DEPARTMENT OF COMPUTER APPLICATIONS

CADX118 - CLOUD SERVICES FOR DATA SCIENCE LABORATORY

B.S. Abdur Rahman
Crescent
Institute of Science & Technology
Deemed to be University u/s 3 of the UGC Act, 1956

LAB RECORD

NAME : _____

RRN : _____

LAB : CADX118 - CLOUD SERVICES FOR DATA SCIENCE LABORATORY

B.S. Abdur Rahman Crescent Institute of Science & Technology

Deemed to be University u/s 3 of the UGC Act, 1956

DEPARTMENT OF COMPUTER APPLICATIONS

ACADEMIC YEAR (JULY 2024 - DECEMBER 2024)

COURSE CODE : CADX118

COURSE NAME : CLOUD SERVICES FOR DATA SCIENCE LABORATORY

PROGRAMME : BCA (DATA SCIENCE)

SEMESTER : V

B.S. Abdur Rahman Crescent Institute of Science & Technology

Deemed to be University u/s 3 of the UGC Act, 1956

BONAFIDE CERTIFICATE

This is a Certified Record Book of _____

RRN: _____ **submitted for the Semester End Practical Examination held on** _____, **for the CADX118 - CLOUD SERVICES FOR DATA SCIENCE LABORATORY during 2024 - 2025.**

.....

Signature of Faculty

INDEX

Ex.No.	Date	List of Experiments	Page No	Signature
1.		Create an account in amazon web services (AWS)		
2.		Explore the storage space of your EC2 instance using Amazon EBS.		
3.		Website Launching		
4.		Create a student attendance maintenance system using AWS.		
5.		Develop the cloud-enabled Elastic File System.		
6.		Develop the mass e-mailing using AWS Lambda.		
7.		Explore a static website with S3 and CloudFront.		
8.		Create a Text-to-speech Conversion with Sage Maker.		
9.		Implement the N vision enables machine learning services for text extraction from images.		
10.		Create a chatbot applications using Artificial Intelligence (AI) in AWS.		

EX.NO : 1

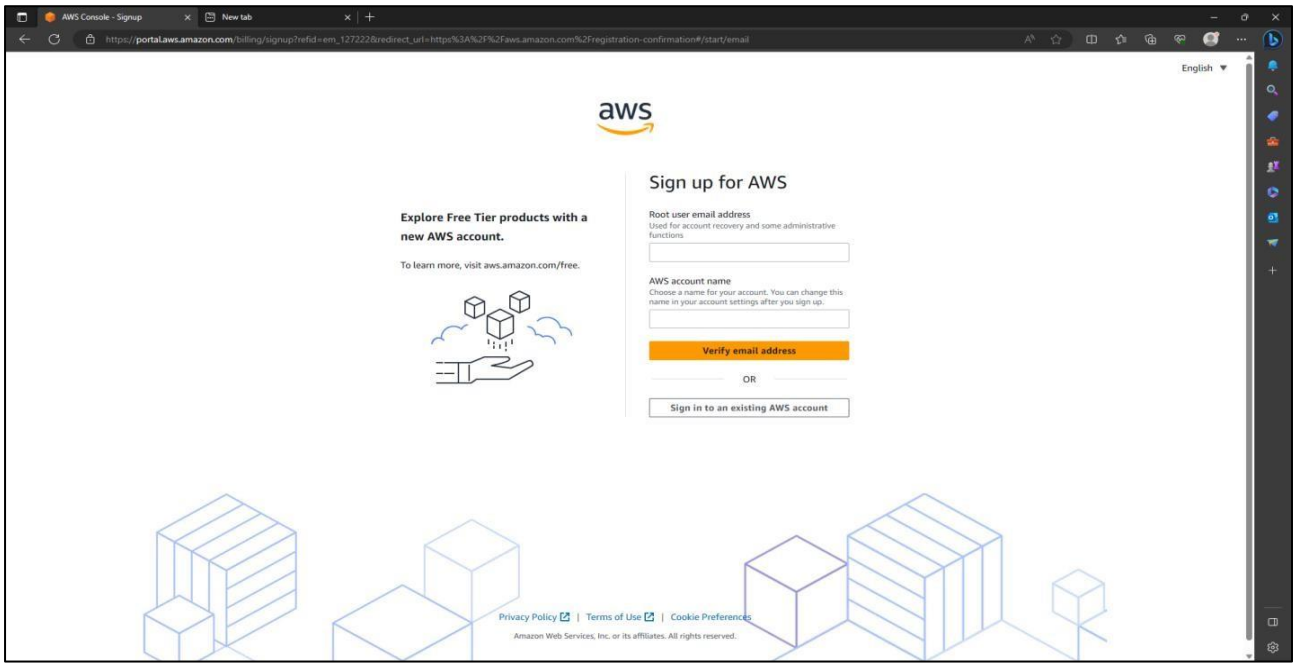
DATE :

CREATE AN ACCOUNT IN AMAZON WEB SERVICES (AWS)

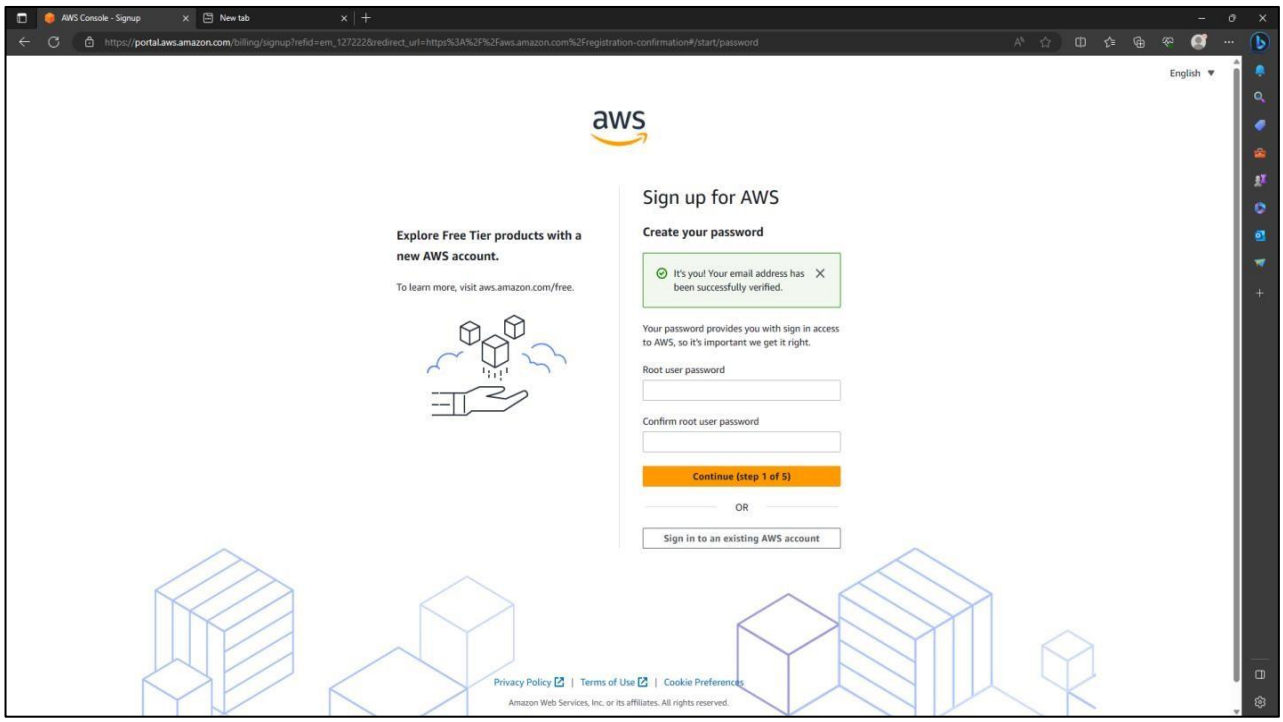
Step 1: Go to web browser and type the following URL:

https://aws.amazon.com/free/?all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc&awsf.Free%20Tier%20Types=*all&awsf.Free%20Tier%20Categories=*all

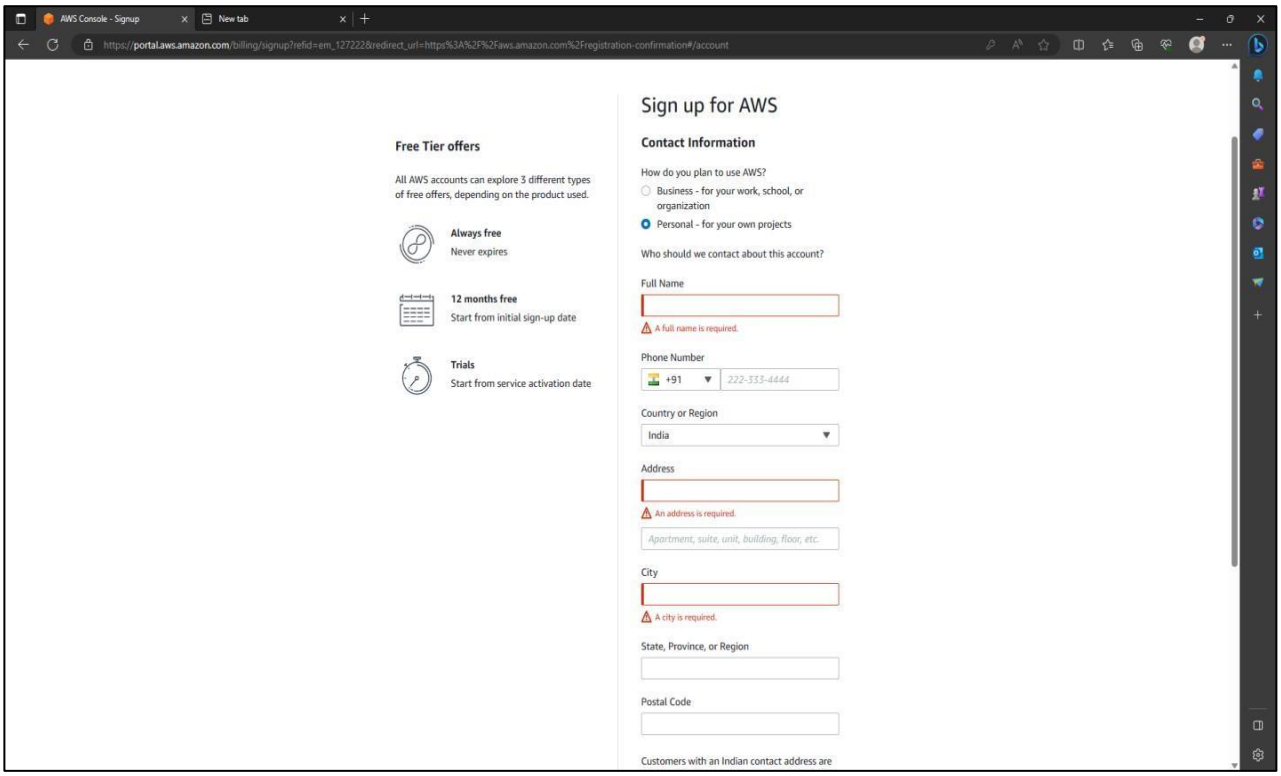
Step 2: Enter your email id and the verification code that has been sent to your account



