

# Retail Data Management Project Report

**Name:** Ali Hamza Shaikh

**Project Title:** Retail Data Processing using AWS Glue

---

## 1. Project Overview

This project was developed to enhance ABC Retail's data management by using AWS Glue to automate ETL (Extract, Transform, Load) processes. The key objectives included cleaning and transforming raw data, joining datasets, extracting numerical values from text, and producing a summary of average sales by product category and shipping mode.

---

## 2. AWS Services Used

- **Amazon S3:** To store input and output data files.
  - **AWS Glue:** For data cataloguing, transformation, and ETL job orchestration.
  - **IAM:** To manage permissions for the Glue job.
- 

## 3. Datasets Used

- **Product Details CSV:** Contains information about product categories, IDs, and descriptions.
  - **Transactions CSV:** Contains sales records, including dates, quantities, discounts, profits, and shipping modes.
- 

## 4. Steps Performed

### Step 1: S3 Bucket Setup

- Created bucket: etl-cep-01-ali
- Subfolders:

- transaction-files/ (uploaded transactions.csv)
- product-files/ (uploaded product details.csv)
- Created output bucket: etl-cep-output-01

### **Step 2: Create AWS Glue Database**

- Name: abc-retail

### **Step 3: Create Classifiers**

- **cust\_classifier:** For product data (CSV with headers)
- **txnClass:** For transaction data with manual headers

### **Step 4: Create IAM Role**

- Role Name: glue-role
- Permissions: AdministratorAccess

### **Step 5: Create Crawlers**

- Crawler for transactions:
  - Source: etl-cep-01-ali/transaction-files/
  - Classifier: txnClass
- Crawler for product details:
  - Source: etl-cep-01-ali/product-files/
  - Classifier: cust\_classifier

### **Step 6: Create Visual ETL Job**

- **Join:** Inner join on Product ID
- **Drop Fields:** Removed duplicate Product ID
- **Regex Extractor:** Extracted numeric value from Sales column using regex `(\d+(\.\d+)?)`
- **Aggregate:**
  - Grouped by: Product Category, Ship Mode

- Aggregated field: Sales
- Aggregation: Average
- **Output:** Saved as .parquet to etl-cep-output-01-ali

## Step 7: Output Verification

- Used S3 Select to query .parquet output file
  - Set output format to CSV
- 

## 5. Key Learnings

- Understood the role of Glue classifiers in schema detection
  - Applied regular expressions for data cleaning
  - Used Visual ETL to create a no-code data pipeline
  - Implemented grouping and aggregation for summary reporting
- 

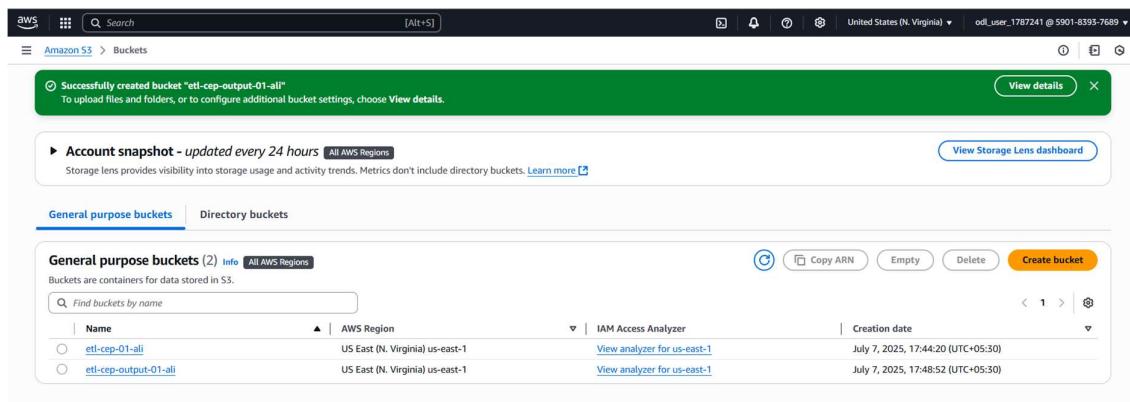
## 6. Screenshots

### Screenshot Checklist for AWS Glue ETL Retail Project

#### ◆ S3 Setup

##### 1. S3 Bucket Created

*Screenshot of the bucket list showing etl-cep-01-ali and etl-cep-output-01-ali.*



The screenshot shows the AWS S3 Buckets list page. At the top, there's a success message: "Successfully created bucket 'etl-cep-output-01-ali'. To upload files and folders, or to configure additional bucket settings, choose View details." Below this, there's an "Account snapshot - updated every 24 hours" section. The main table lists "General purpose buckets" (2) under the "All AWS Regions" tab. The table has columns for Name, AWS Region, IAM Access Analyzer, and Creation date. The two buckets listed are "etl-cep-01-ali" and "etl-cep-output-01-ali", both located in "US East (N. Virginia) us-east-1". The "etl-cep-01-ali" bucket was created on July 7, 2025, at 17:44:20 (UTC+05:30), and the "etl-cep-output-01-ali" bucket was created on July 7, 2025, at 17:48:52 (UTC+05:30). Action buttons for each row include "View details", "Copy ARN", "Empty", "Delete", and "Create bucket".

