**PROJECT with EC2, SQS AND RDS**

**Managing Tightly coupled Architecture using AWS SQS.**

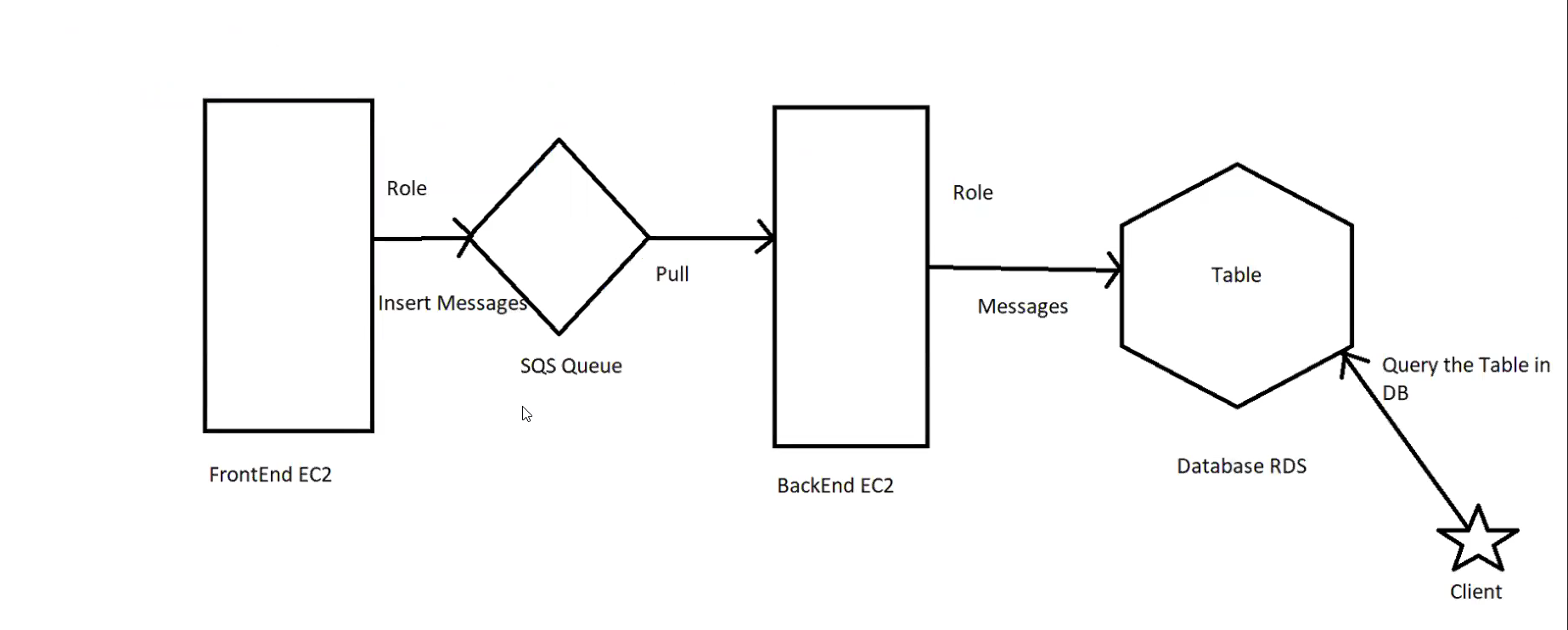
1.Create MYSQL DB using RDS service.

2.Create SQS Queue.

3.Connect through SQL client and create a Table.

4.Create Frontend and Backend EC2 servers.

5.Connect the SQS Queue with the frontend server to insert entries into MySQL DB in RDS service.



**I)Create Security Group**

1.In Inbound, Allow MYSQL/AURORA traffic at port 3306 from Anywhere IPV4

2.we will attach this SG to our mysql Db later.

