Batch Time Analysis of Transactional Data

Course-end Project 1

Description

Lenodo is a multinational e-commerce organization that sells products directly to consumers. The database administrator exports the data every night in a CSV file, but this export functionality is unused. Lenodo wants to use this data to uncover insights about the most-sold item and the countries where customers have bought this item.

You are a data analytics consultant, and you're asked to provide valuable insights and statistics across products, brands, categories, segments to the marketing, product, sales, and procurement teams and inform them about which product has the highest amount of sales and which product and its marketing needs the most improvement. These statistics will help to run effective digital marketing campaigns. The scope of this project is limited to data engineering and analysis.

Objective:

To use AWS Big Data stack for data engineering to analyze transactions, uncover patterns, and share actionable insights

Steps to perform:

1. Create an S3 bucket with a unique name and upload the CSV file to the S3 bucket (ensure that the file is in UTF-8 format only)
2. Create a crawler to crawl the CSV data and generate a metadata catalog
3. Create a Glue job to transform the data into the Parquet format as CSV is not optimal for data warehouse queries
4. Add another crawler to crawl the Parquet data files to generate the metadata catalog of the Parquet file in order to query it with Athena
5. Query the data to identify the best-selling item and countries where customers have bought the most-sold item using Athena

1. Setup AWS S3 Bucket:

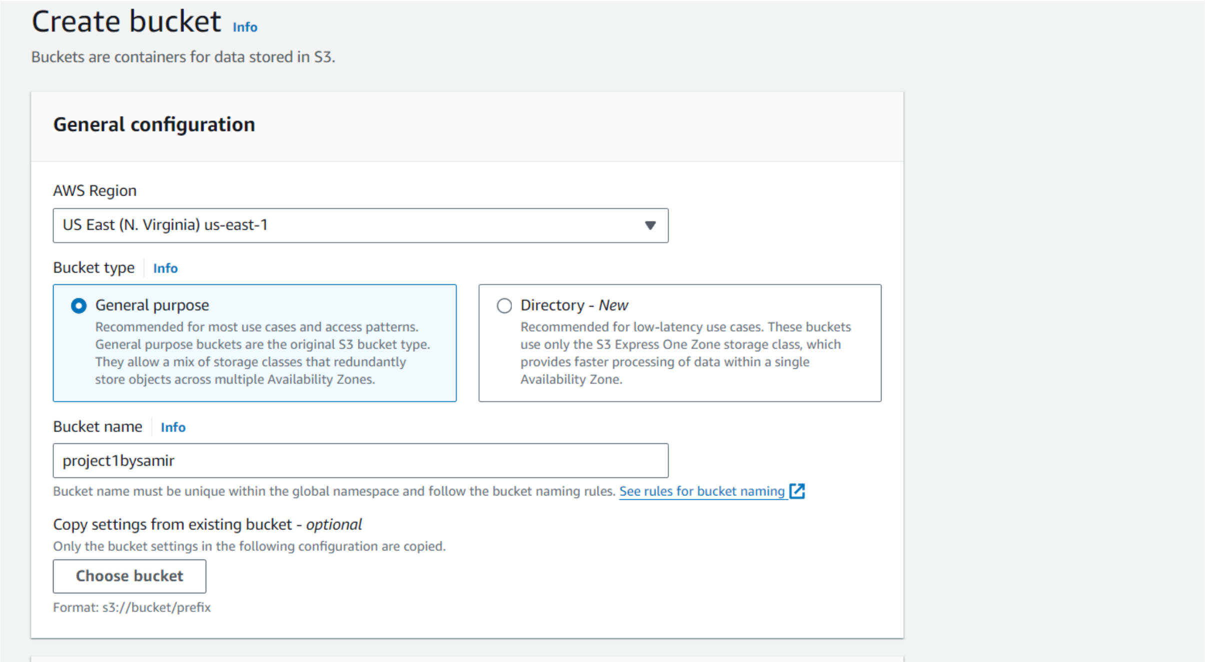
Objective: Store the CSV data file securely and reliably in the cloud.

Action Items:

Go to the AWS Management Console, navigate to S3, and create a new bucket.

Ensure the name is unique, follows AWS naming conventions, and is region-appropriate for your analysis needs.

Upload the CSV file to the newly created S3 bucket. Make sure the CSV file is encoded in UTF-8 to avoid any compatibility issues.



Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region: US East (N. Virginia) us-east-1

Bucket type Info

● General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

○ Directory - *New*
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

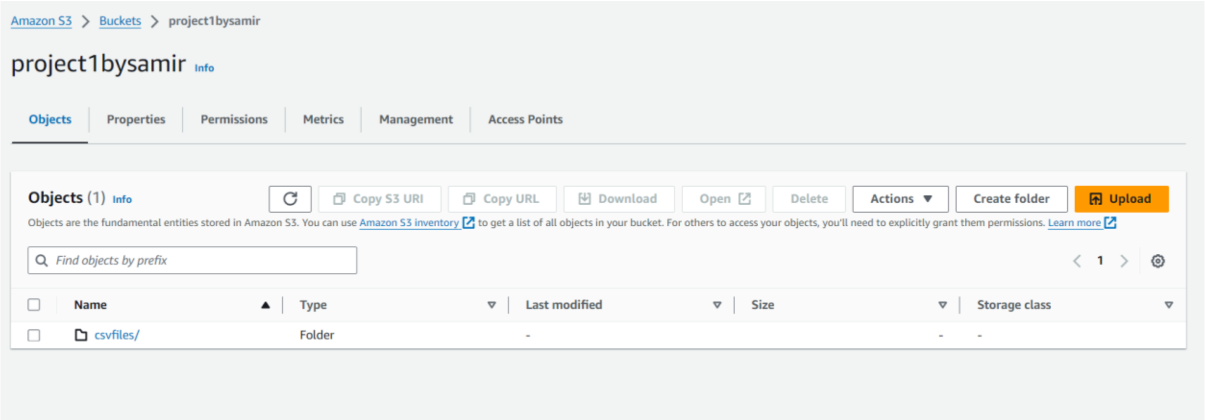
Bucket name Info: project1bysamir

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix



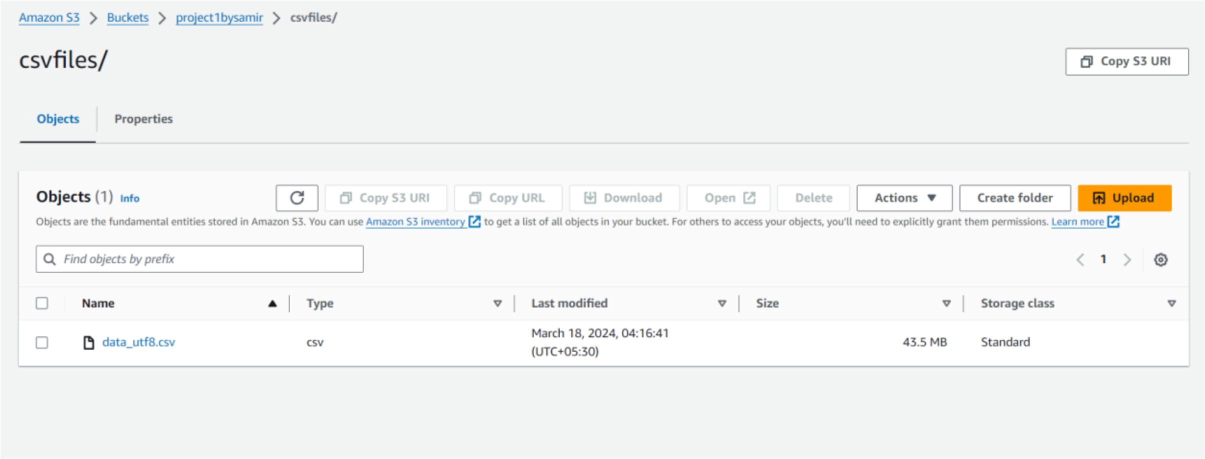
Amazon S3 > Buckets > project1bysamir

project1bysamir Info

Objects (1) Info

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
□	csvfiles/	Folder	-	-	-



csvfiles/

Objects (1) Info

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
□	data_utf8.csv	csv	March 18, 2024, 04:16:41 (UTC+05:30)	43.5 MB	Standard

2. Create and Run AWS Glue Crawler for CSV Data:

Objective: Automatically discover and catalog metadata from the CSV data in S3.

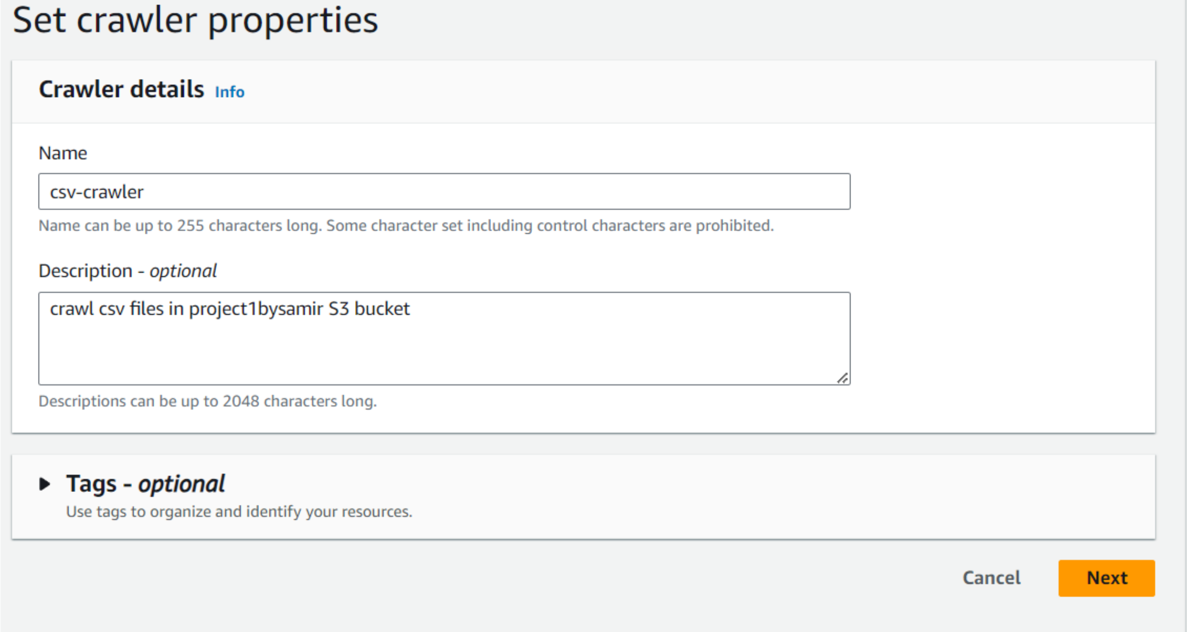
Action Items:

In AWS Glue, create a new crawler. Set the data store to point to the S3 bucket where the CSV file is stored.