Name	AWS Region	IAM Access Analyzer	Creation date
etl-cep-01-ali	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	July 7, 2025, 17:44:20 (UTC+05:30)
etl-cep-output-01-ali	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	July 7, 2025, 17:48:52 (UTC+05:30)

## 2. S3 Subfolders Uploaded

Inside *etl-cep-01*, show *product-files/* and *transaction-files/* each containing their CSV file.

The screenshot shows the AWS S3 console with the path `Buckets > etl-cep-01-ali > transaction-files/`. The **Objects** tab is selected, displaying one object: `transactions.csv`. The object details are as follows:

Name	Type	Last modified	Size	Storage class
<code>transactions.csv</code>	csv	July 7, 2025, 17:46:12 (UTC+05:30)	4.8 MB	Standard

The screenshot shows the AWS S3 console with the path `Buckets > etl-cep-01-ali > product-files/`. The **Objects** tab is selected, displaying one object: `product.details.csv`. The object details are as follows:

Name	Type	Last modified	Size	Storage class
<code>product.details.csv</code>	csv	July 7, 2025, 17:46:50 (UTC+05:30)	1.2 KB	Standard

## ◆ Glue Database and Classifiers

### 3. Glue Database Created

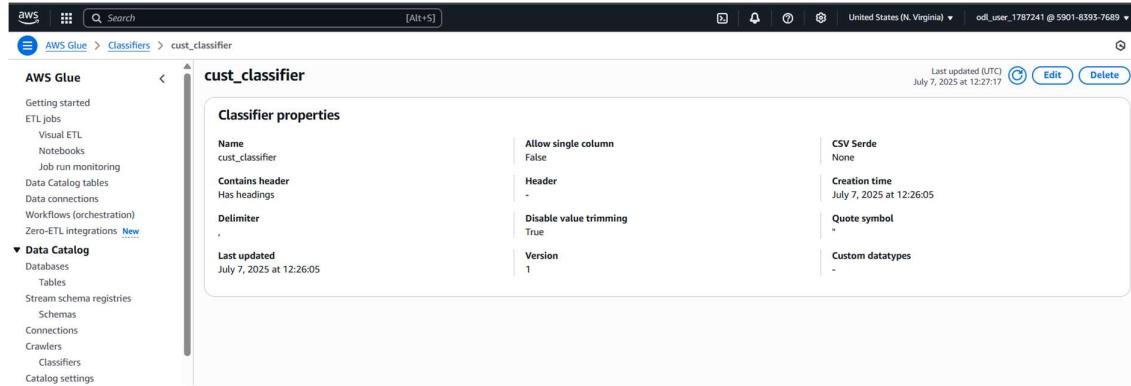
Screenshot of database named *abc-retail* in AWS Glue > Databases.

The screenshot shows the AWS Glue console with the path `Databases`. A blue banner at the top left announces new optimization features for Apache Iceberg tables. The **Databases** section displays one database:

Name	Description	Location URI	Created on (UTC)
<code>abc-retail</code>	-	-	July 7, 2025 at 12:20:49

#### 4. Classifier: cust\_classifier Created

Screenshot of classifier for product data in AWS Glue > Classifiers.

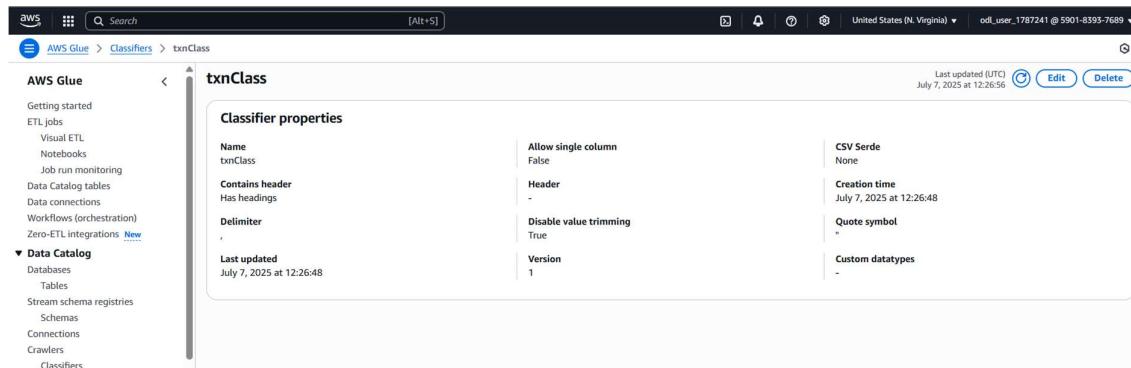


The screenshot shows the AWS Glue Classifier properties for 'cust\_classifier'. The 'Classifier properties' section includes:

Name	Value
Name	cust_classifier
Contains header	Has headings
Delimiter	,
Last updated	July 7, 2025 at 12:26:05
Allow single column	False
Header	-
Disable value trimming	True
Version	1
CSV Serde	None
Creation time	July 7, 2025 at 12:26:05
Quote symbol	"
Custom datatypes	-

#### 5. Classifier: txnClass Created

Screenshot of classifier for transaction data.



