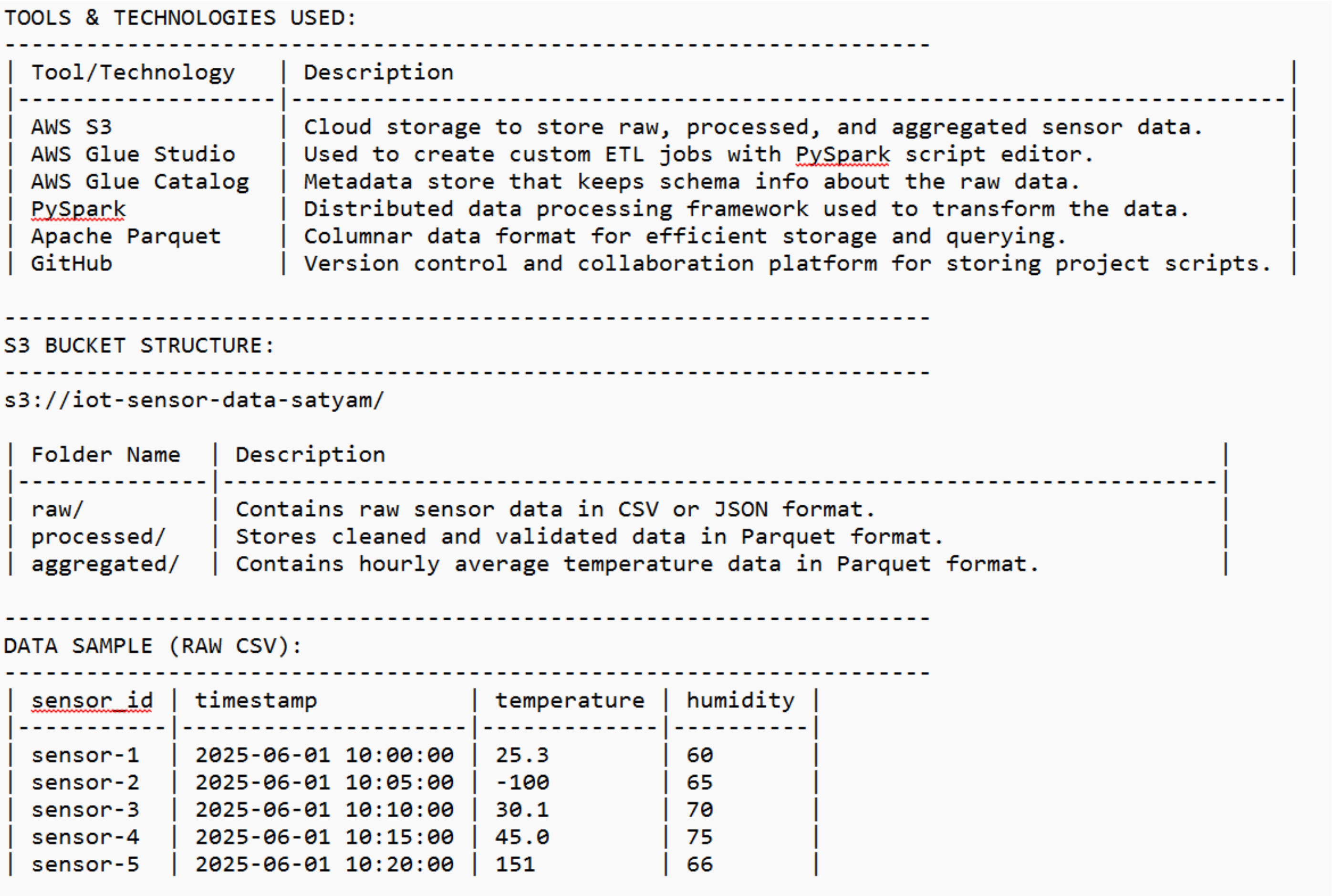
IoT Sensor Data Processor

Short Description

This project demonstrates how to build a complete ETL (Extract, Transform, Load) pipeline using AWS Glue and Amazon S3 to process IoT sensor data. The pipeline reads raw sensor logs (temperature, humidity, timestamps), cleans invalid records, performs transformations, and calculates hourly average temperatures. The cleaned and aggregated data is stored in Parquet format for efficient querying and future analytics.



```
TOOLS & TECHNOLOGIES USED:
--------------------------------------------------------------------
| Tool/Technology    | Description                                                         |
|-------------------|---------------------------------------------------------------------|
| AWS S3             | Cloud storage to store raw, processed, and aggregated sensor data.  |
| AWS Glue Studio    | Used to create custom ETL jobs with PySpark script editor.          |
| AWS Glue Catalog   | Metadata store that keeps schema info about the raw data.           |
| PySpark            | Distributed data processing framework used to transform the data.  |
| Apache Parquet     | Columnar data format for efficient storage and querying.            |
| GitHub             | Version control and collaboration platform for storing project scripts. |

--------------------------------------------------------------------
S3 BUCKET STRUCTURE:
--------------------------------------------------------------------
s3://iot-sensor-data-satyam/

| Folder Name   | Description                                                  |
|--------------|--------------------------------------------------------------|
| raw/          | Contains raw sensor data in CSV or JSON format.              |
| processed/    | Stores cleaned and validated data in Parquet format.         |
| aggregated/   | Contains hourly average temperature data in Parquet format.  |

--------------------------------------------------------------------
DATA SAMPLE (RAW CSV):
--------------------------------------------------------------------
| sensor_id | timestamp            | temperature | humidity |
|-----------|---------------------|-------------|----------|
| sensor-1  | 2025-06-01 10:00:00 | 25.3        | 60       |
| sensor-2  | 2025-06-01 10:05:00 | -100        | 65       |
| sensor-3  | 2025-06-01 10:10:00 | 30.1        | 70       |
| sensor-4  | 2025-06-01 10:15:00 | 45.0        | 75       |
| sensor-5  | 2025-06-01 10:20:00 | 151         | 66       |
```

