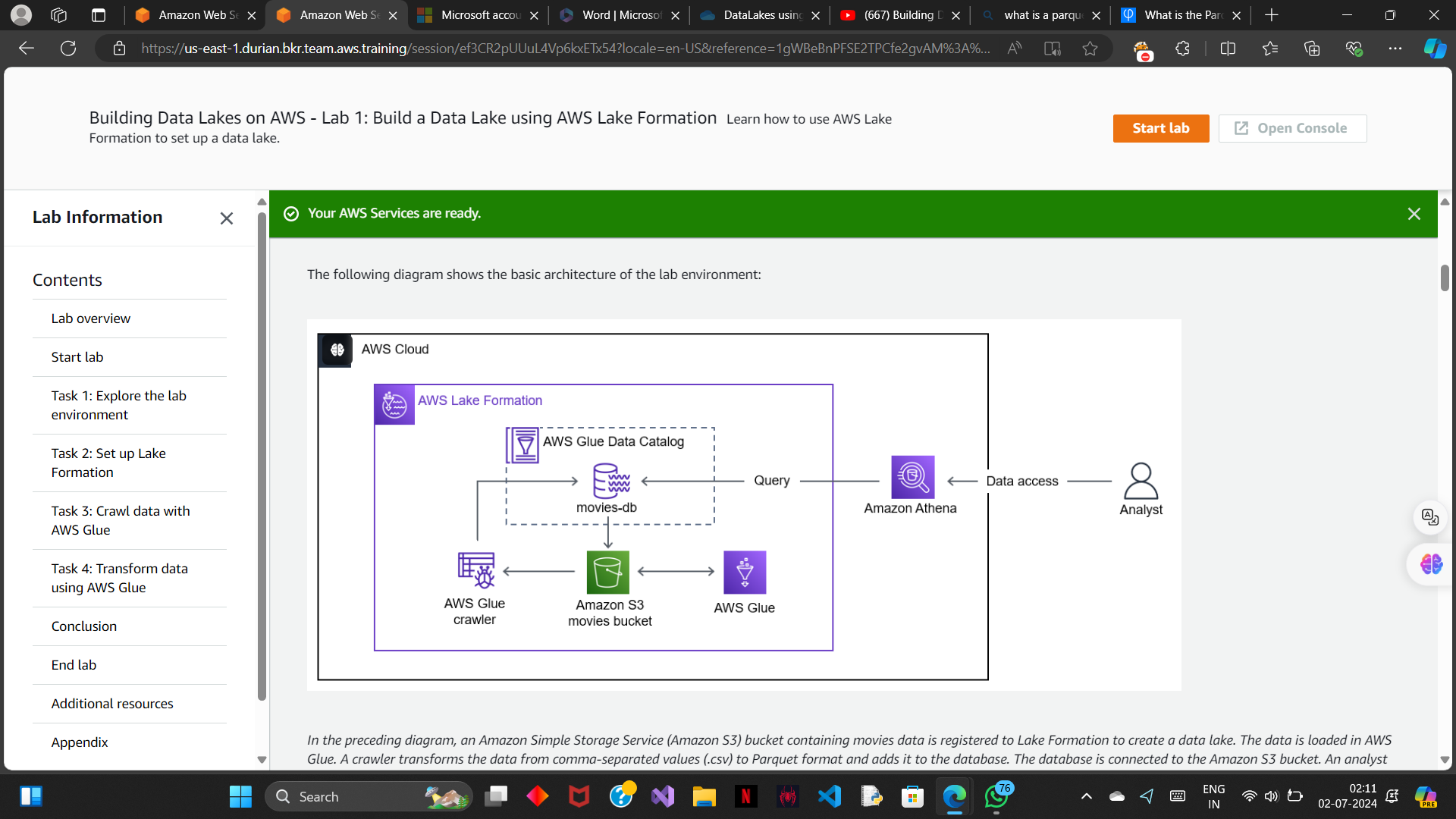
Aim:

In this lab, you work through the basic steps to register storage with Lake Formation and define a crawler to crawl the initial dataset. You then create an AWS Glue job to generate a version of the dataset in a partitioned Apache Parquet format. You also use Amazon Athena to validate the tables, their schema, and the ability to access the underlying data source through the data catalog.



AWS glue:

1> format

2> partition

3> compress

AWS Glue is a serverless data integration and ETL[extract,transform,load] service that helps discover, prepare, move, and integrate data for analytics and machine learning (ML). Administrators can use the new setup tool to grant IAM roles and users access to AWS Glue and their data, as well as a default role for running jobs.

AWS crawler:

1>converts .csv into apache parquet(columnar data)(easy and efficient to compress)

2>Populates aws glue data catalog

Athena :

1> Enables us to query the data from the data lake. It uses the aws data catalog to read the database.