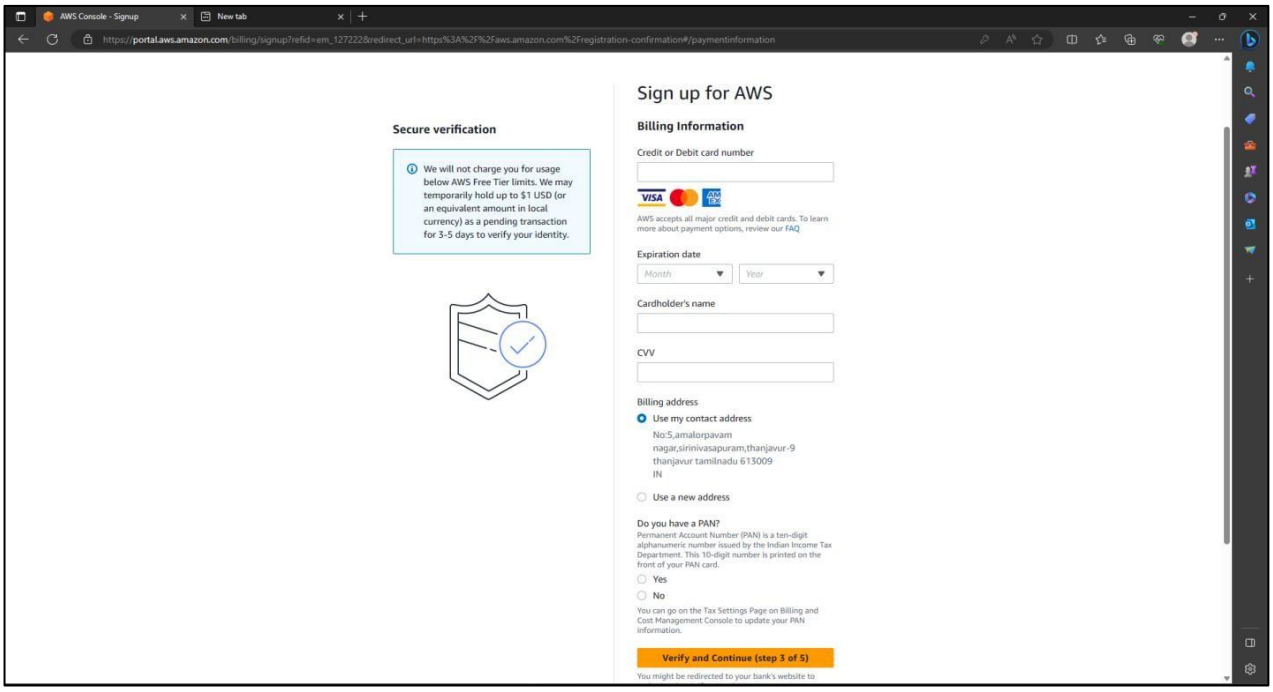
Step 3: Now enter a new password for your AWS account



Step 4: Now enter the basic detail for the creation of AWS account



Step 5: Enter the bank details for creating AWS account



Step 6: Select the free tier option and click next and then enter the verification code that has been sent to your email id.

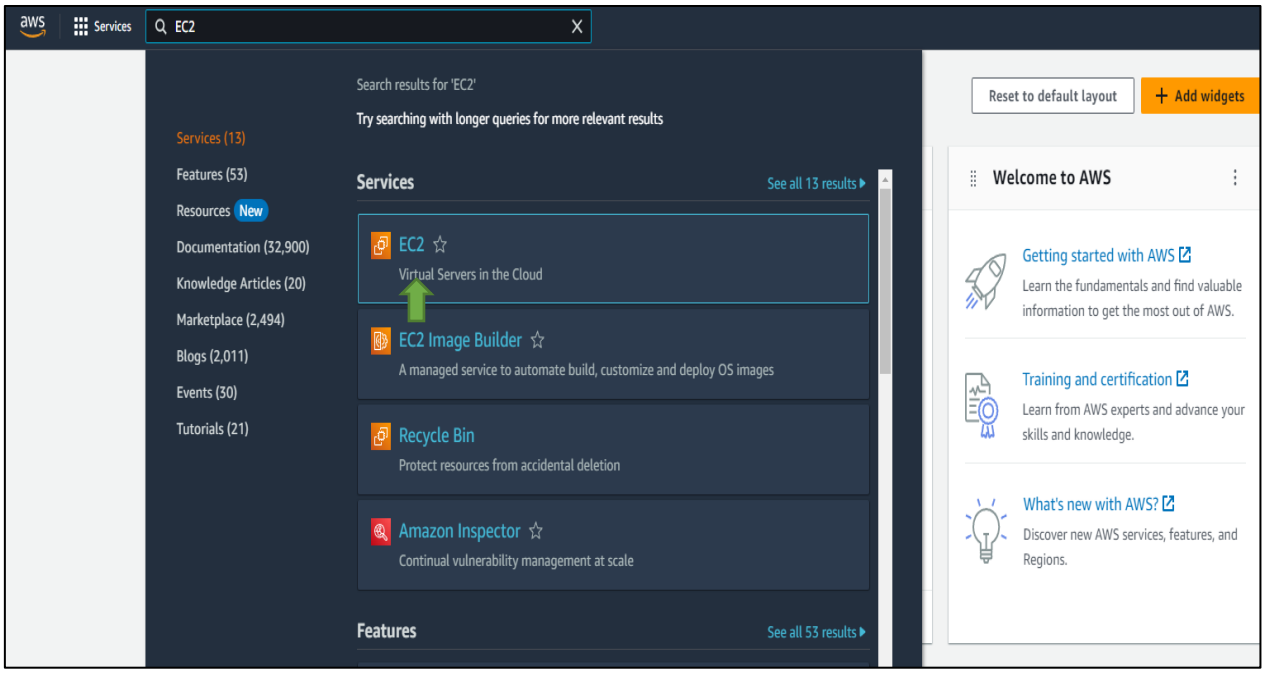
Now you are all set!

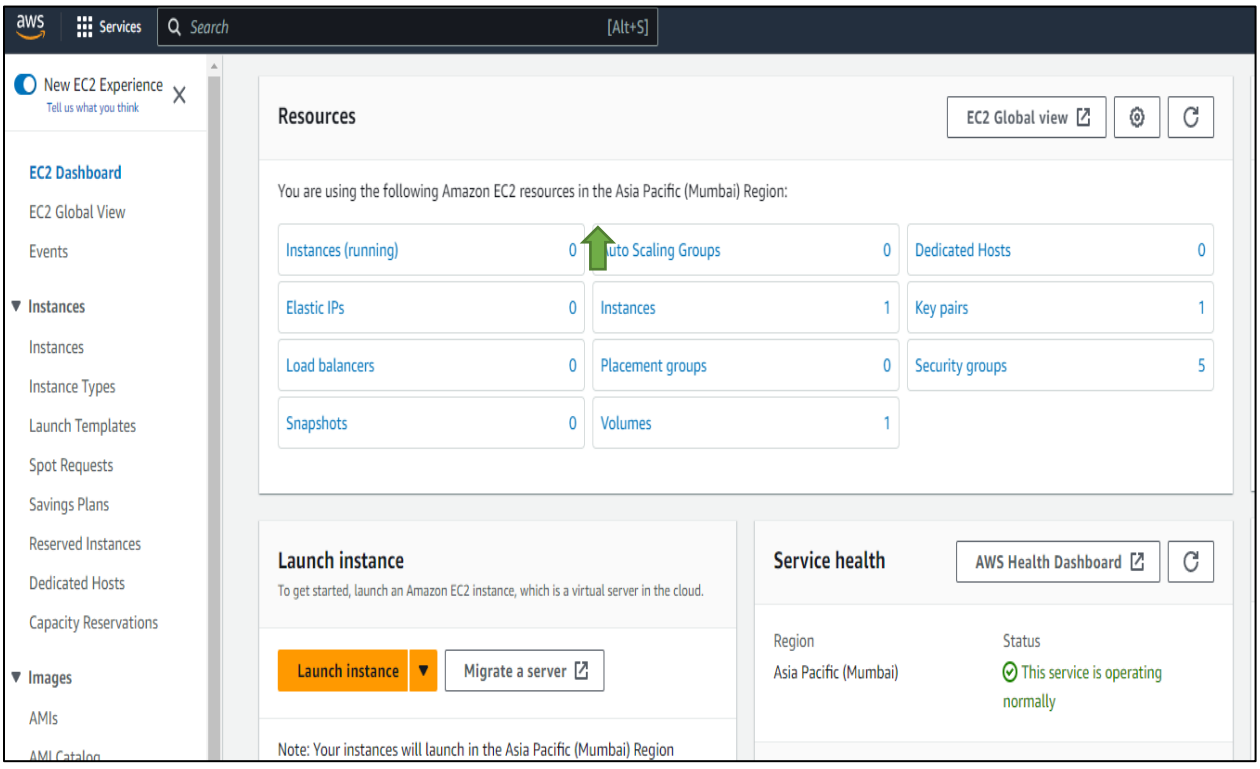
EX.NO : 2

DATE :

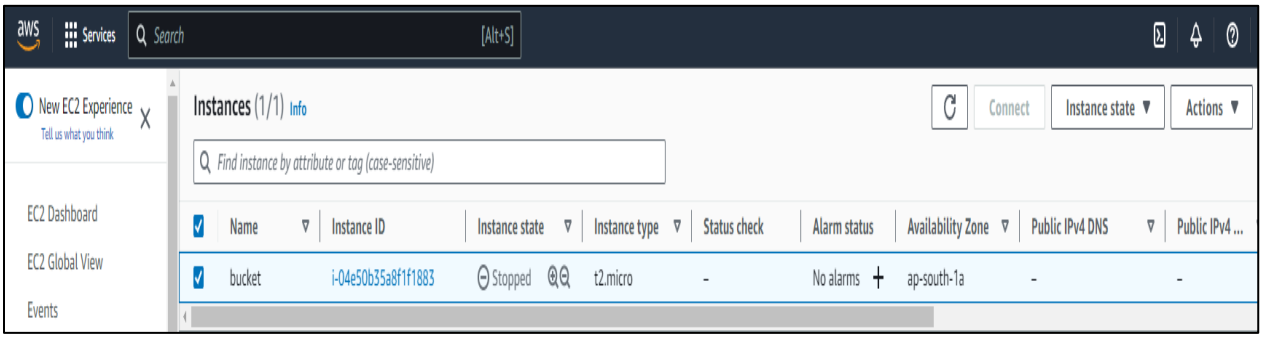
EXPLORE THE STORAGE SPACE OF YOUR EC2 INSTANCE USING AMAZON EBS.