ETL Workflow

Step 1: Raw Data Ingestion

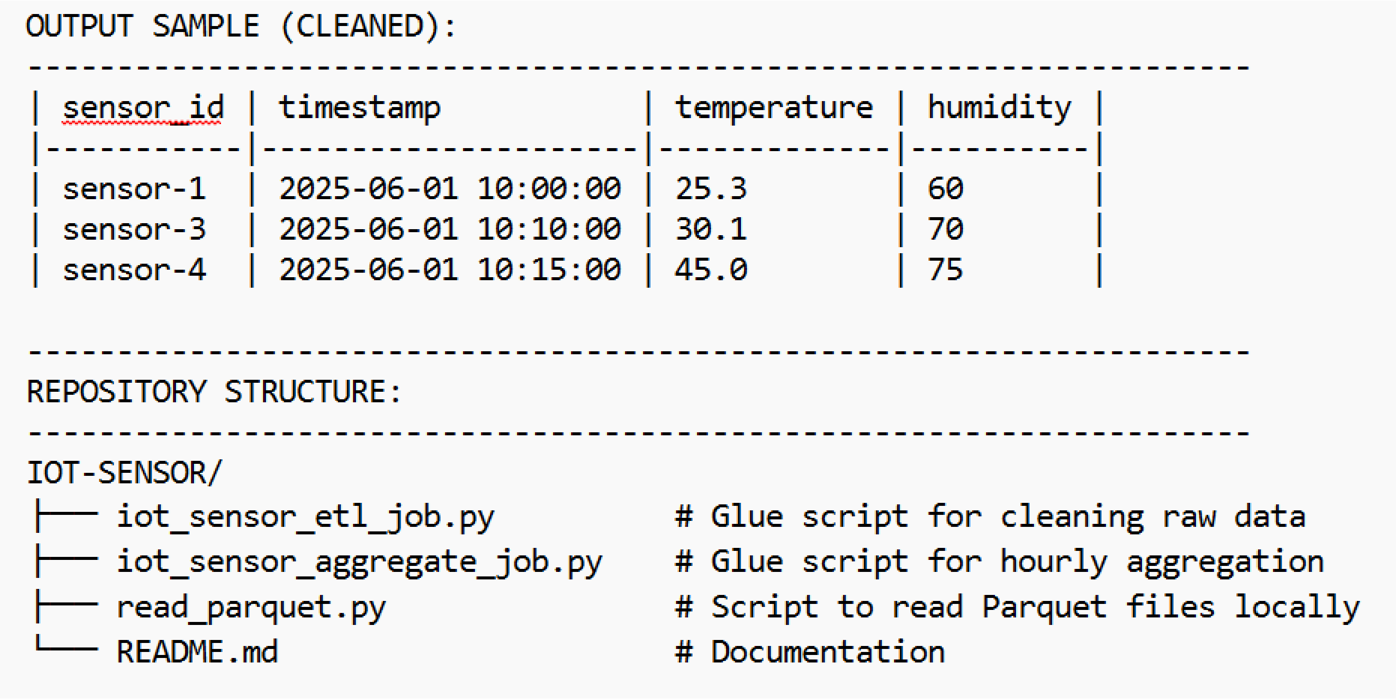
- Upload sensor logs to: s3://iot-sensor-data-satyam/raw/

Step 2: Cleaning & Transformation (ETL Job 1)

- Read data from AWS Glue Data Catalog
- Filter out invalid temperature readings (below -50 or above 150)
- Convert timestamp to datetime format
- Cast humidity from long to double
- Save cleaned data to s3://iot-sensor-data-satyam/processed/ as Parquet

Step 3: Aggregation (ETL Job 2)

- Read cleaned data from processed folder
- Extract hour from each timestamp
- Compute average temperature for each hour
- Save output to s3://iot-sensor-data-satyam/aggregated/ in Parquet



```
OUTPUT SAMPLE (CLEANED):
---------------------------------------------------------------------
| sensor_id | timestamp           | temperature | humidity |
|-----------|---------------------|-------------|----------|
| sensor-1  | 2025-06-01 10:00:00 | 25.3        | 60       |
| sensor-3  | 2025-06-01 10:10:00 | 30.1        | 70       |
| sensor-4  | 2025-06-01 10:15:00 | 45.0        | 75       |

---------------------------------------------------------------------
REPOSITORY STRUCTURE:
---------------------------------------------------------------------
IOT-SENSOR/
├── iot_sensor_etl_job.py          # Glue script for cleaning raw data
├── iot_sensor_aggregate_job.py    # Glue script for hourly aggregation
├── read_parquet.py                # Script to read Parquet files locally
└── README.md                      # Documentation
```

Project Benefits

- **Performance**: Fast reads and queries via Parquet
- **Cost-Efficient**: Only valid, transformed data is saved
- **Flexible**: Fully custom ETL logic using PySpark
- **Scalable**: Easily expandable to more sensors or metrics
- **Analytics Ready**: Compatible with Athena, QuickSight, or Python tools

