student data management system.



**Amazon Athena: Querying and Analyzing Data**

The screenshot below (not embedded, but described for documentation) shows the Amazon Athena query editor being used to analyze data stored in the S3-backed data warehouse.



Key details:

- Athena allows users to run SQL queries directly on data stored in S3, making it easy to analyze large datasets without the need for complex infrastructure.

- The example query (`SELECT \* FROM warehouse`) retrieves all records from the `warehouse` table in the `dw` database.

- The table schema includes fields such as `student\_id`, `name`, `maths`, `class`, `created\_at`, and `eventtype`.

- Query results are displayed instantly, and can be downloaded as CSV for further analysis or reporting.

- Athena provides a serverless, pay-per-query solution for interactive data exploration and business intelligence.

This integration enables fast, flexible, and cost-effective analytics on the processed student data, supporting data-driven decision making.