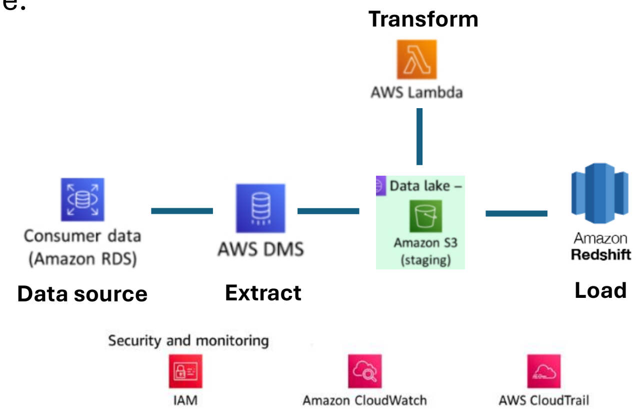
Data Pipelines Projects on AWS

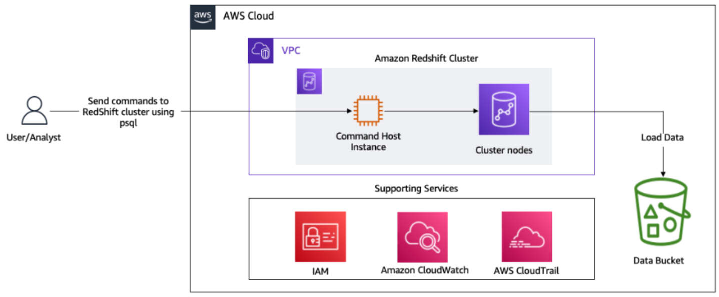
1. ETL pipeline:



Steps:

1. Created an RDS instance with MySQL, set up tables, and inserted records using Sqlectron.
2. Set up AWS DMS with a replication instance and configured role to manage network resources in my default VPC.
3. Configured RDS as the source endpoint.
4. Created an S3 bucket named `datapipeline-staging-datalake`.
5. Created an IAM role (`dms-s3-access-role`) for DMS to interact with the S3 bucket.
6. Defined source and target endpoints in DMS.
7. Created a database migration task to replicate data, resulting in a CSV file in the S3 bucket.
8. Transformed the CSV file using AWS Lambda to add column names and remove duplicates.
9. Created a Redshift cluster (`orders-cluster`) and loaded data from S3 using the COPY command.
10. Executed SQL queries using Redshift Query Editor v2.

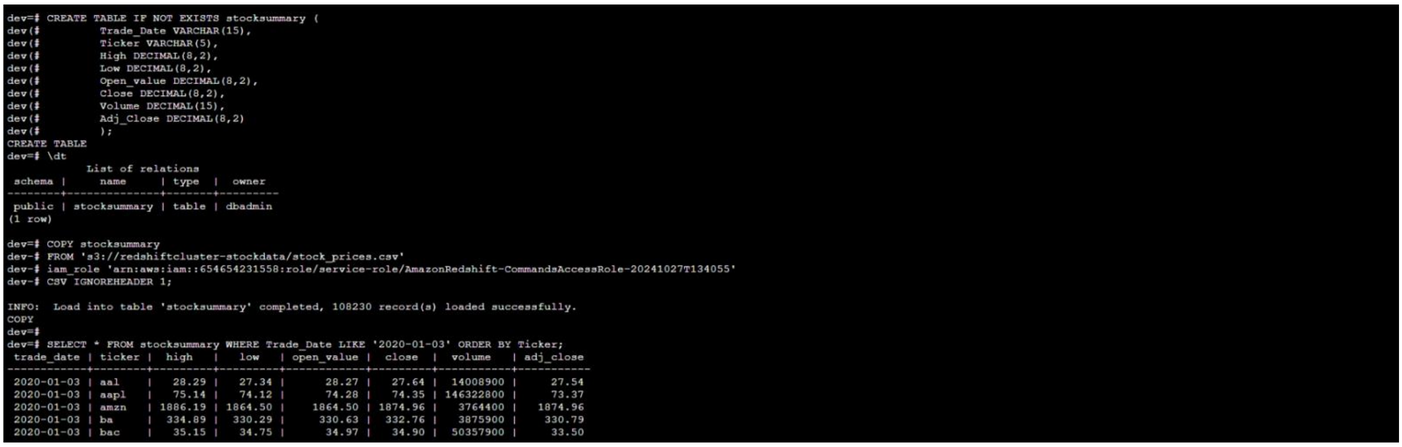
2. Load and Query Data in Redshift Cluster using Command Host and psql.
   AWS Lab replication on personal AWS account.