The screenshot shows the AWS Glue Classifier properties for 'txnClass'. The 'Classifier properties' section includes:

Name	Value
Name	txnClass
Contains header	Has headings
Delimiter	,
Last updated	July 7, 2025 at 12:26:48
Allow single column	False
Header	-
Disable value trimming	True
Version	1
CSV Serde	None
Creation time	July 7, 2025 at 12:26:48
Quote symbol	"
Custom datatypes	-

## ◆ IAM Role

### 6. IAM Role: glue-role Created

*Screenshot showing the new IAM role and the AdministratorAccess policy attached.*

The screenshot shows the AWS IAM Roles page. A new role named "glue-role" has been created. The role's ARN is arn:aws:iam::590183937689:role/glue-role, and its maximum session duration is set to 1 hour. The "Permissions" tab is selected, showing that the "AdministratorAccess" policy is attached to it. The "AdministratorAccess" policy is described as "AWS managed - job function".

## ◆ Crawlers

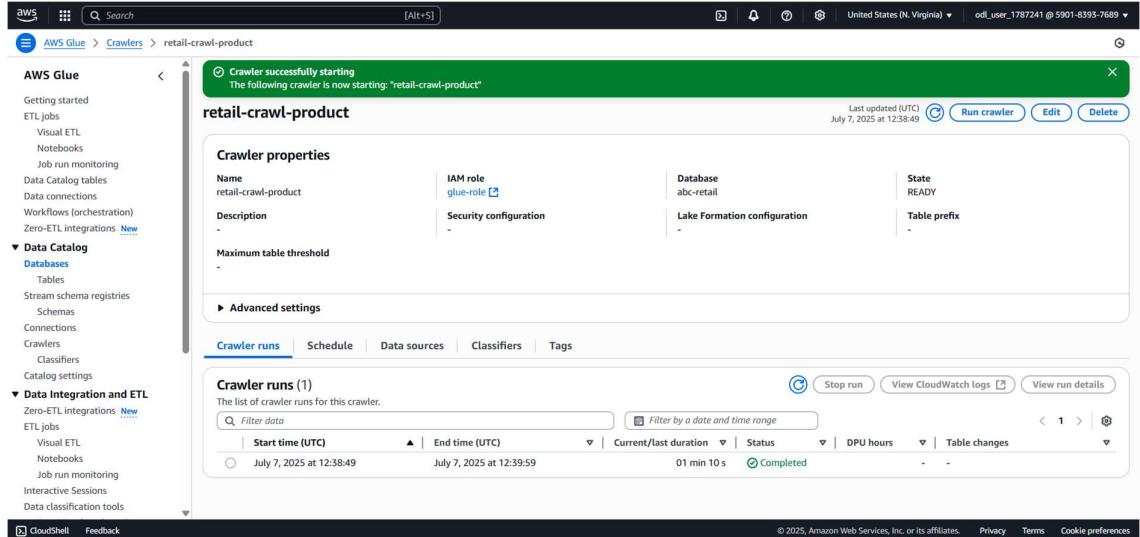
### 7. Crawler for Transactions Created

*Screenshot of crawler settings with txnClass, transaction path, and abc-retail DB selected.*

The screenshot shows the AWS Glue Crawlers page. A crawler named "retail-crawl-txn" is currently starting, as indicated by the green status bar message "Crawler successfully starting". The crawler's properties are displayed, including its name ("retail-crawl-txn"), IAM role ("glue-role"), database ("abc-retail"), and state ("READY"). The "Crawler runs" section shows a single run that completed successfully on July 7, 2025, at 12:34:23, with a duration of 01 min 14 s.

## 8. Crawler for Product Details Created

*Screenshot of second crawler settings with cust\_classifier, product path selected.*



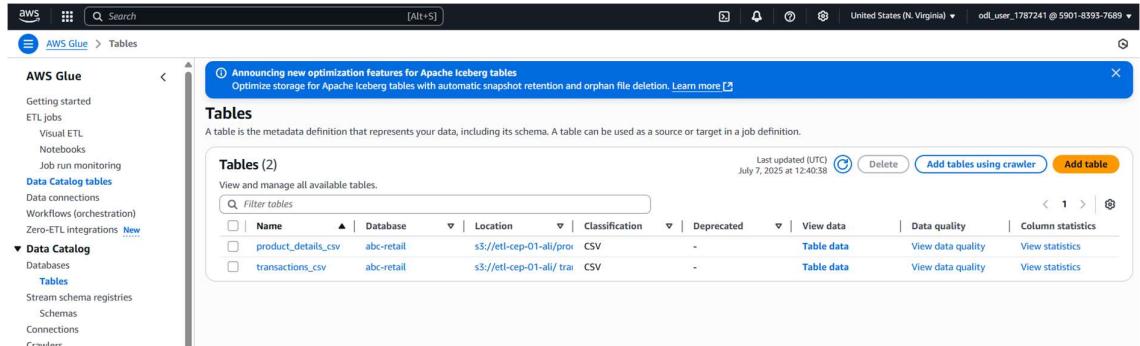
The screenshot shows the AWS Glue Crawler configuration page for the crawler named "retail-crawl-product". The crawler is currently starting successfully. The properties listed include:

- Name: retail-crawl-product
- IAM role: glue-role
- Description: -
- Security configuration: -
- Database: abc-retail
- Lake Formation configuration: -
- State: READY
- Table prefix: -

The "Advanced settings" section is collapsed. Below the properties, there are tabs for "Crawler runs", "Schedule", "Data sources", "Classifiers", and "Tags". The "Crawler runs" tab shows one run completed on July 7, 2025, at 12:38:49, with a duration of 01 min 10 s.