Choose an IAM role that has permissions to access both AWS Glue services and the S3 bucket.

Configure the crawler to run on demand or schedule it as needed. Once configured, run the crawler.

Upon completion, the crawler will create a database and table(s) in the AWS Glue Data Catalog, representing the structure of your CSV data.



Set crawler properties

Crawler details

Name

csv-crawler

Name can be up to 255 characters long. Some character set including control characters are prohibited.

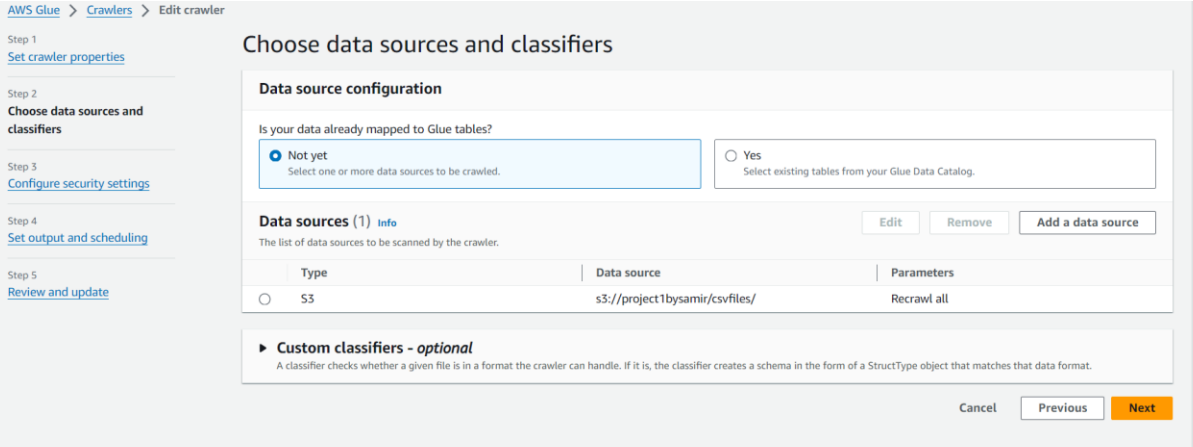
Description - *optional*

crawl csv files in project1bysamir S3 bucket

Descriptions can be up to 2048 characters long.

▶ Tags - *optional*

Use tags to organize and identify your resources.



AWS Glue > Crawlers > Edit crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and update

Choose data sources and classifiers

Data source configuration

Is your data already mapped to Glue tables?

● Not yet
Select one or more data sources to be crawled.

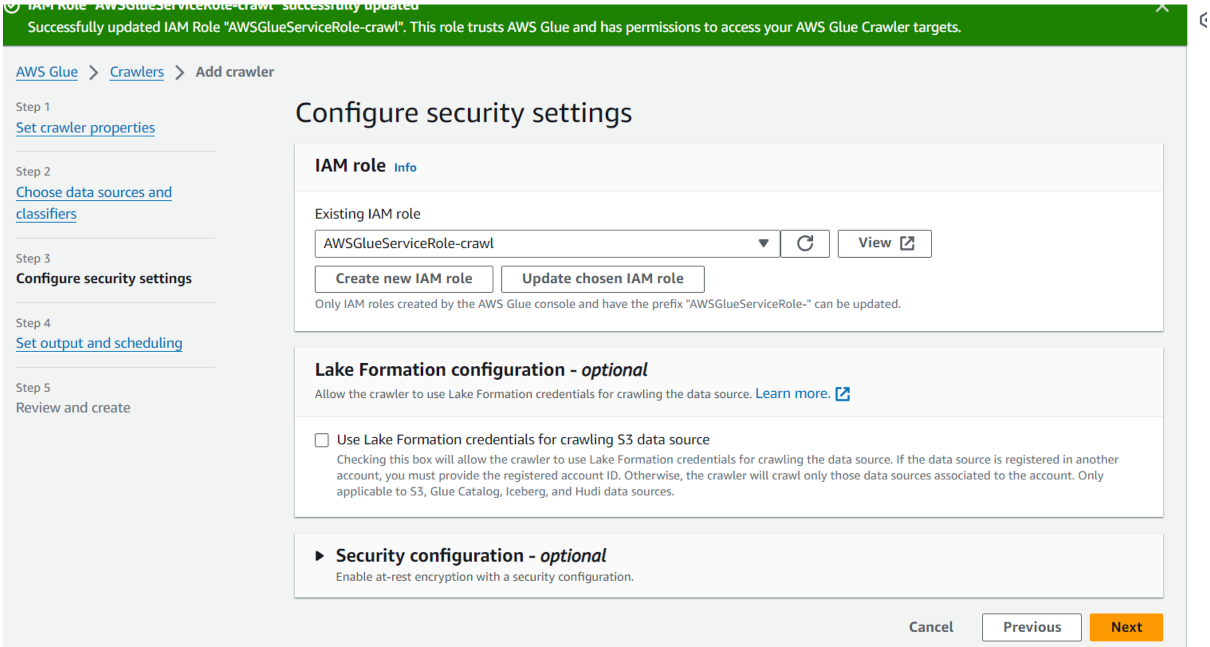
○ Yes
Select existing tables from your Glue Data Catalog.

Data sources (1) Info

The list of data sources to be scanned by the crawler.

	Type	Data source	Parameters
○	S3	s3://project1bysamir/csvfiles/	Recrawl all

▸ **Custom classifiers - *optional***
A classifier checks whether a given file is in a format the crawler can handle. If it is, the classifier creates a schema in the form of a StructType object that matches that data format.



IAM Role "AWSGlueServiceRole-crawl" successfully updated
Successfully updated IAM Role "AWSGlueServiceRole-crawl". This role trusts AWS Glue and has permissions to access your AWS Glue Crawler targets.

AWS Glue > Crawlers > Add crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and create

Configure security settings

IAM role Info

Existing IAM role
AWSGlueServiceRole-crawl

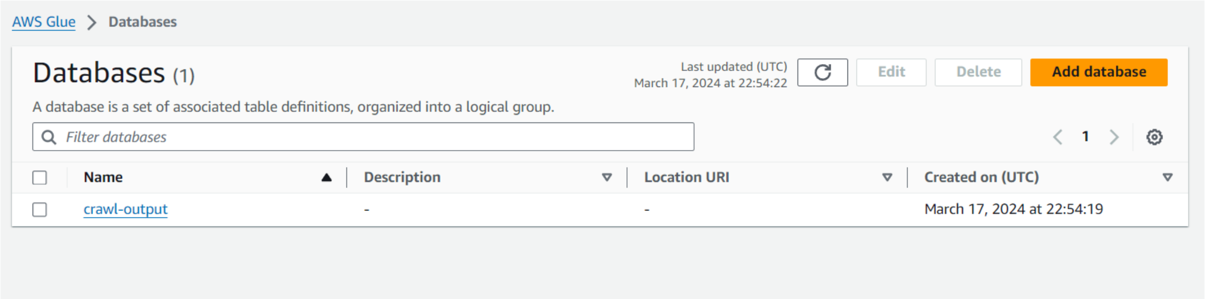
Only IAM roles created by the AWS Glue console and have the prefix "AWSGlueServiceRole-" can be updated.

Lake Formation configuration - *optional*

Allow the crawler to use Lake Formation credentials for crawling the data source. Learn more.

□ Use Lake Formation credentials for crawling S3 data source
Checking this box will allow the crawler to use Lake Formation credentials for crawling the data source. If the data source is registered in another account, you must provide the registered account ID. Otherwise, the crawler will crawl only those data sources associated to the account. Only applicable to S3, Glue Catalog, Iceberg, and Hudi data sources.

▸ **Security configuration - *optional***
Enable at-rest encryption with a security configuration.



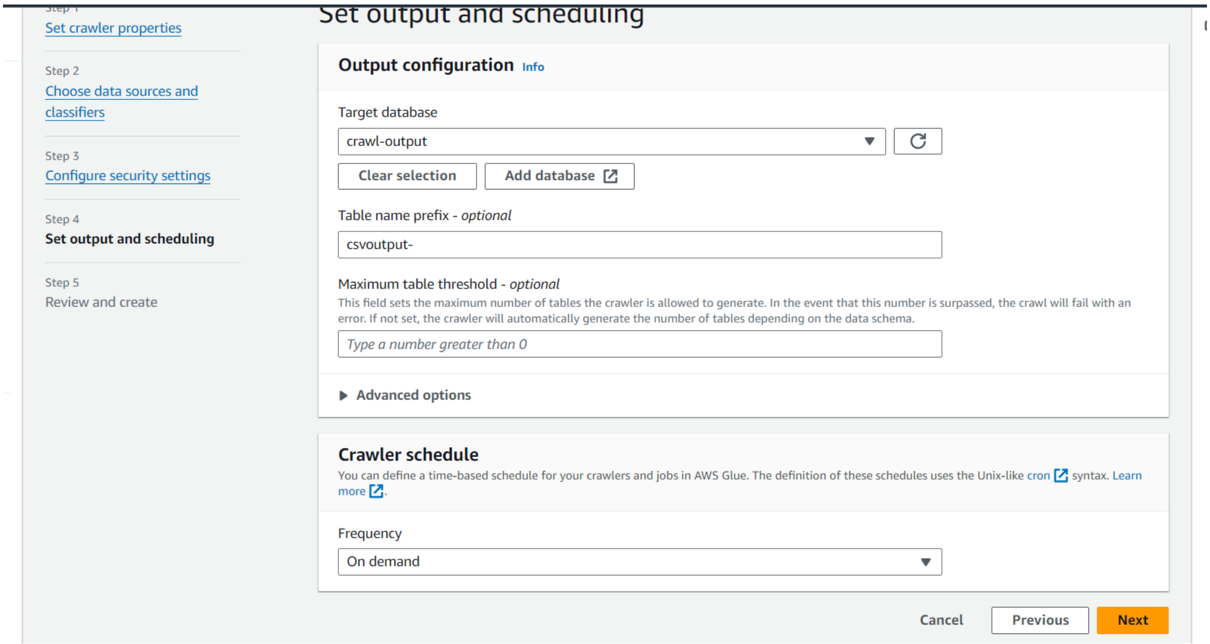
AWS Glue > Databases

Databases (1)

Last updated (UTC)
March 17, 2024 at 22:54:22

A database is a set of associated table definitions, organized into a logical group.

