AUTOMATED CLOUD BASED ETL PIPELINE WITH RDS,S3,GLUE AND SNS INTEGRATION

COACH : LEON YIGAL BENJAMIN

MENTOR : BADRI PRASAD BALAKRISHNAN

TEAM MEMBERS :-

RAHUL V - 227633

PRAVEEN G - 2276623

DHAMOTHARAN A S - 2276706

VISHWA R - 2276634

LIKITHA CHINTAPALLI - 2276784

BU/SL : Data
Technology : AWS

ABSTRACT :-

This project scenario involves the implementation of an Extract, Transform, and Load (ETL) process in the AWS cloud. The project begins by creating an Amazon RDS database and establishing a connection with MySQL. A snapshot of the RDS database is exported to an Amazon S3 bucket using an IAM role for Backup and security. The exported CSV files are then uploaded to an S3 bucket. When a file is uploaded to an S3 bucket, the AWS lambda gets triggered, and it calls the Step function State machine. So the step-function state machine automatically orchestrates the glue crawler, and the AWS Glue crawler is used to scan and catalog the CSV files, storing the crawled data in a newly created database in the AWS Glue Data Catalog. The CSV files are transformed to Parquet format using the visual script editor in AWS Glue and stored in a separate S3 bucket. The Glue crawler is again automatically done by step-function, where step-function will get triggered by lambda whenever a file uploads in the S3 bucket. Then it will scan and crawl the Parquet files, storing the data automatically in another database in the Data Catalog. Data transformation is performed using the visual script editor and AWS Glue DataBrew, and the transformed data is loaded into a new S3 bucket. EventBridge is integrated to trigger the Simple Notification Service (SNS) for email notifications upon successful completion of the Glue job, providing timely updates on job status. This ETL project in the AWS cloud enables efficient data extraction, transformation, and loading, leveraging the scalability, reliability, and flexibility of AWS services.

Introduction :-

Purpose of the Document :

The purpose of this document is to provide an abstract or summary of a project that involves the extraction, transformation, and loading processes using AWS services such as RDS,Glue,Databrew,S3, Eventbridge, and SNS. The document aims to highlight the key components of the project, including its objectives, the technologies involved, and the benefits it offers.

Project Overview :-

This project aims to create the database in the RDS and extract that database by using snapshots, then by using the AWS glue, just crawling and storing it in the data catalog, and then doing the transformation, which is concat,split,drop, join, and dateformat. Then storing the result in the S3 bucket. If the glue job is run, it will send the email to the user by using SNS.

AWS – AMAZON WEB SERVICES

Amazon Web Services (AWS) is a comprehensive and widely adopted cloud computing platform offered by Amazon. It provides a vast array of on-demand computing resources and services, delivered over the internet, to help businesses and individuals

build and run their applications, store and analyze data, and scale their operations with ease.
This project utilizes the following AWS services :-

1. **Amazon Ec2 :** Amazon Elastic Compute Cloud (Amazon EC2) is a web service provided by Amazon Web Services (AWS) that offers resizable compute capacity in the cloud. It allows you to quickly provision virtual servers, known as EC2 instances, and run applications and workloads on them.
2. **AWS RDS** : Amazon Relational Database Service (Amazon RDS) is a collection of managed services that makes it simple to set up, operate, and scale databases in the cloud.. process by monitoring the S3 bucket for new files and triggering the Glue job. Lambda functions are also utilized to crawl the destination bucket after the conversion is completed.

3. **Iam Roles :** AWS Identity and Access Management (IAM) roles are a fundamental component of the AWS security model. They enable you to securely delegate permissions to AWS resources, such as EC2 instances, Lambda functions, or users within your AWS account.

4. **Amazon S3** : Amazon Simple Storage Service (S3) is an object storage service that offers industry-leading scalability, durability, and security. It is used as the storage infrastructure for the CSV files, as well as the destination bucket for the transformed Parquet files.

5. **AWS KMS :** AWS Key Management Service (KMS) is a fully managed service provided by Amazon Web Services (AWS) that

allows you to create and control the encryption keys used to encrypt your data. KMS helps you protect sensitive information by providing a secure and scalable solution for key management.

6. **AWS Glue** : AWS Glue is a fully managed extract, transform, and load (ETL) service that simplifies the process of preparing and transforming data for analytics. It is used in this project to convert the CSV files to Parquet format and optimize the data for efficient processing and storage.
7. AWS Lambda : AWS Lambda is a serverless compute service provided by Amazon Web Services (AWS). It allows you to run your code without provisioning or managing servers, enabling you to focus on writing and deploying your applications rather than worrying about infrastructure.

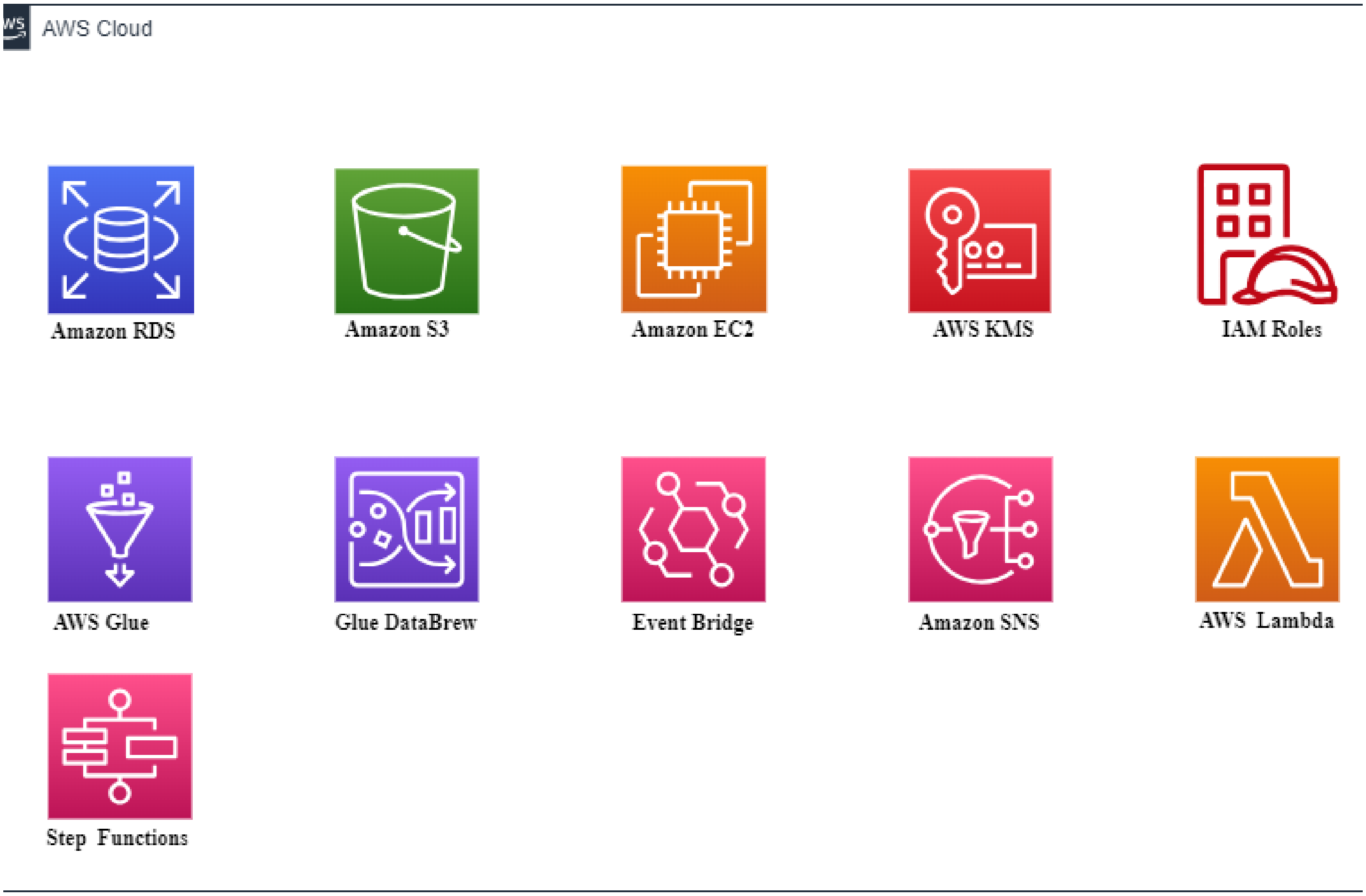
8. AWS Step Function : AWS Step Functions is a serverless workflow orchestration service provided by Amazon Web Services (AWS). It allows you to coordinate and visualize the execution of multiple AWS Lambda functions, AWS Batch jobs, Amazon ECS tasks, and other serverless or traditional compute resources as a series of steps, forming a workflow.