## 9. Crawler Run Completed

*Screenshot showing success of both crawler runs and tables created in the Data Catalog.*



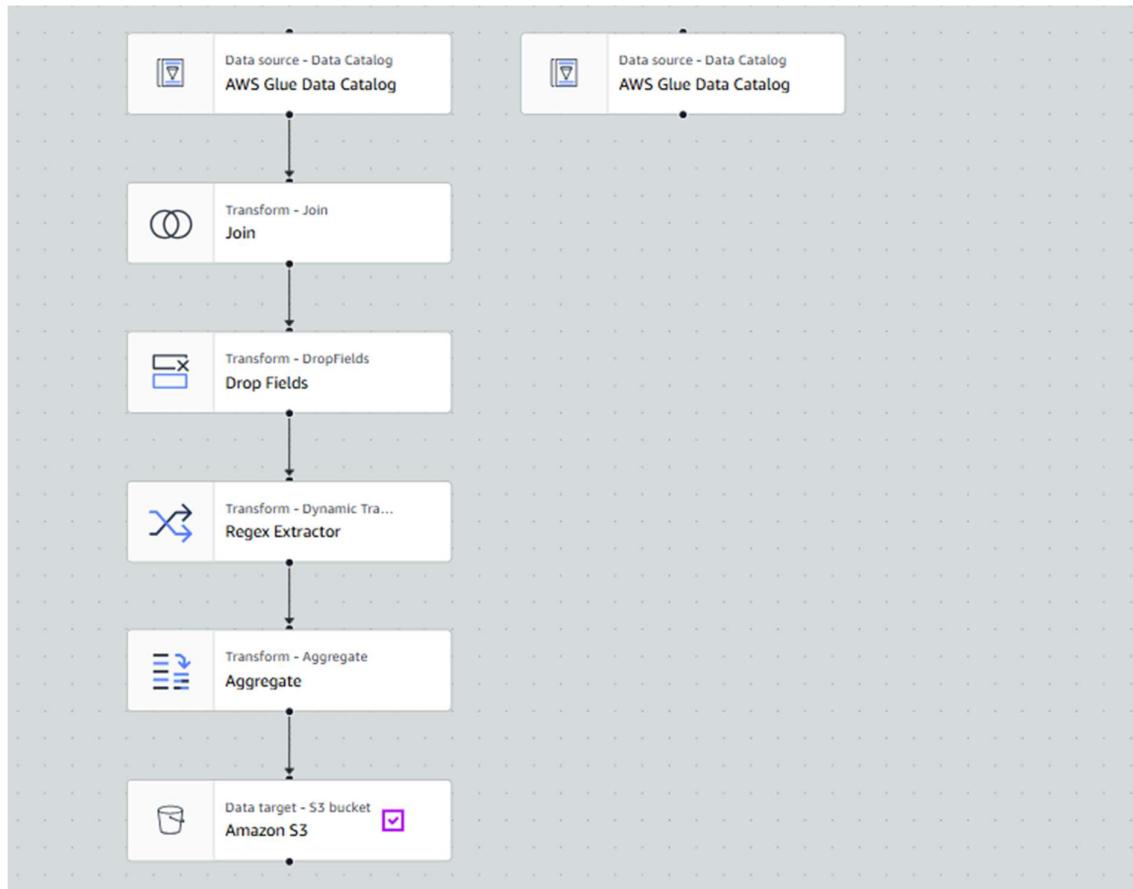
The screenshot shows the AWS Glue Data Catalog tables page. It displays two tables: "product\_details\_csv" and "transactions\_csv", both located in the "abc-retail" database. The tables are defined with the following schema:

Name	Database	Location	Classification	Deprecated	View data	Data quality	Column statistics
product_details_csv	abc-retail	s3://etl-cep-01-ali/prod	CSV	-	Table data	View data quality	View statistics
transactions_csv	abc-retail	s3://etl-cep-01-ali/tran	CSV	-	Table data	View data quality	View statistics