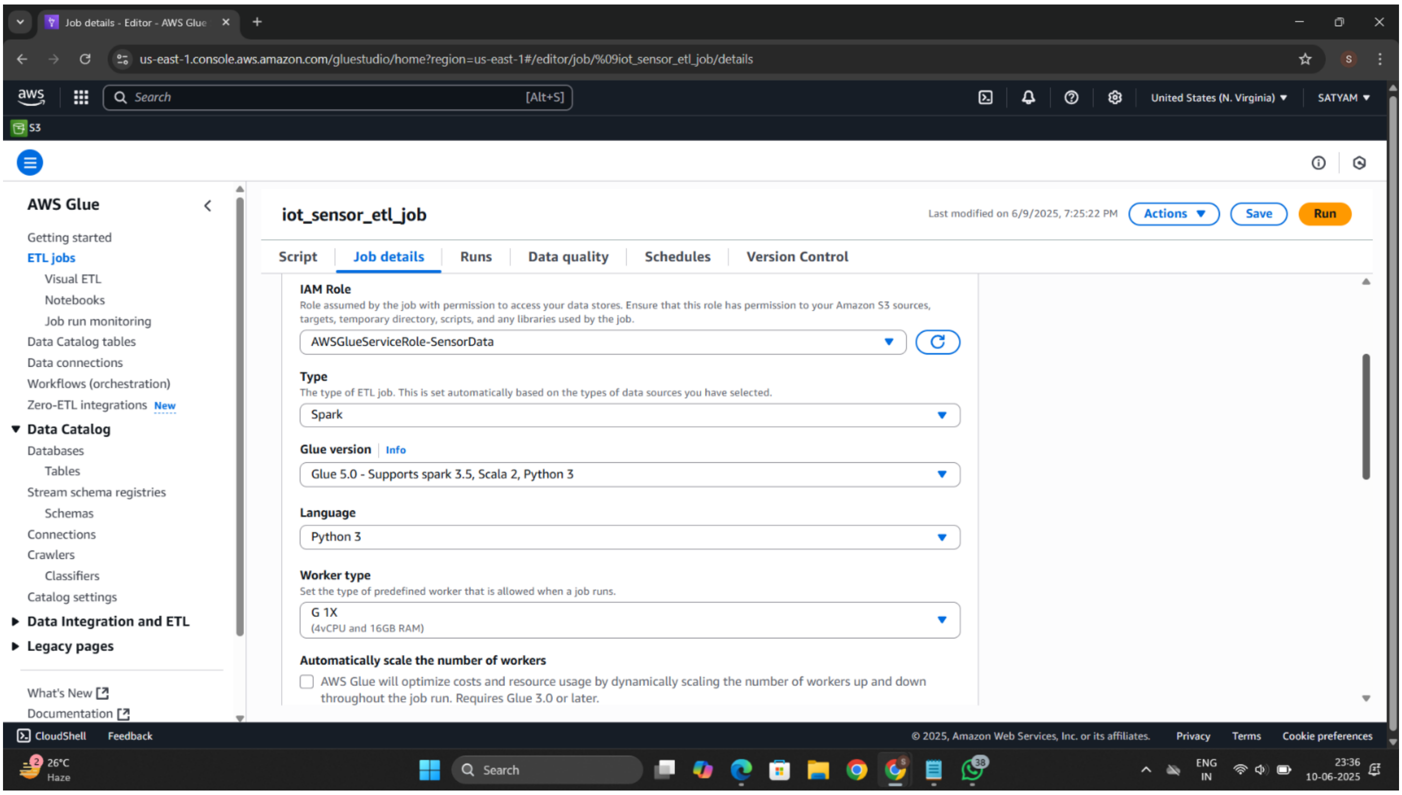
Use Cases

- Real-time environment monitoring
- Industrial IoT sensor data analysis
- Smart home automation
- Weather pattern tracking
- Anomaly detection in sensor readings

Project Repository

Explore the full project code and scripts here:

GitHub Repository: https://github.com/Satyam25613/IOT-SENSOR



iot_sensor_etl_job

IAM Role
Role assumed by the job with permission to access your data stores. Ensure that this role has permission to your Amazon S3 sources, targets, temporary directory, scripts, and any libraries used by the job.
AWSGlueServiceRole-SensorData

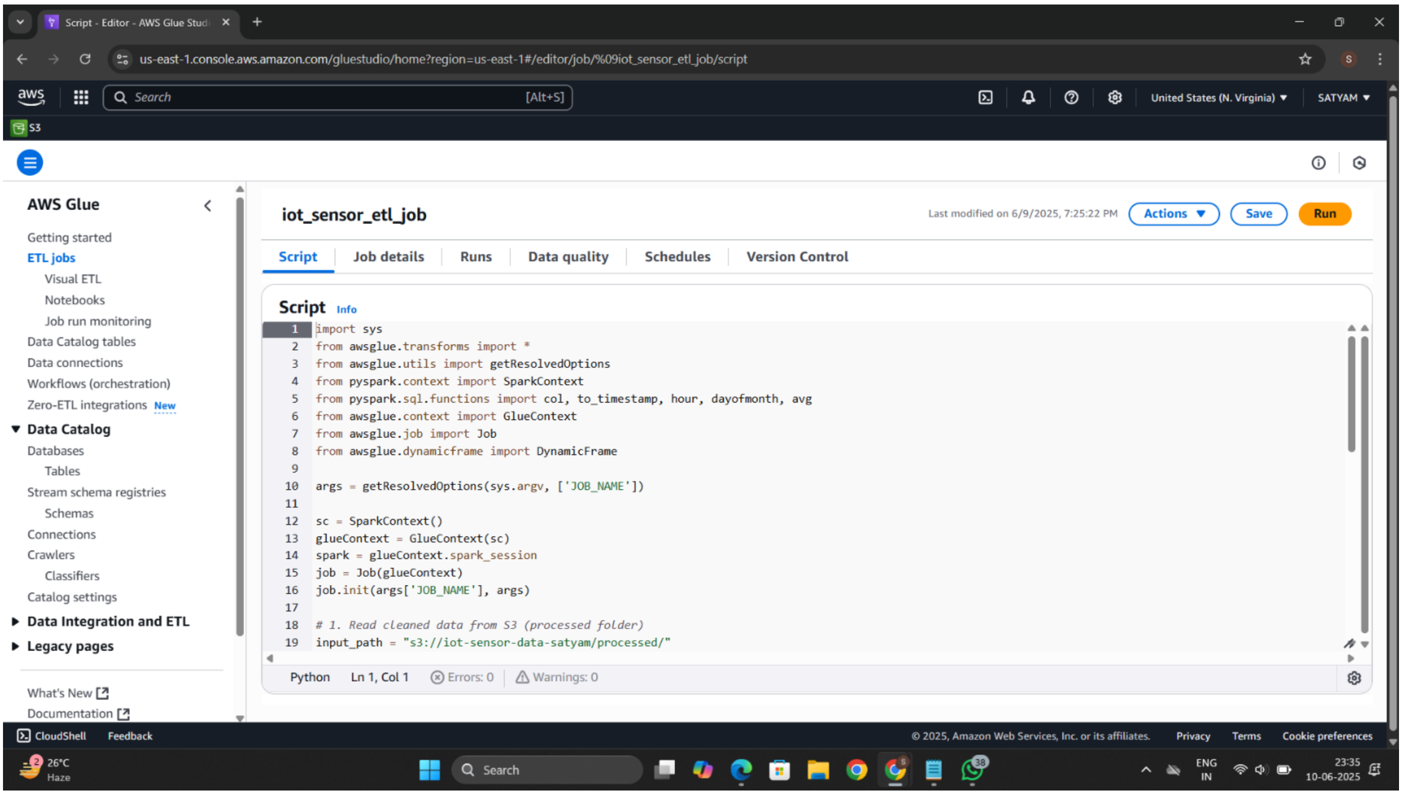
Type
The type of ETL job. This is set automatically based on the types of data sources you have selected.
Spark