9. **Glue DataBrew** : AWS Glue DataBrew is a new visual data preparation tool that makes it easy for data analysts and data scientists to clean and normalize data to prepare it for analytics and machine learning.

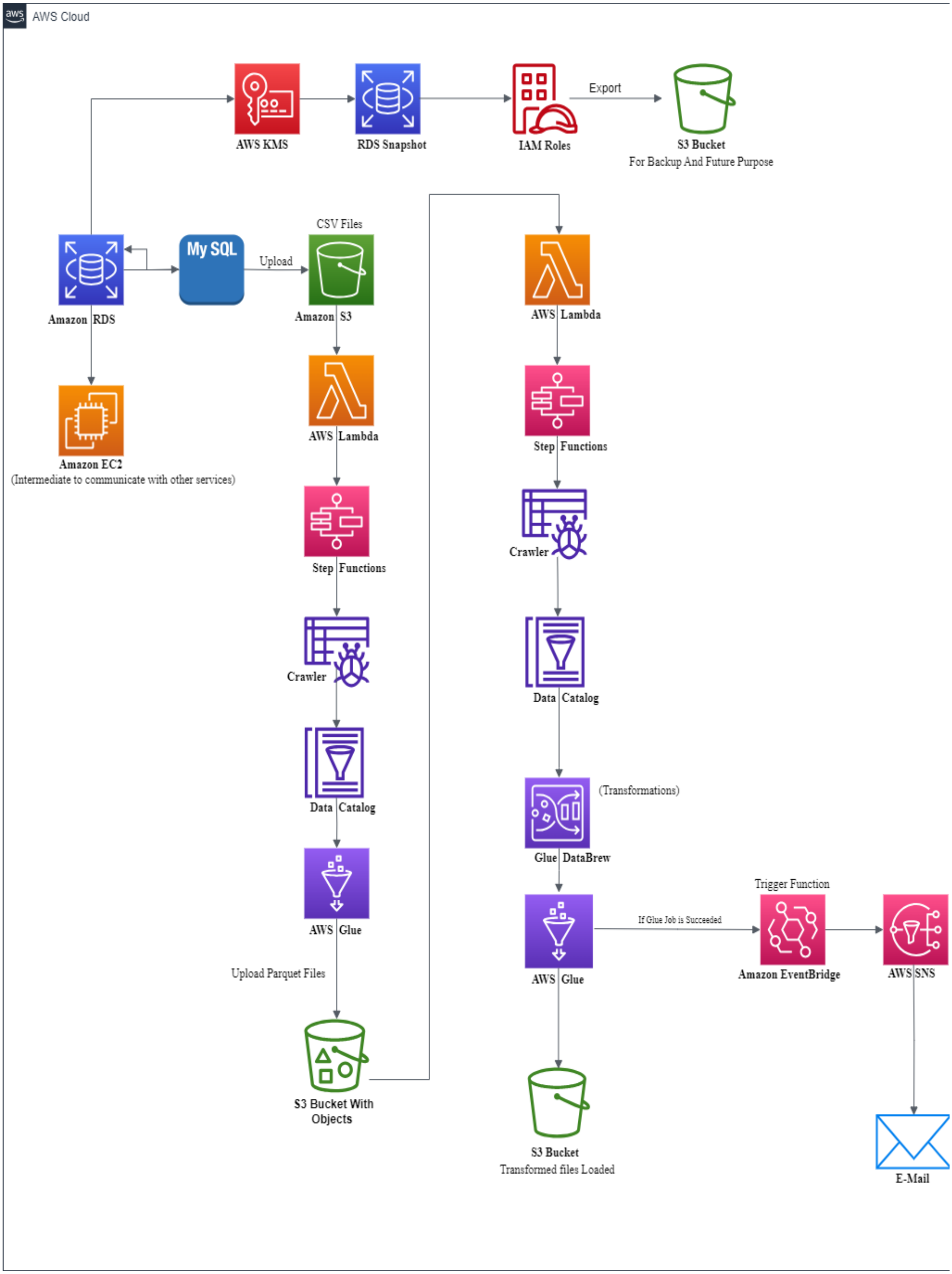
10. **Amazon SNS** : Amazon Simple Notification Service (SNS) is a fully managed messaging service that enables the sending of notifications to subscribers. In this project, SNS is used to send notifications to subscribers about the completion of the Glue job, providing timely updates on the status of the conversion process.

11. Eventbridge : EventBridge is a fully managed event bus service provided by Amazon Web Services (AWS). It enables the creation, routing, and processing of events between different applications or services within an AWS environment or across various software systems.

SERVICES USED :



ARCHITECTURE :



Implementation of Project Architecture :

1. **Create IAM Role** :

AWS Identity and Access Management (IAM) is a web service provided by Amazon Web Services (AWS) that allows you to manage access to AWS resources securely. IAM enables you to create and manage AWS users, groups, and permissions to control who can access specific AWS resources and what actions they can perform

Users : IAM users are entities within your AWS account that represent individual people or applications requiring access to AWS resources. Each user is assigned a unique set of security credentials (access key ID and secret access key) for programmatic access and a password or a password policy for AWS Management Console access.

Groups : IAM groups are collections of IAM users. You can manage permissions for multiple users collectively by attaching policies to a group rather than individual users.

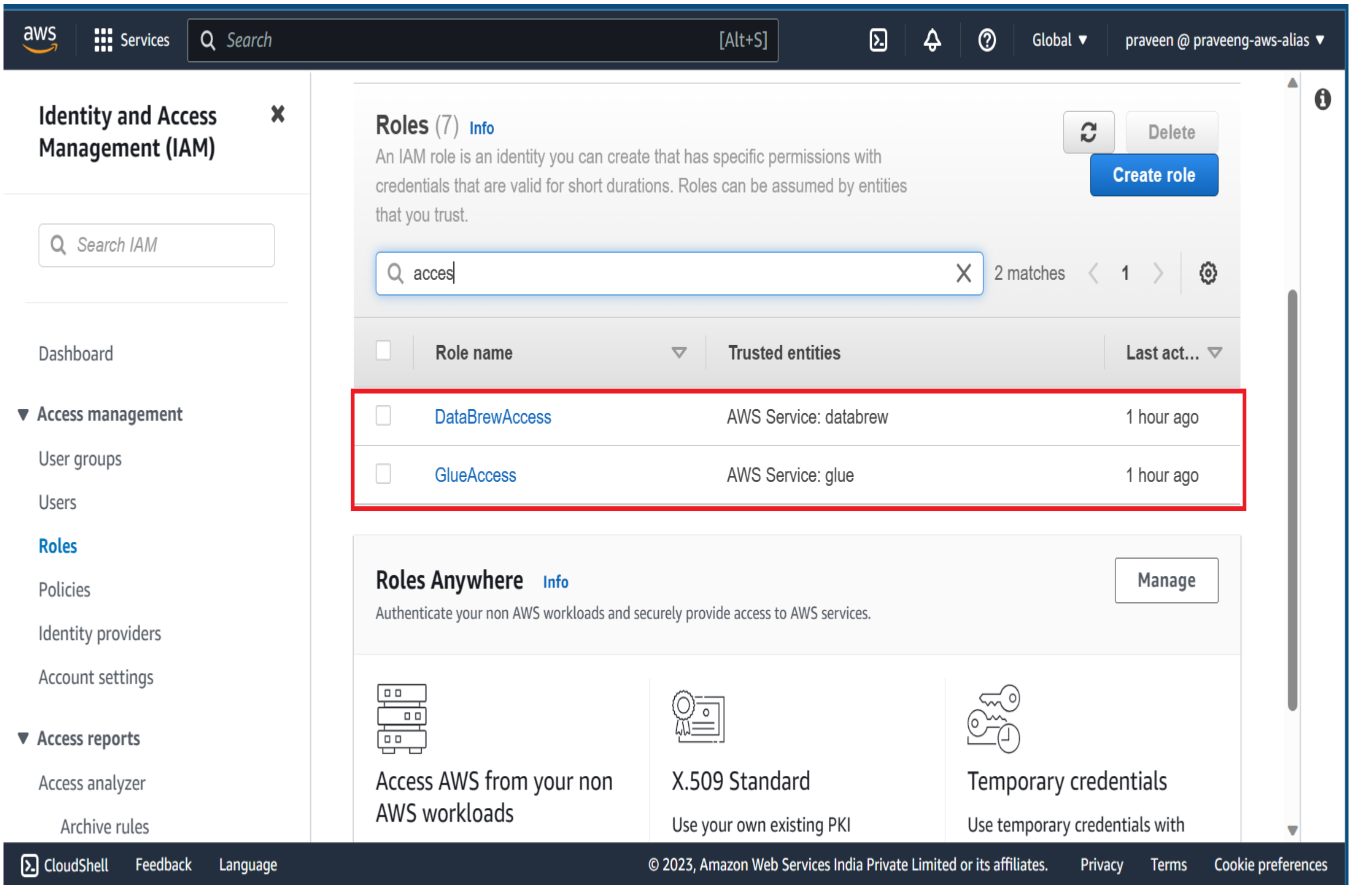
Roles : IAM roles are similar to users but are not associated with a specific identity. Roles define a set of permissions that determine what actions can be performed and what resources can be accessed.

