

School of Computer Science, Engineering and Applications(SCSEA)

B.C.A. TY (SCSEA)

Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav **PRN:** 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages

Open 3 tabs

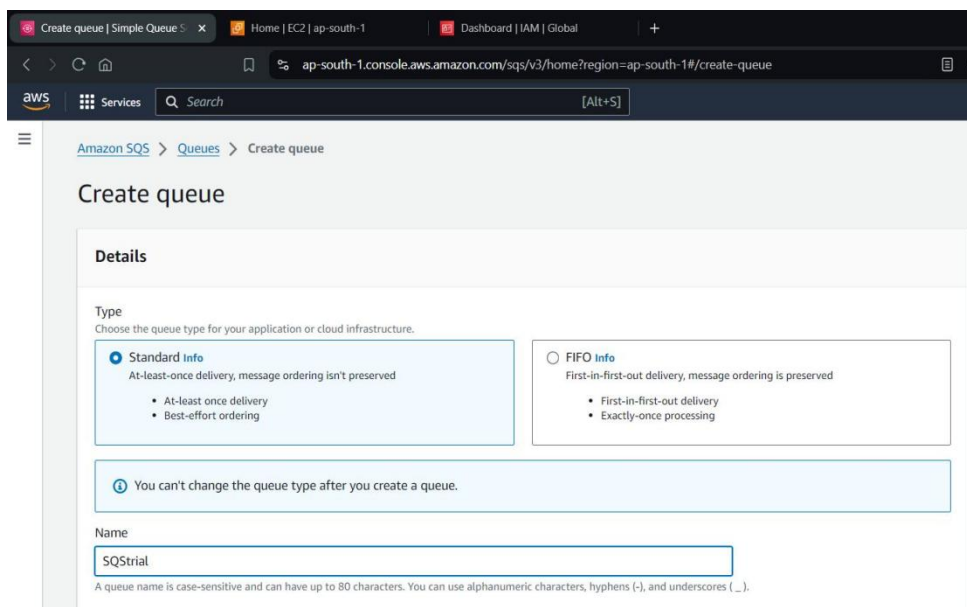
(1)--sqs , (2)--ec2 , (3)--iam

Step 1: create an sqs queue

Create queue-- name--

Keep rest as default configuration

Create queue



Step 2: create ec2 instance

Name-- name instance , AMI-- ubuntu , Instance type-- t2.micro

Keypair-- new

School of Computer Science, Engineering and Applications(SCSEA)

B.C.A. TY (SCSEA)

Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav

PRN: 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages

Network setting--edit

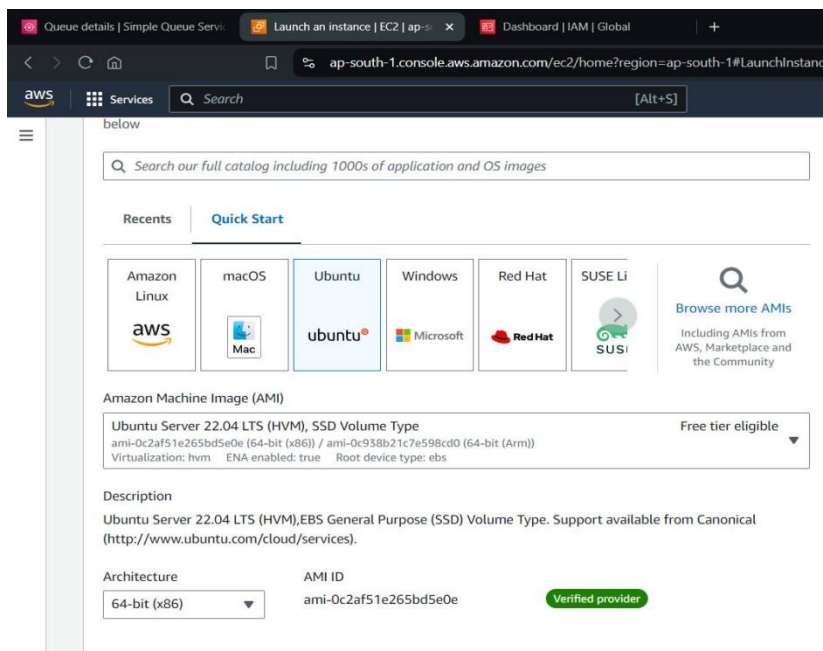
Auto assign public ip--enable

Add security group rule

Type(custom) port(3000)

Source(anywhere)