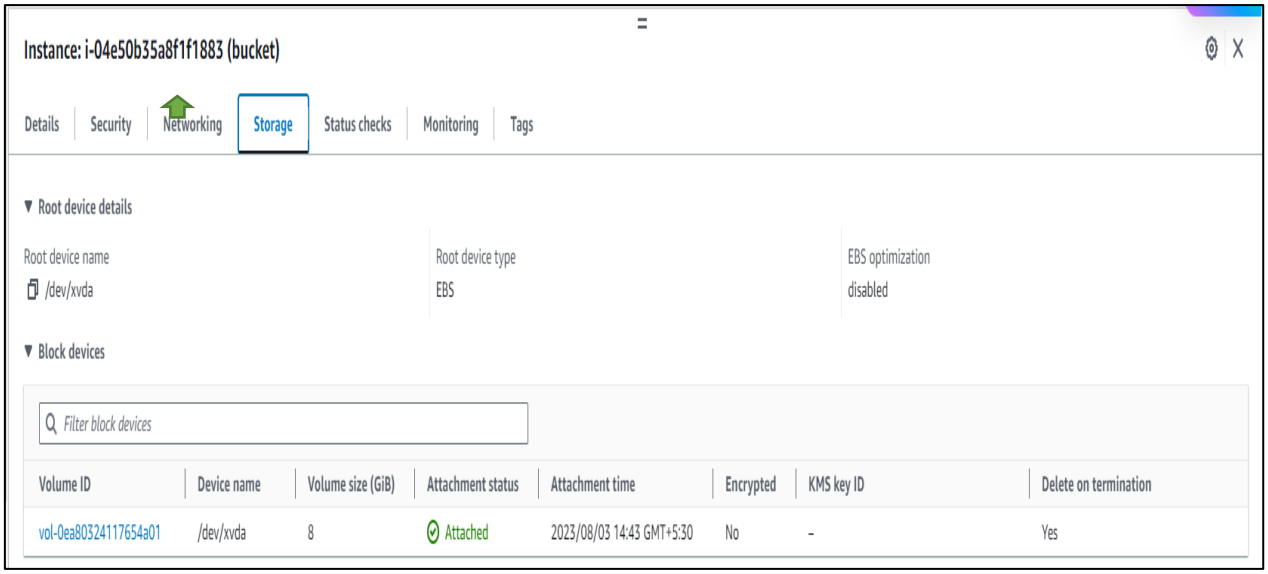
Step 1: Log-in to your AWS account and go to EC2 instance dashboard and view your instance



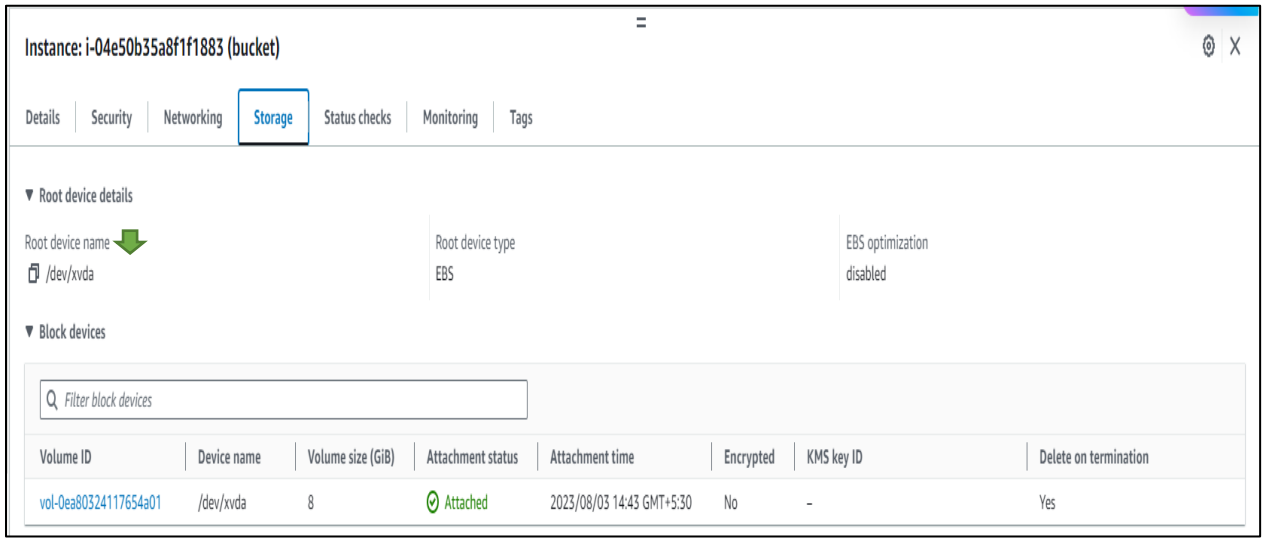


Step 2: Now select the instance you want to explore and go to storage option in details

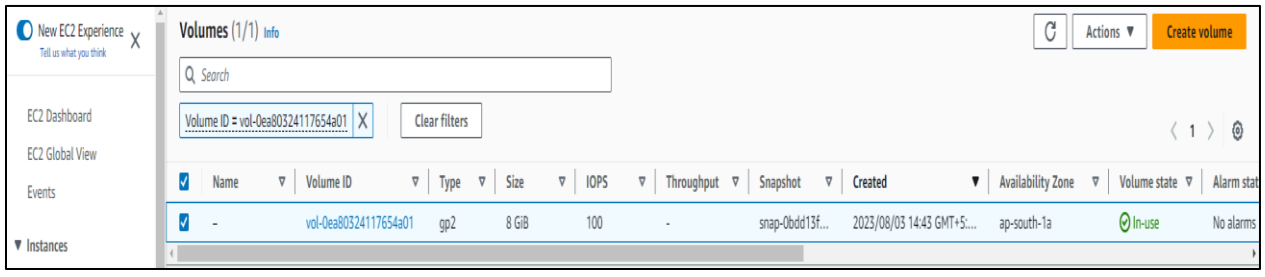




Step 3: Now click the volume ID to open the EBS volume dashboard



Step 4: Now select the volume to view its details



Volume ID: vol-0ea80324117654a01

Details | Status checks | Monitoring | Tags

Volume ID vol-0ea80324117654a01	Size 8 GiB	Type gp2	Volume status Okay
AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. | Learn more	Volume state In-use	IOPS 100	Throughput -
Encryption Not encrypted	KMS key ID -	KMS key alias -	KMS key ARN -
Fast snapshot restored No	Snapshot snap-0bdd13fb26cf68c1c	Availability Zone ap-south-1a	Created Thu Aug 03 2023 14:43:43 GMT+0530 (India Standard Time)
Multi-Attach enabled No	Attached Instances i-04e50b35a8f1f1883 (bucket): /dev/xvda (attached)	Outposts ARN -	

Volume ID: vol-0ea80324117654a01

Details | Status checks | Monitoring | Tags

Volume status Okay	Availability Zone ap-south-1a
I/O status Enabled	I/O performance Not applicable
I/O status updated on Thu Aug 03 2023 14:43:43 GMT+0530 (India Standard Time)	I/O performance updated on -
Description -	Description This feature only applies to attached io1, io2, and gp3 volumes.
Auto-enabled I/O Enabled	

For more information about working with volume status checks and events, see Monitor the status of your volumes in the amazon EC2 user guide.
If you need technical assistance with your volume, post your issue to the Developer Forums or visit our Support Center.

Volume ID: vol-0ea80324117654a01

Details | Status checks | Monitoring | Tags

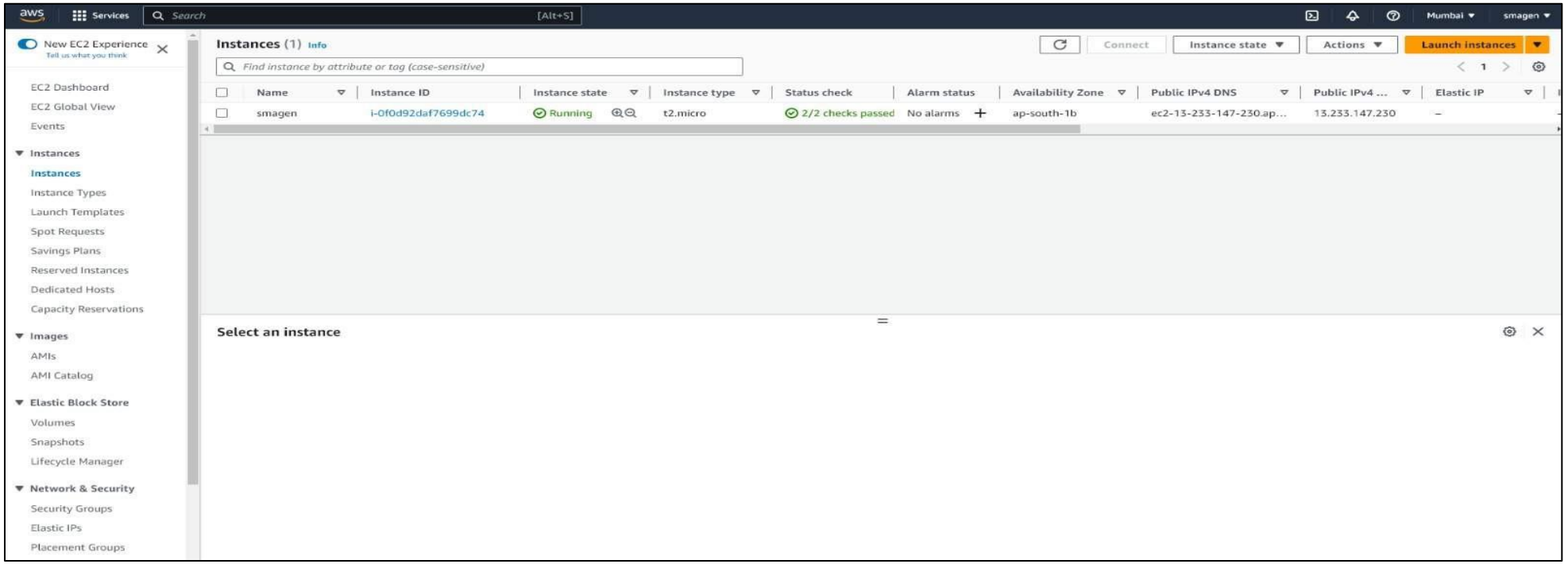
1h 3h 12h 1d 3d 1w Custom | Add to dashboard