◆ ETL Visual Job

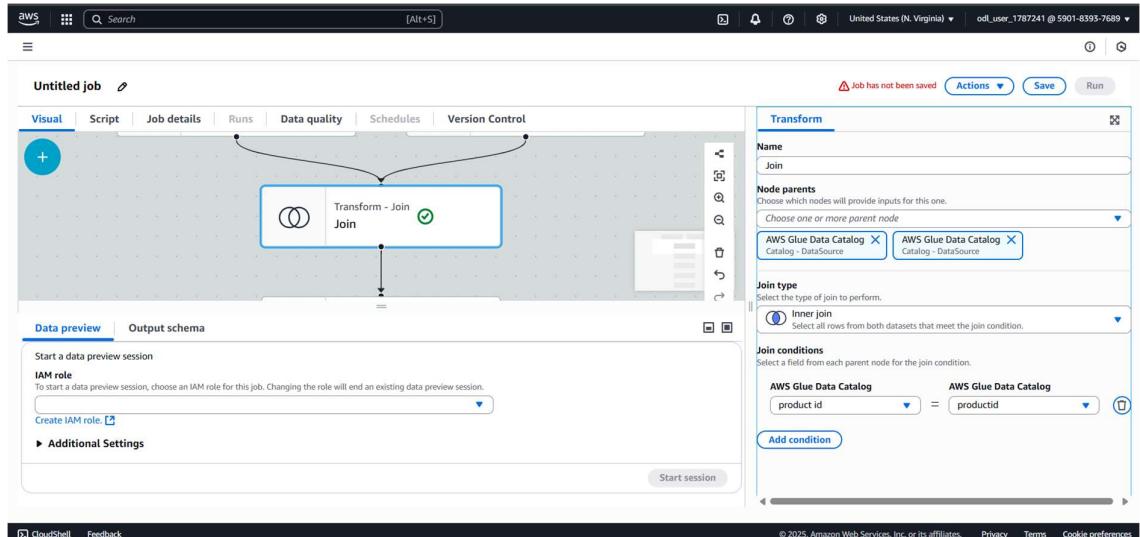
10.  **Visual ETL Canvas (Before Linking)**

*Screenshot of AWS Glue Visual job canvas with unlinked nodes: 2 Data Catalog nodes, Join, Drop Fields, Regex Extractor, Aggregate, and S3 Target.*



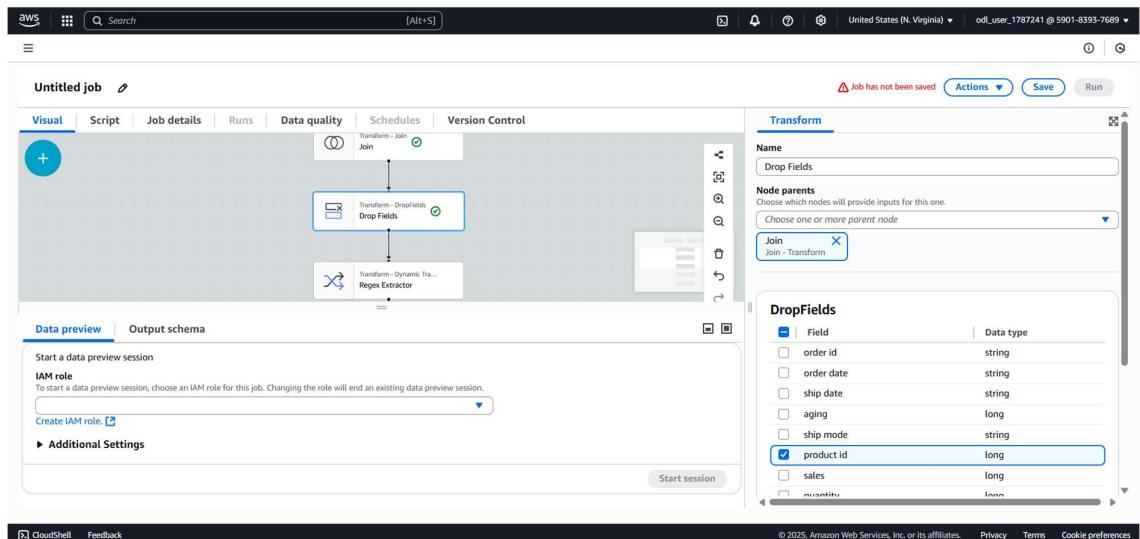
## 11. Join Node Configuration

Screenshot showing Join type as "Inner Join" with Product ID fields selected from both sides.



## 12. Drop Fields Configuration

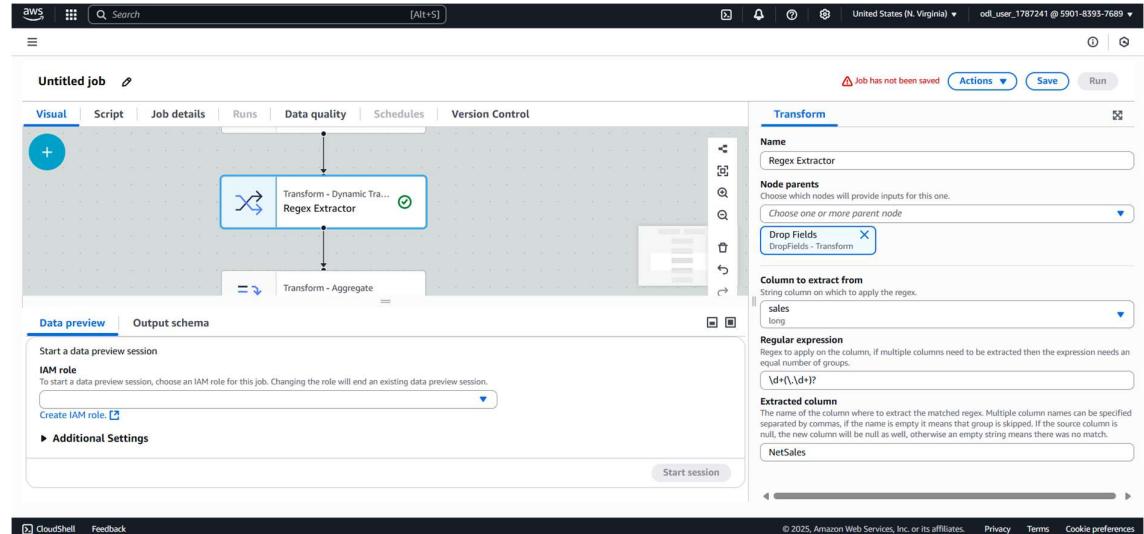
Screenshot where one Product ID is dropped to remove duplicates.