TASK 1(asset discovery):

ASSETS:

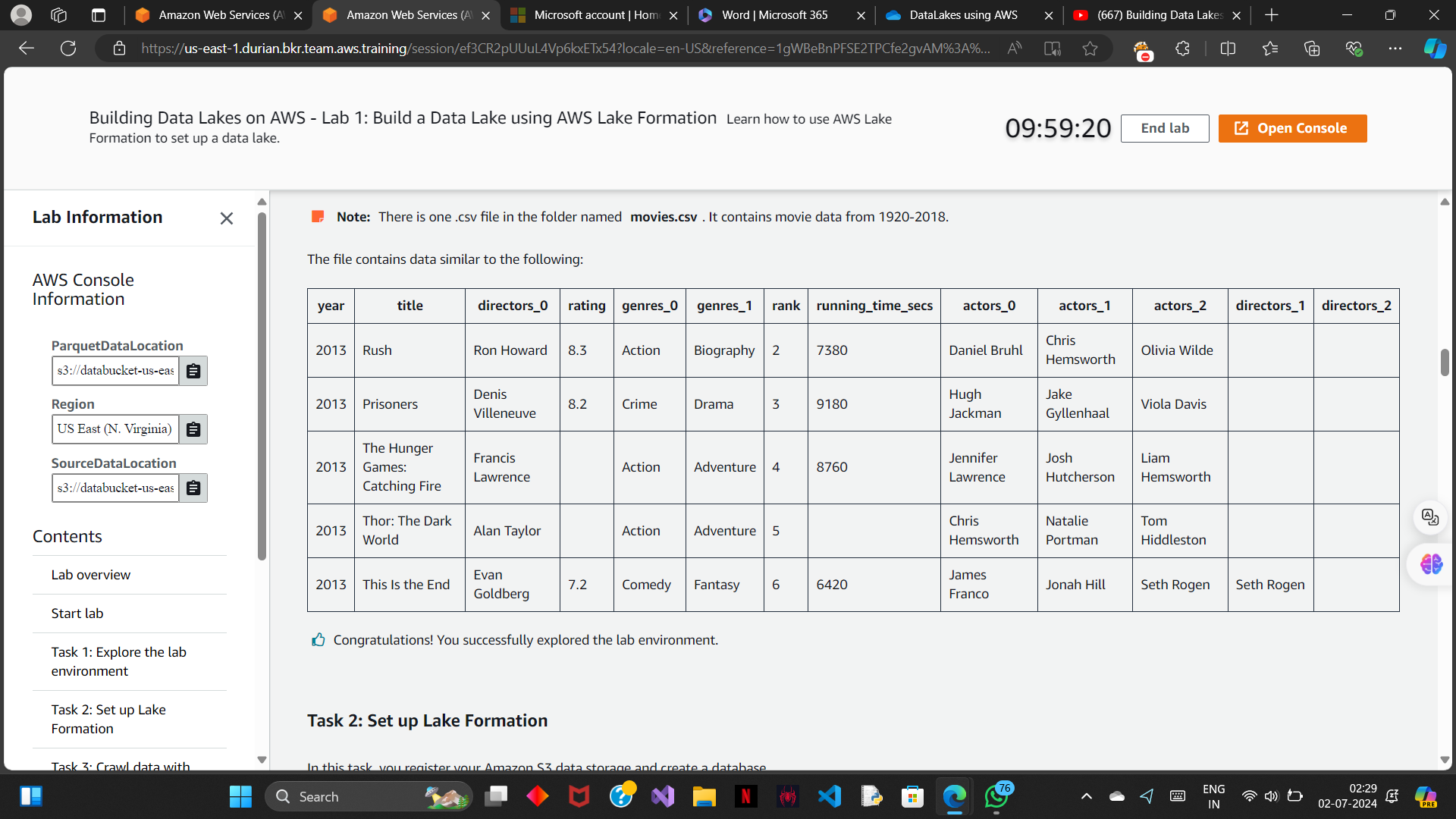
1> parquet data location : s3://databucket-us-east-1-9562181575680315/data/movies\_parquet/

2> source data location :

s3://databucket-us-east-1-9562181575680315/data/

Note-> The **data/** folder contains your dataset. The **results/** folder stores the results of your Athena queries. You will specify the **results/** folder as the query results location in Athena later in this lab.

3> There is one .csv file in the folder named **movies.csv**. It contains movie data from 1920- 2018.