**II)RDS creation**

1.Create a RDS

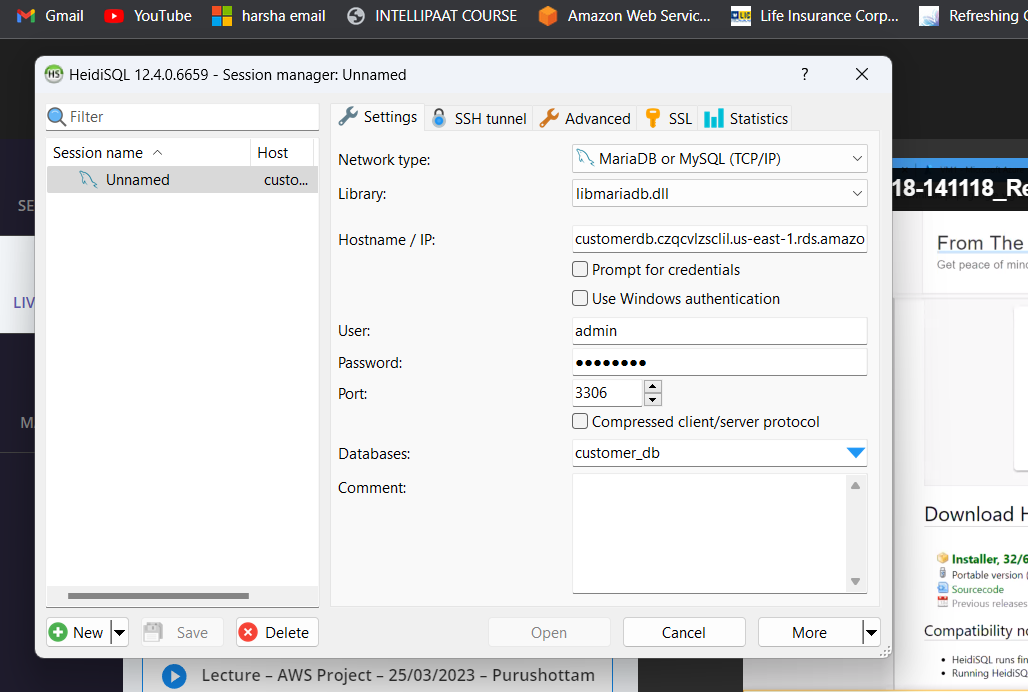
2.In connectivity section, give public access YES, because we want to use this DB directly from our Personal LAPTOP.

3.In VPC SG , attach the previously created SG

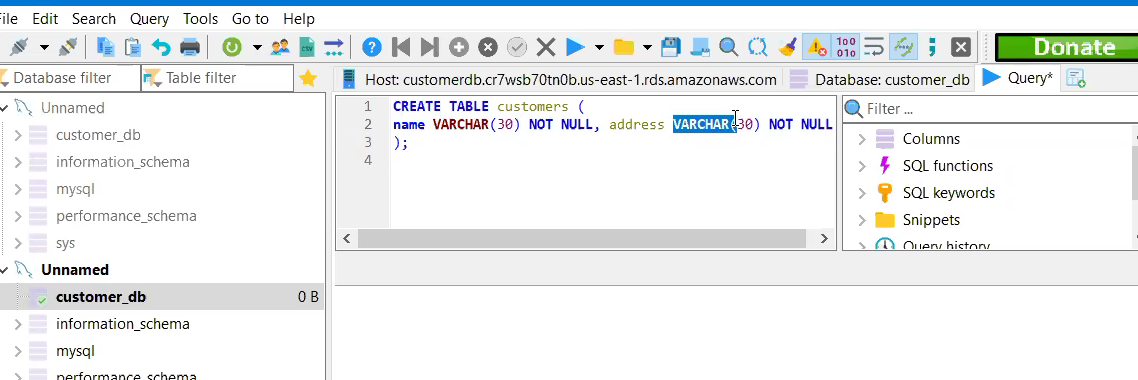
**III)Install Heidi SQL in PC**

**1.connect Heidi sql to the RDS DB by mentioning the user id and password which were defined for RDS.**

**2.in Host IP give the RDS endpoint.**

****

**3.Give the following commands and run the query to create a table**

****

**IV)Create SQS :**

**1.choose standard**

**2.Note down the Queue URL and ARN, we will need it later on.**

**V)Create IAM polices**

1.create a policy with the following JSON

***SQS\_SendMessage\_Policy***

***{***

***"Version": "2012-10-17", "Statement": [{***

***"Sid":"SQSSendMessage",***

***"Effect": "Allow",***

***"Action": "sqs:SendMessage",***

***"Resource": "arn:***GIVE YOUR ARN HERE***"***

***}] }***

2.CREATE A ROLE and attach the SQS send policy to this role , that give EC2 permissions to send messages in the Q.