### 13. Regex Extractor Node Settings

Screenshot showing Sales column cleaned using regex:

$\backslash\backslash d+(\backslash\backslash .\backslash\backslash d+)?$ , and new column as NetSales.



The screenshot shows the AWS Glue Data Catalog job editor interface. A job named "Untitled job" is displayed with a single step consisting of a "Transform - Dynamic Transform" node followed by a "Regex Extractor" node. The "Regex Extractor" node has its "Regular expression" field set to `\d+(\.\d+)?`. The "Extracted column" field is set to "NetSales". The "Column to extract from" field is set to "sales". The "Drop Fields" field is set to "DropFields - Transform". The "Name" field for the node is "Regex Extractor". The "Data preview" tab is selected, showing a preview of the transformed data. The "Output schema" tab is also visible. The IAM role dropdown is populated with "Create IAM role." and "Additional Settings" are expanded.

14.  **Aggregate Node Settings**

*Screenshot showing group by: Product Category, Ship Mode; aggregate: AVG(Sales).*

Last modified on 7/7/2025, 6:49:42 PM Actions ▾ Save Run

**Transform**

**Name**  
Aggregate

**Node parents**  
Choose which nodes will provide inputs for this one.  
Choose one or more parent node

**Regex Extractor** X  
DynamicTransform - Transform

▶ Aggregate [Info](#)

**Fields to group by - optional**  
Select the fields you would like to group your rows by, so the aggregation would be done for each unique group.  
Choose one or more fields

product category X ship mode X

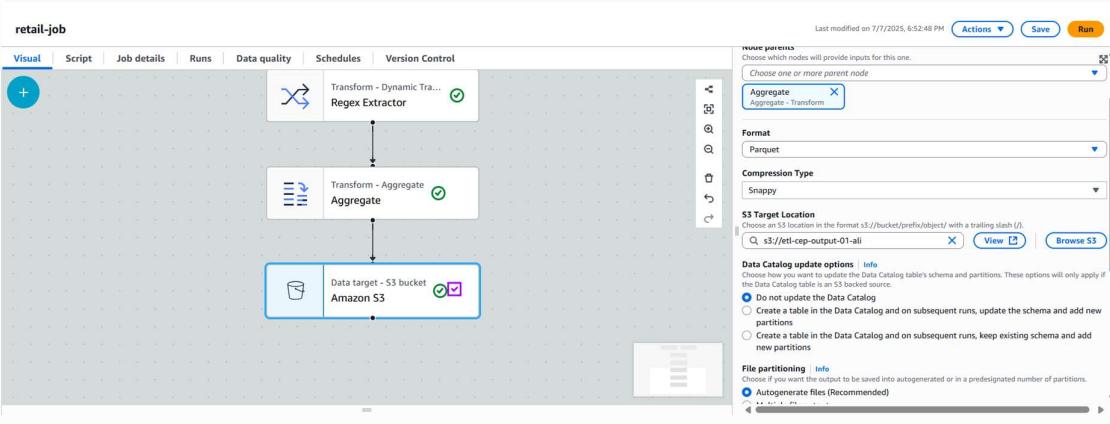
**Aggregation**  
Select fields and functions to aggregate.

**Field to aggregate** sales ▼ **Aggregation function** avg ▼ Info

trash can icon

## 15. ETL Output Node Setup

Screenshot showing target S3 bucket selected: etl-cep-output-01-ali.