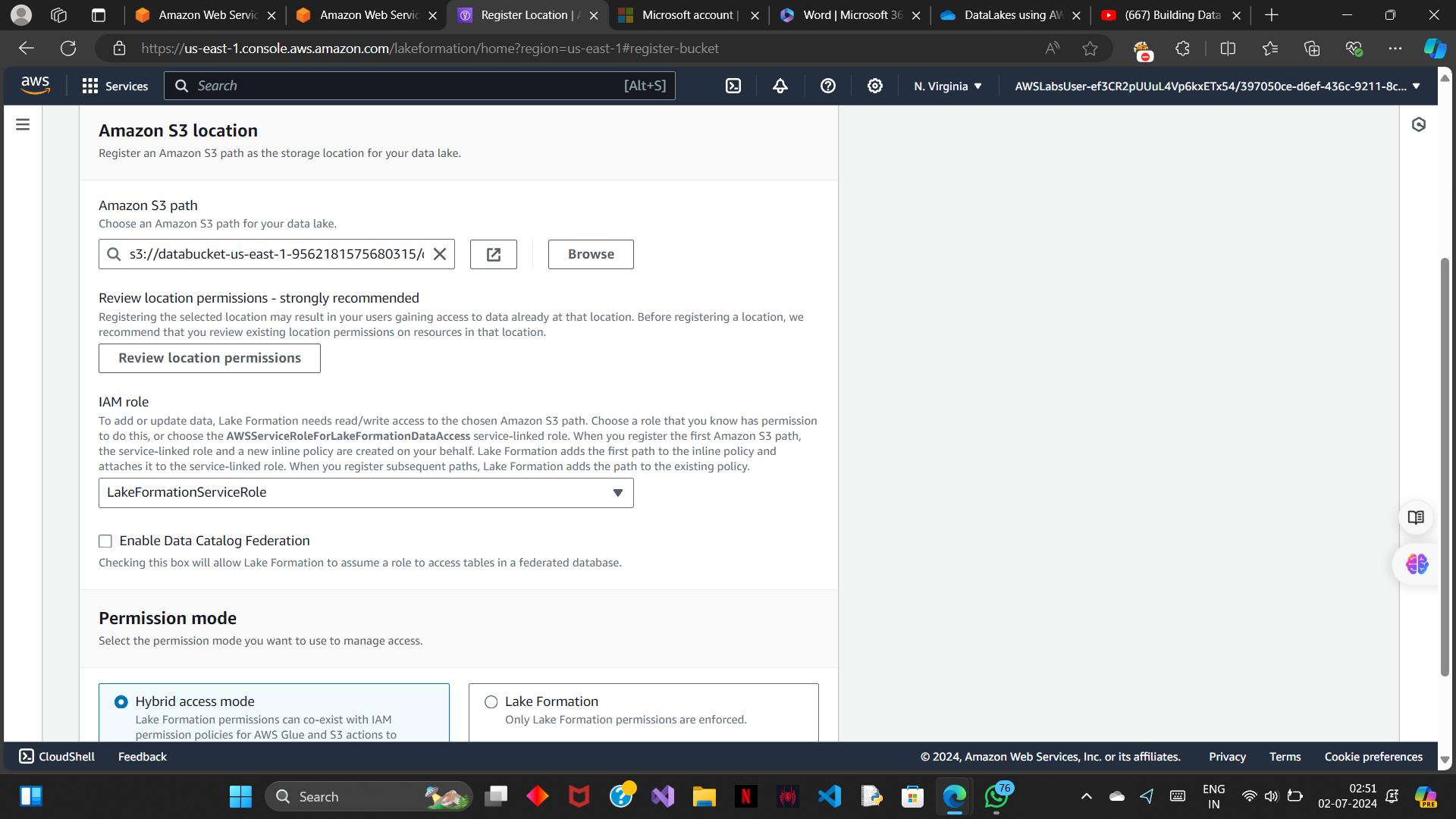
TASK 2(set up lake formation):

TASK 2.1 [Register your amazon s3 storage]:

Lake Formation manages access to designated storage locations in Amazon S3. Register the storage locations you want to be part of the data lake.