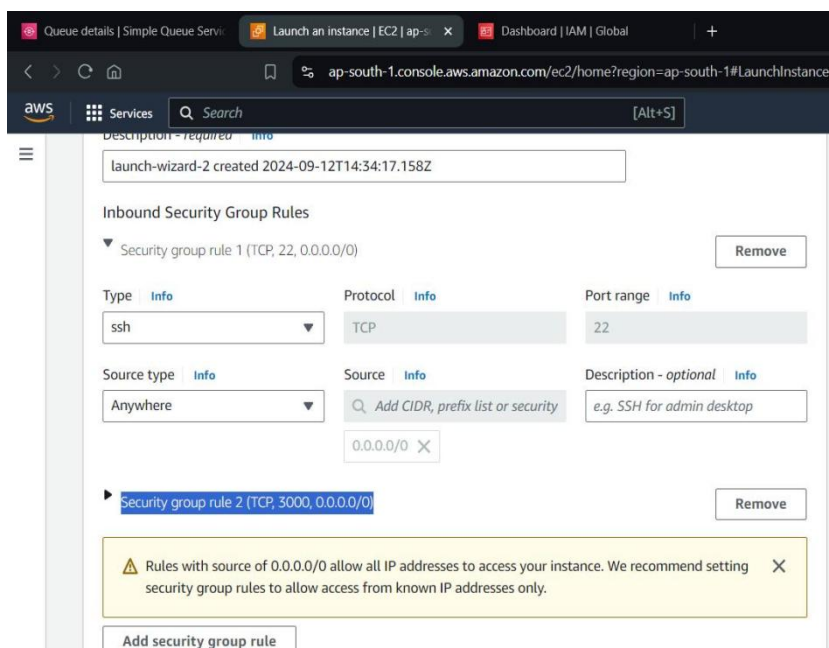
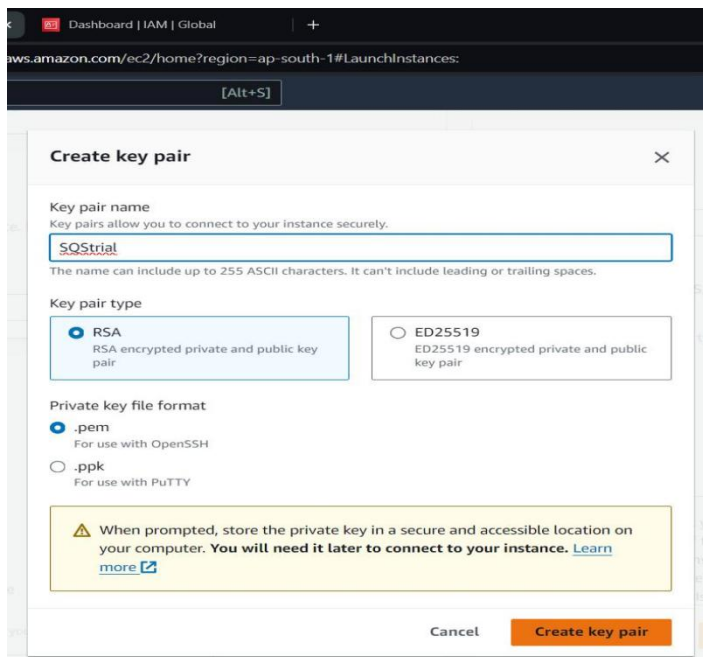
Launch.



School of Computer Science, Engineering and Applications(SCSEA)
B.C.A. TY (SCSEA)
Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav **PRN:** 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages



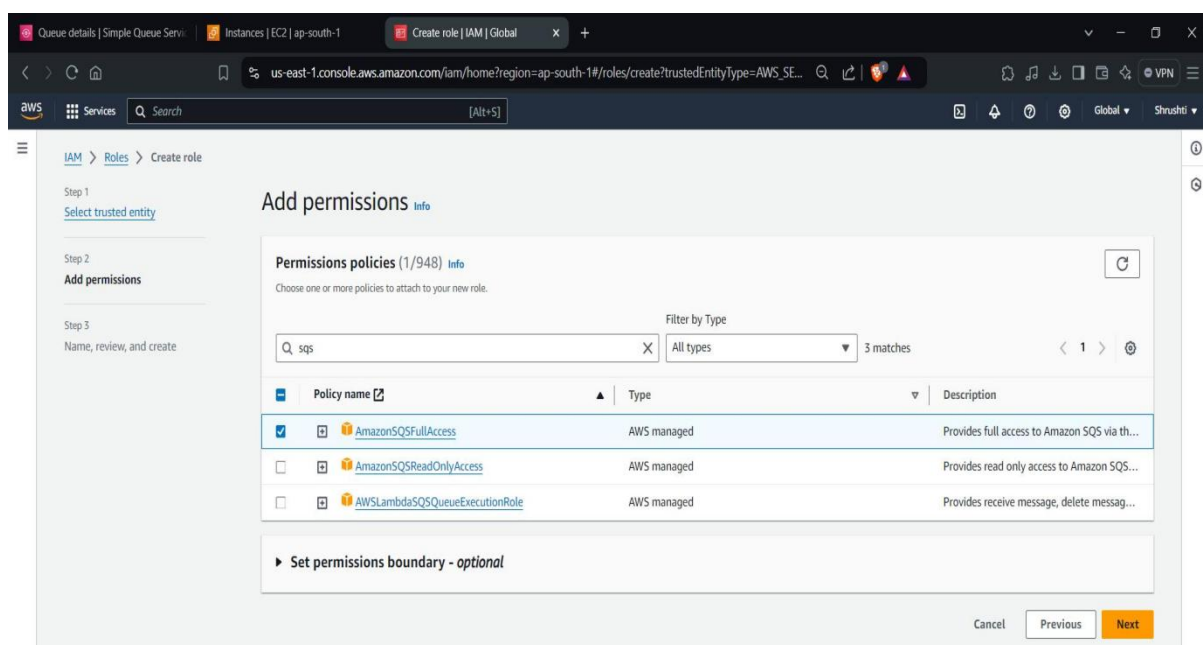
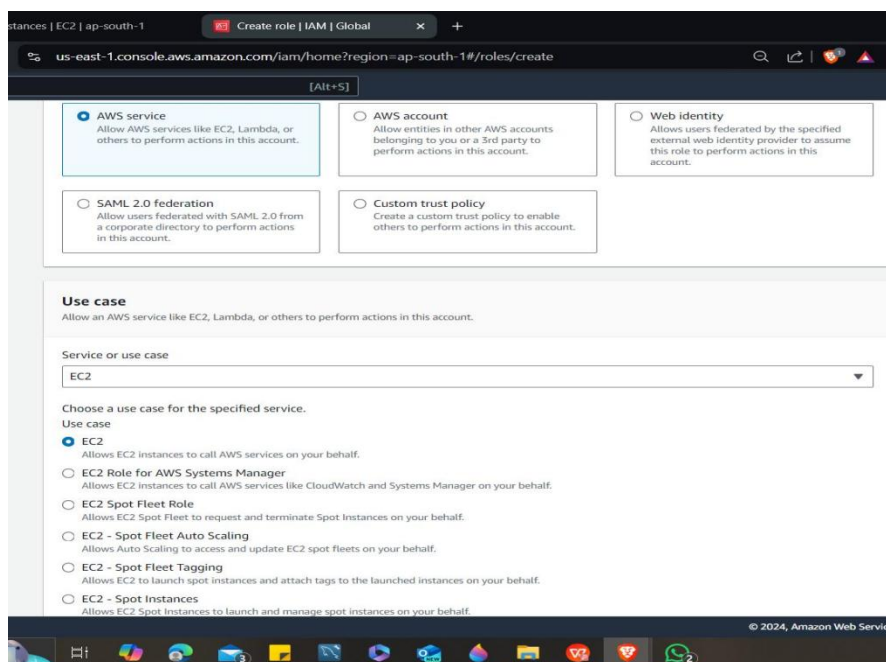
launch