Glue version Info
Glue 5.0 - Supports spark 3.5, Scala 2, Python 3

Language
Python 3

Worker type
Set the type of predefined worker that is allowed when a job runs.
G 1X
(4vCPU and 16GB RAM)

Automatically scale the number of workers
□ AWS Glue will optimize costs and resource usage by dynamically scaling the number of workers up and down throughout the job run. Requires Glue 3.0 or later.



iot_sensor_etl_job

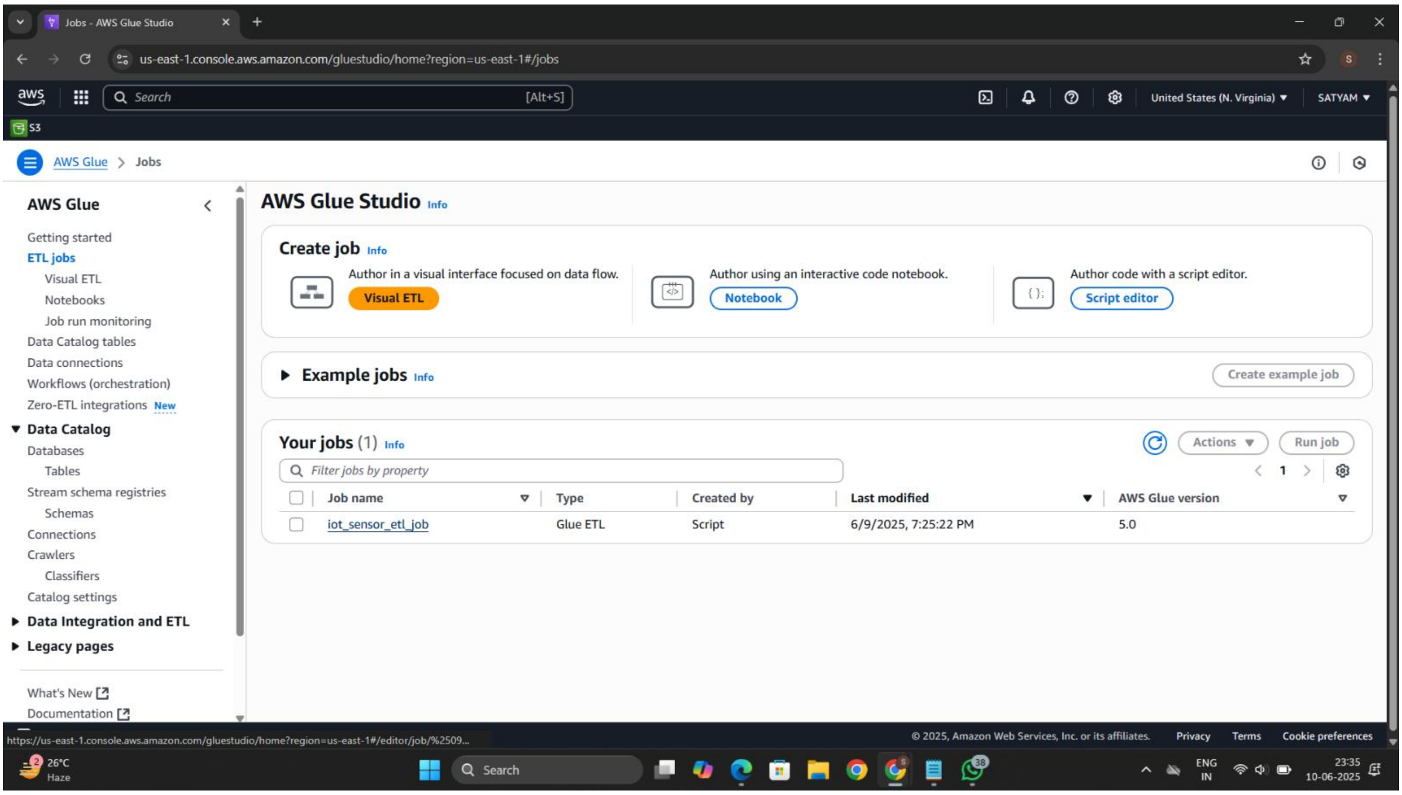
Script Info

```python
import sys
from awsglue.transforms import *
from awsglue.utils import getResolvedOptions
from pyspark.context import SparkContext
from pyspark.sql.functions import col, to_timestamp, hour, dayofmonth, avg
from awsglue.context import GlueContext
from awsglue.job import Job
from awsglue.dynamicframe import DynamicFrame

args = getResolvedOptions(sys.argv, ['JOB_NAME'])

sc = SparkContext()
glueContext = GlueContext(sc)
spark = glueContext.spark_session
job = Job(glueContext)
job.init(args['JOB_NAME'], args)

# 1. Read cleaned data from S3 (processed folder)
input_path = "s3://iot-sensor-data-satyam/processed/"
```



AWS Glue Studio Info

Create job Info

Author in a visual interface focused on data flow.