ROLE 1 : GlueAccess

DESCRIPTION : The purpose of this role is to grant specific permissions that align with the needs of the user or service accessing Glue.

PERMISSION : Administrator Access

PURPOSE : The purpose of granting "Administrator access" to AWS Glue is to provide users or roles with full administrative privileges and permissions within the AWS Glue service. By assigning the Administrator access level, users can have complete control over all aspects of AWS Glue, including managing databases, tables, jobs, triggers, and connections.



Roles (7) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

acces — 2 matches — 1

□	Role name	Trusted entities	Last act...
□	DataBrewAccess	AWS Service: databrew	1 hour ago
□	GlueAccess	AWS Service: glue	1 hour ago

Roles Anywhere Info

Authenticate your non AWS workloads and securely provide access to AWS services.

Access AWS from your non AWS workloads

X.509 Standard

Use your own existing PKI

Temporary credentials

Use temporary credentials with

© 2023, Amazon Web Services India Private Limited or its affiliates.

ROLE 2 : **DataBrewAcess**

DESCRIPTION : The "DataBrewRole" is an IAM role designed to provide specific permissions required for AWS DataBrew, a data preparation and data blending service offered by AWS. The role is tailored to meet the needs and actions associated with utilizing DataBrew effectively.

PERMISSION : Administrator Access

PURPOSE : Granting "Administrator Access" permission within AWS provides users or roles with full administrative privileges and permissions across all services and resources within the AWS account. The purpose of granting Administrator access is to allow complete control and management over the entire AWS environment.

ROLE 3 : **RDS-S3-ExportRole:**

DESCRIPTION : We need to grant specific permissions that allow the role to interact with Athena database and perform necessary actions.

PERMISSION 1 : **S3 Full Access :**Provides full access to Amazon s3 via the AWS Management Consoles3 full access.

STEP 1: EC2 INSTANCE CREATION

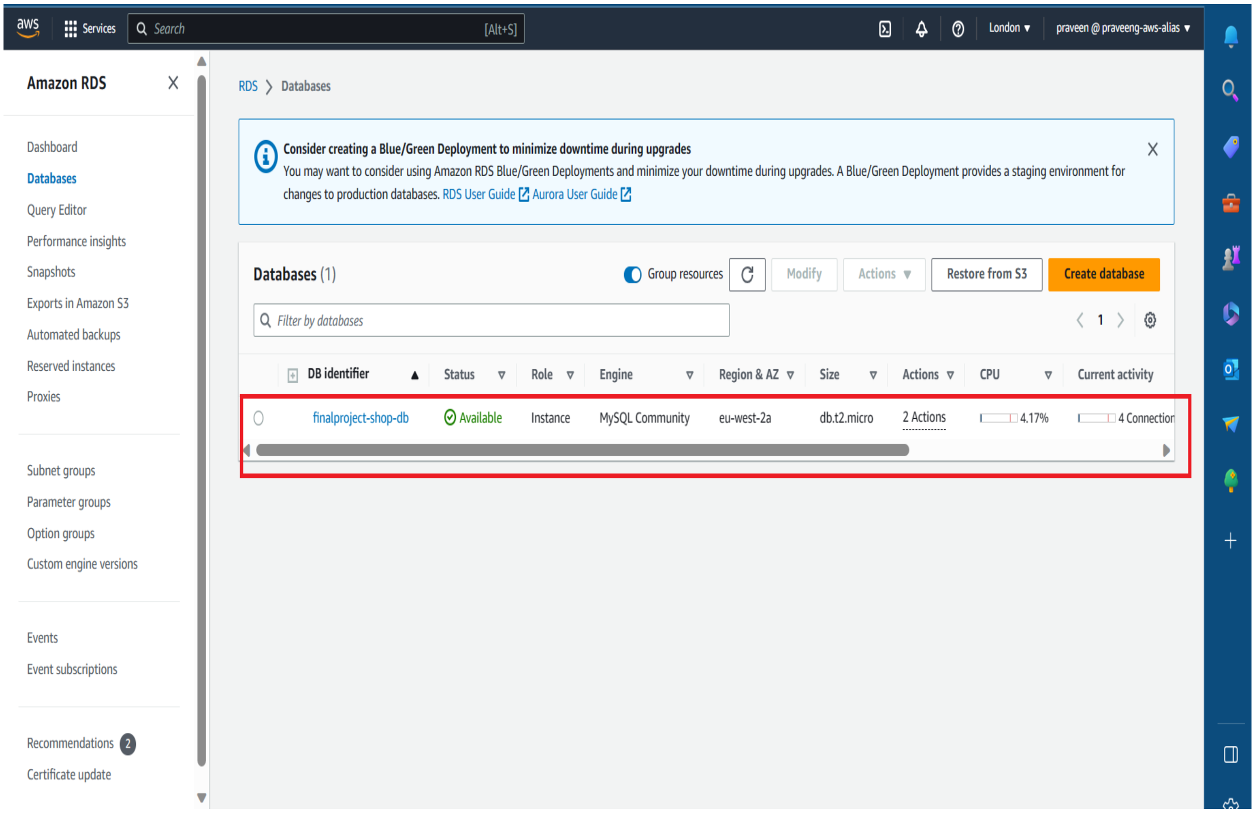
Instance Name : Rds-london

Creating an instance in Amazon Ec2 (Elastic Compute Cloud) allows to provision and manage virtual servers in the cloud.

STEP 2 : RDS DATABASE CREATION

Database Name : finalproject-shop-db

Created An Database Using Amazon RDS



Consider creating a Blue/Green Deployment to minimize downtime during upgrades

You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. RDS User Guide Aurora User Guide

	DB identifier ▲	Status	Role	Engine	Region & AZ	Size	Actions	CPU	Current activity
○	finalproject-shop-db	Available	Instance	MySQL Community	eu-west-2a	db.t2.micro	2 Actions	4.17%	4 Connection

STEP 3 : CONNECTING THE DATABASE IN MYSQL WORKBENCH USING END-POINT AND CREDENTIALS

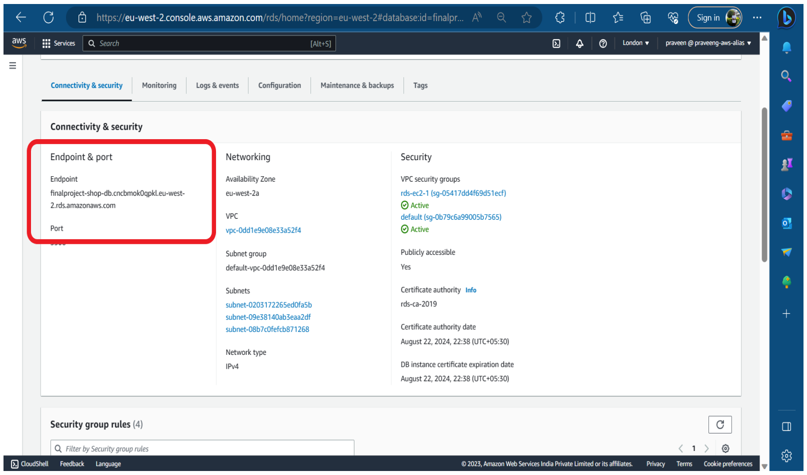
TABLES :

1. CUSTOMER - CustomerID,FirstName,LastName,Address

2. ORDER - OrderID,OrderDate,CustomerID

3. ORDER DETAILS - OrderItemID,OrderID,ProductID,Quantity

4. PRODUCT - ProductID,ProductName,Price



https://eu-west-2.console.aws.amazon.com/rds/home?region=eu-west-2#database:id=finalpr...