□	Name	Description	Location URI	Created on (UTC)
□	crawl-output	-	-	March 17, 2024 at 22:54:19


Step 1
Set crawler properties
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create

Set output and scheduling

Output configuration Info

Target database: crawl-output

Table name prefix - *optional*: csvoutput-

Maximum table threshold - *optional*

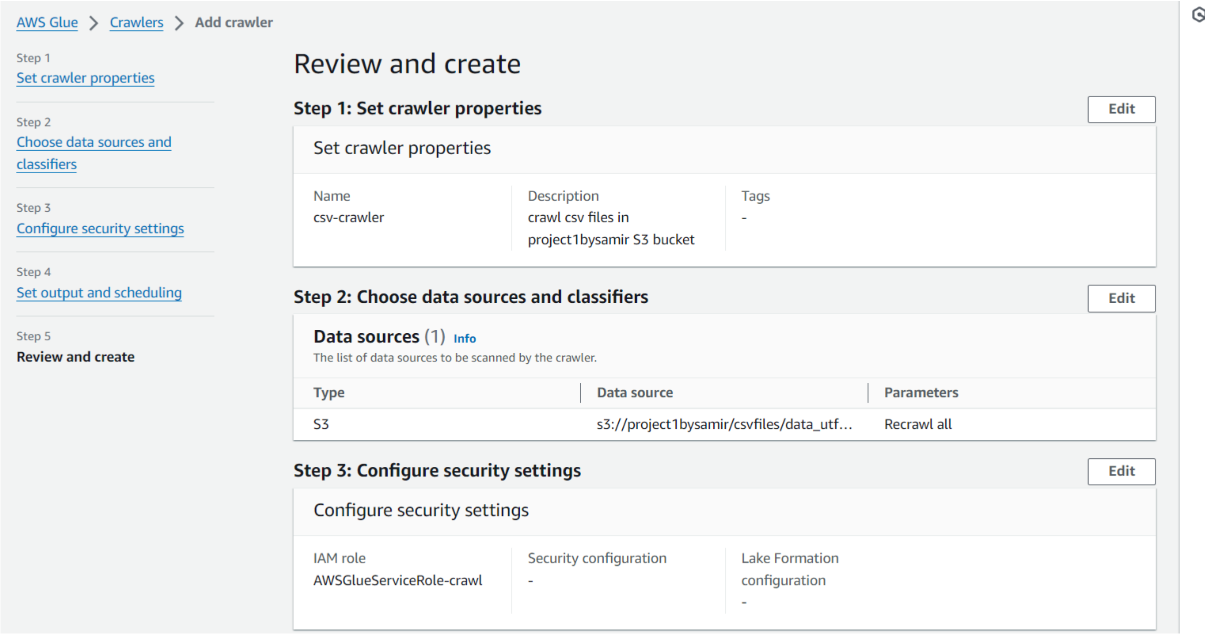
This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.

Type a number greater than 0

Crawler schedule

You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. Learn more.

Frequency: On demand


AWS Glue > Crawlers > Add crawler
Step 1
Set crawler properties
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create

Review and create

Step 1: Set crawler properties

Set crawler properties

Name	Description	Tags
csv-crawler	crawl csv files in project1bysamir S3 bucket	-

Step 2: Choose data sources and classifiers

Data sources (1) Info

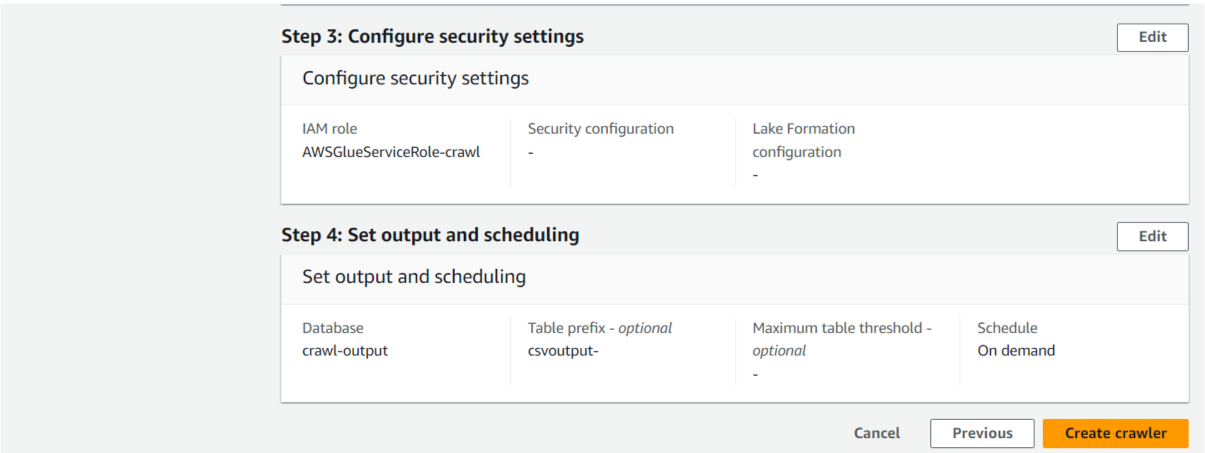
The list of data sources to be scanned by the crawler.

Type	Data source	Parameters
S3	s3://project1bysamir/csvfiles/data_utf…	Recrawl all

Step 3: Configure security settings

Configure security settings

IAM role	Security configuration	Lake Formation configuration
AWSGlueServiceRole-crawl	-	-



Step 3: Configure security settings

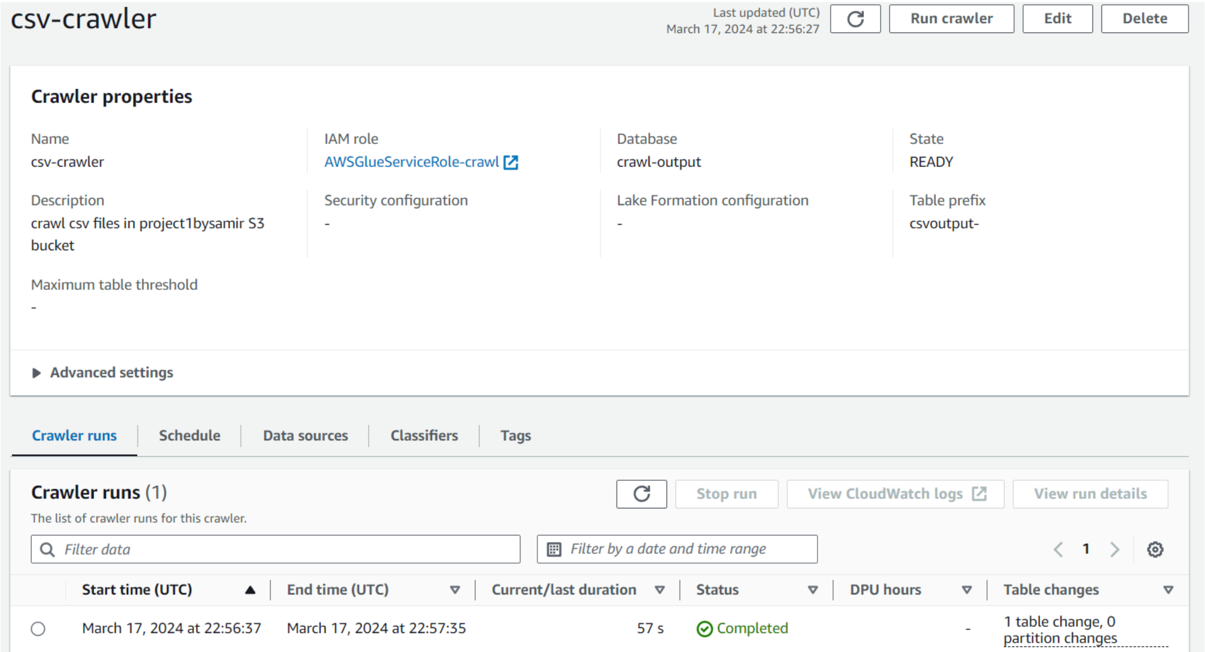
Configure security settings

IAM role	Security configuration	Lake Formation configuration
AWSGlueServiceRole-crawl	-	-

Step 4: Set output and scheduling

Set output and scheduling

Database	Table prefix - *optional*	Maximum table threshold - *optional*	Schedule
crawl-output	csvoutput-	-	On demand



csv-crawler

Last updated (UTC)
March 17, 2024 at 22:56:27

Crawler properties

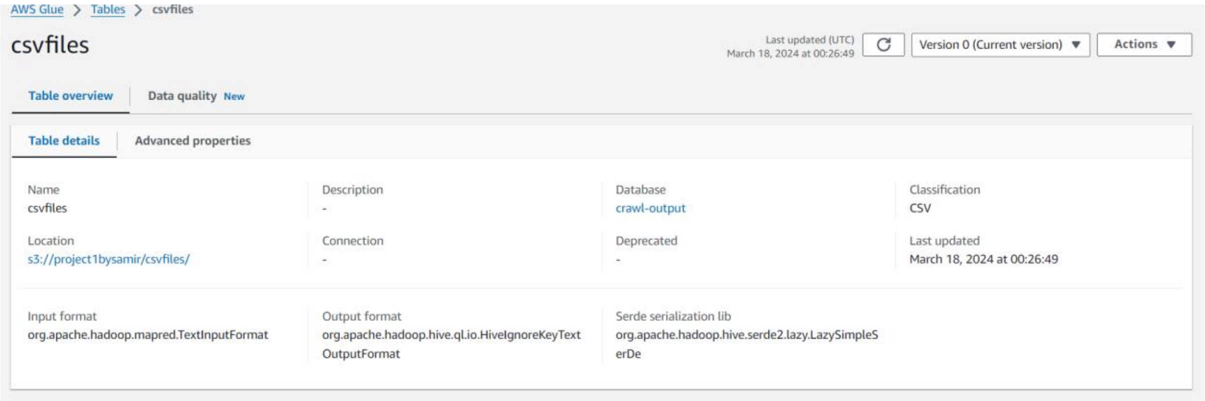
Name csv-crawler	IAM role AWSGlueServiceRole-crawl	Database crawl-output	State READY
Description crawl csv files in project1bysamir S3 bucket	Security configuration -	Lake Formation configuration -	Table prefix csvoutput-
Maximum table threshold -			

▸ Advanced settings

Crawler runs (1)

The list of crawler runs for this crawler.

	Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes
○	March 17, 2024 at 22:56:37	March 17, 2024 at 22:57:35	57 s	✓ Completed	-	1 table change, 0 partition changes

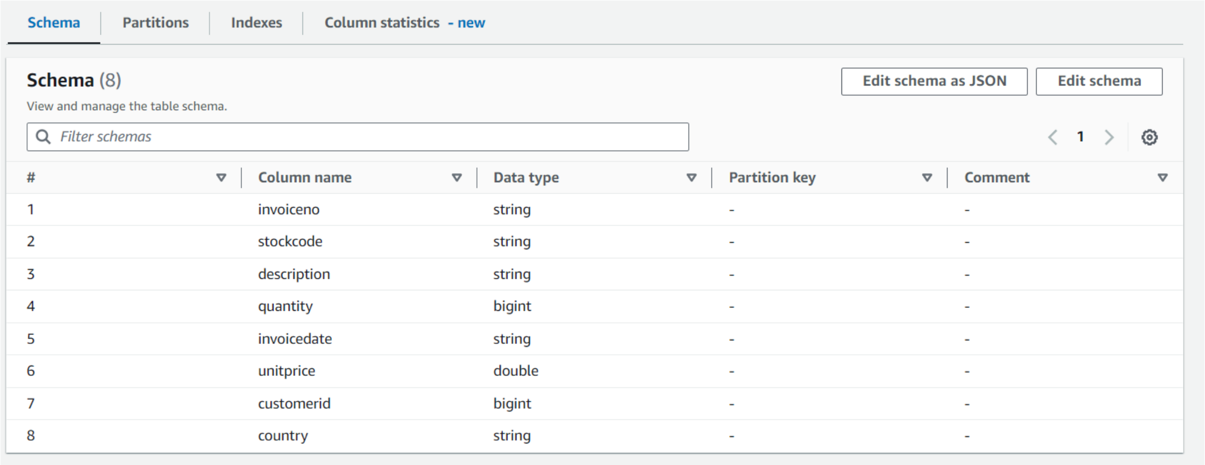


AWS Glue > Tables > csvfiles

csvfiles

Last updated (UTC)
March 18, 2024 at 00:26:49

Name csvfiles	Description -	Database crawl-output	Classification CSV
Location s3://project1bysamir/csvfiles/	Connection -	Deprecated -	Last updated March 18, 2024 at 00:26:49
Input format org.apache.hadoop.mapred.TextInputFormat	Output format org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat	Serde serialization lib org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe	



Schema (8)

View and manage the table schema.

#	Column name	Data type	Partition key	Comment
1	invoiceno	string	-	-
2	stockcode	string	-	-
3	description	string	-	-
4	quantity	bigint	-	-
5	invoicedate	string	-	-
6	unitprice	double	-	-
7	customerid	bigint	-	-
8	country	string	-	-

3. Transform Data with AWS Glue Job:

Objective: Convert CSV data into Parquet format for efficient querying and analysis.

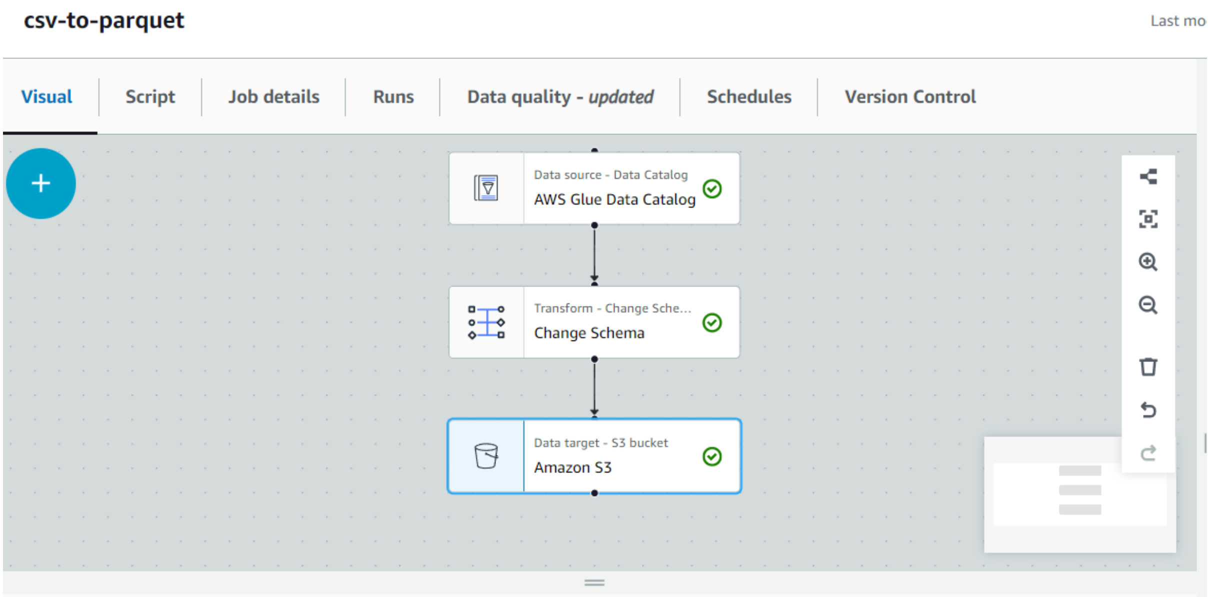
Action Items:

Create an AWS Glue job specifying the source data (the table generated by the crawler), the target format (Parquet), and the target location (a new or existing S3 bucket).

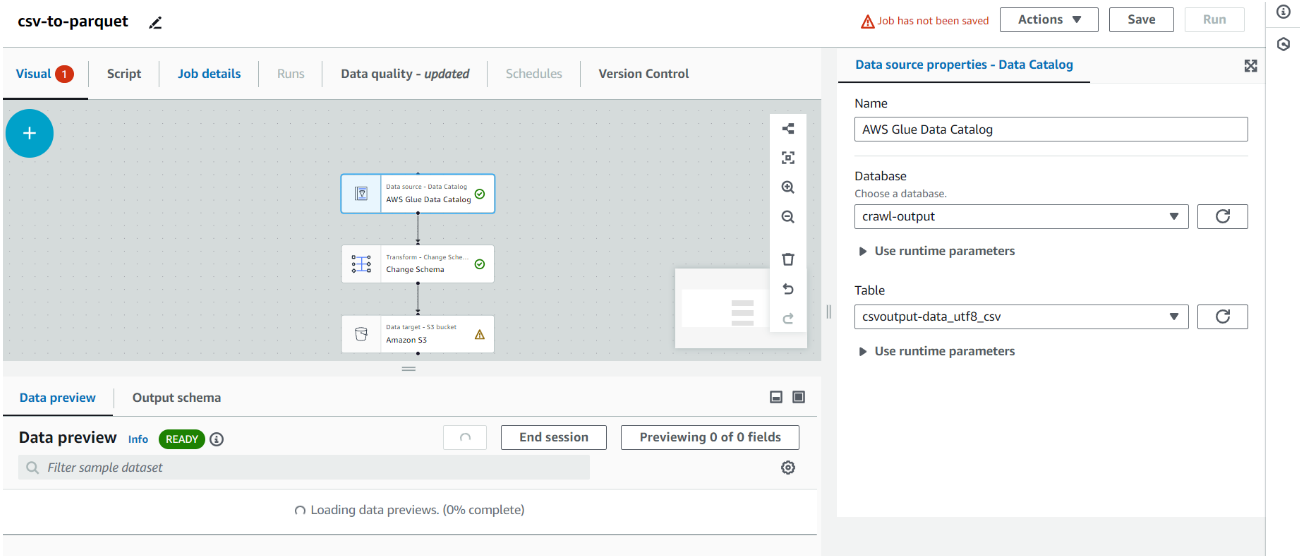
Choose or create an appropriate IAM role with the necessary permissions for the Glue job.

Write or generate the transformation script. AWS Glue can auto-generate a script for converting formats with minimal adjustments.