School of Computer Science, Engineering and Applications(SCSEA)
B.C.A. TY (SCSEA)
Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav **PRN:** 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages

Step 3: create IAM role

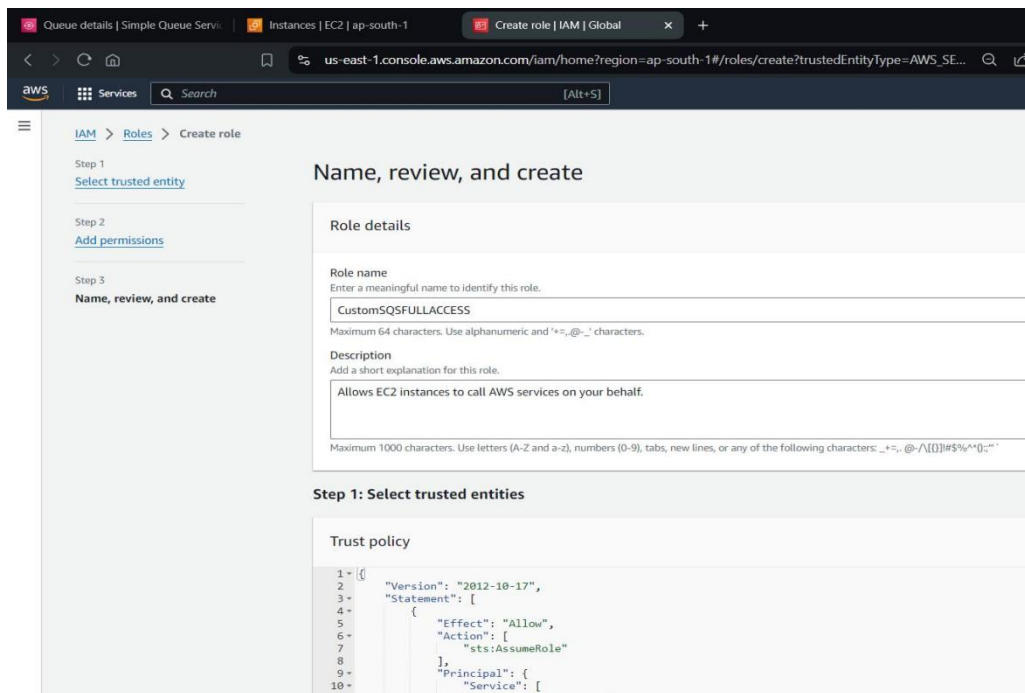


Policy name	Type	Description
<input checked="" type="checkbox"/> AmazonSQSFullAccess	AWS managed	Provides full access to Amazon SQS via th...
<input type="checkbox"/> AmazonSQSReadOnlyAccess	AWS managed	Provides read only access to Amazon SQS...
<input type="checkbox"/> AWSLambdaSQSQueueExecutionRole	AWS managed	Provides receive message, delete messag...

School of Computer Science, Engineering and Applications(SCSEA)
B.C.A. TY (SCSEA)
Subject: Advance Cloud Computing(ACC)

Name of the Student: **Shrushti Krishna Shrivastav** **PRN: 20220801024**

Title of Practicle : **AWS Simple Queue Service (SQS) For Sending Messages**



The screenshot shows the AWS IAM console 'Create role' wizard. The browser tabs include 'Queue details | Simple Queue Service', 'Instances | EC2 | ap-south-1', and 'Create role | IAM | Global'. The URL is 'us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/roles/create?trustedEntityType=AWS_SE...'. The left sidebar shows the navigation menu with 'IAM > Roles > Create role' selected. The main content area is titled 'Name, review, and create' and includes a 'Role details' section with fields for 'Role name' (CustomSQSFULLACCESS) and 'Description' (Allows EC2 instances to call AWS services on your behalf.). Below this is 'Step 1: Select trusted entities' and a 'Trust policy' section showing a JSON policy snippet.

```
1- {
2-   "Version": "2012-10-17",
3-   "Statement": [
4-     {
5-       "Effect": "Allow",
6-       "Action": [
7-         "sts:AssumeRole"
8-       ],
9-       "Principal": {
10-        "Service": [
```

After creating the rule , navigate back to previously created ec2 instance.