EX.NO : 3

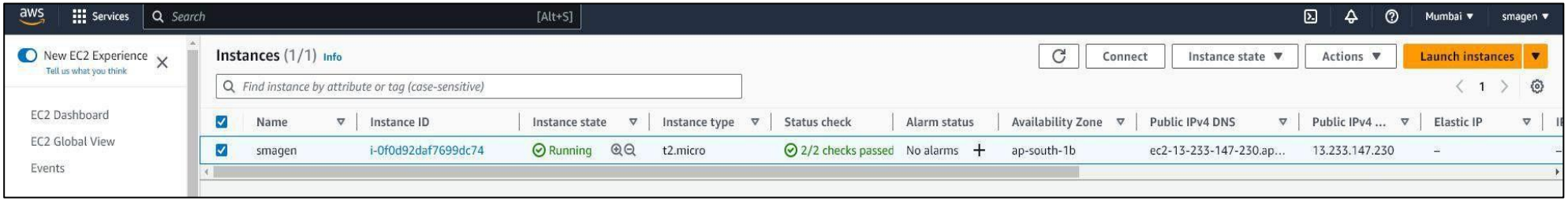
DATE :

WEBSITE LAUNCHING

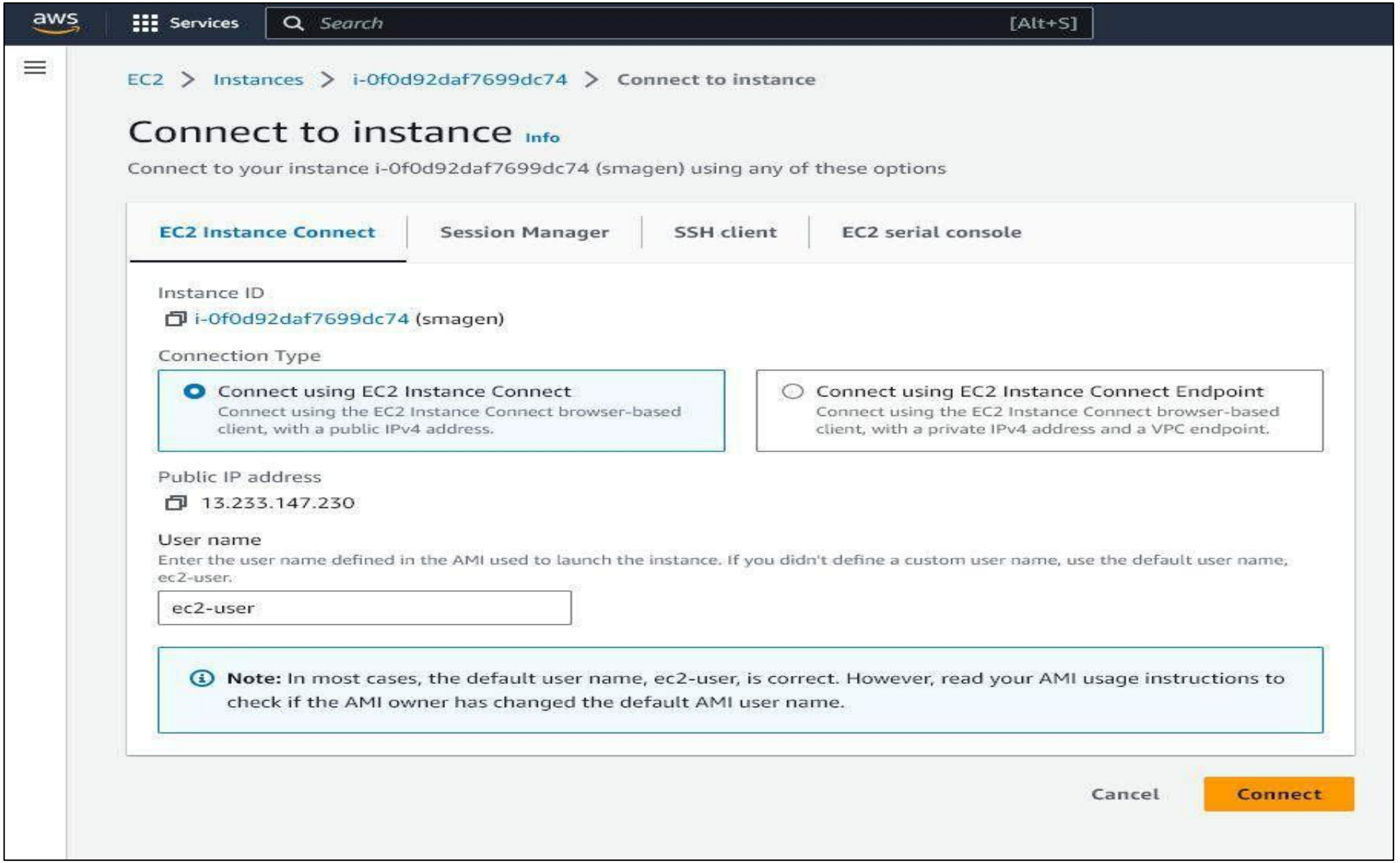
Step 1: Select the instance you created



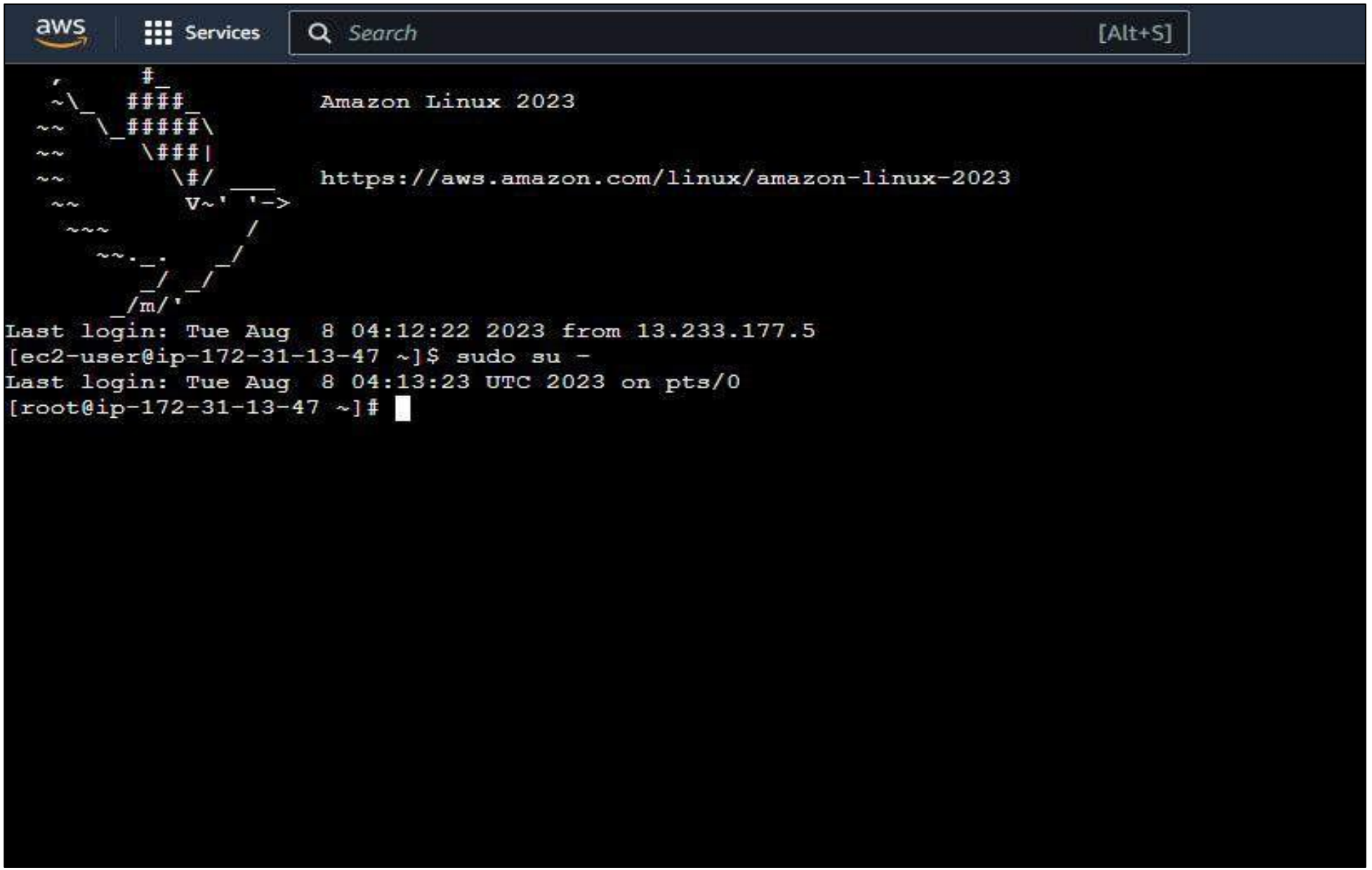
Step 2: Click the connect option



Step 3: Connecting to the instance " **click the connect option** "



Step 4 : Type "sudo su -"



Step 5 : Installing the httpd using this command "**yum install httpd**"

```
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Tue Aug  8 04:12:22 2023 from 13.233.177.5
[ec2-user@ip-172-31-13-47 ~]$ sudo su -
Last login: Tue Aug  8 04:13:23 UTC 2023 on pts/0
[root@ip-172-31-13-47 ~]# yum update -y
Last metadata expiration check: 0:33:20 ago on Tue Aug  8 04:11:36 2023.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-13-47 ~]# yum install httpd
Last metadata expiration check: 0:34:03 ago on Tue Aug  8 04:11:36 2023.
Dependencies resolved.
================================================================================
 Package                 Architecture        Version                 Repository      Size
================================================================================
Installing:
 httpd                   x86_64              2.4.56-1.amzn2023       amazonlinux     48 k
Installing dependencies:
 apr                     x86_64              1.7.2-2.amzn2023.0.2    amazonlinux    129 k
 apr-util                x86_64              1.6.3-1.amzn2023.0.1    amazonlinux     98 k
 generic-logos-httpd     noarch              18.0.0-12.amzn2023.0.3  amazonlinux     19 k
 httpd-core              x86_64              2.4.56-1.amzn2023       amazonlinux    1.4 M
 httpd-filesystem        noarch              2.4.56-1.amzn2023       amazonlinux     15 k
 httpd-tools             x86_64              2.4.56-1.amzn2023       amazonlinux     82 k
 libbrotli               x86_64              1.0.9-4.amzn2023.0.2    amazonlinux    315 k
 mailcap                 noarch              2.1.49-3.amzn2023.0.3   amazonlinux     33 k
Installing weak dependencies:
 apr-util-openssl        x86_64              1.6.3-1.amzn2023.0.1    amazonlinux     17 k
 mod_http2               x86_64              2.0.11-2.amzn2023       amazonlinux    150 k
 mod_lua                 x86_64              2.4.56-1.amzn2023       amazonlinux     62 k

Transaction Summary
================================================================================
Install  12 Packages

Total download size: 2.3 M
Installed size: 6.9 M
Is this ok [y/N]:
```