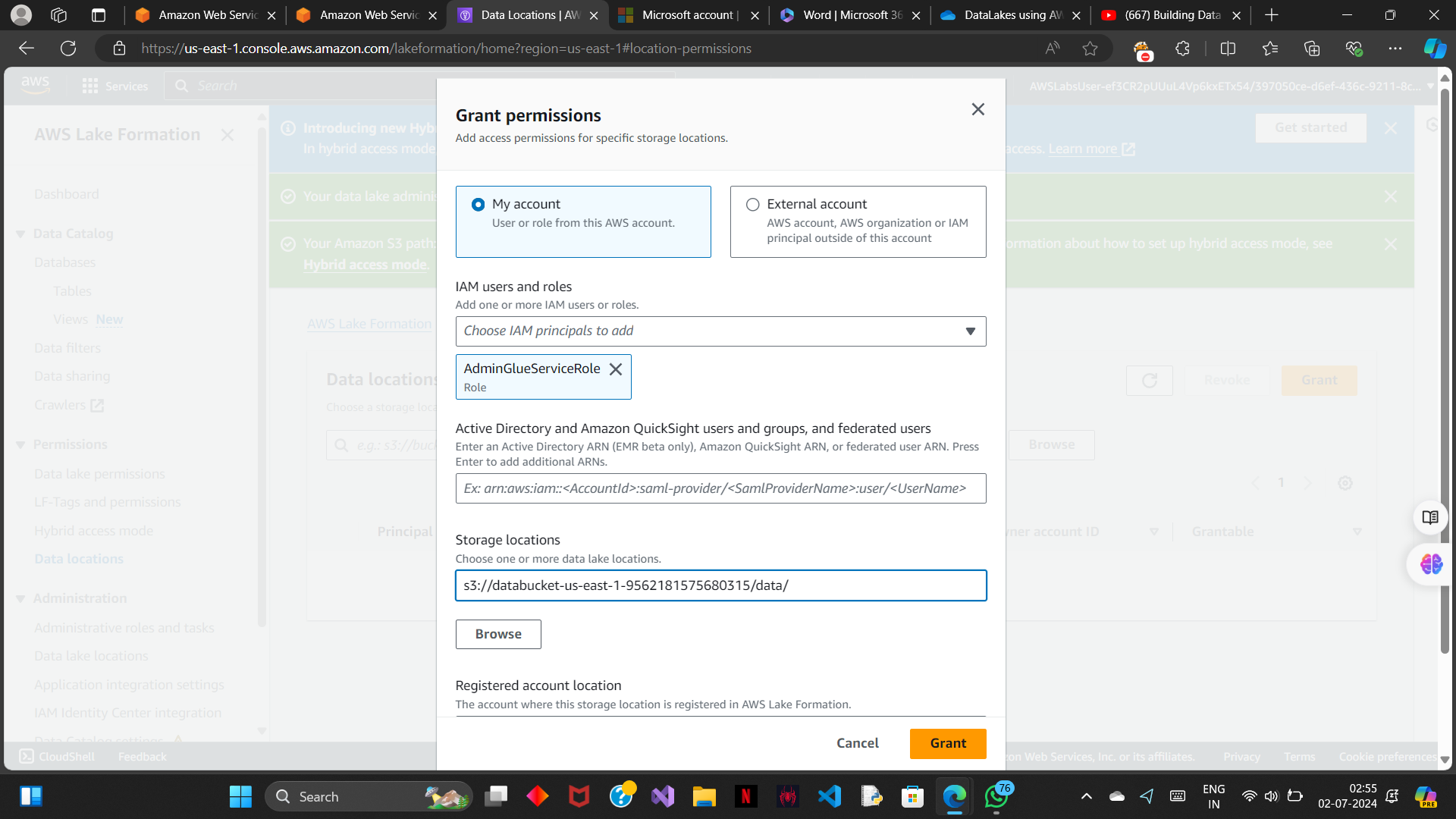
1> search and choose AWS lake formation. (make sure add myself is selected and then click on get started).

2> In the left navigation pane, in the **Administration** section, choose **Data lake locations**



**Note->** The **LakeFormationService** role was created at the start of the lab and includes the **s3:PutObject**, **s3:GetObject**, **s3:DeleteObject**, and **s3:ListBucket** actions for your **DataBucket.**

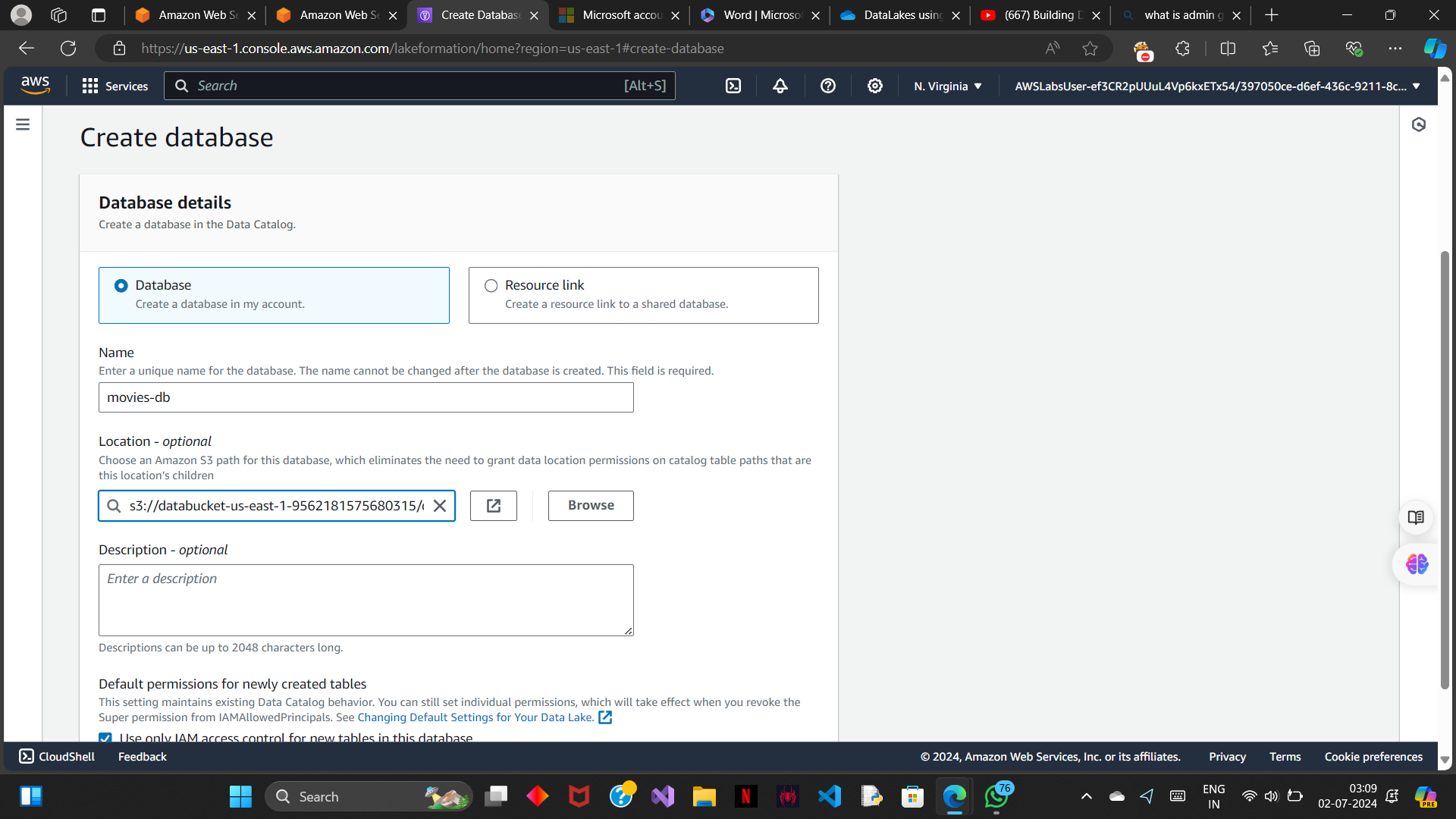
3> grant permissions (choose admin glue service role)