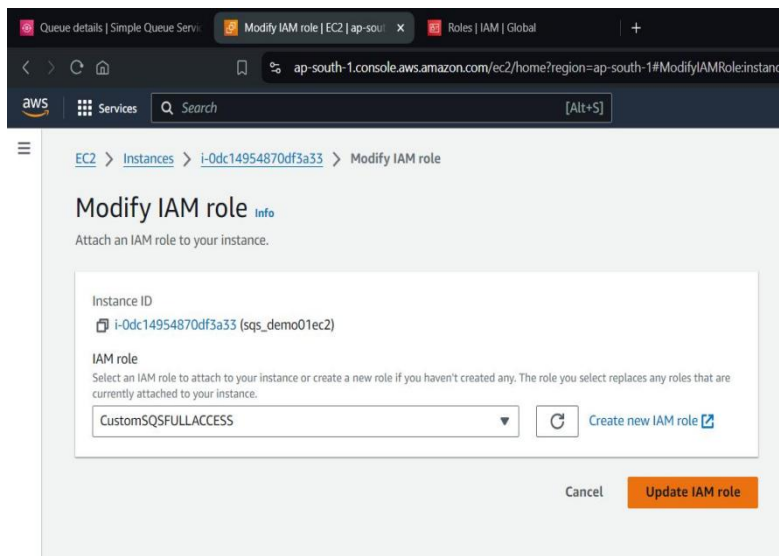
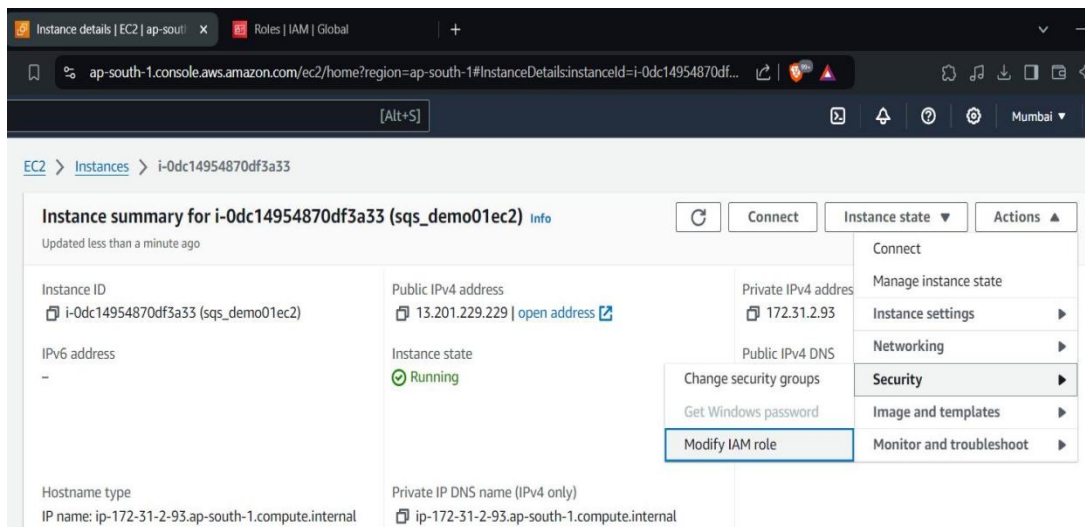
Click on 'instance-id' -- action -- security -- modify iam role --select the created role --

then -update iam role.

School of Computer Science, Engineering and Applications(SCSEA)
B.C.A. TY (SCSEA)
Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav **PRN:** 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages



Step4: connect ec2 instance



School of Computer Science, Engineering and Applications(SCSEA)

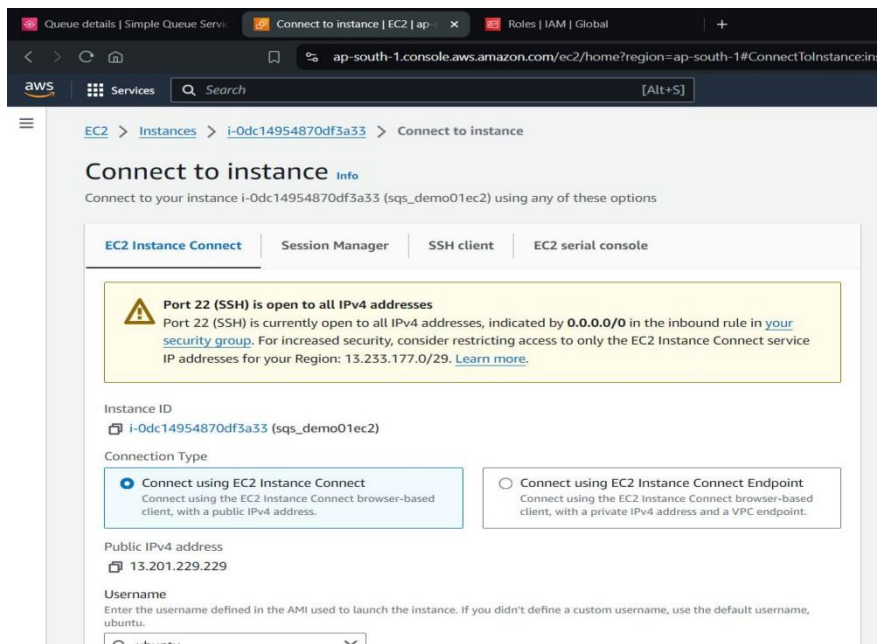
B.C.A. TY (SCSEA)

Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav

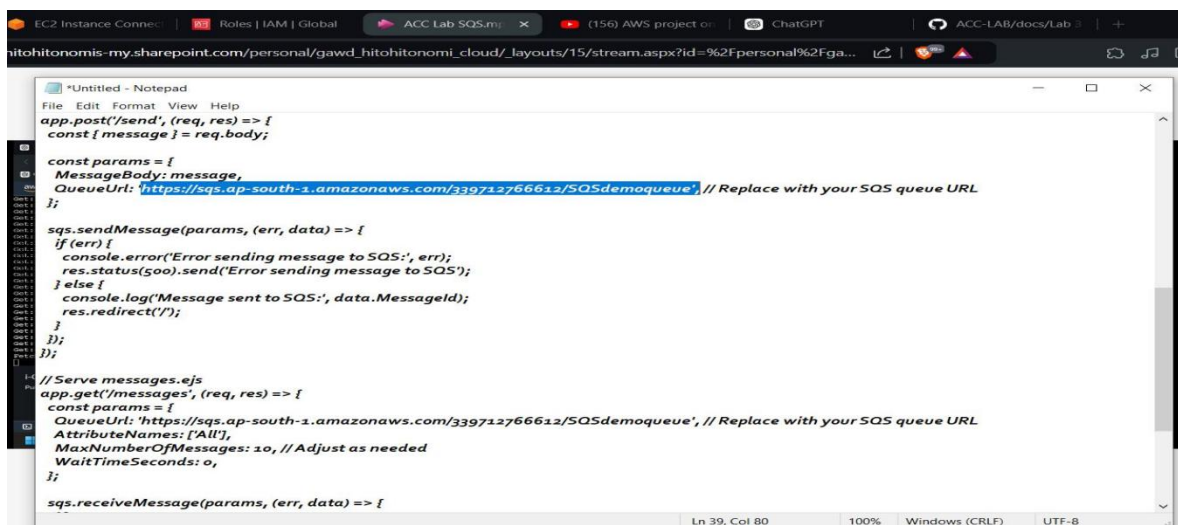
PRN: 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages



Step5: edit

Now copy paste the app.js file to notepad and replace the placeholder sqs url with your actual sqs queue url.(total 2 url to be replaced)





School of Computer Science, Engineering and Applications(SCSEA)

B.C.A. TY (SCSEA)

Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav **PRN:** 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages

Step6: ec2 connect

Back to ec2 connect and type the command

:: Sudo apt update && sudo apt upgrade -y && sudo apt install nodejs -y &&
sudo apt install npm -y

(in between a purple box will apperar a lot just press 'enter')

Once done

Follow these to create node.js file::::

```
No VM guests are running outdated hypervisor (qemu)
ubuntu@ip-172-31-1-145:~$
ubuntu@ip-172-31-1-145:~$
ubuntu@ip-172-31-1-145:~$ nano ~/app.js
```

```
i-05d5d633e2b2bb6af (SQSTRIAL)
```


School of Computer Science, Engineering and Applications(SCSEA)

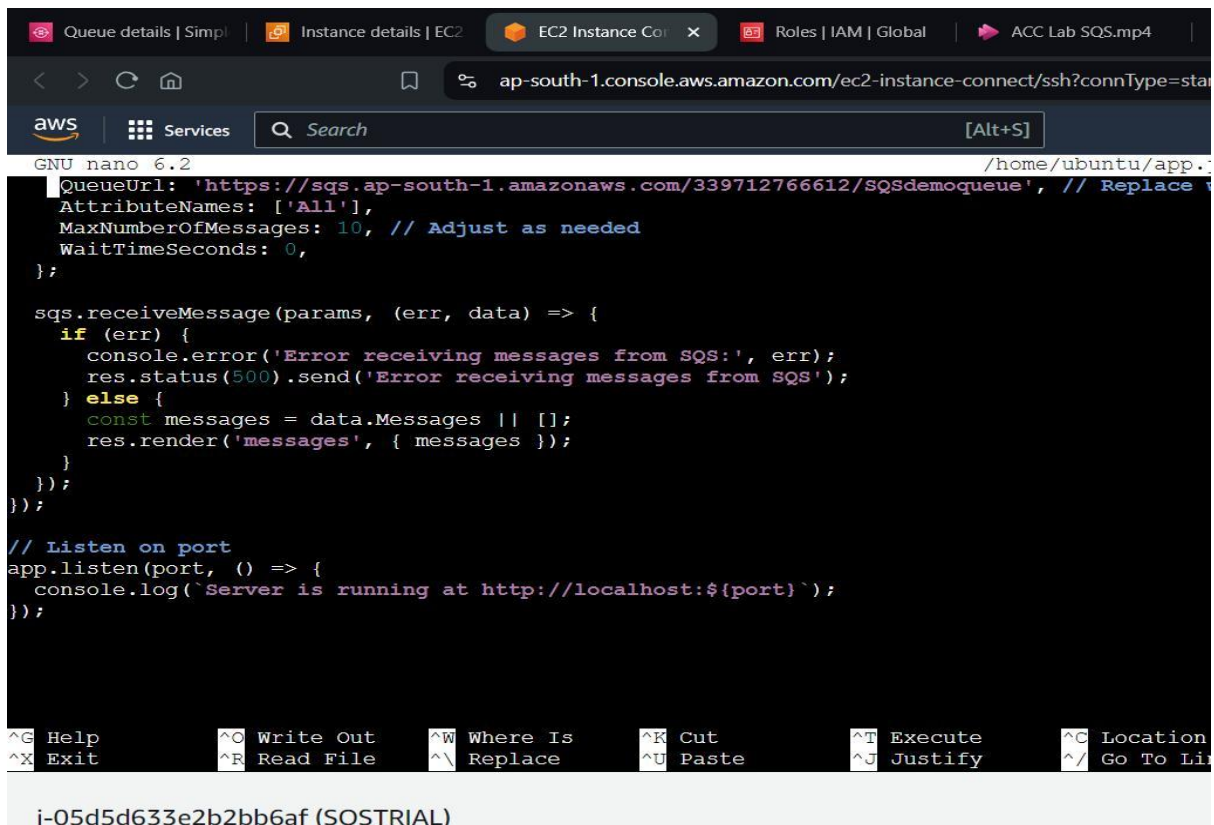
B.C.A. TY (SCSEA)

Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav

PRN: 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages



```
GNU nano 6.2 /home/ubuntu/app.js
QueueUrl: 'https://sqs.ap-south-1.amazonaws.com/339712766612/SQSDemoQueue', // Replace with your QueueUrl
AttributeNames: ['All'],
MaxNumberOfMessages: 10, // Adjust as needed
WaitTimeSeconds: 0,
};

sqs.receiveMessage(params, (err, data) => {
  if (err) {
    console.error('Error receiving messages from SQS:', err);
    res.status(500).send('Error receiving messages from SQS');
  } else {
    const messages = data.Messages || [];
    res.render('messages', { messages });
  }
});
});

// Listen on port
app.listen(port, () => {
  console.log(`Server is running at http://localhost:${port}`);
});
```

i-05d5d633e2b2bb6af (SQSTRIAL)

This will create app.js file and in editor paste the updated file.

To exit editor :

Ctrl+o-- enter-- ctrl+x

Done.

Now create index.html file::::



School of Computer Science, Engineering and Applications(SCSEA)

B.C.A. TY (SCSEA)

Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav

PRN: 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages

```
No user sessions are running outdated binaries.  
  
No VM guests are running outdated hypervisor (qemu) binaries on this  
ubuntu@ip-172-31-1-145:~$  
ubuntu@ip-172-31-1-145:~$  
ubuntu@ip-172-31-1-145:~$ nano ~/app.js  
ubuntu@ip-172-31-1-145:~$ nano ~/index.html
```

The screenshot shows the AWS Management Console for an SQS queue. The top navigation bar includes 'Queue details | Send', 'Instance details | View', and the IP address '3.109.158.15'. The main content area displays the 'nano' text editor with the following HTML code:

```
GNU nano 6.2  
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>SQS Example</title>  
</head>  
<body>  
  <h1>Home Page</h1>  
  <p>Welcome to the home page!</p>  
  <a href="/send">Go to Send Page</a>  
  <br>  
  <a href="/messages">View Messages</a>  
</body>  
</html>
```

Create send.html and add content::::

School of Computer Science, Engineering and Applications(SCSEA)
B.C.A. TY (SCSEA)
Subject: Advance Cloud Computing(ACC)

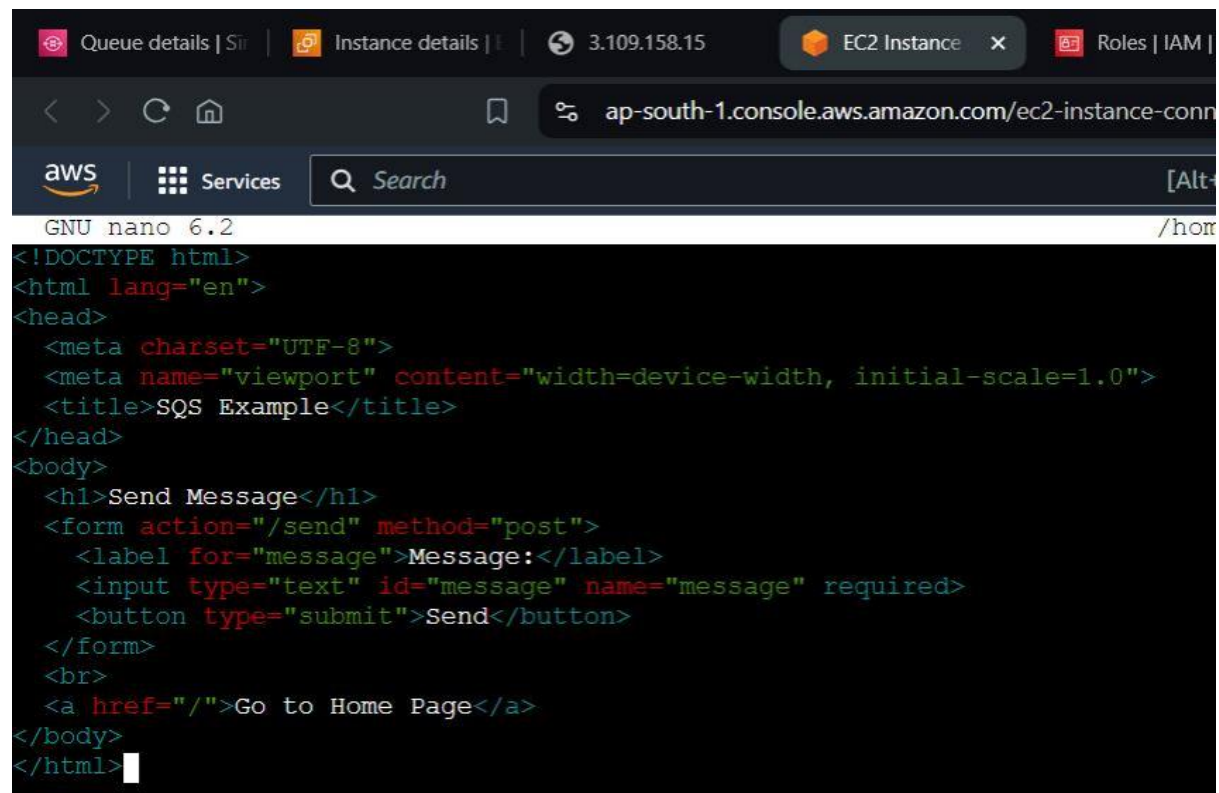
Name of the Student: Shrushti Krishna Shrivastav **PRN:** 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages

```
No VM guests are running outdated hypervisor (qemu)
ubuntu@ip-172-31-1-145:~$
ubuntu@ip-172-31-1-145:~$
ubuntu@ip-172-31-1-145:~$ nano ~/app.js
ubuntu@ip-172-31-1-145:~$ nano ~/index.html
ubuntu@ip-172-31-1-145:~$ nano ~/send.html
```