Step 6 : Checking the status of Apache http server using this command " **systemctl status httpd** "

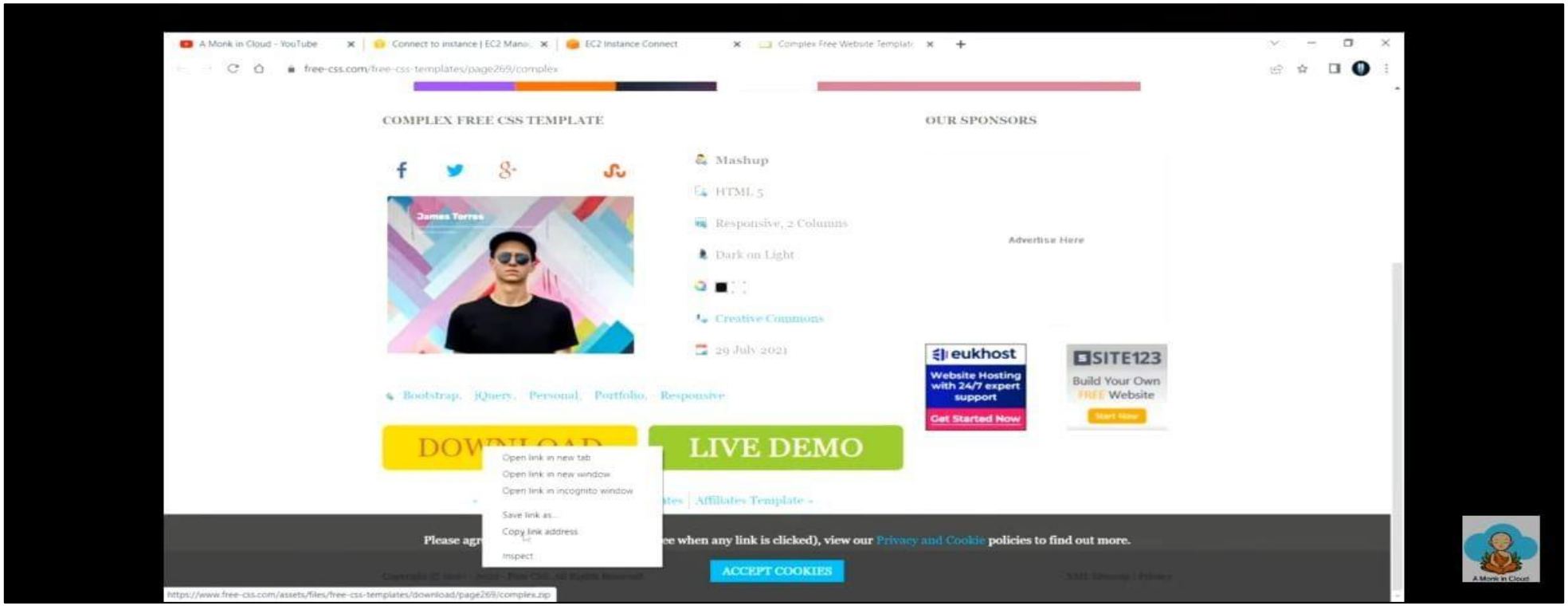
```
[root@ip-172-31-13-47 ~]# systemctl status httpd
○ httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
     Active: inactive (dead)
       Docs: man:httpd.service(8)
[root@ip-172-31-13-47 ~]#
```

Step 7 : creating directory name as temp " **mkdir temp** "

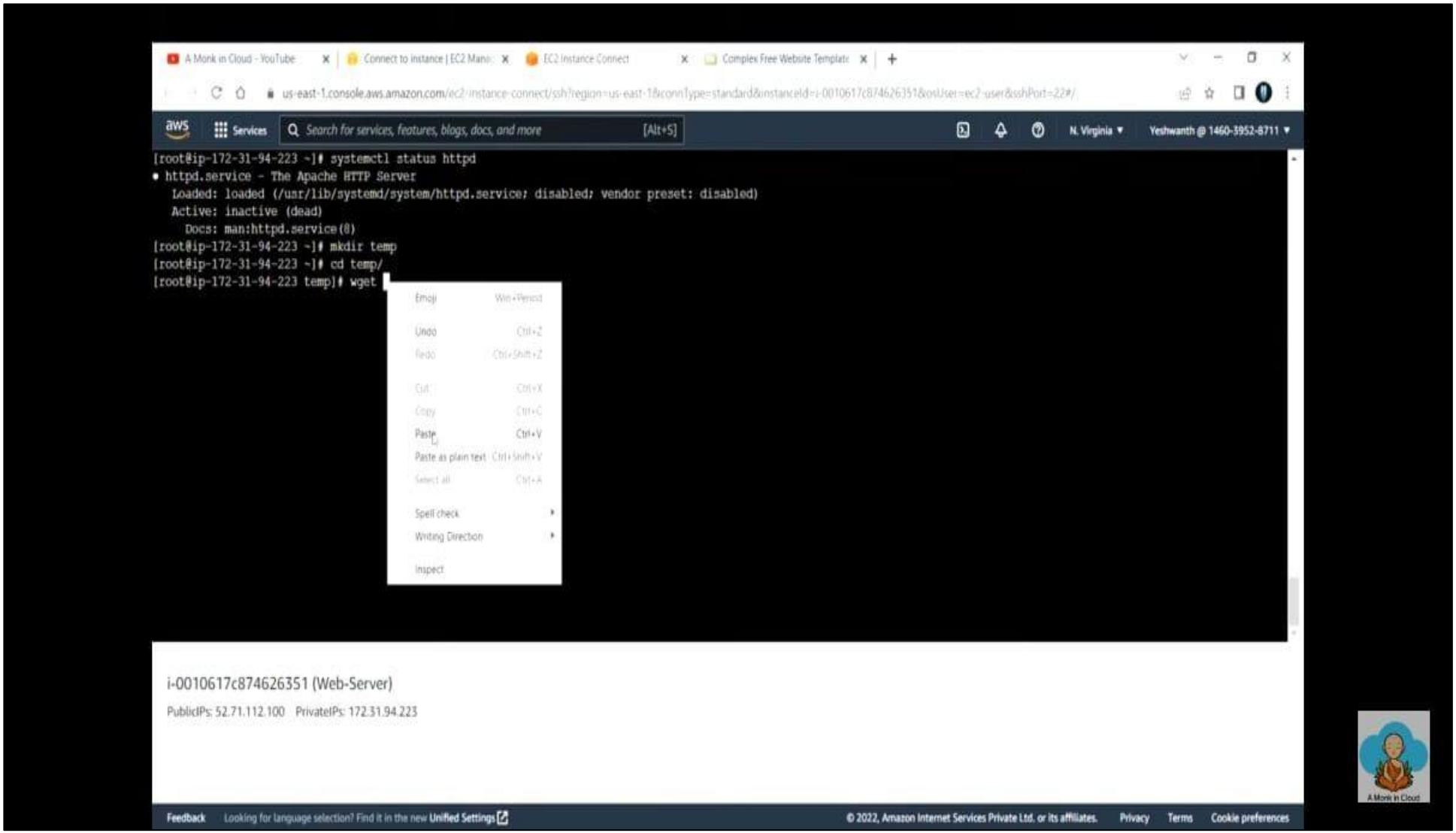
Step 8 : using cd command changing the directory " **cd temp** "

```
[root@ip-172-31-13-47 ~]# mkdir temp
[root@ip-172-31-13-47 ~]# cd temp
[root@ip-172-31-13-47 temp]# wget https://www.free-css.com/assets/files/free-css-templates/download/page294/woody.zip
--2023-08-08 04:49:56--  https://www.free-css.com/assets/files/free-css-templates/download/page294/woody.zip
Resolving www.free-css.com (www.free-css.com)... 217.160.0.242, 2001:8d8:100f:f000::28f
Connecting to www.free-css.com (www.free-css.com)|217.160.0.242|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 961484 (939K) [application/zip]
Saving to: 'woody.zip'

woody.zip                                   100%[=========================================================================================================================>] 938.95K  1.17MB/s    in 0.8s

2023-08-08 04:49:57 (1.17 MB/s) - 'woody.zip' saved [961484/961484]

[root@ip-172-31-13-47 temp]#
```