TASK 2.2 [Create a database]:

Lake Formation uses the AWS Glue Data Catalog to store metadata about data lakes, data sources, transforms, and targets. Metadata about data sources and targets is in the form of databases and tables. Tables store information about the underlying data, including schema information, partition information, and data location. Databases are collections of tables.

1>choose databases under the data catalogue section and create a database.



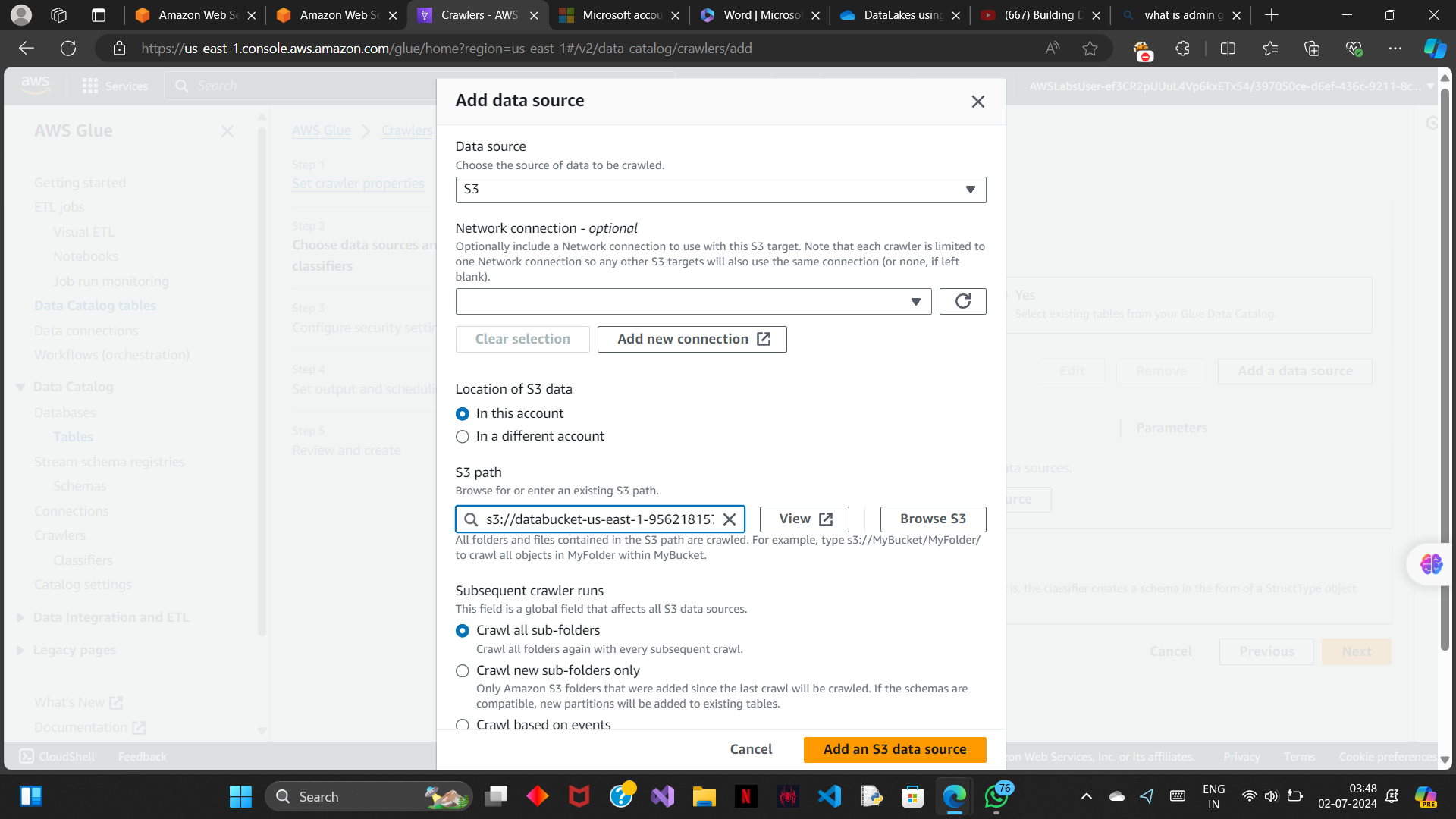
TASK 3 (Crawl data with AWS glue):