Run the Glue job to transform the CSV data into Parquet format.



csv-to-parquet

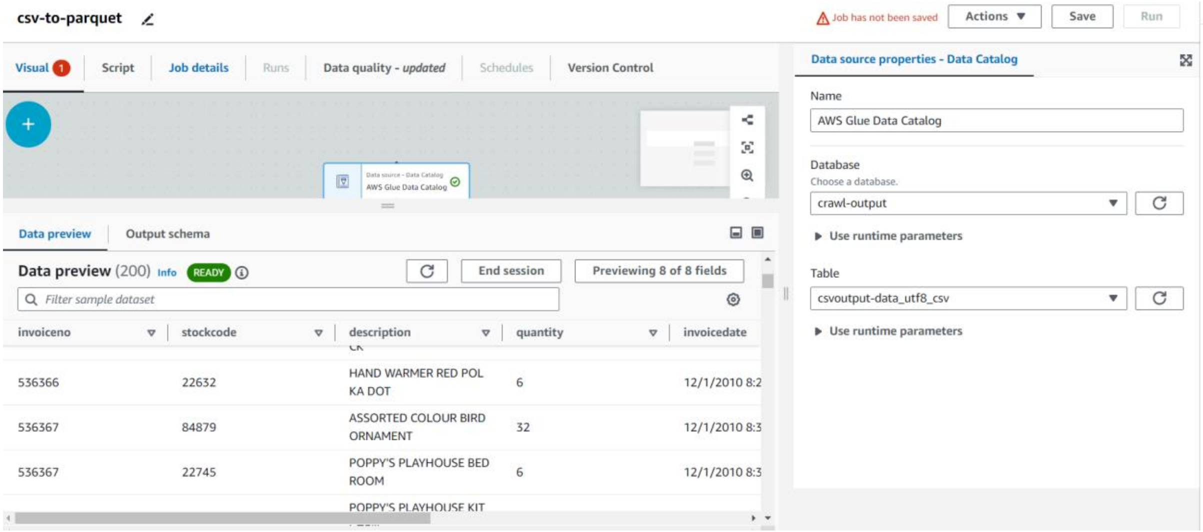


csv-to-parquet

Name: AWS Glue Data Catalog

Database — Choose a database.: crawl-output

Table: csvoutput-data_utf8_csv



csv-to-parquet

Job has not been saved | Actions ▼ | Save | Run

Visual 1 | Script | Job details | Runs | Data quality - *updated* | Schedules | Version Control

Data source properties - Data Catalog

Name: AWS Glue Data Catalog

Database
Choose a database.
crawl-output

▶ Use runtime parameters

Table: csvoutput-data_utf8_csv

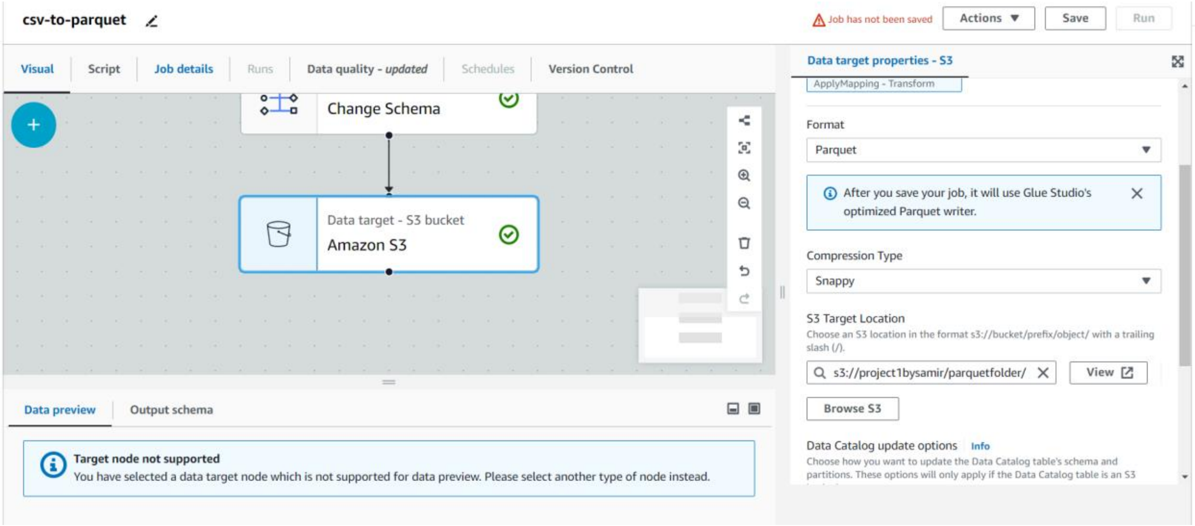
▶ Use runtime parameters

Data preview | Output schema

Data preview (200) Info READY | End session | Previewing 8 of 8 fields

Filter sample dataset

invoiceno	stockcode	description	quantity	invoicedate
536366	22632	HAND WARMER RED POLKA DOT	6	12/1/2010 8:2
536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	12/1/2010 8:3
536367	22745	POPPY'S PLAYHOUSE BED ROOM	6	12/1/2010 8:3
		POPPY'S PLAYHOUSE KIT		



csv-to-parquet

Job has not been saved | Actions ▼ | Save | Run

Visual | Script | Job details | Runs | Data quality - *updated* | Schedules | Version Control

Data target properties - S3

ApplyMapping - Transform

Format: Parquet

After you save your job, it will use Glue Studio's optimized Parquet writer.

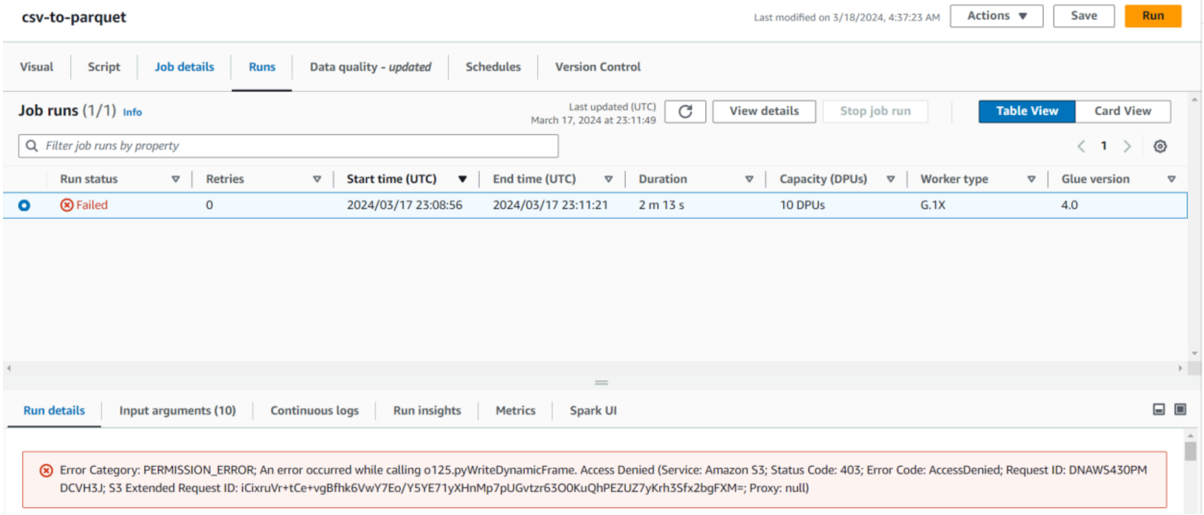
Compression Type: Snappy

S3 Target Location
Choose an S3 location in the format s3://bucket/prefix/object/ with a trailing slash (/).
s3://project1bysamir/parquetfolder/ | View | Browse S3

Data Catalog update options Info
Choose how you want to update the Data Catalog table's schema and partitions. These options will only apply if the Data Catalog table is an S3

Data preview | Output schema

Target node not supported
You have selected a data target node which is not supported for data preview. Please select another type of node instead.



csv-to-parquet

Last modified on 3/18/2024, 4:37:23 AM | Actions ▼ | Save | Run

Visual | Script | Job details | Runs | Data quality - *updated* | Schedules | Version Control

Job runs (1/1) Info

Last updated (UTC) March 17, 2024 at 23:11:49 | View details | Stop job run | Table View | Card View

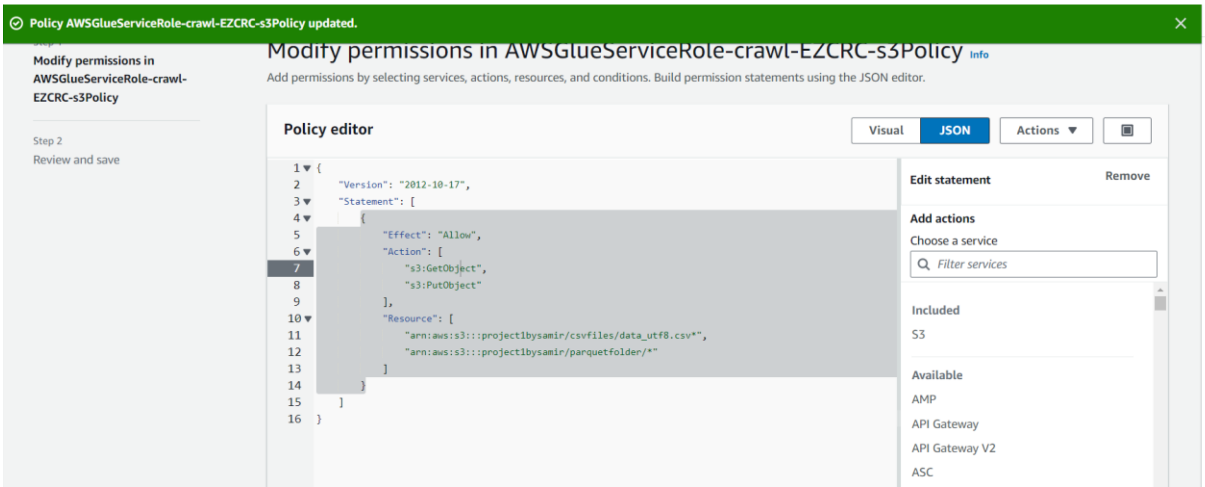
Filter job runs by property

	Run status	Retries	Start time (UTC)	End time (UTC)	Duration	Capacity (DPUs)	Worker type	Glue version
●	Failed	0	2024/03/17 23:08:56	2024/03/17 23:11:21	2 m 13 s	10 DPUs	G.1X	4.0

Run details | Input arguments (10) | Continuous logs | Run insights | Metrics | Spark UI

```
Error Category: PERMISSION_ERROR; An error occurred while calling o125.pyWriteDynamicFrame. Access Denied (Service: Amazon S3; Status Code: 403; Error Code: AccessDenied; Request ID: DNAWS430PM DCVH3J; S3 Extended Request ID: iCixruVr+tCe+vgBfhk6VwY7Eo/Y5YE71yXHnMp7pUGvtzr63O0KuQhPEZUZ7yKrh3Sfx2bgFXM=; Proxy: null)
```

The Glue Job fails as Access was denied for the folder “Parquet” in the S3 bucket created in previous step. Add permissions, and the job is completed successfully.



Policy AWSGlueServiceRole-crawl-EZCRC-s3Policy updated.