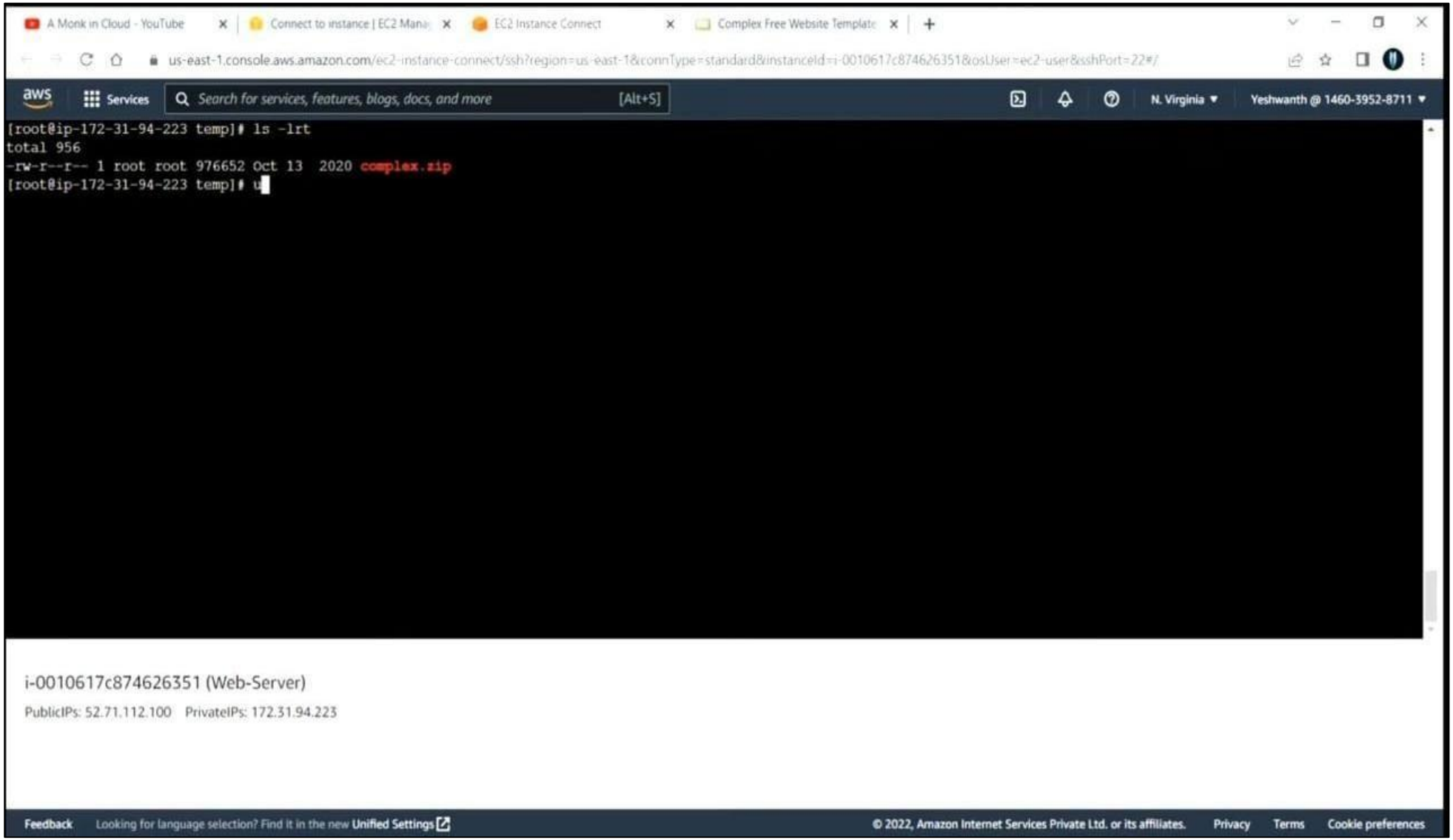
Step 9 : Copy the link address of the css template website



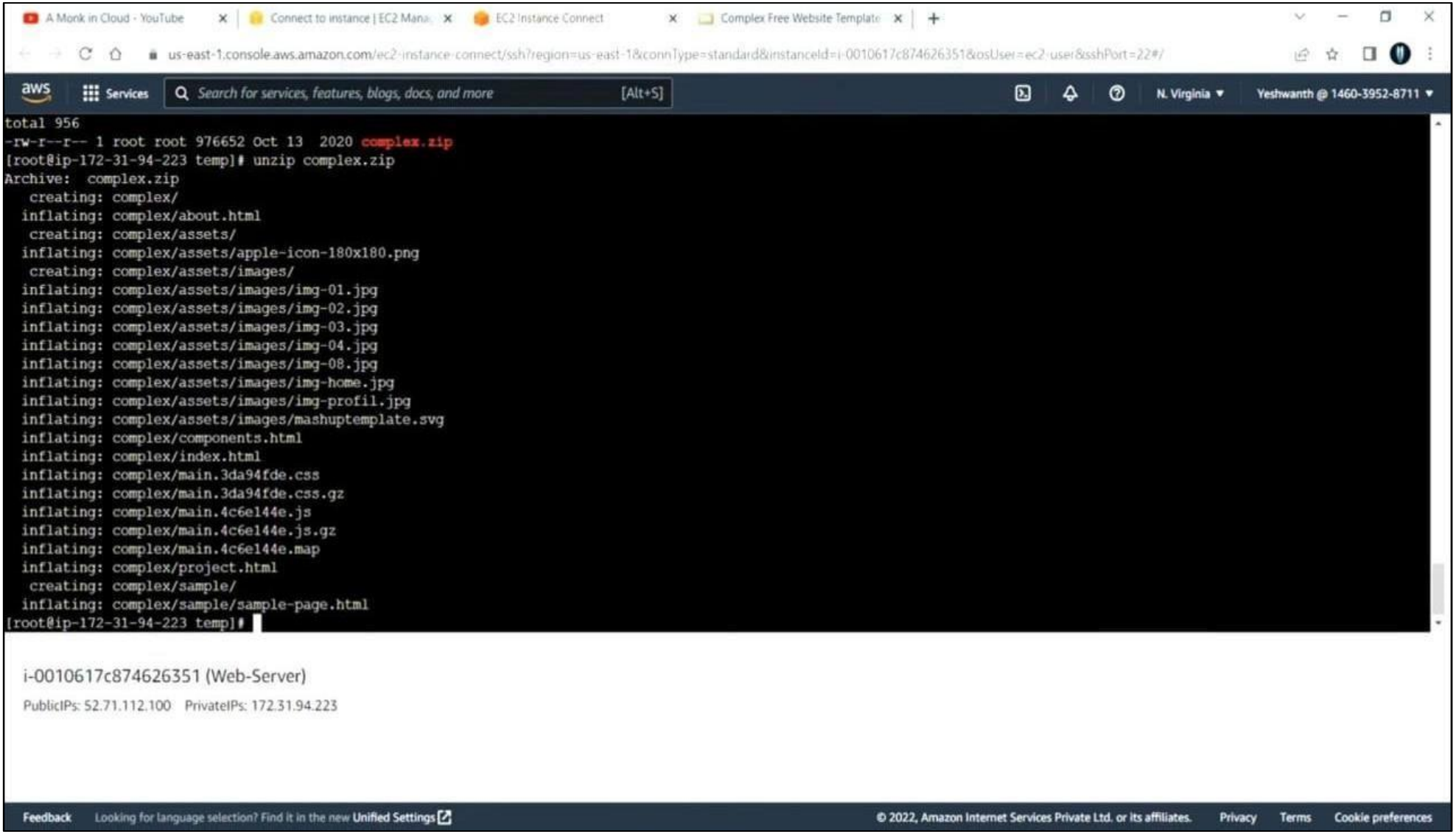
Step 10 : Download the complex file using this command " **wget (paste the link)** "



Step 11 : To view the file use " **ls -lrt** "



Step 12 : Unzip the complex file using this command " **unzip complex.zip** "

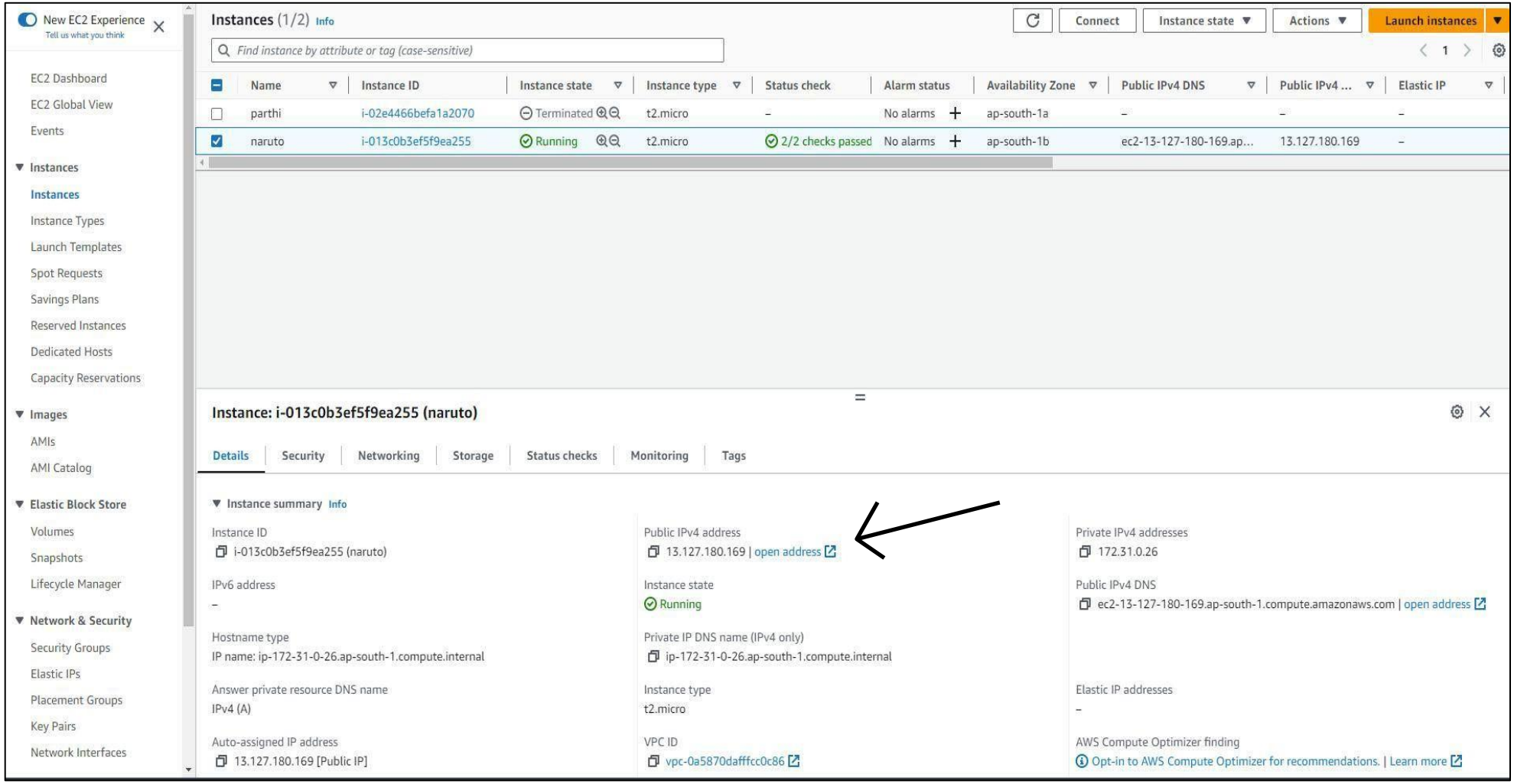


Step 13: Move the file to html using this commad " **mv * /var/www/html/** "