Connectivity & security

Endpoint & port

Endpoint
finalproject-shop-db.cncbmok0qpkl.eu-west-2.rds.amazonaws.com

Port

Networking

Availability Zone
eu-west-2a

VPC
vpc-0dd1e9e08e33a52f4

Subnet group
default-vpc-0dd1e9e08e33a52f4

Subnets
subnet-0203172265ed0fa5b
subnet-09e38140ab3eaa2df
subnet-08b7c0fefcb871268

Network type
IPv4

Security

VPC security groups
rds-ec2-1 (sg-05417dd4f69d51ecf)
Active
default (sg-0b79c6a99005b7565)
Active

Publicly accessible
Yes

Certificate authority Info
rds-ca-2019

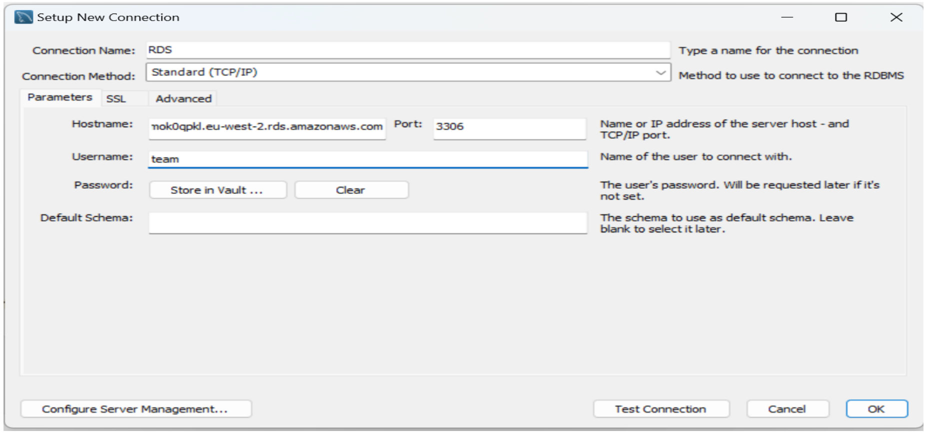
Certificate authority date
August 22, 2024, 22:38 (UTC+05:30)

DB instance certificate expiration date
August 22, 2024, 22:38 (UTC+05:30)

Security group rules (4)

Filter by Security group rules

© 2023, Amazon Web Services India Private Limited or its affiliates.



Setup New Connection

Connection Name: RDS — Type a name for the connection

Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

Hostname: nok0qpkl.eu-west-2.rds.amazonaws.com Port: 3306 — Name or IP address of the server host - and TCP/IP port.

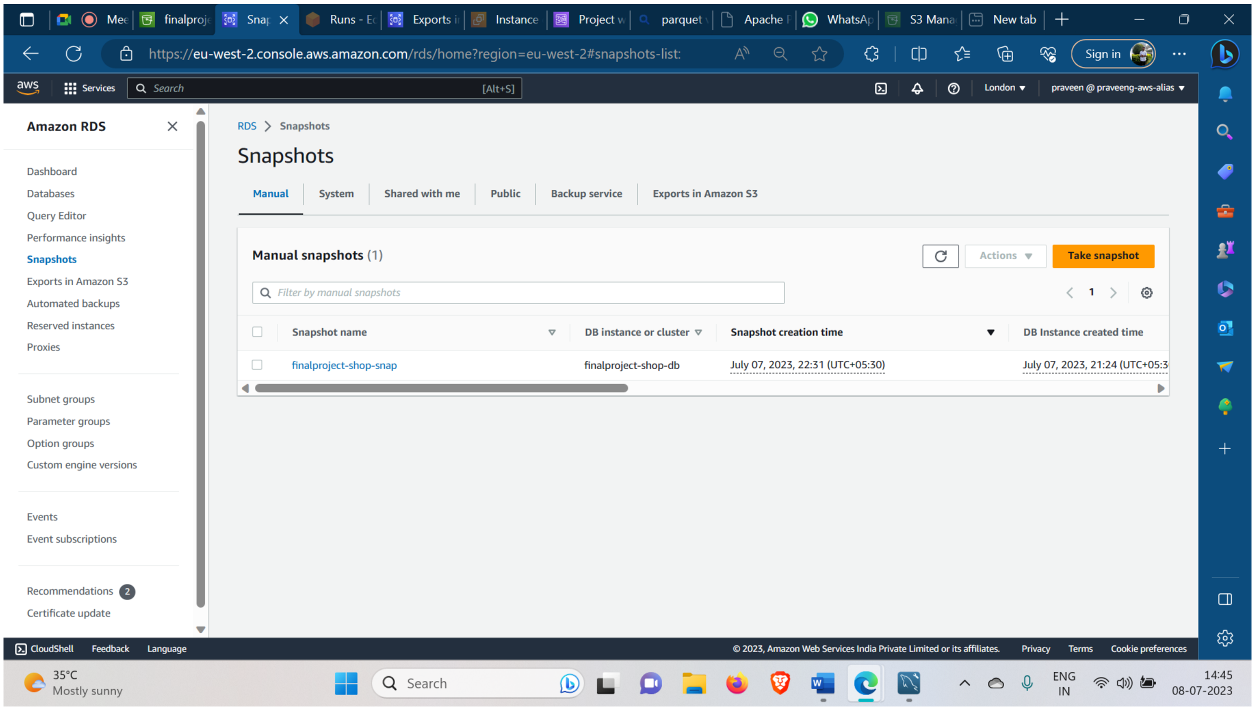
Username: team — Name of the user to connect with.

Password: — The user's password. Will be requested later if it's not set.

Default Schema: — The schema to use as default schema. Leave blank to select it later.

STEP 4 : CREATED A SNAPSHOT USING RDS

NAME : finalproject-shop-snap



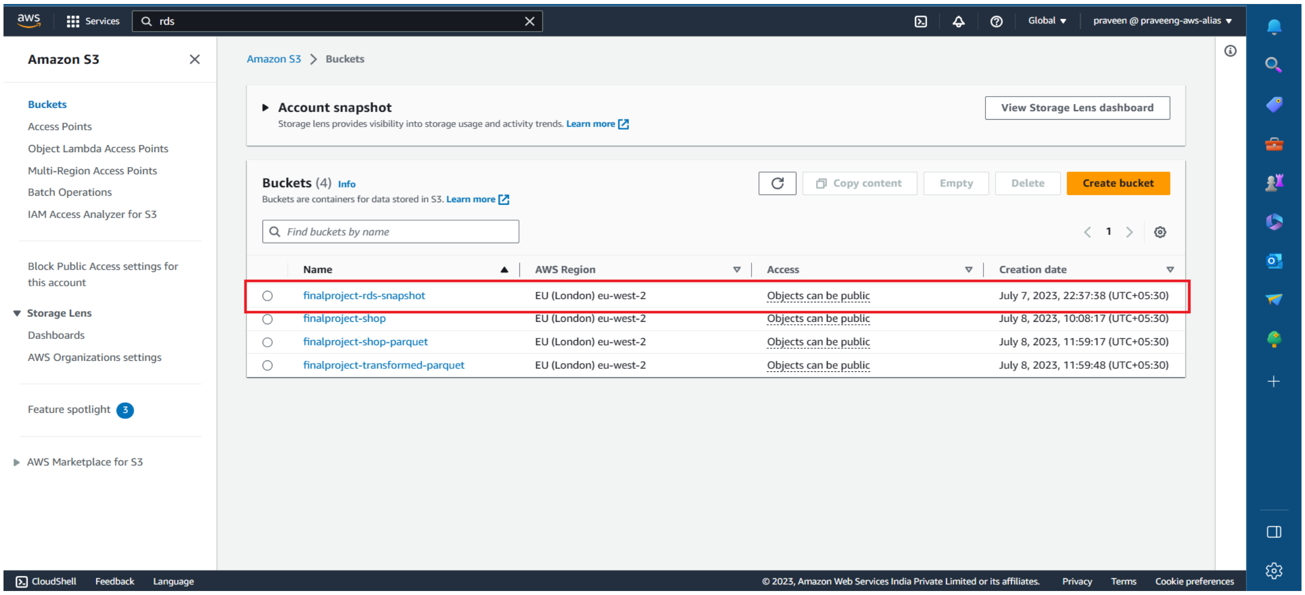
Snapshots

Manual snapshots (1)

Snapshot name	DB instance or cluster	Snapshot creation time	DB Instance created time
finalproject-shop-snap	finalproject-shop-db	July 07, 2023, 22:31 (UTC+05:30)	July 07, 2023, 21:24 (UTC+05:3

STEP 5 : EXPORTED RDS SNAPSHOT TO S3 BUCKET IN PARQUET FORMAT USING IAM ROLE

Target Name : finalproject-rds-snapshot


rds
Global
praveen @ praveeng-aws-alias
Amazon S3
Buckets
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
AWS Organizations settings
Feature spotlight 3
AWS Marketplace for S3
Amazon S3 > Buckets
View Storage Lens dashboard
Copy content
Empty
Delete
Create bucket
Find buckets by name
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences

Account snapshot

Storage lens provides visibility into storage usage and activity trends. Learn more