Author using an interactive code notebook.

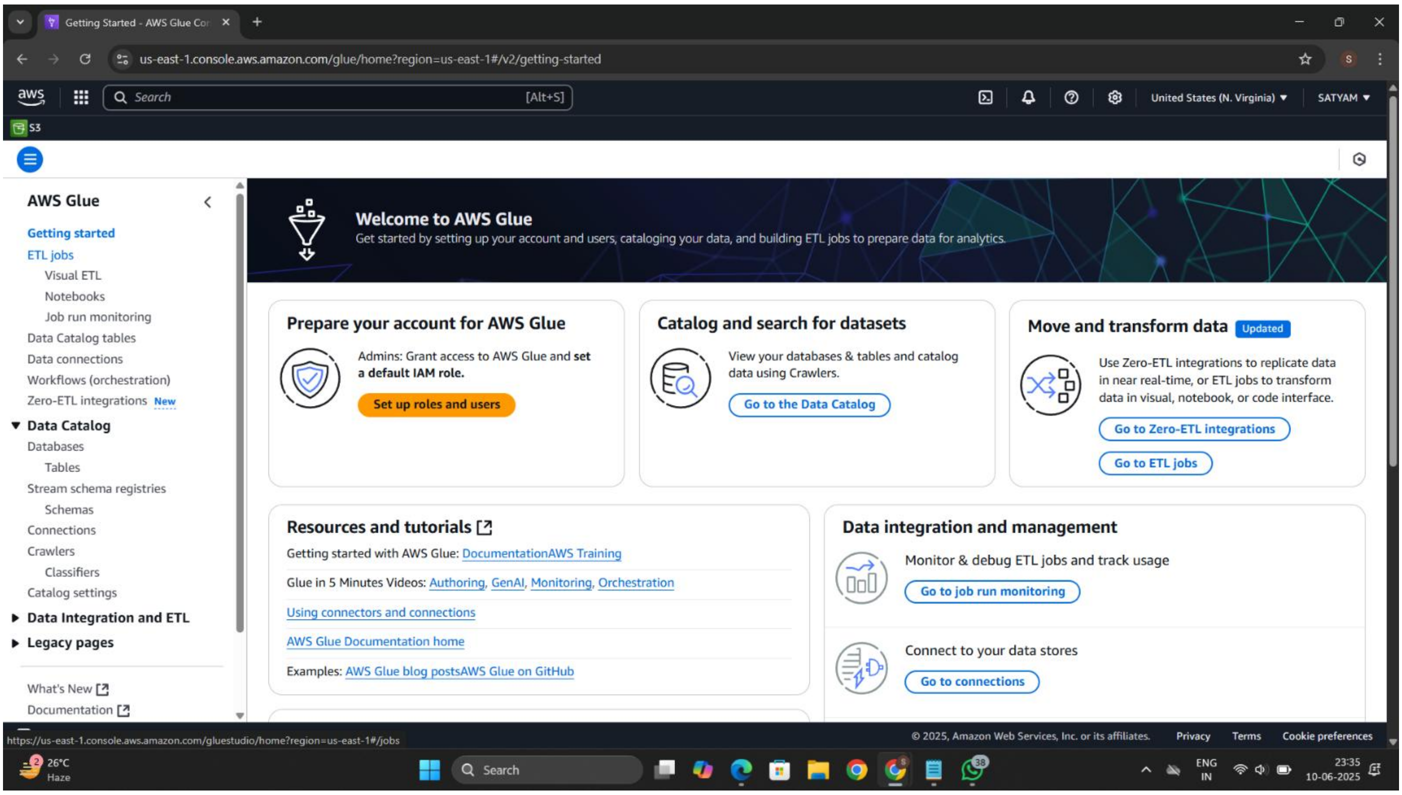
Author code with a script editor.

▶ Example jobs Info

Your jobs (1) Info

Filter jobs by property

□	Job name	Type	Created by	Last modified	AWS Glue version
□	iot_sensor_etl_job	Glue ETL	Script	6/9/2025, 7:25:22 PM	5.0



Welcome to AWS Glue

Get started by setting up your account and users, cataloging your data, and building ETL jobs to prepare data for analytics.

Prepare your account for AWS Glue

Admins: Grant access to AWS Glue and **set a default IAM role.**

Catalog and search for datasets

View your databases & tables and catalog data using Crawlers.

Move and transform data

Use Zero-ETL integrations to replicate data in near real-time, or ETL jobs to transform data in visual, notebook, or code interface.