Modify permissions in AWSGlueServiceRole-crawl-EZCRC-s3Policy

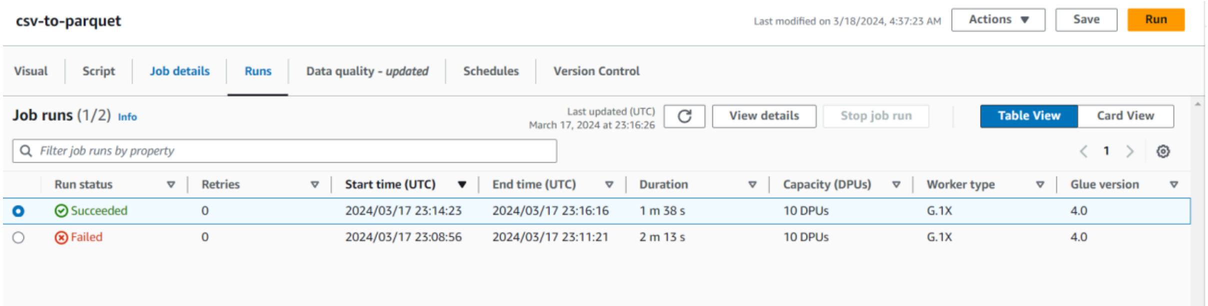
Step 2
Review and save

Modify permissions in AWSGlueServiceRole-crawl-EZCRC-s3Policy Info

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::project1bysamir/csvfiles/data_utf8.csv*",
                "arn:aws:s3:::project1bysamir/parquetfolder/*"
            ]
        }
    ]
}
```



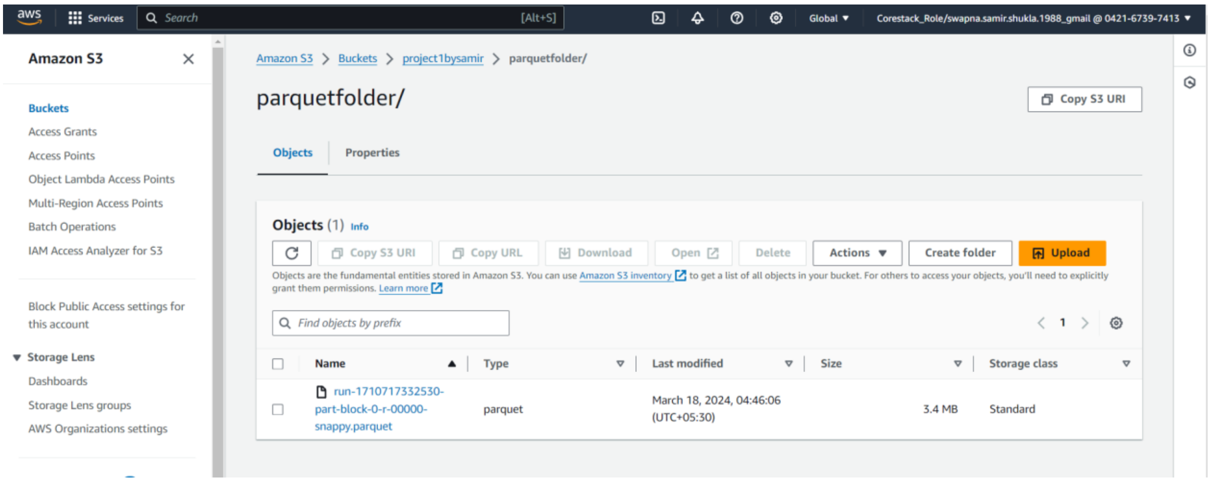
csv-to-parquet

Last modified on 3/18/2024, 4:37:23 AM

Job runs (1/2) Info

Last updated (UTC)
March 17, 2024 at 23:16:26

	Run status	Retries	Start time (UTC)	End time (UTC)	Duration	Capacity (DPUs)	Worker type	Glue version
●	Succeeded	0	2024/03/17 23:14:23	2024/03/17 23:16:16	1 m 38 s	10 DPUs	G.1X	4.0
○	Failed	0	2024/03/17 23:08:56	2024/03/17 23:11:21	2 m 13 s	10 DPUs	G.1X	4.0



aws
Services
Search
[Alt+S]
Global
Corestack_Role/swapna.samir.shukla.1988_gmail @ 0421-6739-7413
Amazon S3
Buckets
Access Grants
Access Points
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

Amazon S3 > Buckets > project1bysamir > parquetfolder/

parquetfolder/

Objects (1) Info

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
□	run-1710717332530-part-block-0-r-00000-snappy.parquet	parquet	March 18, 2024, 04:46:06 (UTC+05:30)	3.4 MB	Standard

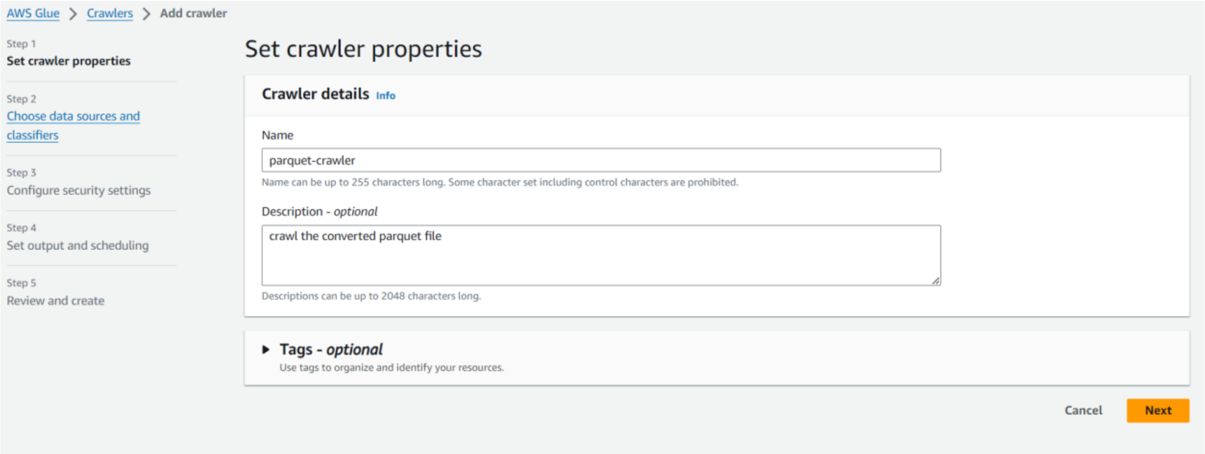
4. Crawl Transformed Data:

Objective: Catalog the metadata of the Parquet data files for querying.

Action Items:

Repeat the crawling process for the S3 location where the Parquet files are stored. This will allow AWS Athena to query the data efficiently.

Ensure the new crawler specifies the Parquet data's location, and after running, check that the Data Catalog contains the metadata for your Parquet files.



AWS Glue > Crawlers > Add crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and create

Set crawler properties

Crawler details Info

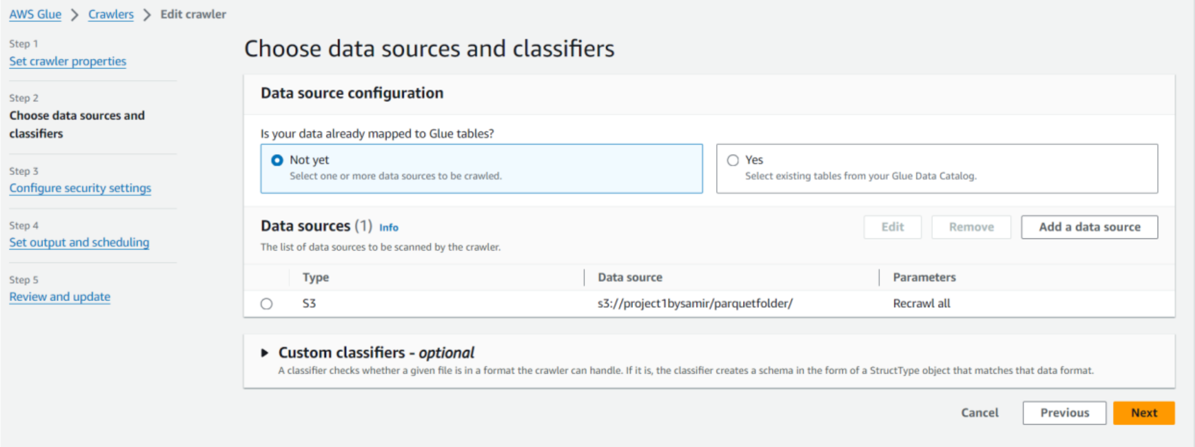
Name: parquet-crawler

Name can be up to 255 characters long. Some character set including control characters are prohibited.

Description - *optional*: crawl the converted parquet file

Descriptions can be up to 2048 characters long.

▸ Tags - *optional*
Use tags to organize and identify your resources.



AWS Glue > Crawlers > Edit crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and update

Choose data sources and classifiers

Data source configuration

Is your data already mapped to Glue tables?

● Not yet
Select one or more data sources to be crawled.

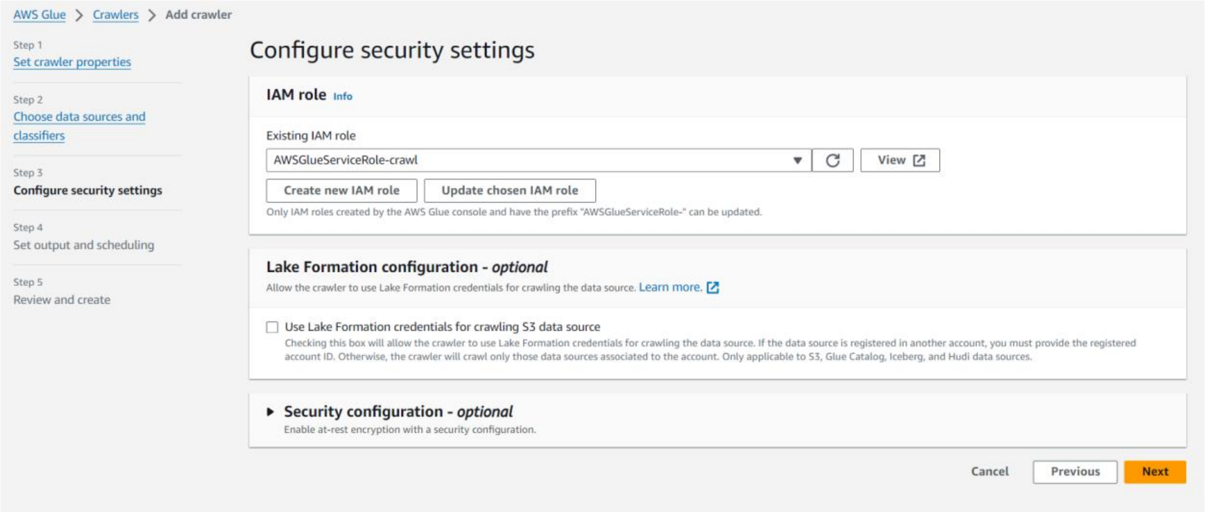
○ Yes
Select existing tables from your Glue Data Catalog.

Data sources (1) Info

The list of data sources to be scanned by the crawler.

	Type	Data source	Parameters
○	S3	s3://project1bysamir/parquetfolder/	Recrawl all

▸ Custom classifiers - *optional*
A classifier checks whether a given file is in a format the crawler can handle. If it is, the classifier creates a schema in the form of a StructType object that matches that data format.



AWS Glue > Crawlers > Add crawler

Step 1
Set crawler properties
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create

Configure security settings

IAM role Info

Existing IAM role

AWSGlueServiceRole-crawl

View
Create new IAM role
Update chosen IAM role

Only IAM roles created by the AWS Glue console and have the prefix "AWSGlueServiceRole-" can be updated.

Lake Formation configuration - *optional*

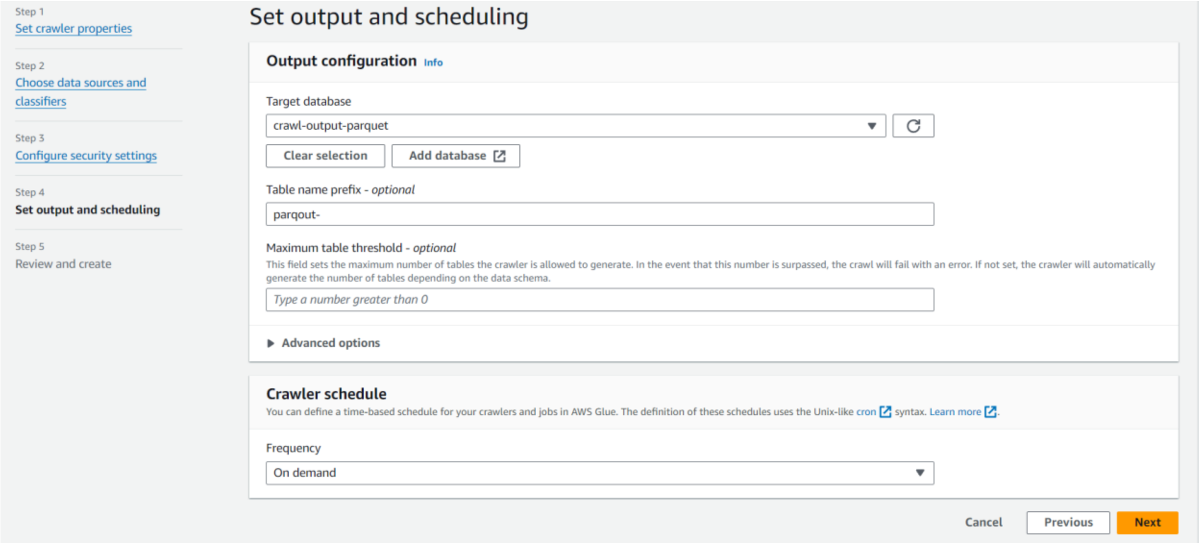