Buckets (4) Info

Buckets are containers for data stored in S3. Learn more

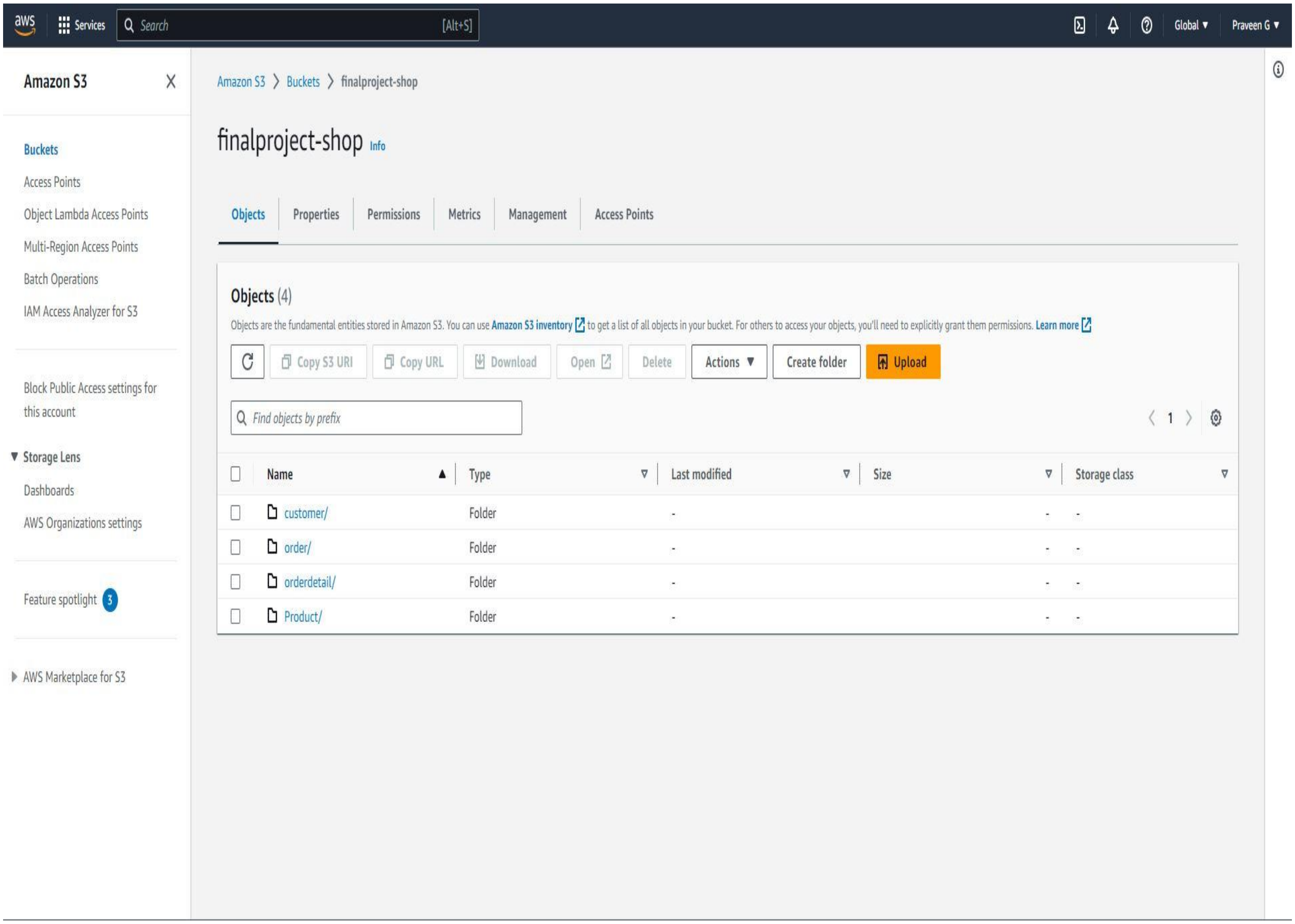
Name	AWS Region	Access	Creation date
finalproject-rds-snapshot	EU (London) eu-west-2	Objects can be public	July 7, 2023, 22:37:38 (UTC+05:30)
finalproject-shop	EU (London) eu-west-2	Objects can be public	July 8, 2023, 10:08:17 (UTC+05:30)
finalproject-shop-parquet	EU (London) eu-west-2	Objects can be public	July 8, 2023, 11:59:17 (UTC+05:30)
finalproject-transformed-parquet	EU (London) eu-west-2	Objects can be public	July 8, 2023, 11:59:48 (UTC+05:30)

STEP 6 : **Uploading the database in S3 - CSV format**

Target Name : finalproject-shop-csv

FOLDER :

1. Customer-csv
2. Orders-csv
3. Orderdetail-csv
4. Product-csv



Amazon S3 > Buckets > finalproject-shop

finalproject-shop Info

Objects (4)

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
□	customer/	Folder	-	-	-
□	order/	Folder	-	-	-
□	orderdetail/	Folder	-	-	-
□	Product/	Folder	-	-	-

STEP 7 : TRIGGER THE S3 BUCKET USING LAMBDA FUNCTION

Function Name 1 : lambda-stepfunction

Function Name 2 : lambda-stepfunc-transformation

CODE :

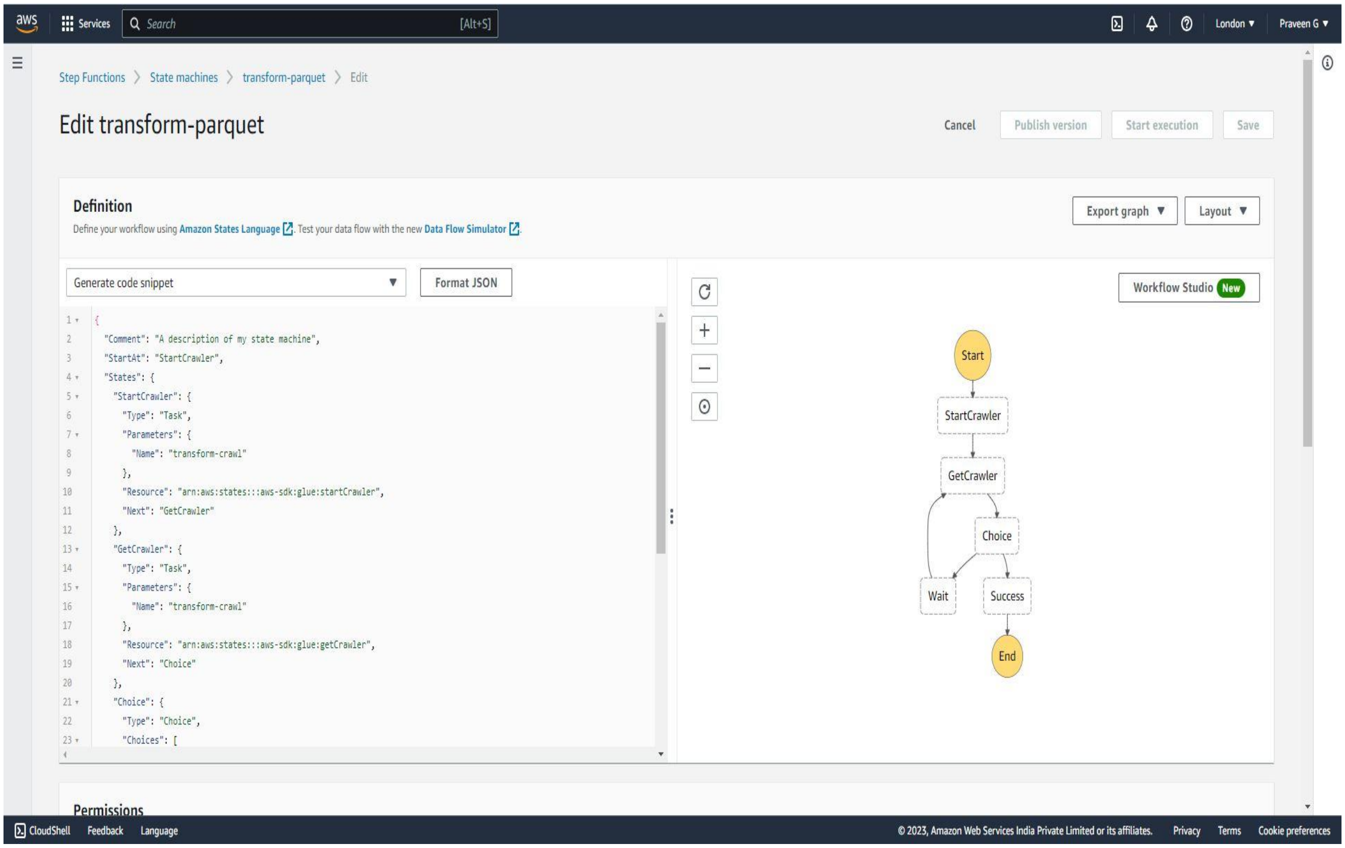
```
import boto3

import time

def lambda_handler(event, context):

    # Sample bucket name

    bucket_name = 'finalproject-shop'

    # Instantiate the AWS Step Functions client

    sf_client = boto3.client('stepfunctions')

   # Generate a unique execution name with a timestamp

    execution_name = f"execution_name_{int(time.time())}"

  # Start the execution of the Step Function state machine

    response = sf_client.start_execution(


stateMachineArn='arn:aws:states:eu-west-2:631707933068:stateMachine:fileformat-csv-parquet',

        name=execution_name,

        input='{"bucket": "' + bucket_name + '"}'

    )
```

```
# Return the execution ARN as the output of the Lambda function
 return {
     'statusCode': 200,
     'body': f"Step Function execution started: {response['executionArn']}"
 }
```