***3***.create a SQS receive and delete policy front backend EC2 instance

***SQS\_ReceiveDeleteMessage\_Policy***

***{***

***"Version": "2012-10-17", "Statement": [{***

***"Sid":"SQSReceiveDeleteMessage",***

***"Effect": "Allow",***

***"Action": ["sqs:ReceiveMessage" , "sqs:DeleteMessage"],***

***"Resource": "arn:aws:sqs:us-east-1:304000509264:CustomerQueue"***

***}] }***

4.CREATE A ROLE and attach the SQS receivedelete policy to this role , that give EC2 backend instance permissions to rceive and delete the messages in Q.

VI)**Create 2 EC2 instances**

1.create a frontend instance and give IAM role [SQS\_Send\_role](https://us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#roles/SQS_Send_role)

2.create a backend instance and give IAM role [SQS\_Recive\_del\_role](https://us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#roles/SQS_Recive_del_role)

3.Give SG of SSH from anywhere for both frontend and backend instances.

**4.Connect to both frontend and backend instances.**

**5.Give following commands in frontend server:**

0.sudo su –

1.apt-get update -y

    2  apt-get install -y python3.10

    3  apt install python3-pip

    4  pip3 install boto3

    5  pip3 install mysql-connector-python

    6  mkdir .aws

    7  echo -e "[default]\nregion=us-east-1" > .aws/config

    8  vi send\_message.py

    9  python3 send\_message.py Praveen,Hyderabad

   10  python3 send\_message.py Prajval,Delhi

   11  python3 send\_message.py Kusuma,Mumbai

   12  ls

   13  vi send\_message.py

   14  history

**On front end instance, the file looks like below:**

import sys

import boto3

sqs = boto3.client('sqs')

queue\_url = 'https://sqs.us-east-1.amazonaws.com/432501396116/CustomerQueue' #Give SQS Q URL

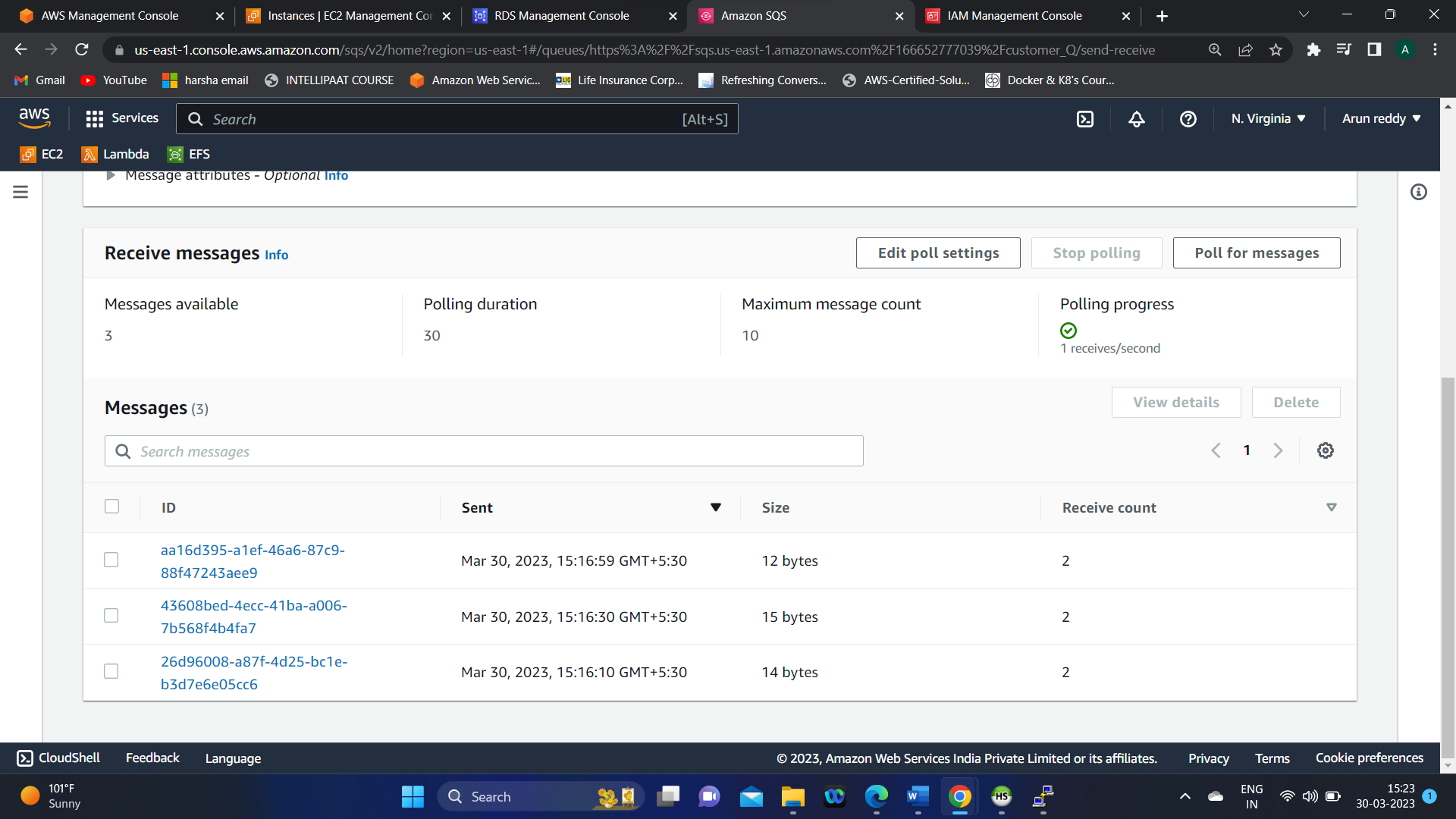
response = sqs.send\_message( QueueUrl=queue\_url, MessageBody=(sys.argv[1]))