Allow the crawler to use Lake Formation credentials for crawling the data source. Learn more.

□ Use Lake Formation credentials for crawling S3 data source

Checking this box will allow the crawler to use Lake Formation credentials for crawling the data source. If the data source is registered in another account, you must provide the registered account ID. Otherwise, the crawler will crawl only those data sources associated to the account. Only applicable to S3, Glue Catalog, Iceberg, and Hudi data sources.

▸ **Security configuration - *optional***

Enable at-rest encryption with a security configuration.



Step 1
Set crawler properties
Step 2
Choose data sources and classifiers
Step 3
Configure security settings
Step 4
Set output and scheduling
Step 5
Review and create

Set output and scheduling

Output configuration Info

Target database

crawl-output-parquet

Table name prefix - *optional*

parqout-

Maximum table threshold - *optional*

This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.

Type a number greater than 0

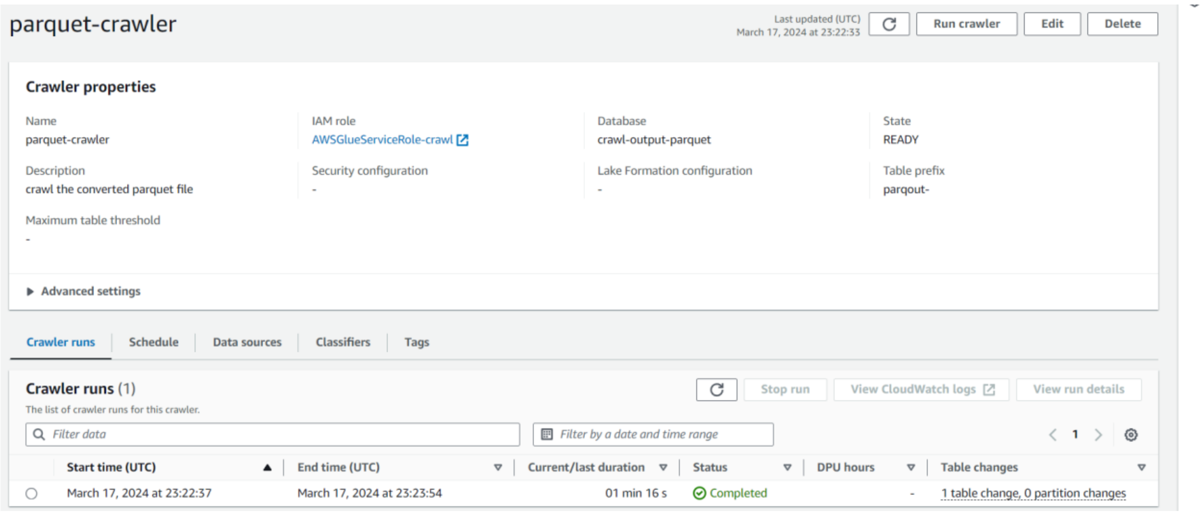
▸ Advanced options

Crawler schedule

You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. Learn more.

Frequency

On demand



parquet-crawler

Last updated (UTC)
March 17, 2024 at 23:22:33

Crawler properties

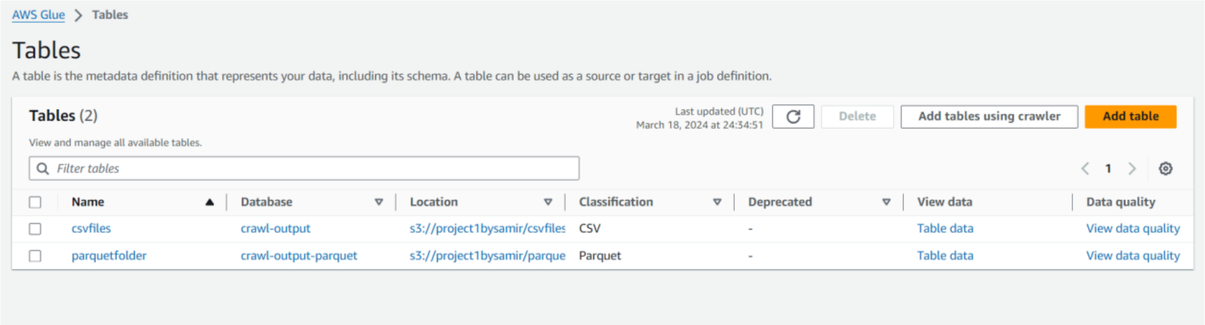
Name parquet-crawler	IAM role AWSGlueServiceRole-crawl	Database crawl-output-parquet	State READY
Description crawl the converted parquet file	Security configuration -	Lake Formation configuration -	Table prefix parqout-
Maximum table threshold -			

▸ Advanced settings

Crawler runs (1)

The list of crawler runs for this crawler.

	Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes
○	March 17, 2024 at 23:22:37	March 17, 2024 at 23:23:54	01 min 16 s	✓ Completed	-	1 table change, 0 partition changes



AWS Glue > Tables

Tables

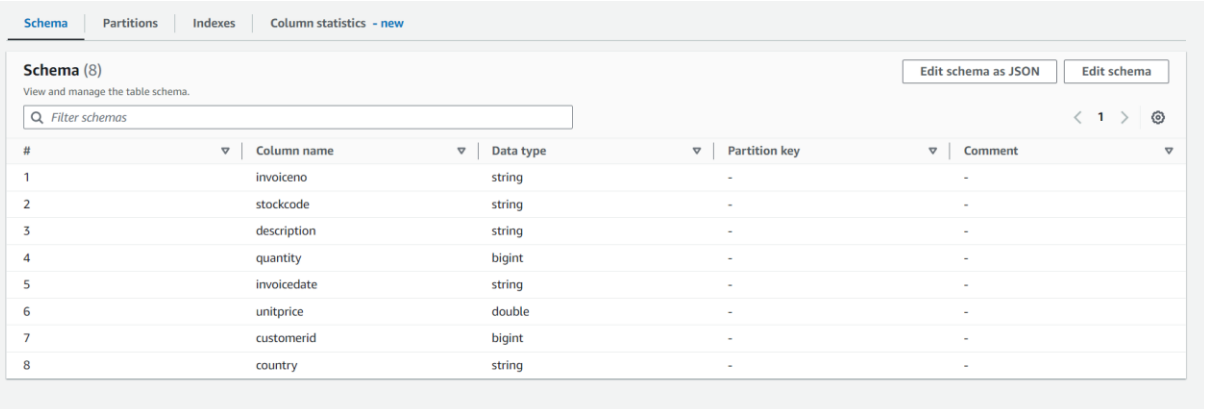
A table is the metadata definition that represents your data, including its schema. A table can be used as a source or target in a job definition.

Tables (2)

View and manage all available tables.

Last updated (UTC) March 18, 2024 at 24:34:51

□	Name	Database	Location	Classification	Deprecated	View data	Data quality
□	csvfiles	crawl-output	s3://project1bysamir/csvfiles	CSV	-	Table data	View data quality
□	parquetfolder	crawl-output-parquet	s3://project1bysamir/parque	Parquet	-	Table data	View data quality



Schema (8)

View and manage the table schema.

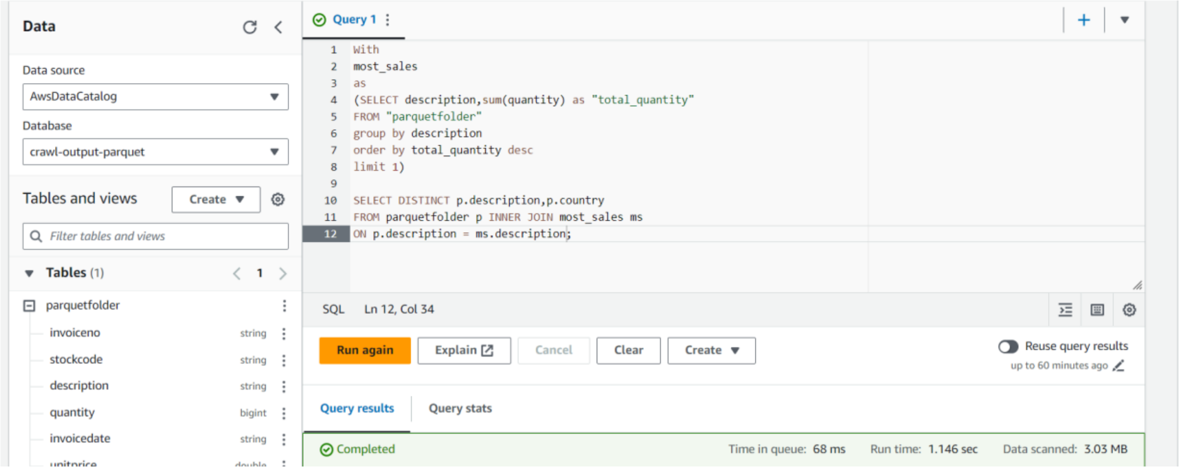
#	Column name	Data type	Partition key	Comment
1	invoiceno	string	-	-
2	stockcode	string	-	-
3	description	string	-	-
4	quantity	bigint	-	-
5	invoicedate	string	-	-
6	unitprice	double	-	-
7	customerid	bigint	-	-
8	country	string	-	-

5. Analyze Data with AWS Athena:

Objective: Query the data to uncover insights such as the best-selling item and the geographical distribution of sales.

Action Items:

Open AWS Athena. Ensure it's configured to use the database generated by the Glue crawlers.



Data source: AwsDataCatalog

Database: crawl-output-parquet

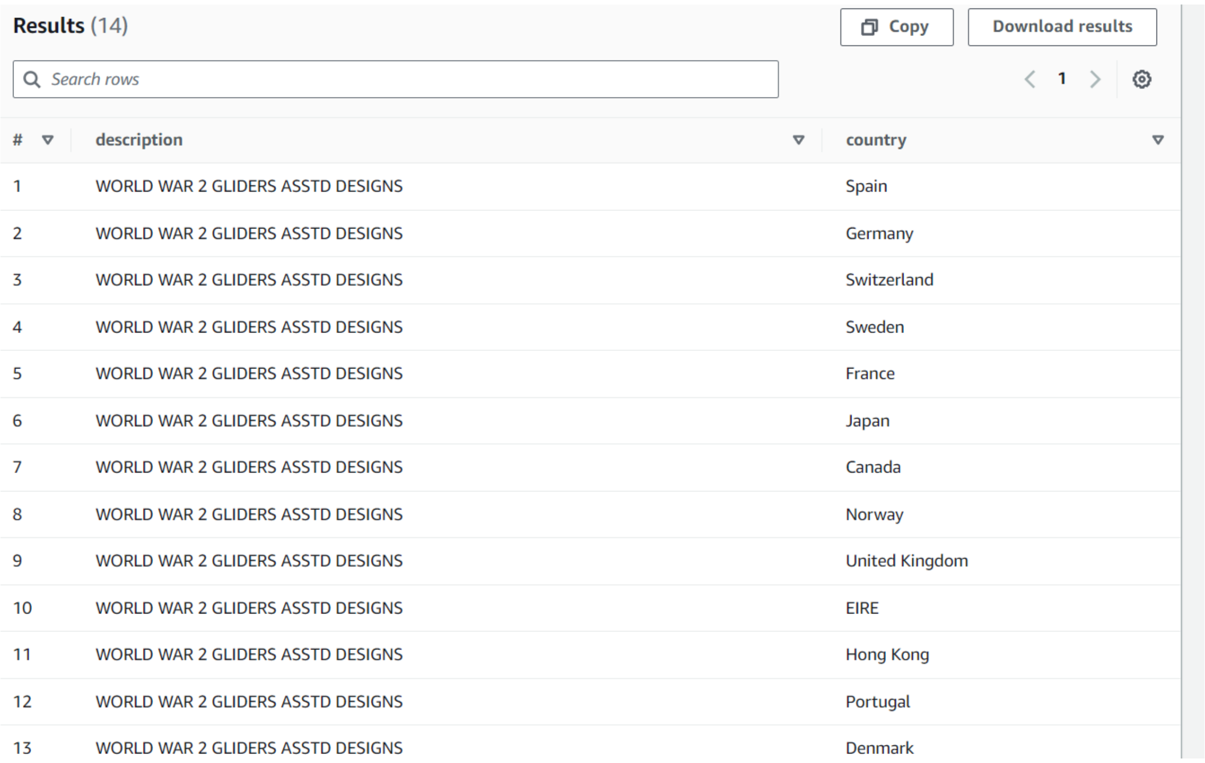
Tables (1)

parquetfolder	
invoiceno	string
stockcode	string
description	string
quantity	bigint
invoicedate	string
unitprice	double

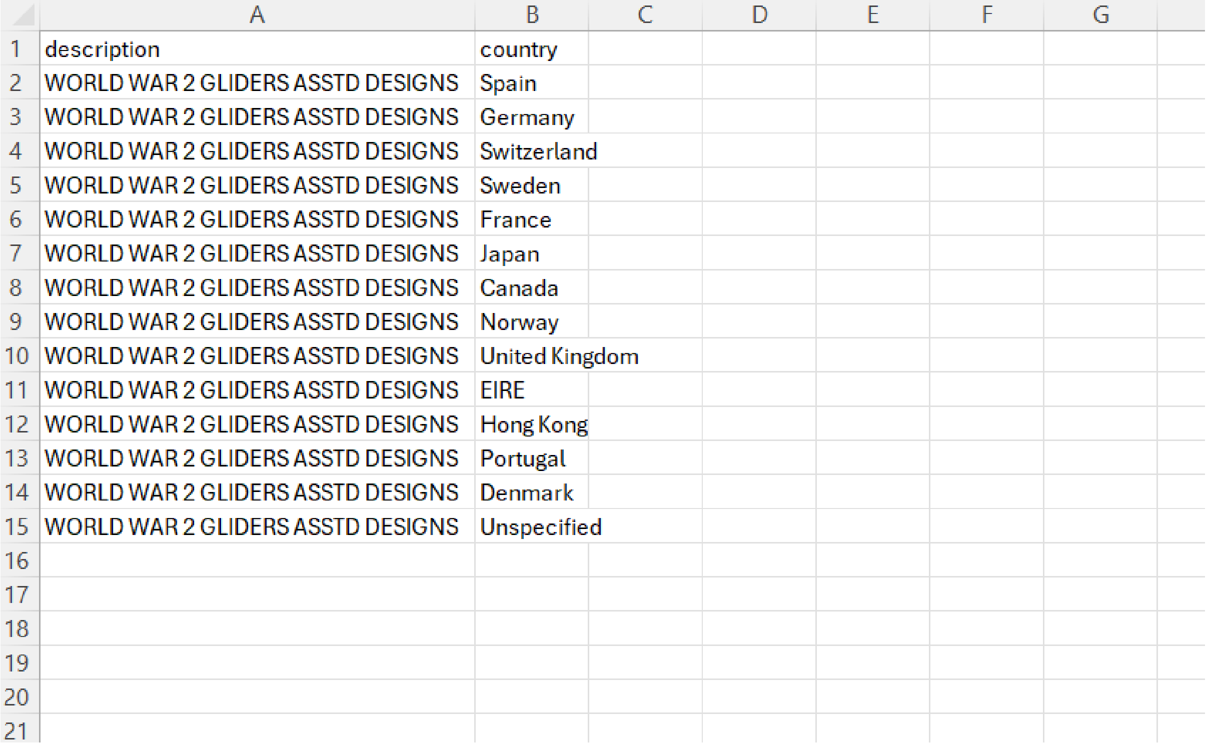
```
With
most_sales
as
(SELECT description,sum(quantity) as "total_quantity"
FROM "parquetfolder"
group by description
order by total_quantity desc
limit 1)