Resources and tutorials

Getting started with AWS Glue: DocumentationAWS Training

Glue in 5 Minutes Videos: Authoring, GenAI, Monitoring, Orchestration

Using connectors and connections

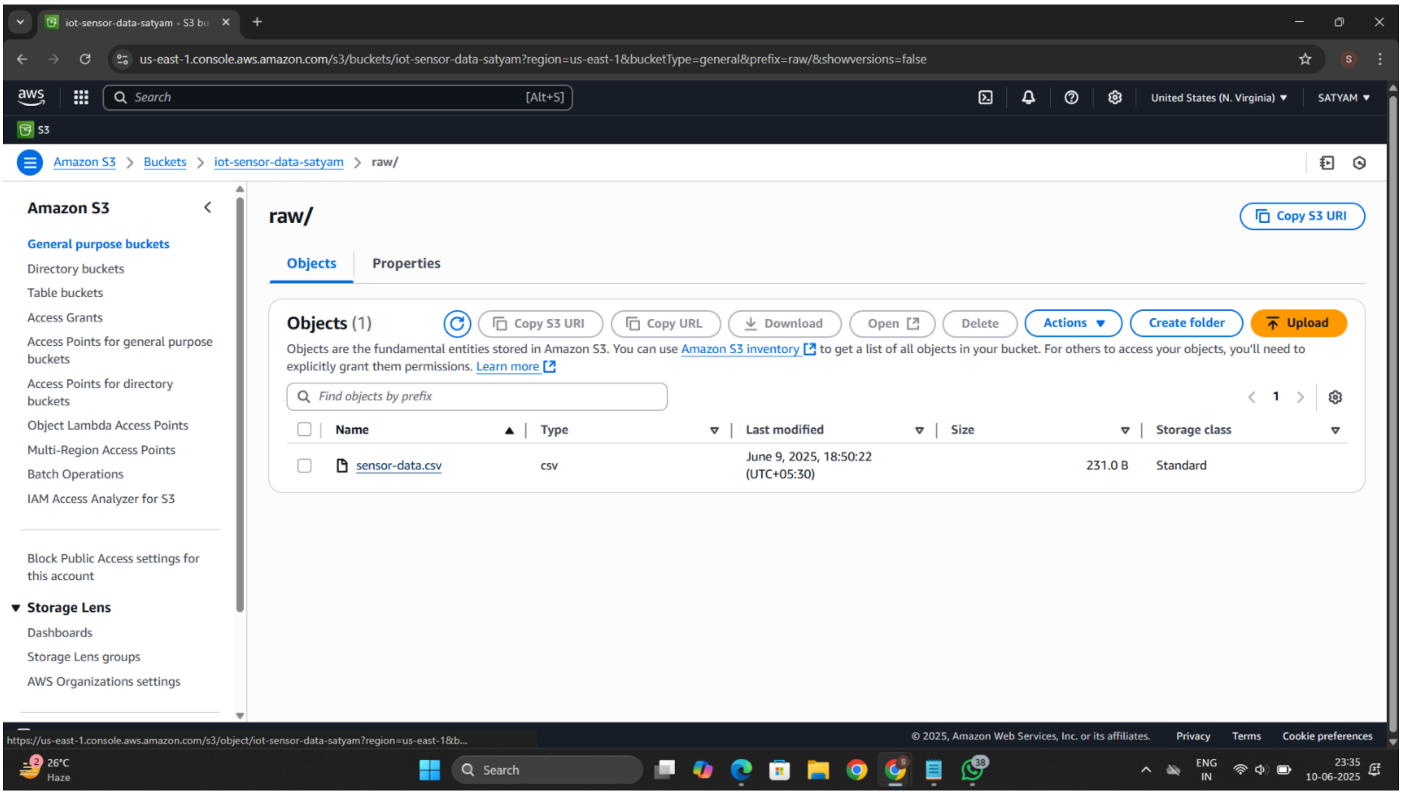
AWS Glue Documentation home

Examples: AWS Glue blog postsAWS Glue on GitHub

Data integration and management

Monitor & debug ETL jobs and track usage

Connect to your data stores



us-east-1.console.aws.amazon.com/s3/buckets/iot-sensor-data-satyam?region=us-east-1&bucketType=general&prefix=raw/&showversions=false

Amazon S3 > Buckets > iot-sensor-data-satyam > raw/

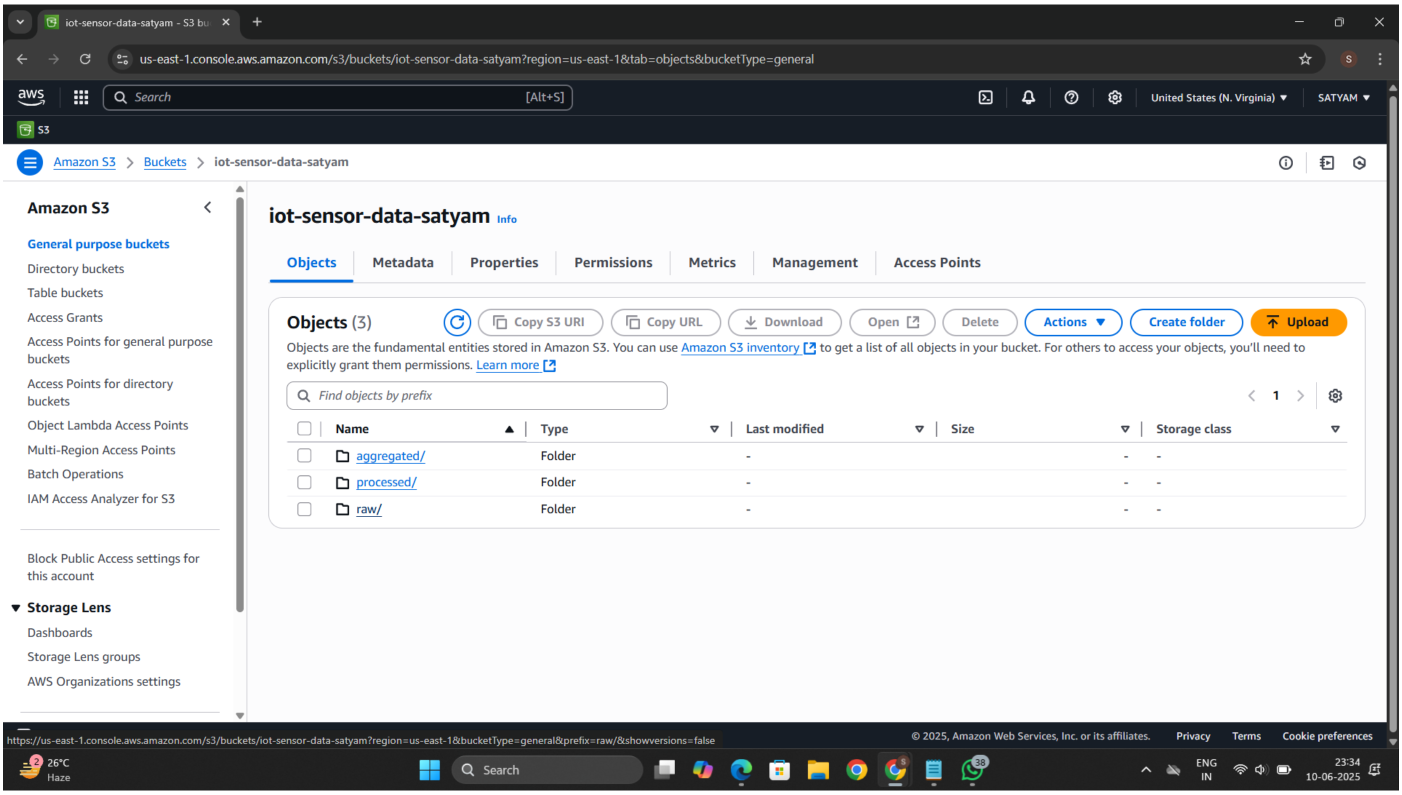
raw/

Objects (1)