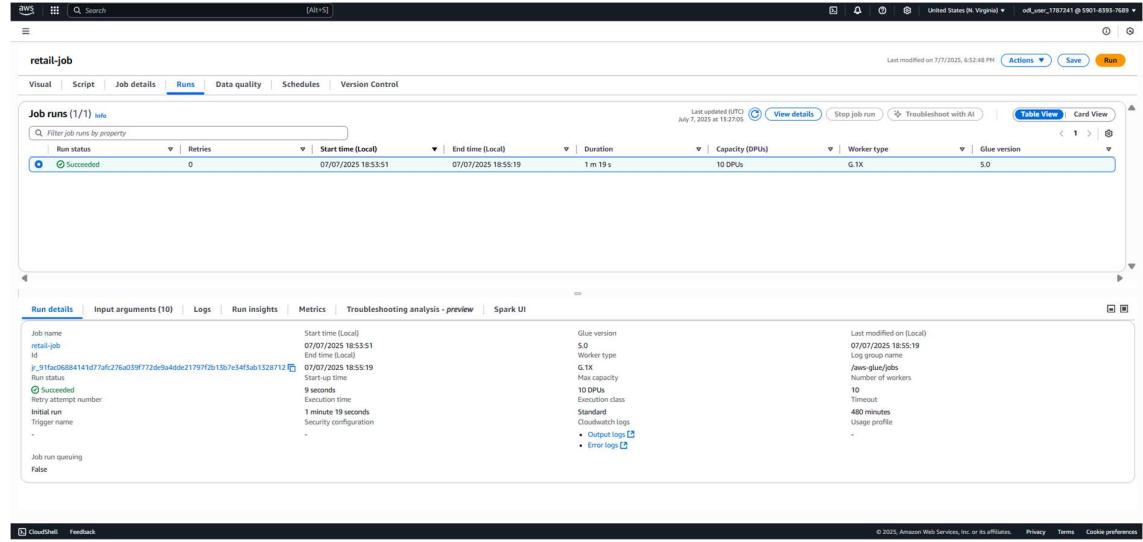
## 16. Job Saved and Run Started

Screenshot showing job name as etl-cep-job and job run in progress.

The screenshot shows the AWS Glue job runs page. A single job run for 'retail-job' is listed under 'Job runs (1 / 1)'. The run status is 'Running' with 0 retries, started at 07/07/2025 18:53:51, and ended at 0 seconds. The 'Metrics' tab is selected, displaying details such as Glue version 5.0, Worker type G.1X, and Capacity 10 DPU. The 'Logs' tab shows log entries for the job run.

## 17. Job Run Completed Successfully

Screenshot of job run status: "Succeeded".



The screenshot shows the AWS Glue Job Run Details page. At the top, there's a navigation bar with tabs: Visual, Script, Job details, Runs (which is selected), Data quality, Schedules, and Version Control. Below the navigation bar, a table displays the job run details:

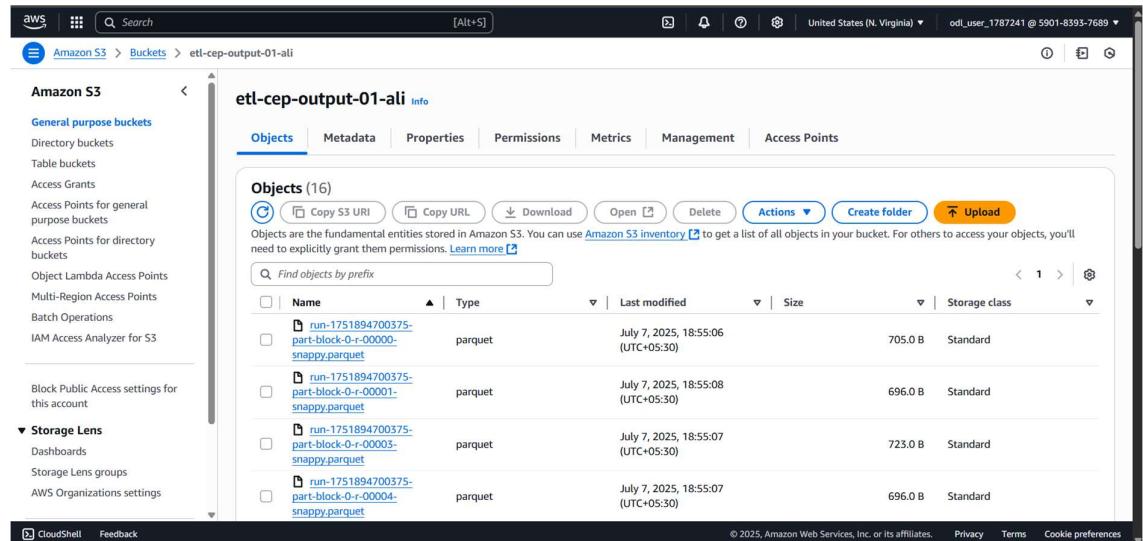
Run status	Retries	Start time (Local)	End time (Local)	Duration	Capacity (DPU)	Worker type	Glue version
SUCCEEDED	0	07/07/2025 18:53:51	07/07/2025 18:55:19	1 m 19 s	10 DPU	G.1X	5.0

Below the table, there are several tabs: Run details, Input arguments (10), Logs, Run insights, Metrics, Troubleshooting analysis - preview, and Spark UI. The Run details tab is active. On the right side of the page, there's a sidebar with various configuration options and metrics.

## ◆ Output and SQL Query

## 18. Output File in Output Bucket

Screenshot showing the .parquet file inside etl-cep-output-01-ali.