Steps:

1. Created `RedshiftCluster-VPC`.
2. Added an EC2 instance to the public subnet for internet access and a Redshift cluster to the private subnet for enhanced security.
3. Created separate security groups: `CommandHost-SG` and `RedshiftCluster-SG`.
4. Created an S3 bucket and uploaded a CSV file with stock data.
5. Launched an EC2 instance environment (Command Host Instance) to send commands to a Redshift cluster database.
6. Logged in to EC2 via SSH to install `psql`.
7. Created a cluster subnet group and a cluster parameter group, adjusting the `statement_timeout` to 1 minute.

8. Created the `stock-cluster`.
9. Loaded data into Redshift using `psql` commands via the CommandHost (screenshots below).

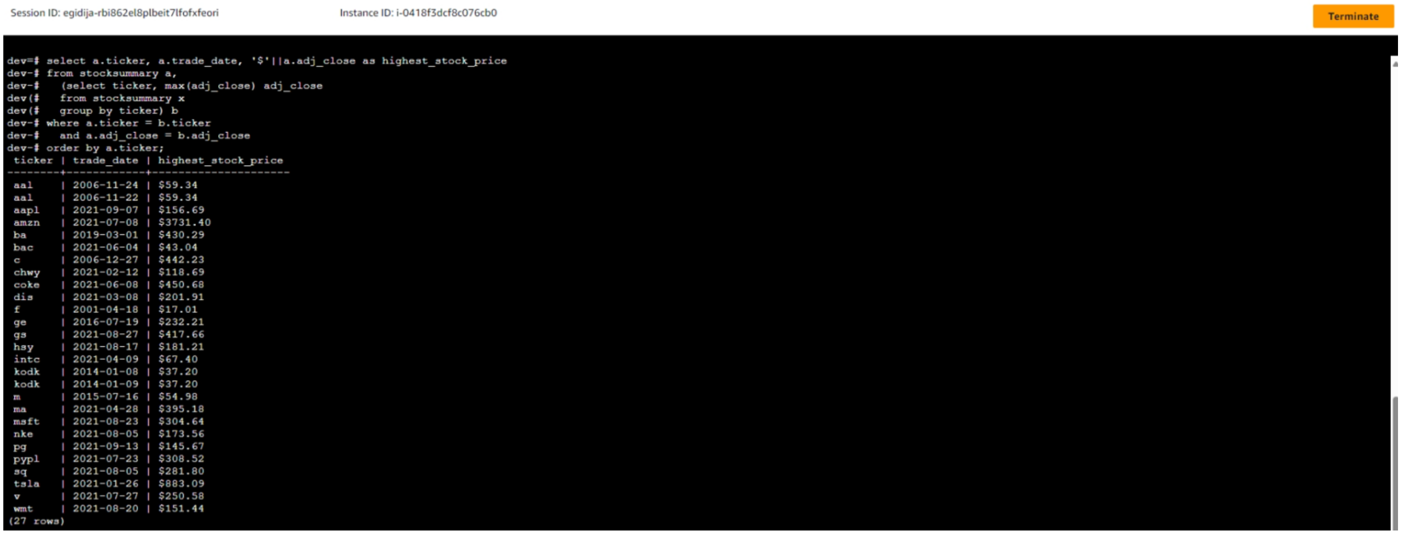


```
dev=# CREATE TABLE IF NOT EXISTS stocksummary (
dev(#          Trade_Date VARCHAR(15),
dev(#          Ticker VARCHAR(5),
dev(#          High DECIMAL(8,2),
dev(#          Low DECIMAL(8,2),
dev(#          Open_value DECIMAL(8,2),
dev(#          Close DECIMAL(8,2),
dev(#          Volume DECIMAL(15),
dev(#          Adj_Close DECIMAL(8,2)
dev(#          );
CREATE TABLE
dev=# \dt
            List of relations
 schema |     name     | type  |  owner
--------+--------------+-------+---------
 public | stocksummary | table | dbadmin
(1 row)

dev=# COPY stocksummary
dev-# FROM 's3://redshiftcluster-stockdata/stock_prices.csv'
dev-# iam_role 'arn:aws:iam::654654231558:role/service-role/AmazonRedshift-CommandsAccessRole-20241027T134055'
dev-# CSV IGNOREHEADER 1;

INFO:  Load into table 'stocksummary' completed, 108230 record(s) loaded successfully.
COPY
dev=#
dev=# SELECT * FROM stocksummary WHERE Trade_Date LIKE '2020-01-03' ORDER BY Ticker;
 trade_date | ticker |  high   |   low   | open_value |  close  |  volume   | adj_close
------------+--------+---------+---------+------------+---------+-----------+-----------
 2020-01-03 | aal    |   28.29 |   27.34 |      28.27 |   27.64 |  14008900 |     27.54
 2020-01-03 | aapl   |   75.14 |   74.12 |      74.28 |   74.35 | 146322800 |     73.37
 2020-01-03 | amzn   | 1886.19 | 1864.50 |    1864.50 | 1874.96 |   3764400 |   1874.96
 2020-01-03 | ba     |  334.89 |  330.29 |     330.63 |  332.76 |   3875900 |    330.79
 2020-01-03 | bac    |   35.15 |   34.75 |      34.97 |   34.90 |  50357900 |     33.50
```