print(response['MessageId'])

+++++++++++++++++++++++++++++++++++++++

+++++++++++++++++++++++++++++++++++++++

Now you can see the messages in SQS Q



**NOW WE ARE ABLE TO PUSH THE MESSAGES INTO SQS Q.**

**6.Give following commands on backend instance :**

**On backend instance**

1  apt-get update -y

    2  apt-get install python3.10 -y

    3  apt install python3-pip

    4  pip install python-pip

    5  apt install python3-pip

    6  pip install boto3

    7  pip install mysql-connector-python

    8  mkdir .aws

    9  echo -e "[default]\nregion=us-east-1" > .aws/config

   10  vi get\_message\_write\_to\_rds.py

   11  python get\_message\_write\_to\_rds.py  --------🡪not valid igore

   12  wget https://downloads.mysql.com/archives/get/p/29/file/mysql-connector-python-py3\_8.0.28-1ubuntu18.04\_amd64.deb

   13  ls

   14  apt-get install mysql-connector-python-py3\_8.0.28-1ubuntu18.04\_amd64.deb --------🡪not valid igore

   15  pwd

   16  apt-get install /root/mysql-connector-python-py3\_8.0.28-1ubuntu18.04\_amd64.deb

16.1)pip3 install mysql-connector

   17  sudo chown -Rv \_apt:root /var/cache/apt/archives/partial/

   18  sudo chmod -Rv 700 /var/cache/apt/archives/partial/

   19  sudo chown -Rv \_apt:root /root/mysql-connector-python-py3\_8.0.28-1ubuntu18.04\_amd64.deb

   20  sudo chmod -Rv 700 /root/mysql-connector-python-py3\_8.0.28-1ubuntu18.04\_amd64.deb

   21  apt-get install /root/mysql-connector-python-py3\_8.0.28-1ubuntu18.04\_amd64.deb

   22  python get\_message\_write\_to\_rds.py

   23  python3 get\_message\_write\_to\_rds.py

   24  pip3 install boto3

   25  apt install python3-pip

   26  pip3 install boto3

   27  python3 get\_message\_write\_to\_rds.py

   28  history

**On backend instance file should look like below:**

import time

import boto3

import mysql.connector

queue\_url = ‘https://sqs.us-east-1.amazonaws.com/166652777039/customer\_Q’

#Specify the database details

host = ‘customerdb.czqcvlzsclil.us-east-1.rds.amazonaws.com’

user = admin

password = ‘admin123’

database= ‘customer\_db’

#Create a SQS Client

sqs = boto3.client('sqs',  region\_name='us-east-1’)