STEP 8 : AUTOMATING THE STEP FUNCTION USING AWS LAMBDA

State Machine 1 : lambda-stepfunc-transformation

State Machine 2 : transform-parquet

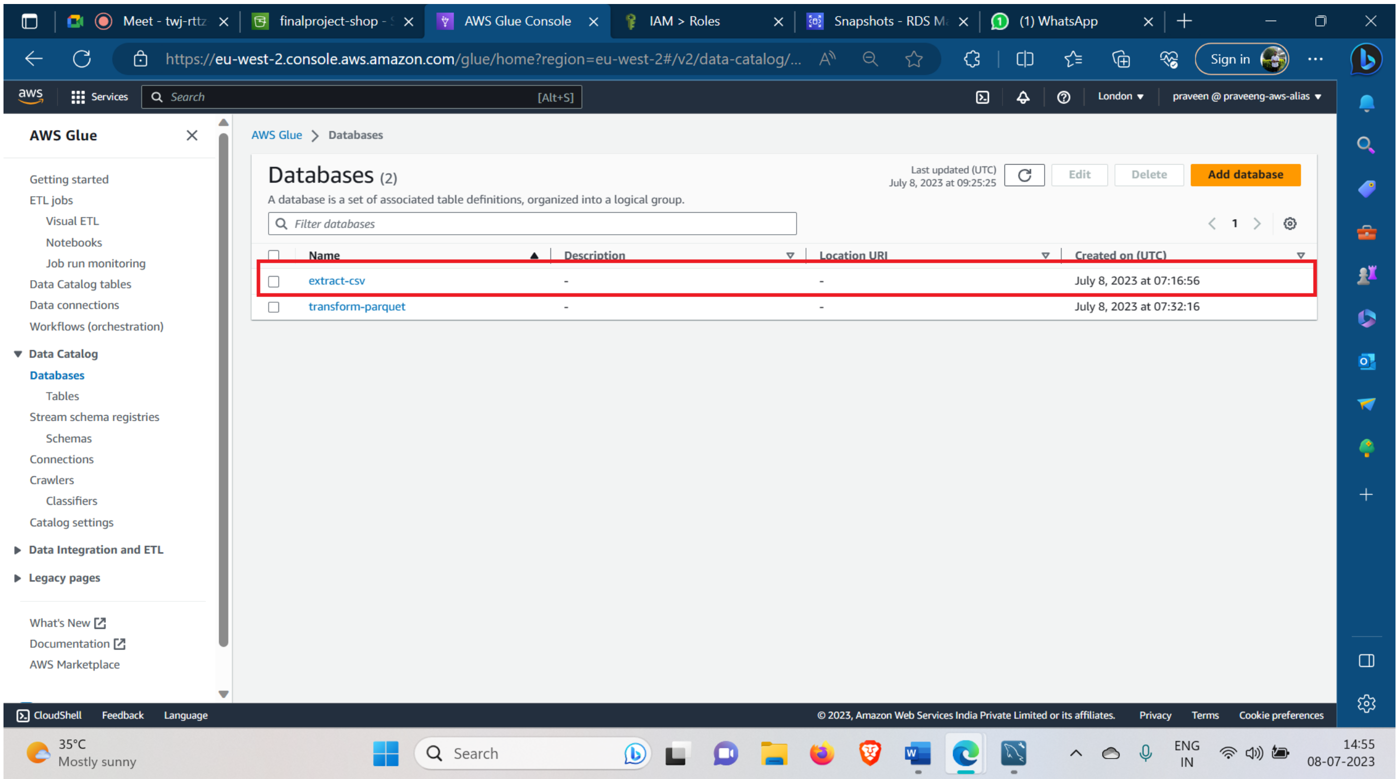


Define your workflow using Amazon States Language. Test your data flow with the new Data Flow Simulator.

```
{
  "Comment": "A description of my state machine",
  "StartAt": "StartCrawler",
  "States": {
    "StartCrawler": {
      "Type": "Task",
      "Parameters": {
        "Name": "transform-crawl"
      },
      "Resource": "arn:aws:states:::aws-sdk:glue:startCrawler",
      "Next": "GetCrawler"
    },
    "GetCrawler": {
      "Type": "Task",
      "Parameters": {
        "Name": "transform-crawl"
      },
      "Resource": "arn:aws:states:::aws-sdk:glue:getCrawler",
      "Next": "Choice"
    },
    "Choice": {
      "Type": "Choice",
      "Choices": [
```

STEP 9 : CRAWL THE DATA FROM S3 AND STORED IN DATA CATALOG

Name : Extract-csv



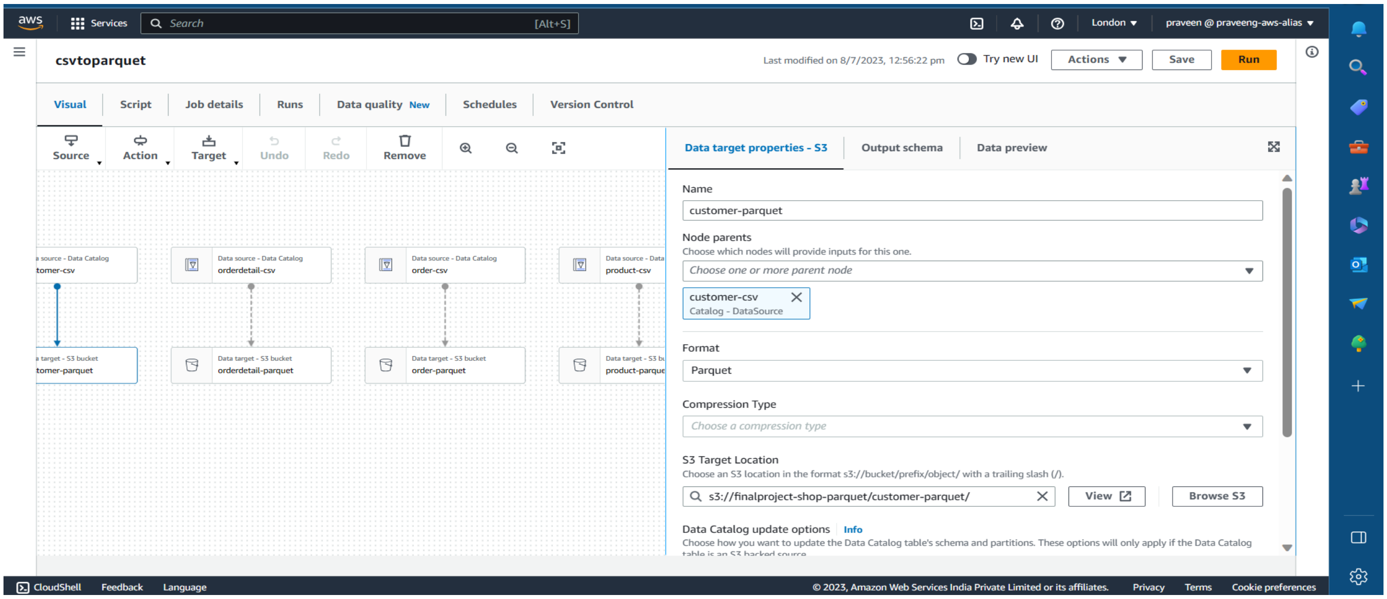
Databases (2)

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

Last updated (UTC)
July 8, 2023 at 09:25:25

Name	Description	Location URI	Created on (UTC)
extract-csv	-	-	July 8, 2023 at 07:16:56
transform-parquet	-	-	July 8, 2023 at 07:32:16

STEP 10 : CHANGING THE FILE FORMAT OF CSV TO PARQUET USING AWS GLUE AND UPLOADING TO S3



csvtoparquet

Last modified on 8/7/2023, 12:56:22 pm

Name
customer-parquet

Node parents
Choose which nodes will provide inputs for this one.
Choose one or more parent node
customer-csv
Catalog - DataSource

Format
Parquet

Compression Type
Choose a compression type

S3 Target Location
Choose an S3 location in the format s3://bucket/prefix/object/ with a trailing slash (/).
s3://finalproject-shop-parquet/customer-parquet/