TASK 3.1 (create a crawler):

A crawler connects to a data store and progresses through a prioritized list of classifiers to determine the schema for your data. It then creates metadata tables in your data catalog.The crawler reads data at the source location and creates tables in the AWS Glue Data Catalog. A table is the metadata definition that represents your data, including its schema. The tables in the AWS Glue Data Catalog do not contain data. Instead, you use these tables as a source or target in an AWS Glue job definition.

1>add a source data location for crawler to crawl.

AWS Glue --> Tables --> Add table with crawler



2> give admin glue service role permission.

3>choose movies-db as the target database.

4> run the crawler

TASK 3.2(run a query):

1>go to athena and launch a query editor

The workgroup stores the Athena results in an Amazon S3 bucket. The primary workgroup saves the results to your data bucket in a *results/* folder.

TASK 4 (transform data using AWS glue):

you transform your data in .csv format to Parquet format. After completing the transformation, you run your crawlers to add the converted data to the new table. Finally, you compare the query performance using the csv file and Parquet data sources.

TASK 4.1 (create an AWS gluejob to transform data):

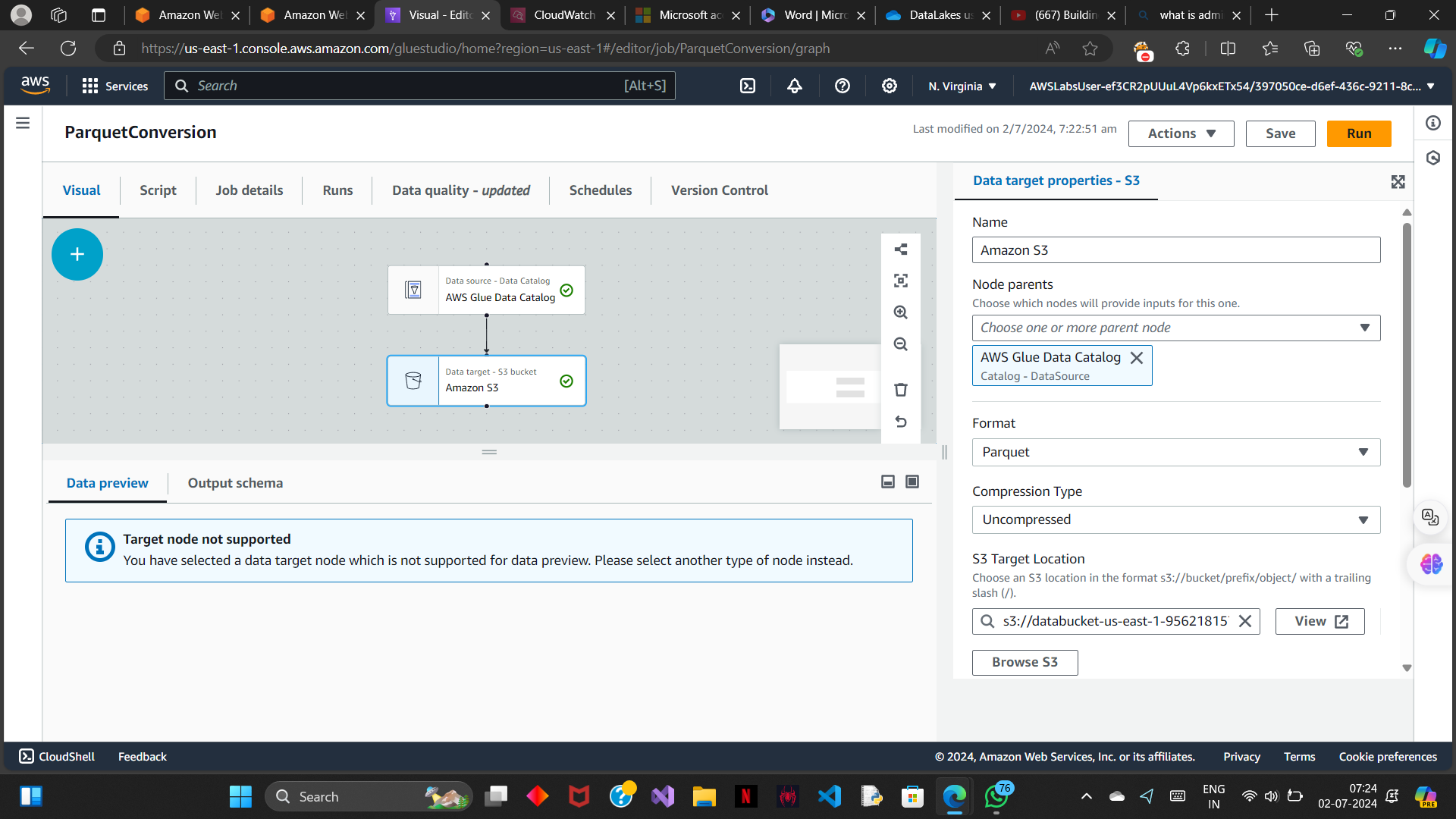
1> under the data integration ETL choose ETL jobs.