#Connect to the RDS MySQL Instance

mydb = mysql.connector.connect(host=host, user=user, password=password, database=database)

mycursor = mydb.cursor()

# Receive message from SQS queue

response = sqs.receive\_message(QueueUrl=queue\_url)

message = response['Messages'][0]

# Delete received message from queue

receipt\_handle = message['ReceiptHandle']

sqs.delete\_message( QueueUrl=queue\_url,ReceiptHandle=receipt\_handle )

print('Received and deleted message: %s' % message["Body"])

#Get the customer name and address from the message

customerDetails = message["Body"]

customerDetailsList = customerDetails.split(',')

name = customerDetailsList[0]

address = customerDetailsList[1]

#Write the record to the database

val = (name, address)

sql = "INSERT INTO customers (name, address) VALUES (%s, %s)"

mycursor.execute(sql, val)

mydb.commit()

print("Record inserted in the DB")

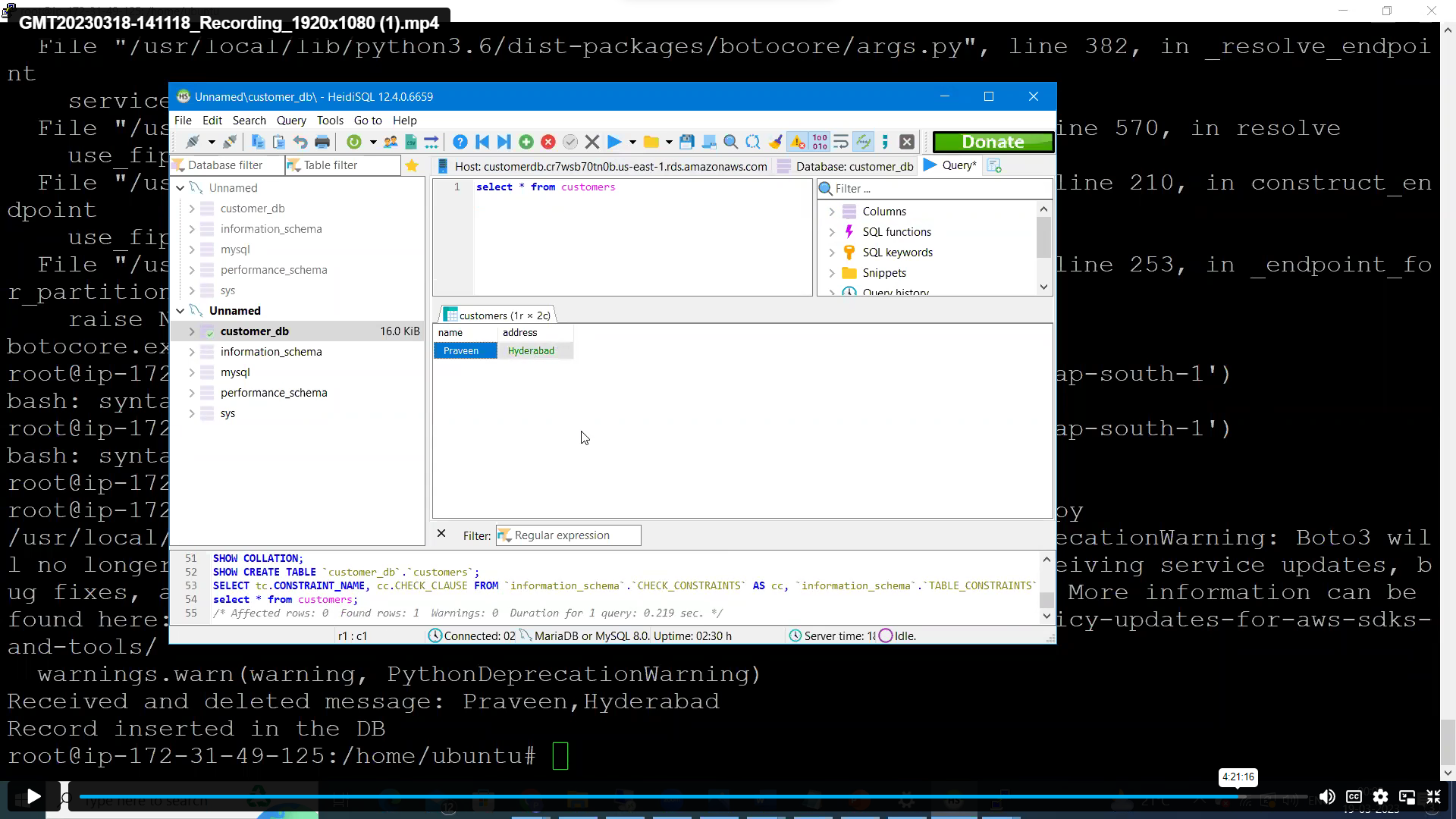
1.untill step 21 executed perfect

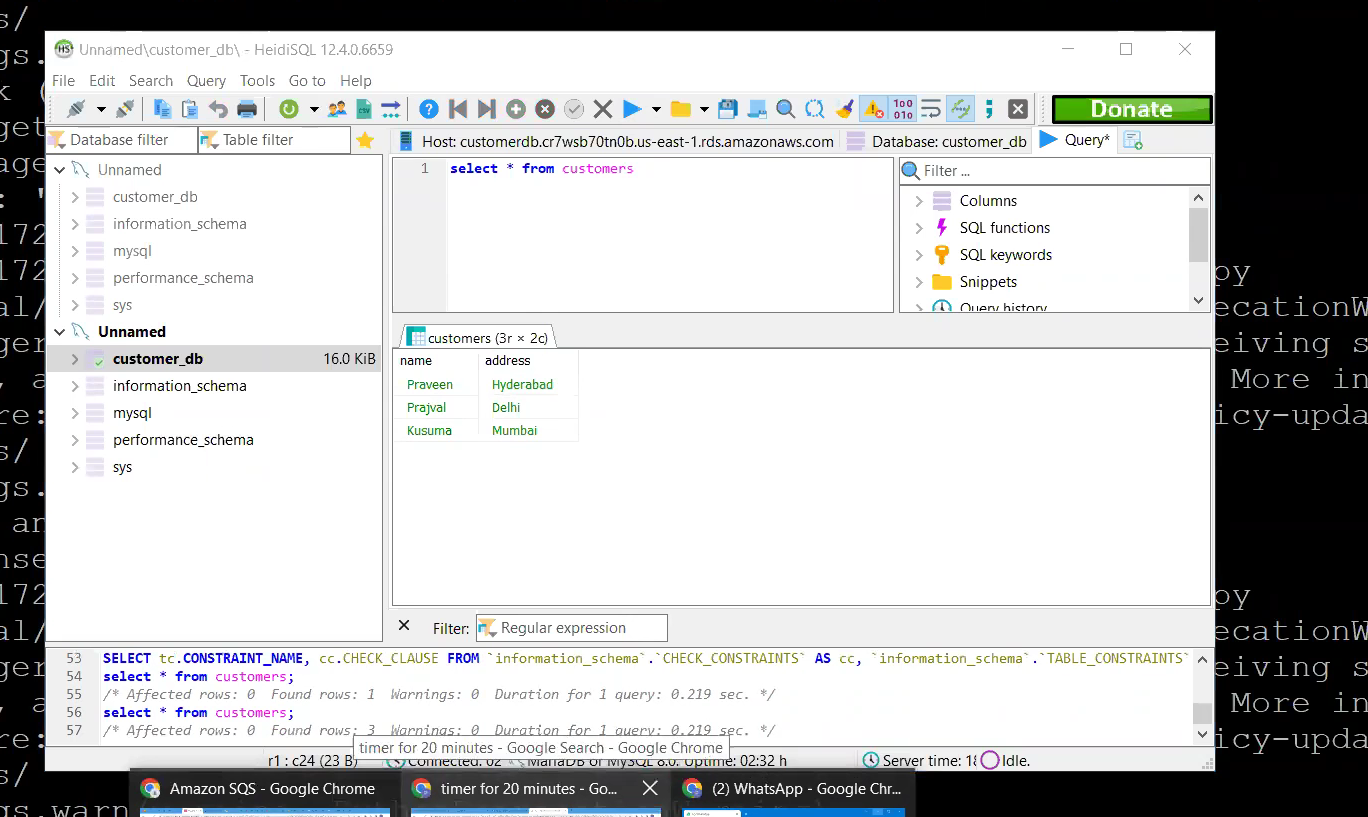
2.some syntax error in scripting

3.once you perform step 22(write message to RDS)