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

Name	Type	Last modified	Size	Storage class
sensor-data.csv	csv	June 9, 2025, 18:50:22 (UTC+05:30)	231.0 B	Standard

iot-sensor-data-satyam - S3 bu
Copy S3 URI
Objects
Properties
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Find objects by prefix
Amazon S3
General purpose buckets
Directory buckets
Table buckets
Access Grants
Access Points for general purpose buckets
Access Points for directory buckets
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
United States (N. Virginia)
SATYAM
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
https://us-east-1.console.aws.amazon.com/s3/object/iot-sensor-data-satyam?region=us-east-1&b...
26°C
Haze
23:35
10-06-2025



us-east-1.console.aws.amazon.com/s3/buckets/iot-sensor-data-satyam?region=us-east-1&tab=objects&bucketType=general

Amazon S3 > Buckets > iot-sensor-data-satyam

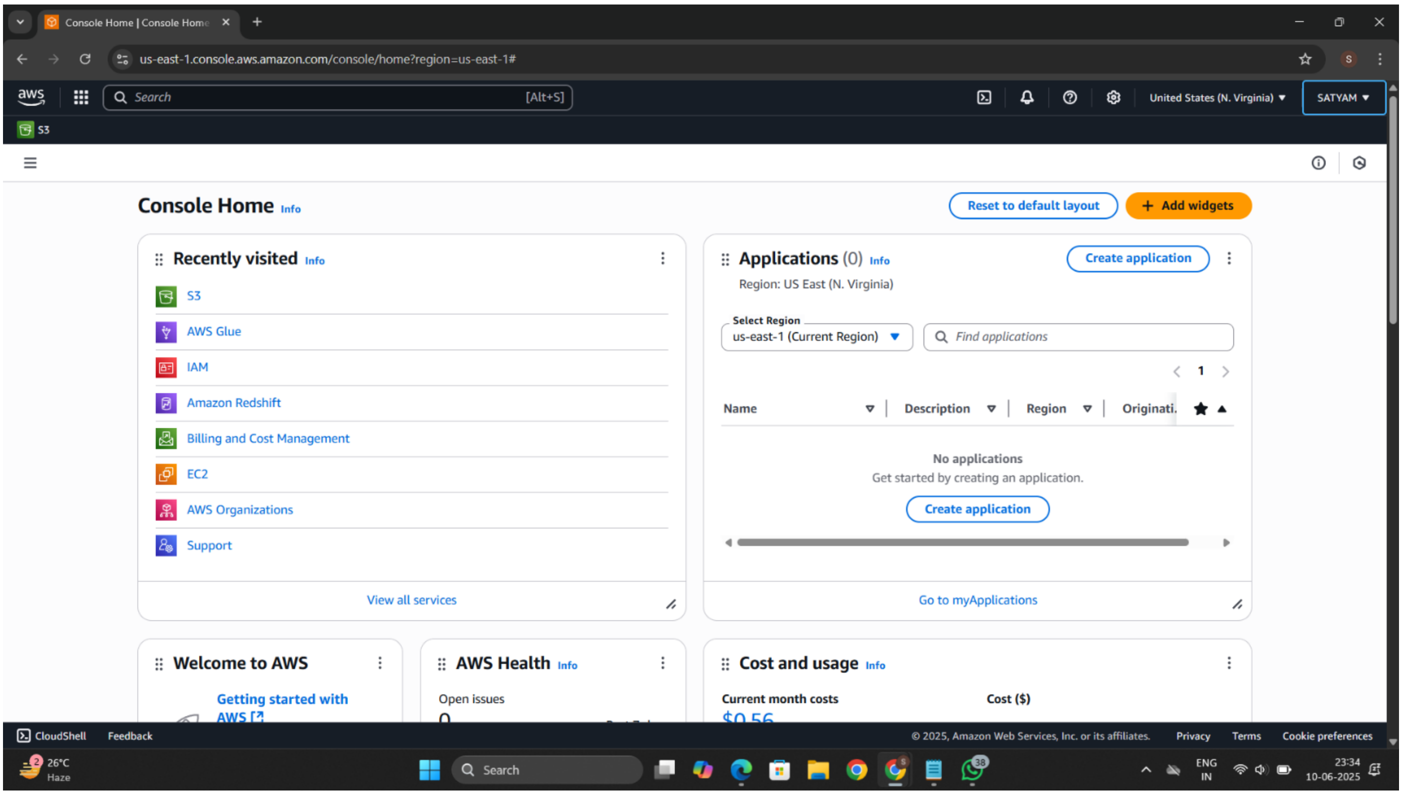
iot-sensor-data-satyam Info

Objects (3)

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

Name	Type	Last modified	Size	Storage class
aggregated/	Folder	-	-	-
processed/	Folder	-	-	-
raw/	Folder	-	-	-

iot-sensor-data-satyam - S3 bu
Objects
Metadata
Properties
Permissions
Metrics
Management
Access Points
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Find objects by prefix
Amazon S3
General purpose buckets
Directory buckets
Table buckets
Access Grants
Access Points for general purpose buckets
Access Points for directory buckets
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
United States (N. Virginia)
SATYAM
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
https://us-east-1.console.aws.amazon.com/s3/buckets/iot-sensor-data-satyam?region=us-east-1&bucketType=general&prefix=raw/&showversions=false
26°C
Haze
23:34
10-06-2025


Console Home | Console Home
us-east-1.console.aws.amazon.com/console/home?region=us-east-1#
Search [Alt+S]
United States (N. Virginia) ▼
SATYAM ▼
S3

Console Home Info

Reset to default layout | + Add widgets

Recently visited Info

- S3
- AWS Glue
- IAM
- Amazon Redshift
- Billing and Cost Management
- EC2
- AWS Organizations
- Support

View all services

Applications (0) Info | Create application

Region: US East (N. Virginia)

Select Region: us-east-1 (Current Region)

Find applications

Name	Description	Region	Originati...