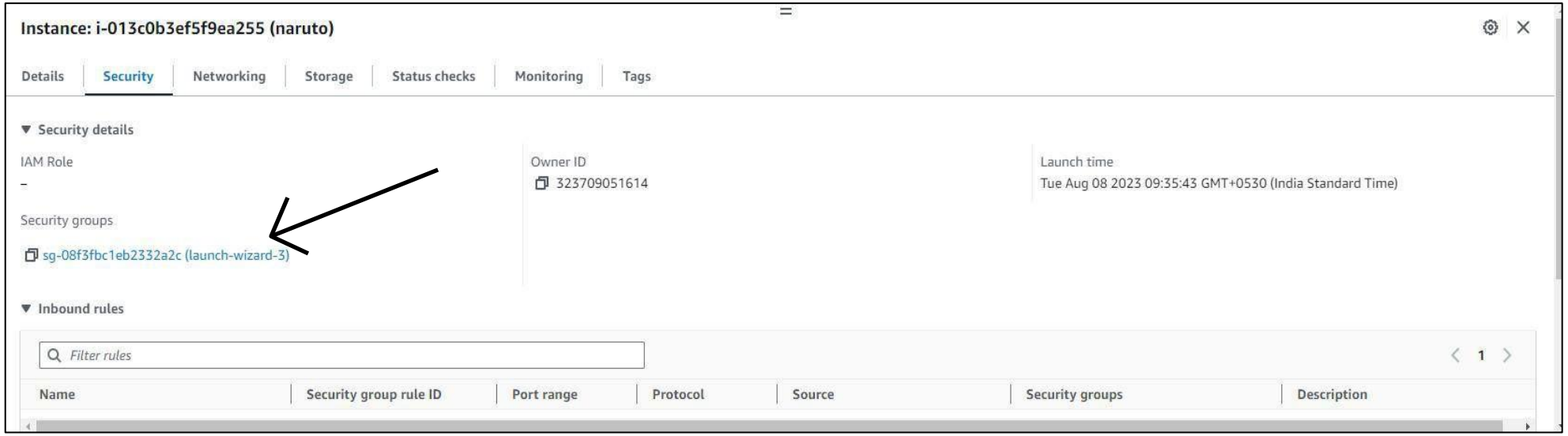
Step 14: " **cd /var/www/html/** "

```
[root@ip-172-31-0-26 complex]# mv * /var/www/html/
[root@ip-172-31-0-26 complex]# cd /var/www/html/
[root@ip-172-31-0-26 html]# ls -lrt
total 424
drwxr-xr-x. 2 root root     30 Aug 28  2017 sample
-rw-r--r--. 1 root root   7122 Aug 28  2017 project.html
-rw-r--r--. 1 root root 111625 Aug 28  2017 main.4c6e144e.map
-rw-r--r--. 1 root root  44095 Aug 28  2017 main.4c6e144e.js.gz
-rw-r--r--. 1 root root 138039 Aug 28  2017 main.4c6e144e.js
-rw-r--r--. 1 root root  14588 Aug 28  2017 main.3da94fde.css.gz
-rw-r--r--. 1 root root  84357 Aug 28  2017 main.3da94fde.css
-rw-r--r--. 1 root root   3739 Aug 28  2017 index.html
-rw-r--r--. 1 root root  10249 Aug 28  2017 components.html
drwxr-xr-x. 3 root root     50 Aug 28  2017 assets
-rw-r--r--. 1 root root   5097 Aug 28  2017 about.html
[root@ip-172-31-0-26 html]#
```

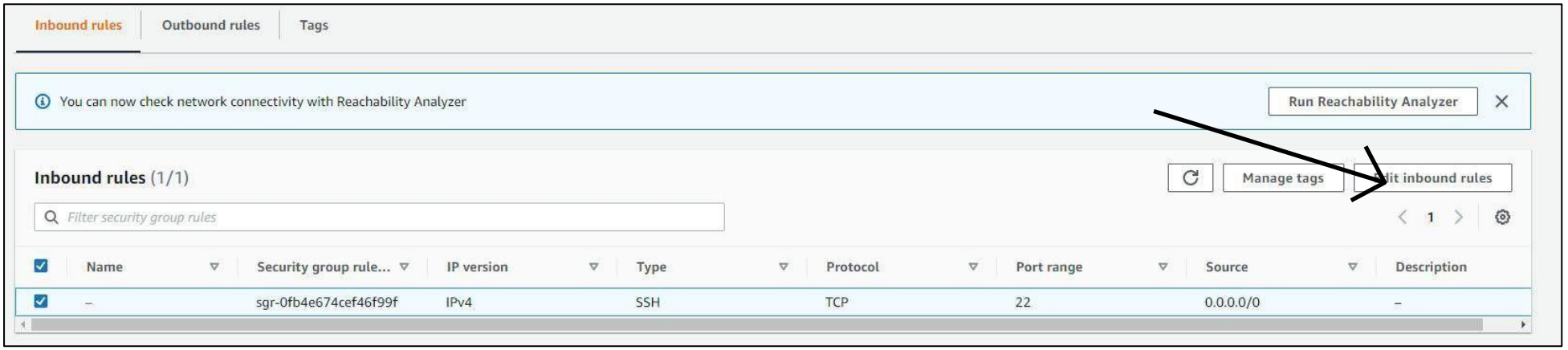
Step 15: Copy the public address and paste in the google



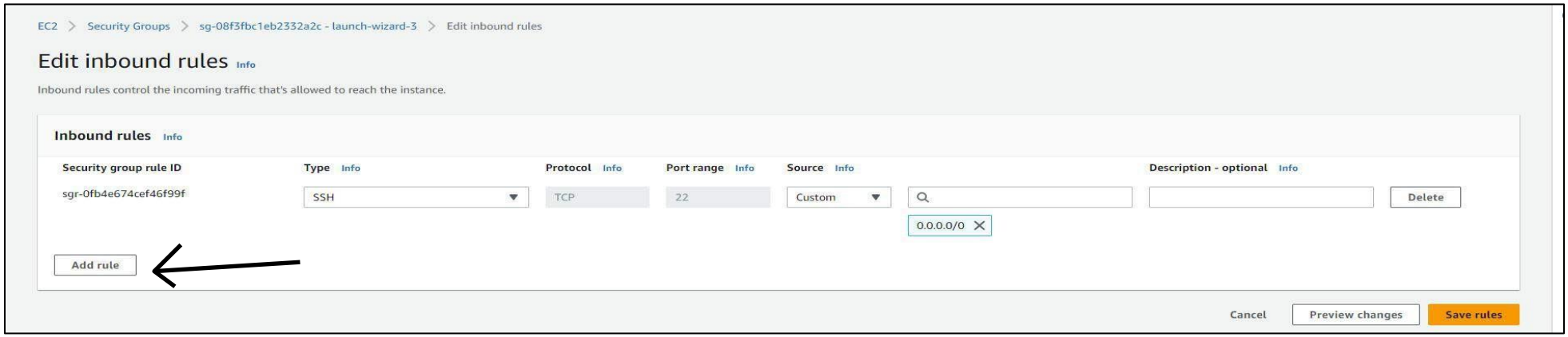
Step 16: Click security option and the security groups link