SELECT DISTINCT p.description,p.country
FROM parquetfolder p INNER JOIN most_sales ms
ON p.description = ms.description;
```

SQL Ln 12, Col 34

Completed Time in queue: 68 ms Run time: 1.146 sec Data scanned: 3.03 MB



#	description	country
1	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Spain
2	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Germany
3	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Switzerland
4	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Sweden
5	WORLD WAR 2 GLIDERS ASSTD DESIGNS	France
6	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Japan
7	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Canada
8	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Norway
9	WORLD WAR 2 GLIDERS ASSTD DESIGNS	United Kingdom
10	WORLD WAR 2 GLIDERS ASSTD DESIGNS	EIRE
11	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Hong Kong
12	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Portugal
13	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Denmark



	A	B	C	D	E	F	G
1	description	country					
2	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Spain					
3	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Germany					
4	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Switzerland					
5	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Sweden					
6	WORLD WAR 2 GLIDERS ASSTD DESIGNS	France					
7	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Japan					
8	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Canada					
9	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Norway					
10	WORLD WAR 2 GLIDERS ASSTD DESIGNS	United Kingdom					
11	WORLD WAR 2 GLIDERS ASSTD DESIGNS	EIRE					
12	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Hong Kong					
13	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Portugal					
14	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Denmark					
15	WORLD WAR 2 GLIDERS ASSTD DESIGNS	Unspecified					
16							
17							
18							
19							
20							
21							