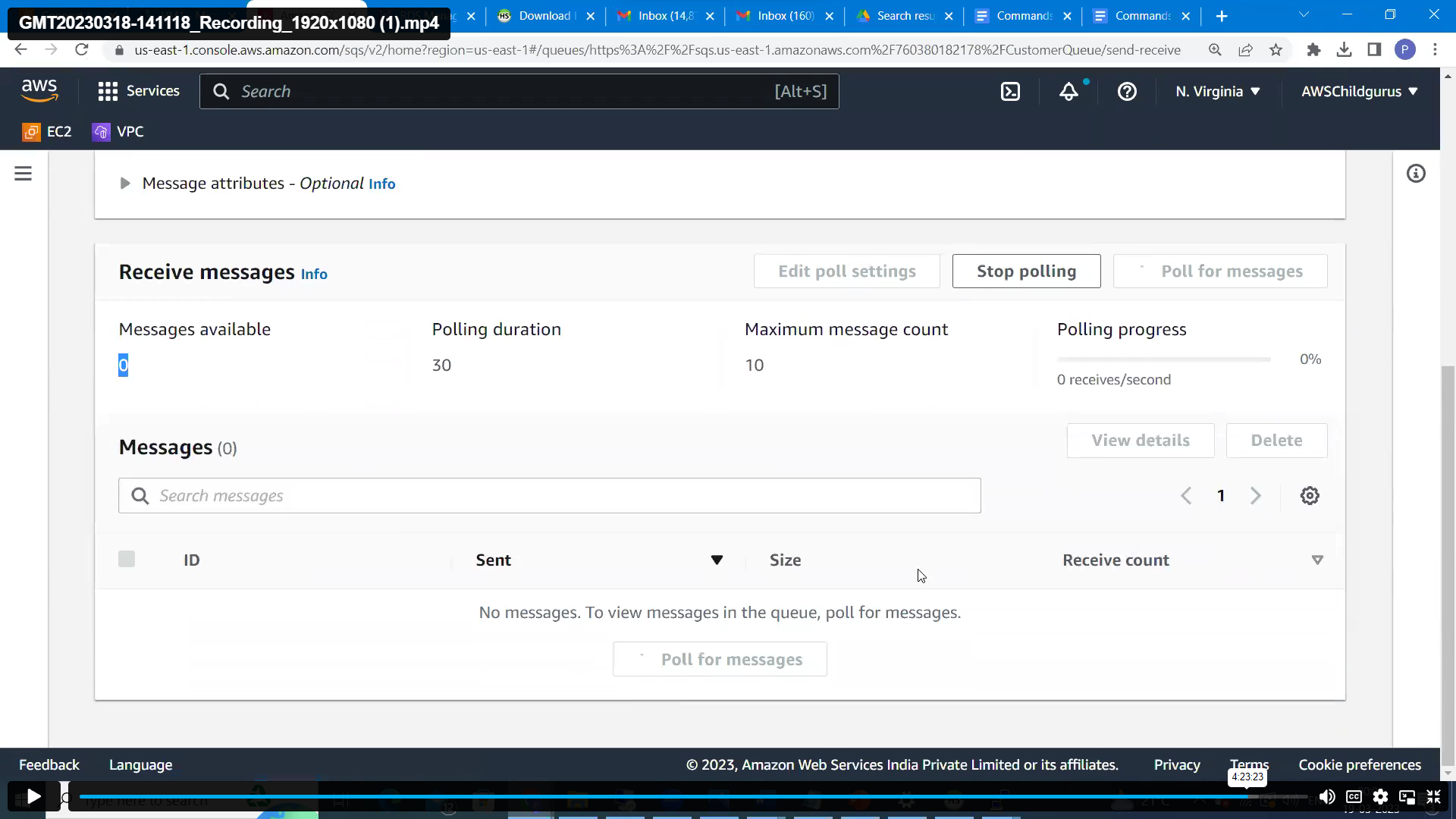
4. you can see the message in Table through HEIDI SQL like below :

5. once it is listed in the table it will be deleted rom the SQS Queue.





No messages are available in SQS Q :



We have created a Database ,And from frontend instance we pushed messages into SQS Q, and then pulled and deleted those messages through Backend instance and written tose records in a Database.

That completes our project…………