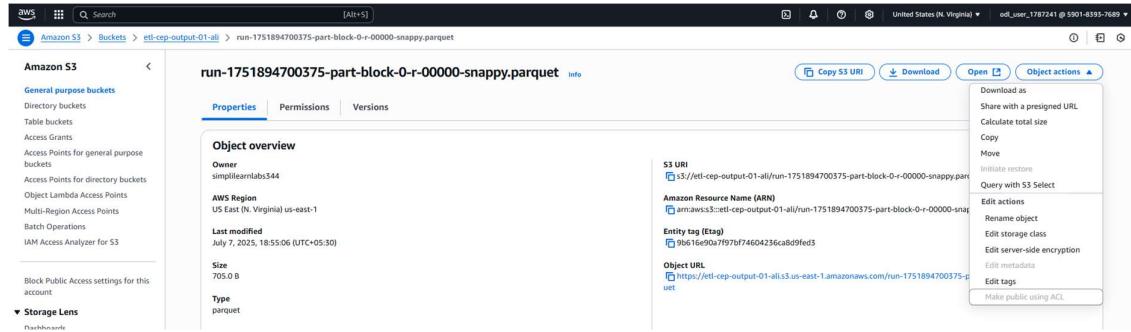


The screenshot shows the Amazon S3 Buckets page. The left sidebar lists General purpose buckets (Directory buckets, Table buckets, Access Grants, Access Points for general purpose buckets, Access Points for directory buckets, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3) and Storage Lens (Dashboards, Storage Lens groups, AWS Organizations settings). The main area shows the contents of the 'etl-cep-output-01-ali' bucket under the 'Objects' tab. There are 16 objects listed, all of which are parquet files with names starting with 'run-'. The table includes columns for Name, Type, Last modified, Size, and Storage class.

Name	Type	Last modified	Size	Storage class
run-1751894700375-part-block-0-r-00000-snappy.parquet	parquet	July 7, 2025, 18:55:06 (UTC+05:30)	705.0 B	Standard
run-1751894700375-part-block-0-r-00001-snappy.parquet	parquet	July 7, 2025, 18:55:08 (UTC+05:30)	696.0 B	Standard
run-1751894700375-part-block-0-r-00003-snappy.parquet	parquet	July 7, 2025, 18:55:07 (UTC+05:30)	723.0 B	Standard
run-1751894700375-part-block-0-r-00004-snappy.parquet	parquet	July 7, 2025, 18:55:07 (UTC+05:30)	696.0 B	Standard

## 19. S3 Select Query Panel

*Screenshot of “Query with S3 Select” screen, with format set to CSV and query executed.*



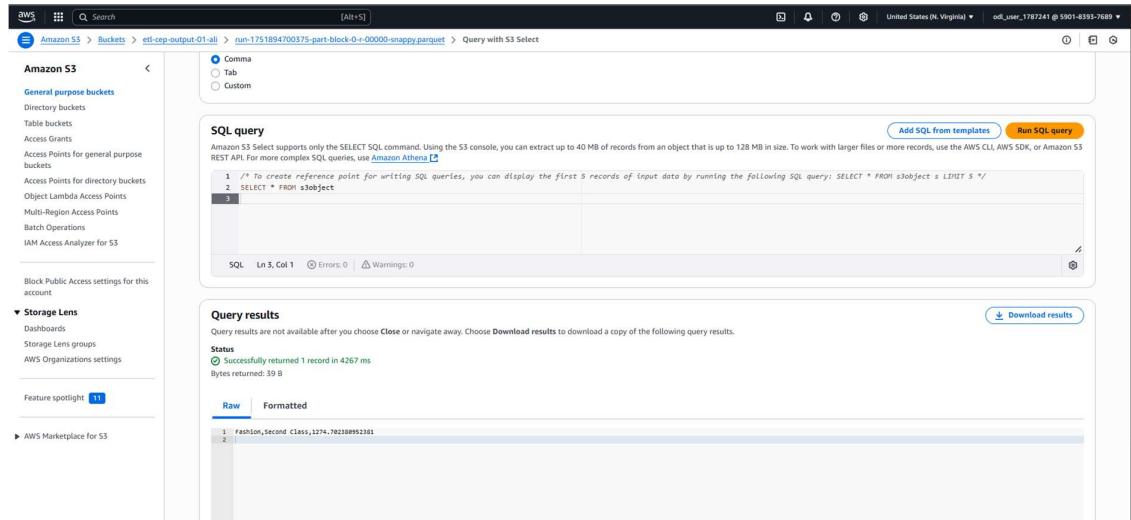
The screenshot shows the AWS S3 console with the path: Amazon S3 > Buckets > etl-cep-output-01-all > run-1751894700375-part-block-0-r-00000-snappy.parquet. The 'Properties' tab is selected. The 'Object overview' section displays the following details:

- Owner: simpleairlabs344
- AWS Region: US East (N. Virginia) us-east-1
- Last modified: July 7, 2025, 18:55:06 (UTC+05:30)
- Size: 705.0 B
- Type: parquet

The 'Actions' menu on the right includes options like Copy, Move, and 'Query with S3 Select'. The 'Object URI' and 'Object URL' are also displayed.

## 20. Query Output Displayed

*Screenshot showing sample rows of output (average sales by product category and ship mode).*



The screenshot shows the 'Query with S3 Select' panel with the path: Amazon S3 > Buckets > etl-cep-output-01-all > run-1751894700375-part-block-0-r-00000-snappy.parquet. The 'SQL query' section contains the following code:

```
1 /* To create reference point for writing SQL queries, you can display the first 5 records of input data by running the following SQL query: SELECT * FROM $object LIMIT 5 */
2 SELECT * FROM $object
3
```

The 'Query results' section shows the output:

Status: Successfully returned 1 record in 4267 ms  
Bytes returned: 39 B

The results are displayed in Raw and Formatted tabs, with the first row being '1 Fashion,Second Class,1274.762380952381'.

## **7. Conclusion**

This project demonstrated how AWS Glue can be effectively used to automate and manage ETL workflows. By integrating and transforming retail datasets, valuable insights were generated to support data-driven decision-making.

---

**[End of Report]**