i-05d5d633e2b2bb6af (SQSTRIAL)

PublicIPs: 3.109.158.15 PrivateIPs: 172.31.1.145



The screenshot shows the AWS Management Console for an EC2 instance named 'i-05d5d633e2b2bb6af (SQSTRIAL)'. The instance is running Ubuntu 6.2. The terminal shows the creation of three files: app.js, index.html, and send.html. The 'send.html' file is open in the nano editor, displaying the following HTML code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>SQS Example</title>
</head>
<body>
  <h1>Send Message</h1>
  <form action="/send" method="post">
    <label for="message">Message:</label>
    <input type="text" id="message" name="message" required>
    <button type="submit">Send</button>
  </form>
  <br>
  <a href="/">Go to Home Page</a>
</body>
</html>
```

Create view directory and message.ejs file

School of Computer Science, Engineering and Applications(SCSEA)

B.C.A. TY (SCSEA)

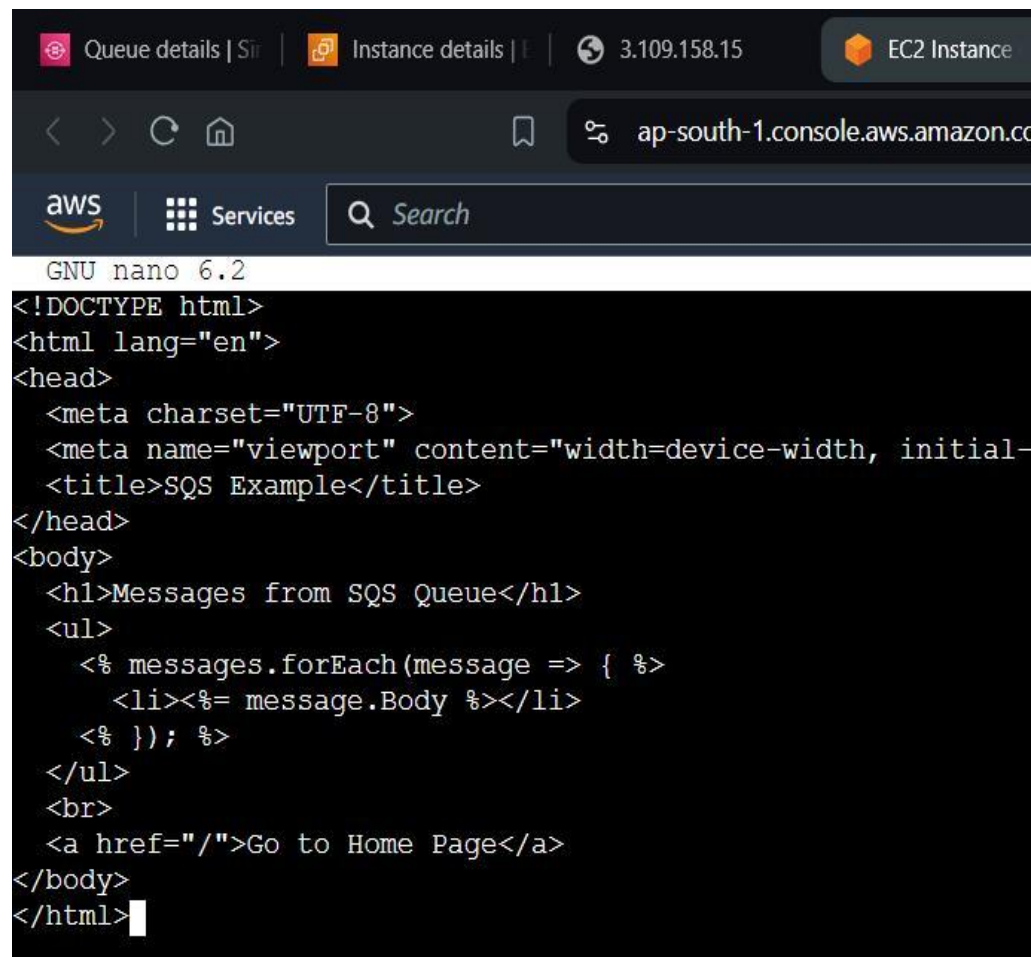
Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav

PRN: 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-1-145:~$
ubuntu@ip-172-31-1-145:~$
ubuntu@ip-172-31-1-145:~$ nano ~/app.js
ubuntu@ip-172-31-1-145:~$ nano ~/index.html
ubuntu@ip-172-31-1-145:~$ nano ~/send.html
ubuntu@ip-172-31-1-145:~$ mkdir ~/views && nano ~/views/messages.ejs
```