SQL query to calculate all time high stock price for each company:



Session ID: egidija-rbi862el8plbeit7lfofxfeori

Instance ID: i-0418f3dcf8c076cb0

```
dev=# select a.ticker, a.trade_date, '$'||a.adj_close as highest_stock_price
dev-# from stocksummary a,
dev-#   (select ticker, max(adj_close) adj_close
dev(#   from stocksummary x
dev(#   group by ticker) b
dev-# where a.ticker = b.ticker
dev-#   and a.adj_close = b.adj_close
dev-# order by a.ticker;
 ticker | trade_date | highest_stock_price
--------+------------+---------------------
 aal    | 2006-11-24 | $59.34
 aal    | 2006-11-22 | $59.34
 aapl   | 2021-09-07 | $156.69
 amzn   | 2021-07-08 | $3731.40
 ba     | 2019-03-01 | $430.29
 bac    | 2021-06-04 | $43.04
 c      | 2006-12-27 | $442.23
 chwy   | 2021-02-12 | $118.69
 coke   | 2021-06-08 | $450.68
 dis    | 2021-03-08 | $201.91
 f      | 2001-04-18 | $17.01
 ge     | 2016-07-19 | $232.21
 gs     | 2021-08-27 | $417.66
 hsy    | 2021-08-17 | $181.21
 intc   | 2021-04-09 | $67.40
 kodk   | 2014-01-08 | $37.20
 kodk   | 2014-01-09 | $37.20
 m      | 2015-07-16 | $54.98
 ma     | 2021-04-28 | $395.18
 msft   | 2021-08-23 | $304.64
 nke    | 2021-08-05 | $173.56
 pg     | 2021-09-13 | $145.67
 pypl   | 2021-07-23 | $308.52
 sq     | 2021-08-05 | $281.80
 tsla   | 2021-01-26 | $883.09
 v      | 2021-07-27 | $250.58
 wmt    | 2021-08-20 | $151.44
(27 rows)
```

3. Created EC2 instance via CloudFormation yaml template

```
Resources:
 MyInstance:
  Type: AWS::EC2::Instance
  Properties:
   AvailabilityZone: us-east-1a
   ImageId: ami-0a3c3a20c09d6f377
   InstanceType: t2.micro
```