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 backed source

STEP 11 : AWS GLUE DATA CATALOG CREATION

Database Name : transform-parquet

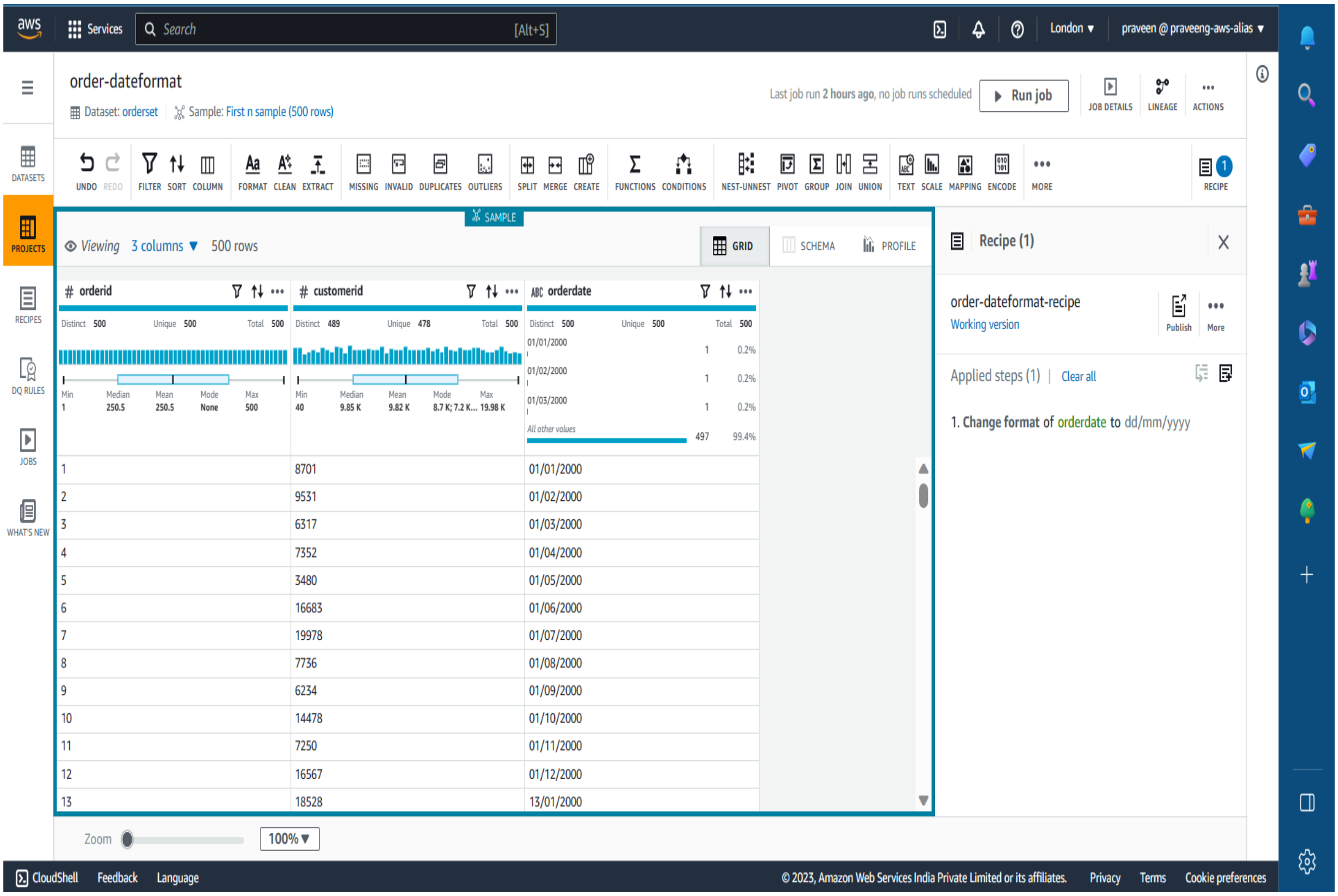
STEP 12 : CRAWL THE DATA FROM S3 AND STORED IN DATA CATALOG

Crawl Name : transformation-crawl

STEP 13 : CHANGING THE DATE FORMAT USING DATABREW-SOURCE DATA CATALOG And UPDATING IN SAME DATA CATALOG

Job : date-format-change

Step name : Change format-of-orderdate-to dd/mm/yyyy



order-dateformat

Dataset: orderset   Sample: First n sample (500 rows)

Last job run 2 hours ago, no job runs scheduled

Viewing 3 columns ▾ 500 rows

# orderid	# customerid	ABC orderdate
Distinct 500 Unique 500 Total 500	Distinct 489 Unique 478 Total 500	Distinct 500 Unique 500 Total 500
Min 1 Median 250.5 Mean 250.5 Mode None Max 500	Min 40 Median 9.85 K Mean 9.82 K Mode 8.7 K; 7.2 K... Max 19.98 K	01/01/2000 1 0.2% 01/02/2000 1 0.2% 01/03/2000 1 0.2% All other values 497 99.4%
1	8701	01/01/2000
2	9531	01/02/2000
3	6317	01/03/2000
4	7352	01/04/2000
5	3480	01/05/2000
6	16683	01/06/2000
7	19978	01/07/2000
8	7736	01/08/2000
9	6234	01/09/2000
10	14478	01/10/2000
11	7250	01/11/2000
12	16567	01/12/2000
13	18528	13/01/2000

Recipe (1)

order-dateformat-recipe

Working version

Applied steps (1) | Clear all

1. Change format of orderdate to dd/mm/yyyy

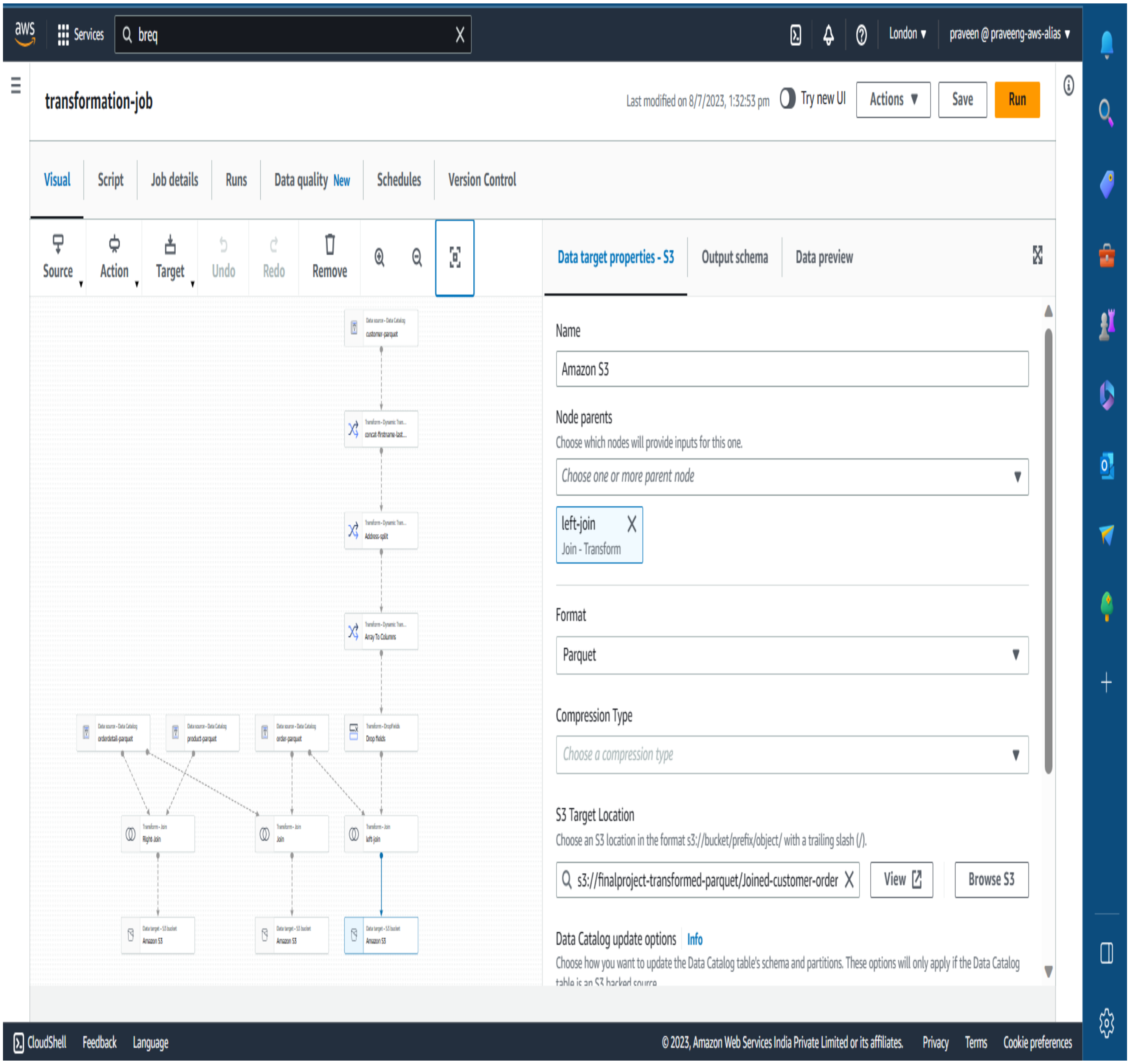
© 2023, Amazon Web Services India Private Limited or its affiliates.