No applications

Get started by creating an application.

Create application

Go to myApplications

Welcome to AWS

Getting started with AWS

AWS Health Info

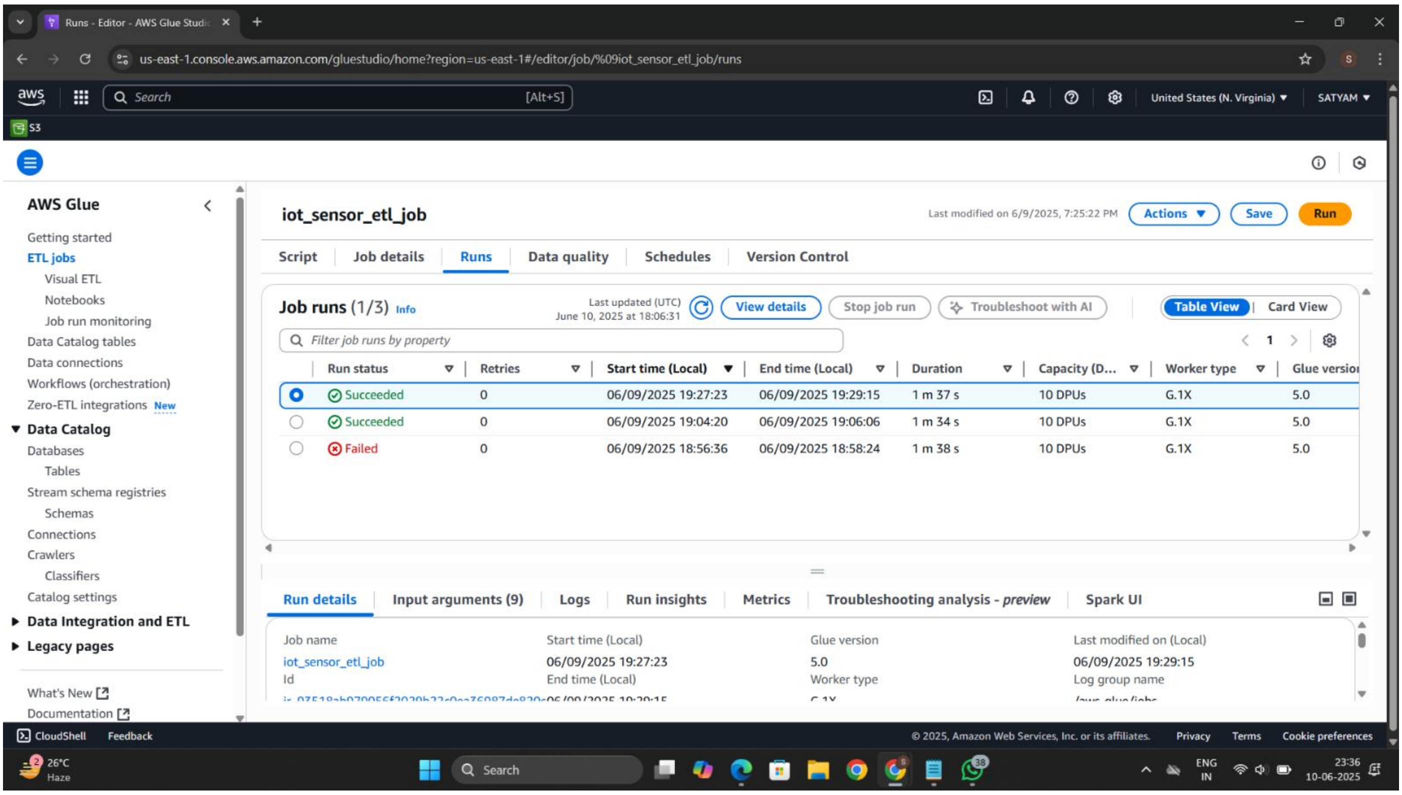
Open issues

0

Cost and usage Info

Current month costs | Cost ($)

$0.56



iot_sensor_etl_job

Last modified on 6/9/2025, 7:25:22 PM | Actions ▼ | Save | Run

Script | Job details | Runs | Data quality | Schedules | Version Control

Job runs (1/3) Info

Last updated (UTC) June 10, 2025 at 18:06:31

View details | Stop job run | Troubleshoot with AI | Table View | Card View

Filter job runs by property

	Run status	Retries	Start time (Local)	End time (Local)	Duration	Capacity (D...	Worker type	Glue version
●	✓ Succeeded	0	06/09/2025 19:27:23	06/09/2025 19:29:15	1 m 37 s	10 DPUs	G.1X	5.0
○	✓ Succeeded	0	06/09/2025 19:04:20	06/09/2025 19:06:06	1 m 34 s	10 DPUs	G.1X	5.0
○	☒ Failed	0	06/09/2025 18:56:36	06/09/2025 18:58:24	1 m 38 s	10 DPUs	G.1X	5.0

Run details | Input arguments (9) | Logs | Run insights | Metrics | Troubleshooting analysis - *preview* | Spark UI

Job name iot_sensor_etl_job	Start time (Local) 06/09/2025 19:27:23	Glue version 5.0	Last modified on (Local) 06/09/2025 19:29:15
Id	End time (Local)	Worker type	Log group name