The screenshot shows the AWS Management Console for an EC2 instance. The top navigation bar includes links for Queue details, Instance details, and the IP address 3.109.158.15. The main content area displays the SQS console interface. Overlaid on this is a terminal window running the GNU nano 6.2 text editor. The editor shows the creation of an HTML file named messages.ejs, which contains a title 'SQS Example' and a list of messages from the SQS queue. The code uses a loop to iterate over messages and display their bodies. A link to the home page is also included.

```
GNU nano 6.2
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
  <title>SQS Example</title>
</head>
<body>
  <h1>Messages from SQS Queue</h1>
  <ul>
    <% messages.forEach(message => { %>
      <li><%= message.Body %></li>
    <% }); %>
  </ul>
  <br>
  <a href="/">Go to Home Page</a>
</body>
</html>
```



School of Computer Science, Engineering and Applications(SCSEA)

B.C.A. TY (SCSEA)

Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav

PRN: 20220801024

Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages

Now install required packages

```
ubuntu@ip-172-31-1-145:~$  
ubuntu@ip-172-31-1-145:~$ nano ~/app.js  
ubuntu@ip-172-31-1-145:~$ nano ~/index.html  
ubuntu@ip-172-31-1-145:~$ nano ~/send.html  
ubuntu@ip-172-31-1-145:~$ mkdir ~/views && nano ~/views/messages.ejs  
ubuntu@ip-172-31-1-145:~$ mkdir ~/views && nano ~/views/messages.ejs  
mkdir: cannot create directory '/home/ubuntu/views': File exists  
ubuntu@ip-172-31-1-145:~$ npm install express aws-sdk body-parser ejs  
npm WARN deprecated querystring@0.2.0: The querystring API is considered Legacy. new code should use the URLSearchParams API instead.  
added 105 packages, and audited 106 packages in 6s  
  
23 packages are looking for funding  
  run `npm fund` for details  
  
found 0 vulnerabilities  
ubuntu@ip-172-31-1-145:~$ node ~/app.js  
Server is running at http://localhost:3000  
(node:13634) NOTE: The AWS SDK for JavaScript (v2) is in maintenance mode.  
  SDK releases are limited to address critical bug fixes and security issues only.  
  
Please migrate your code to use AWS SDK for JavaScript (v3).  
For more information, check the blog post at https://a.co/cUPnyil
```

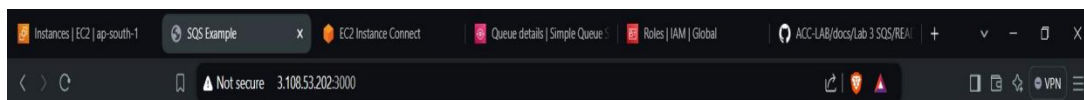
i-05d5d633e2b2bb6af (SQSTRIAL)

PublicIPs: 3.109.158.15 PrivateIPs: 172.31.1.145

Run :: node ~/app.js

Your application should now be running at:::

http:// your-ec2-public-ip:3000



Home Page

Welcome to the home page!

[Go to Send Page](#)

[View Messages](#)



**D Y PATIL
INTERNATIONAL
UNIVERSITY**
AKURDI PUNE

School of Computer Science, Engineering and Applications(SCSEA)

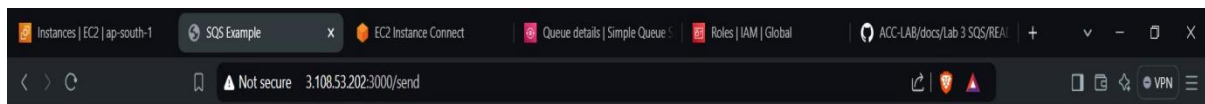
B.C.A. TY (SCSEA)

Subject: Advance Cloud Computing(ACC)

Name of the Student: Shrushti Krishna Shrivastav

PRN: 20220801024

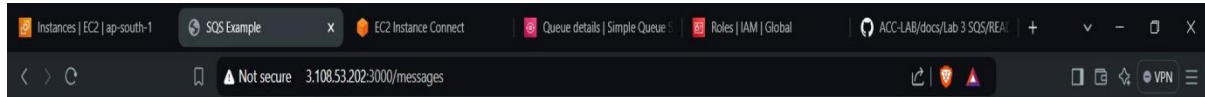
Title of Practicle : AWS Simple Queue Service (SQS) For Sending Messages



Send Message

Message:

[Go to Home Page](#)



Messages from SQS Queue

- hello

[Go to Home Page](#)

Done.