STEP 14 : TRANSFORMATION USING AWS GLUE -VISUALLY STORING IN S3

Job Name : transformation-job

Transformation : Date Format Change ,Split ,Drop ,Concat ,Join

Target Bucket : finalproject-transformed-parquet



Name

Amazon S3

Node parents

Choose which nodes will provide inputs for this one.

Choose one or more parent node ▾

left-join X
Join - Transform

Format

Parquet ▾

Compression Type

Choose a compression type ▾

S3 Target Location

Choose an S3 location in the format s3://bucket/prefix/object/ with a trailing slash (/).

s3://finalproject-transformed-parquet/Joined-customer-order X

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 backed source

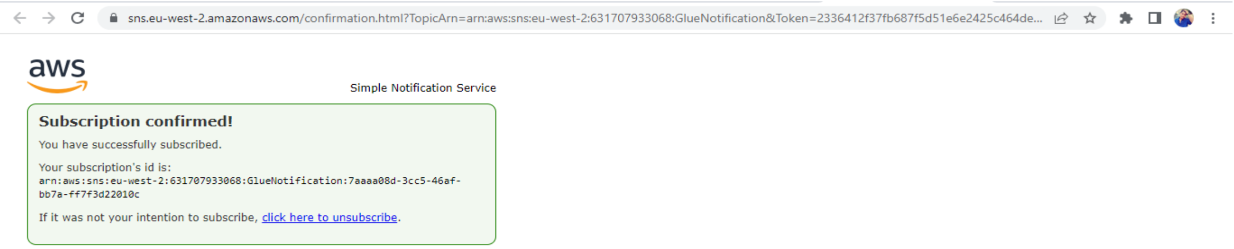
STEP 15 : SNS CREATION (SIMPLE NOTIFICATION SERVICE)

Created a topic named as SNS Glue Notification and added a EMAIL subscription of our personal mail's.So whenever SNS gets triggered it will send a notification.

Topic Name : Glue Notification

Subscription : dhamotharan596@gmail.com

praveen21052002@gmail.com

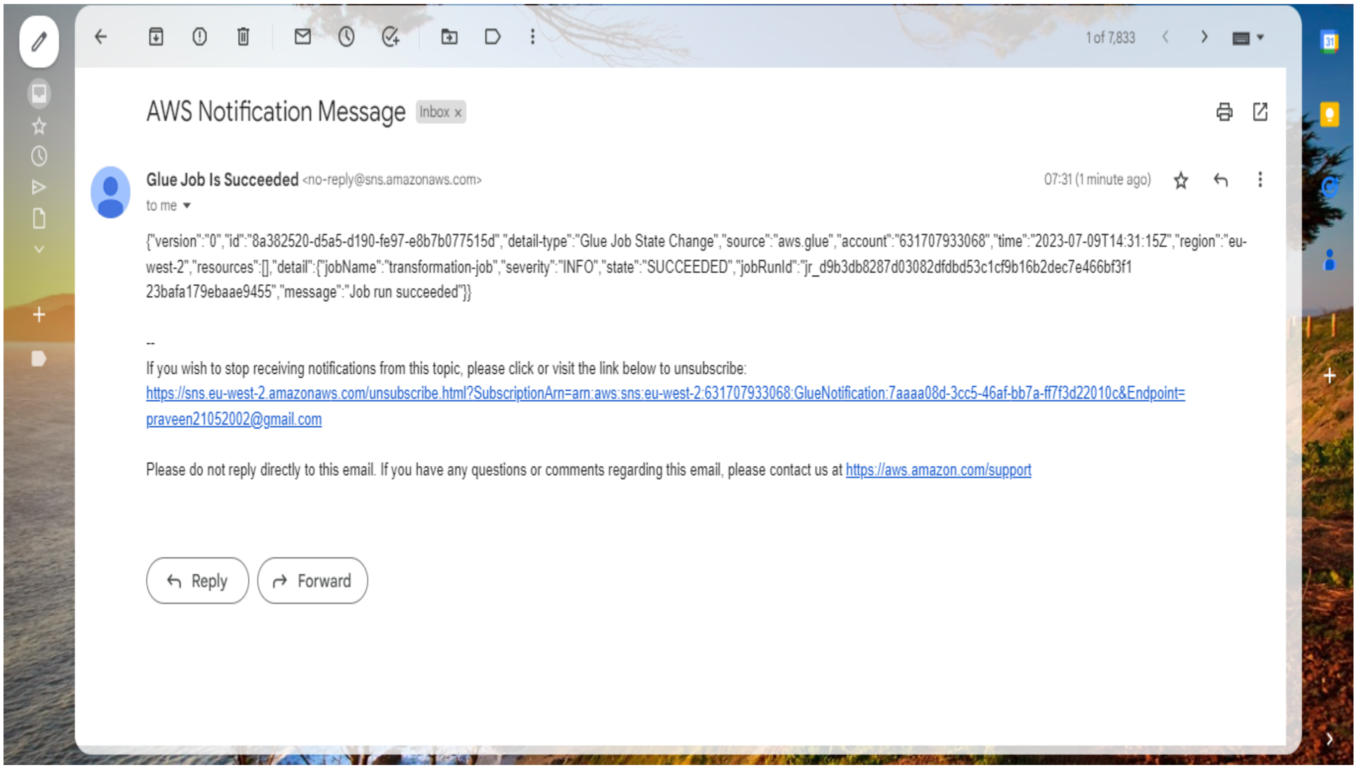

sns.eu-west-2.amazonaws.com/confirmation.html?TopicArn=arn:aws:sns:eu-west-2:631707933068:GlueNotification&Token=2336412f37fb687f5d51e6e2425c464de...
aws
Simple Notification Service

Subscription confirmed!

You have successfully subscribed.

Your subscription's id is:
arn:aws:sns:eu-west-2:631707933068:GlueNotification:7aaaa08d-3cc5-46af-bb7a-ff7f3d22010c

If it was not your intention to subscribe, click here to unsubscribe.



AWS Notification Message

Glue Job Is Succeeded <no-reply@sns.amazonaws.com>
to me

07:31 (1 minute ago)

```
{"version":"0","id":"8a382520-d5a5-d190-fe97-e8b7b077515d","detail-type":"Glue Job State Change","source":"aws.glue","account":"631707933068","time":"2023-07-09T14:31:15Z","region":"eu-west-2","resources":[],"detail":{"jobName":"transformation-job","severity":"INFO","state":"SUCCEEDED","jobRunId":"jr_d9b3db8287d03082dfdbd53c1cf9b16b2dec7e466bf3f1
23bafa179ebaae9455","message":"Job run succeeded"}}
```

--

If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:
https://sns.eu-west-2.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:eu-west-2:631707933068:GlueNotification:7aaaa08d-3cc5-46af-bb7a-ff7f3d22010c&Endpoint=praveen21052002@gmail.com

Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at https://aws.amazon.com/support

STEP 16 : EVENT BRIDGE RULE CREATION

Rule Name : gluesuccess-event

In the event pattern,we chose the source service as glue and wrote the code for event-type gluejob-statechange. If gluejob-statechange succeeds, we set the target to trigger the SNS topic. So it will send a notification to EMAIL as "GLUE JOB IS SUCCEEDED.

JSON Code :-

```
{
  "source": ["aws.glue"],
  "detail-type": ["Glue Job State Change"],
  "detail": {
    "jobName": ["transformation-job"],
    "state": ["SUCCEEDED"]
  }
}
```

Description:

The provided code snippet is an event pattern designed for monitoring the state change of an AWS Glue job.

Conclusion :-

In conclusion, this project showcases the successful implementation of an Extract, Transform, and Load (ETL) process in the AWS cloud. It leverages a range of AWS services including Amazon RDS, S3, AWS Glue, DataBrew, and EventBridge to achieve efficient data extraction, transformation, and loading capabilities.

By establishing a connection with MySQL and exporting a snapshot of the RDS database to an S3 bucket, data backup and security are ensured. The use of the AWS Glue crawler facilitates the scanning, cataloging, and transformation of CSV files into Parquet format, providing organized and refined data. Furthermore, the integration of EventBridge and Simple Notification Service (SNS) allows for timely email notifications upon job completion, ensuring stakeholders are informed.

Overall, this ETL project in the AWS cloud harnesses the scalability, reliability, and flexibility of AWS services, enabling organizations to efficiently manage and process their data.