2>now , create a source using the AWS glue data catalogue.

3>create the target bucket node.

4> update the aws glue job name,role,and script name.

Note -> There are currently 10 workers using the worker type G.1X. For a simple conversion AWS Glue job, 10 workers are more than enough. When setting up an AWS Glue job, you can customize the number of workers and the worker type to fit your use case.

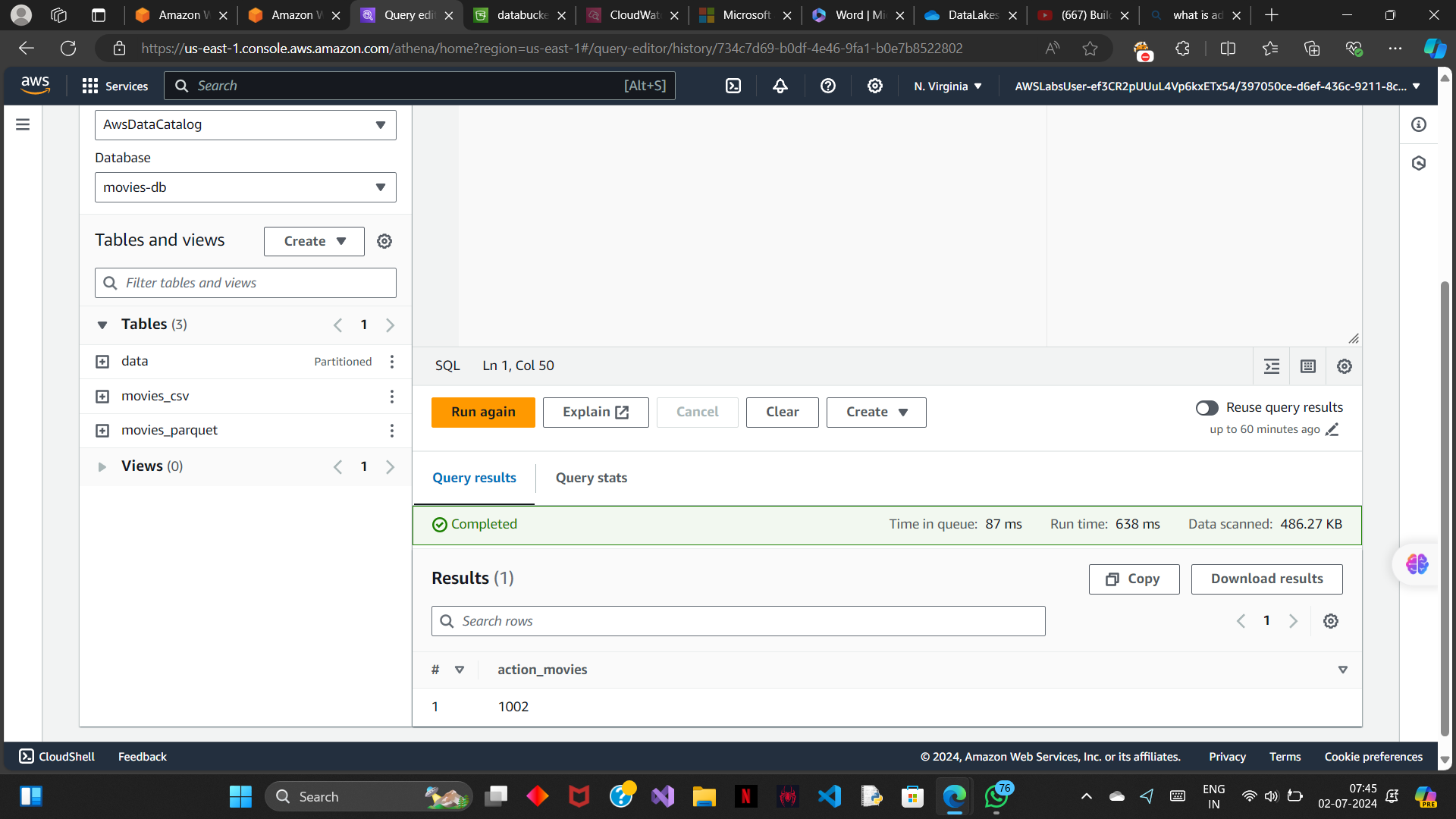


TASK 4.2(run crawlers to add parquet data in your table):

In this task you run your existing crawler to add the table for the newly transformed data.

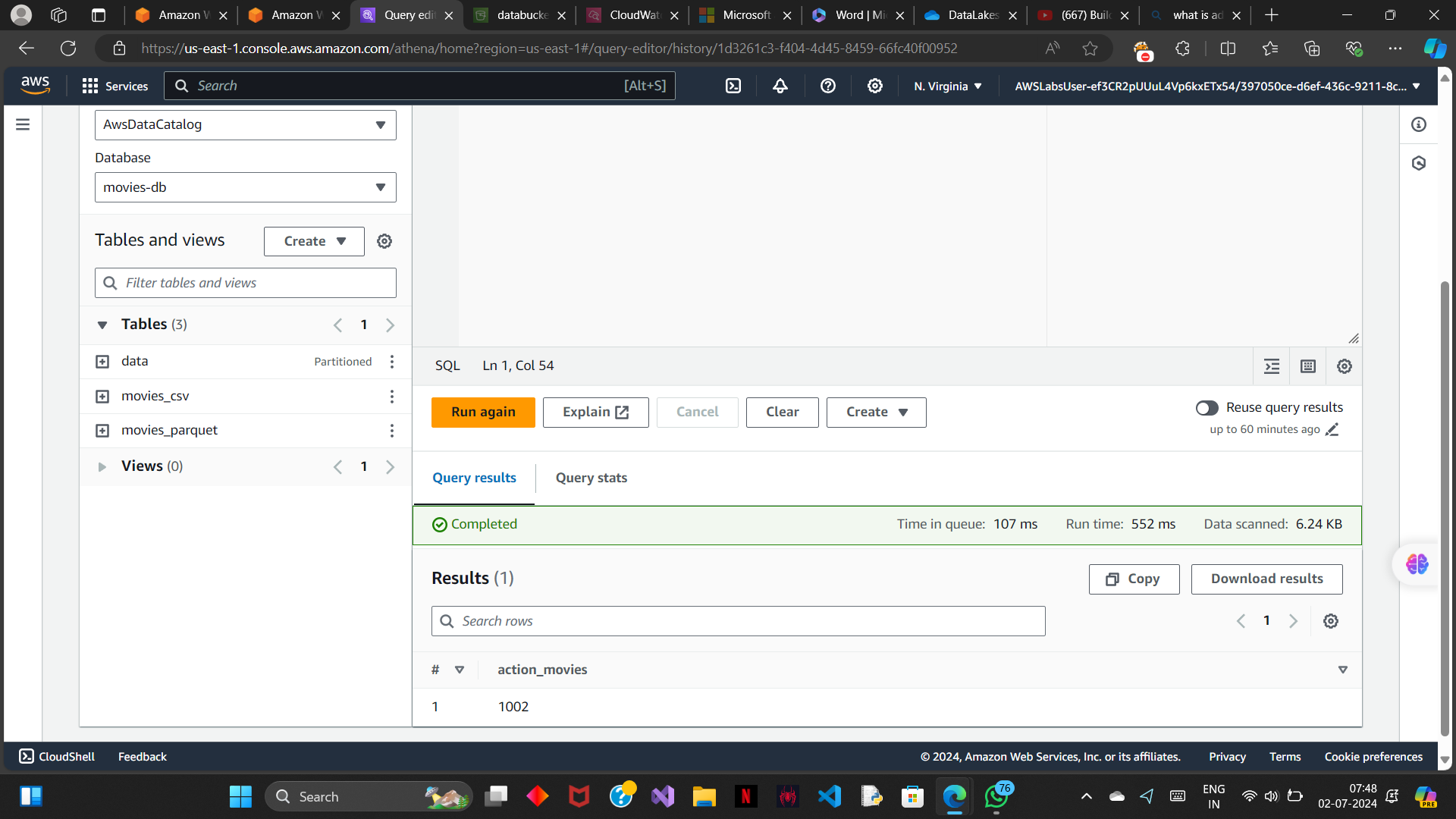
1> run the movies table crawler.

After adding the data compare the runtimes by quering both movies\_csv and movies\_parquet.



The above ss is for movies\_csv .

For movies\_parquet :



We can see that the movies parquet is more compressed and the time required to query it is lesser than the movies\_csv table.

Key takeaways :

1> learned to create a data lake.

2> crawling and populating the data catalogue using AWS glue

3>learned to query the data catalogue via athena.

4> to transform the data from .csv to parquet.