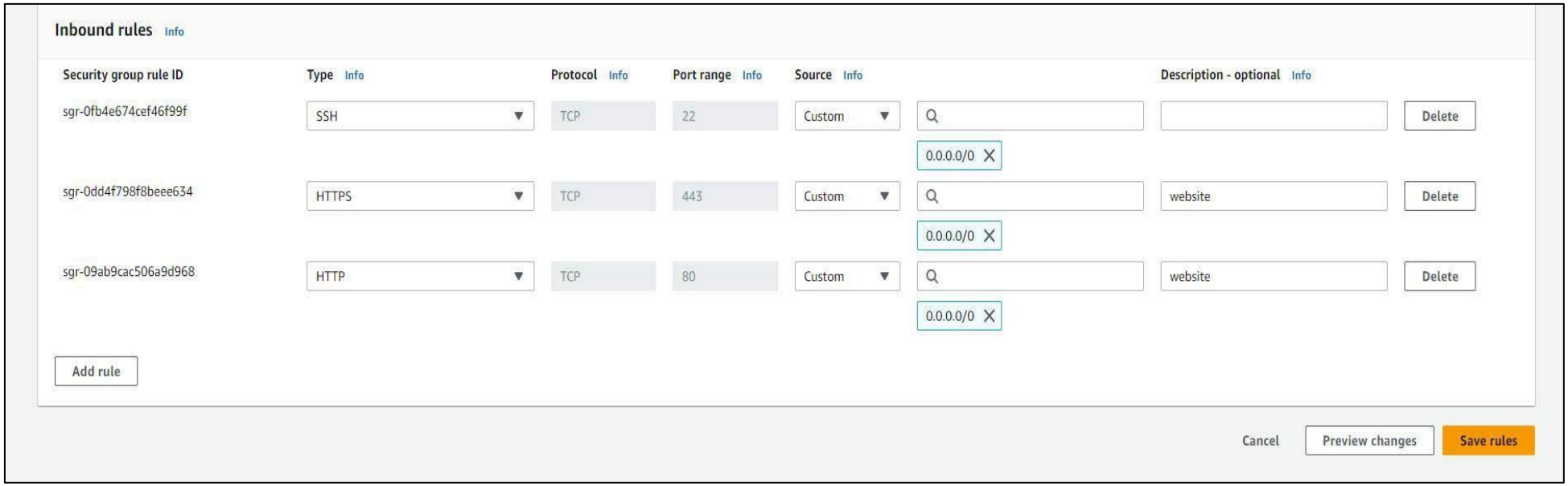
Step 17: click the edit inbound rules



Step 18: Click add rule

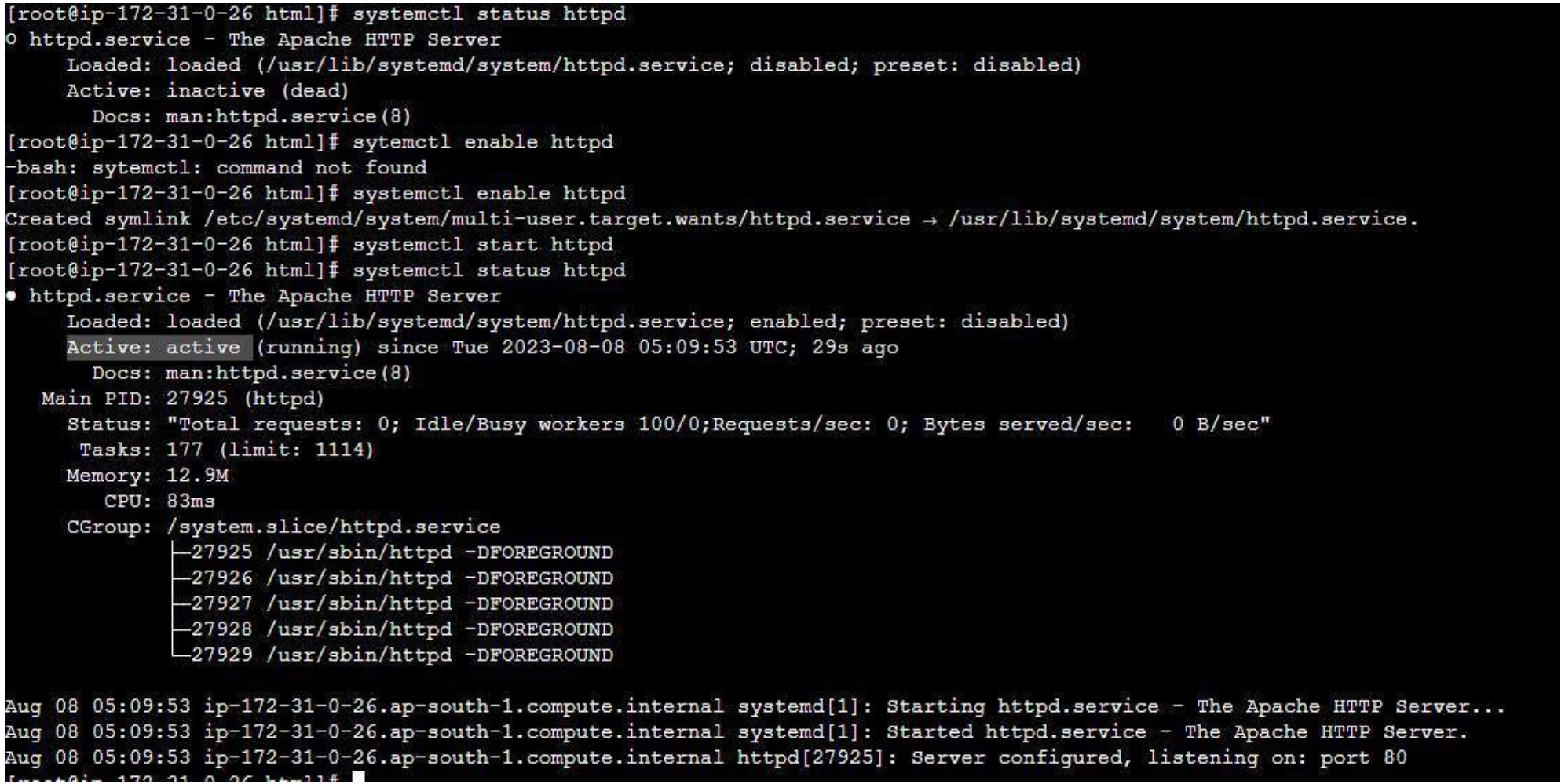


Step 19: Add two rules for " **HTTP** " and " **HTTPS** "

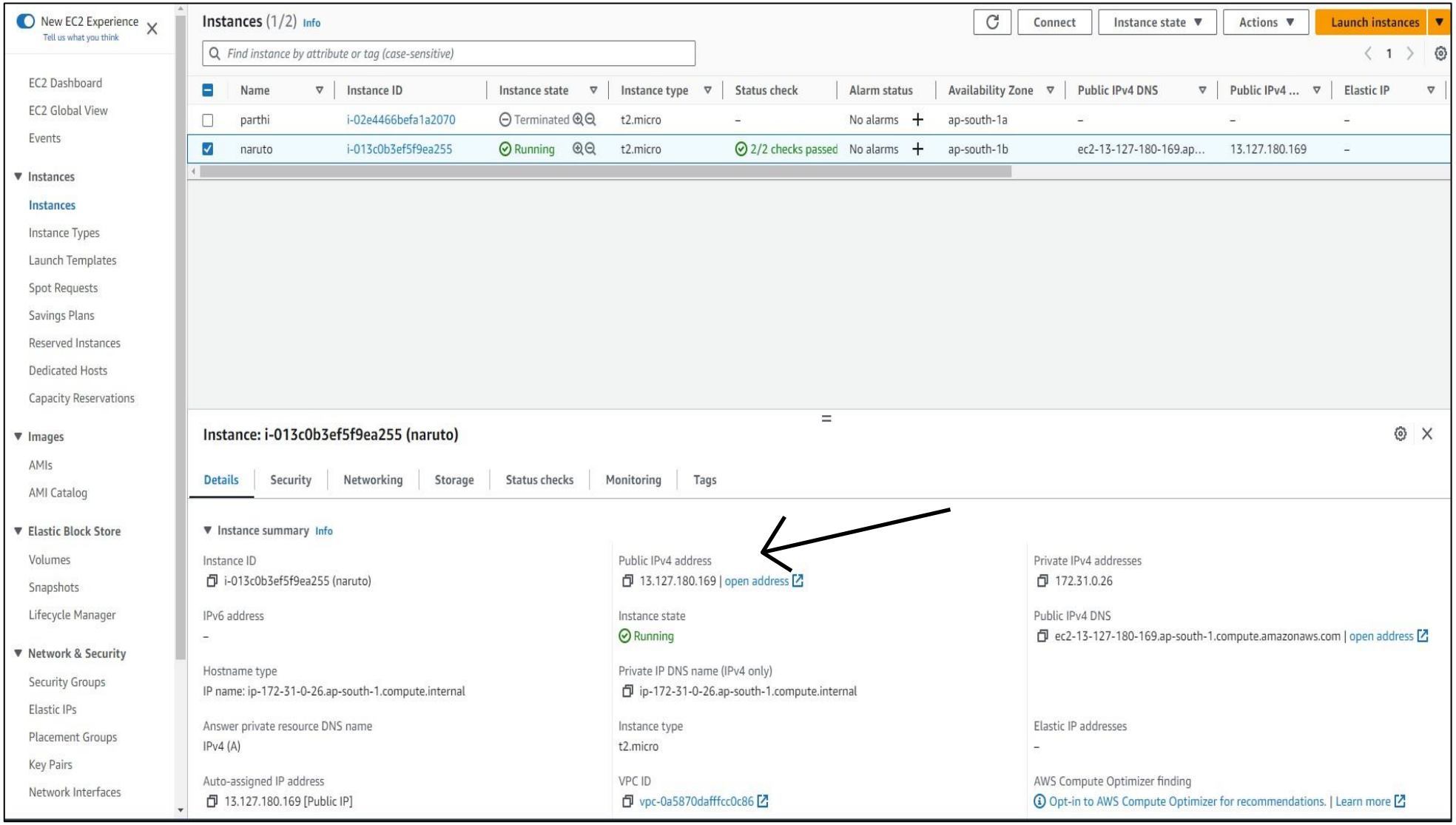


Step 20: Active the httpd with this command " **systemctl enable httpd** "

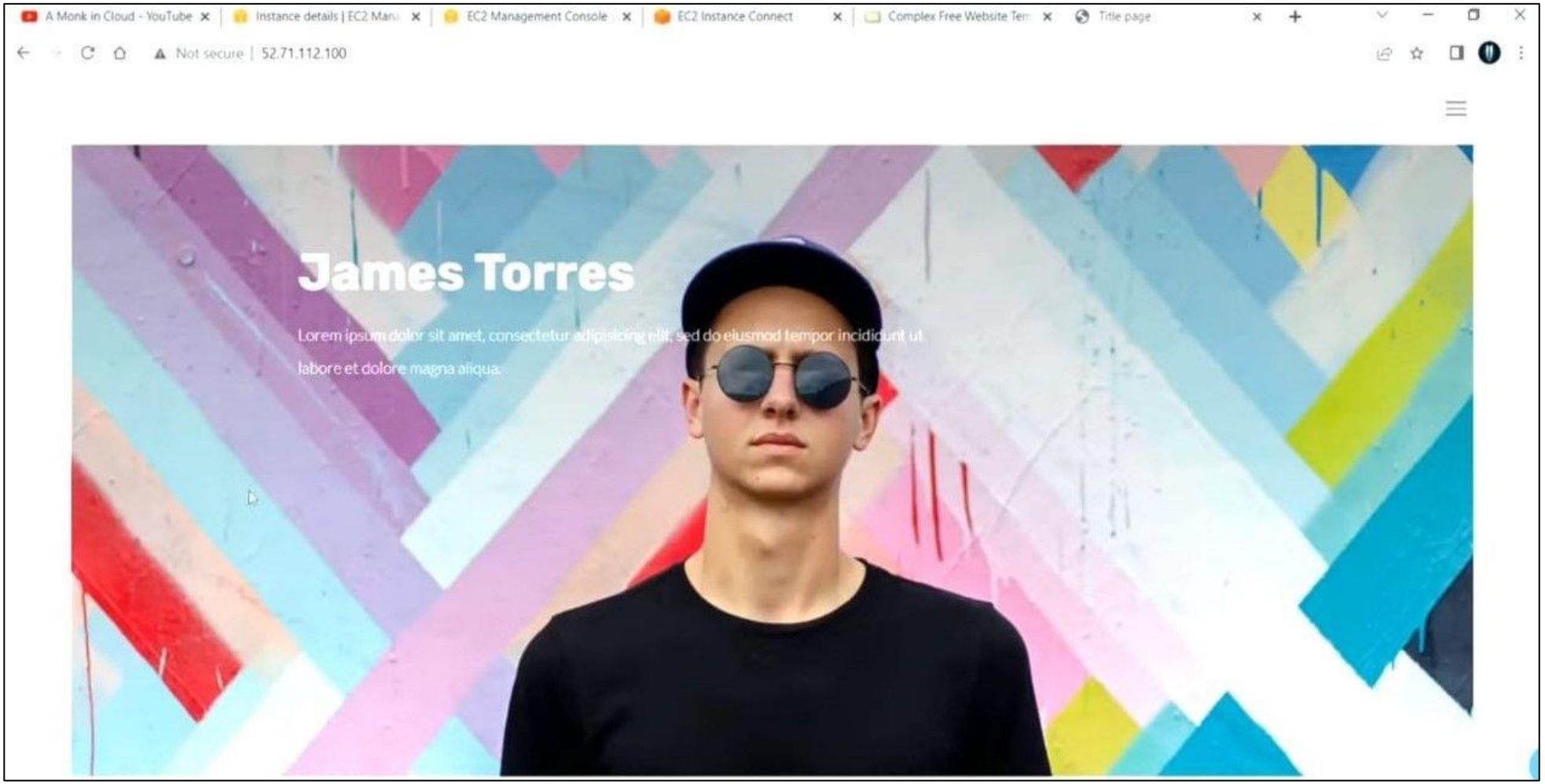
Step 21: Start the httpd with this command " **systemctl start httpd** "



Step 22: Copy the public address



Step 23: Paste in the google website



EX.NO : 4

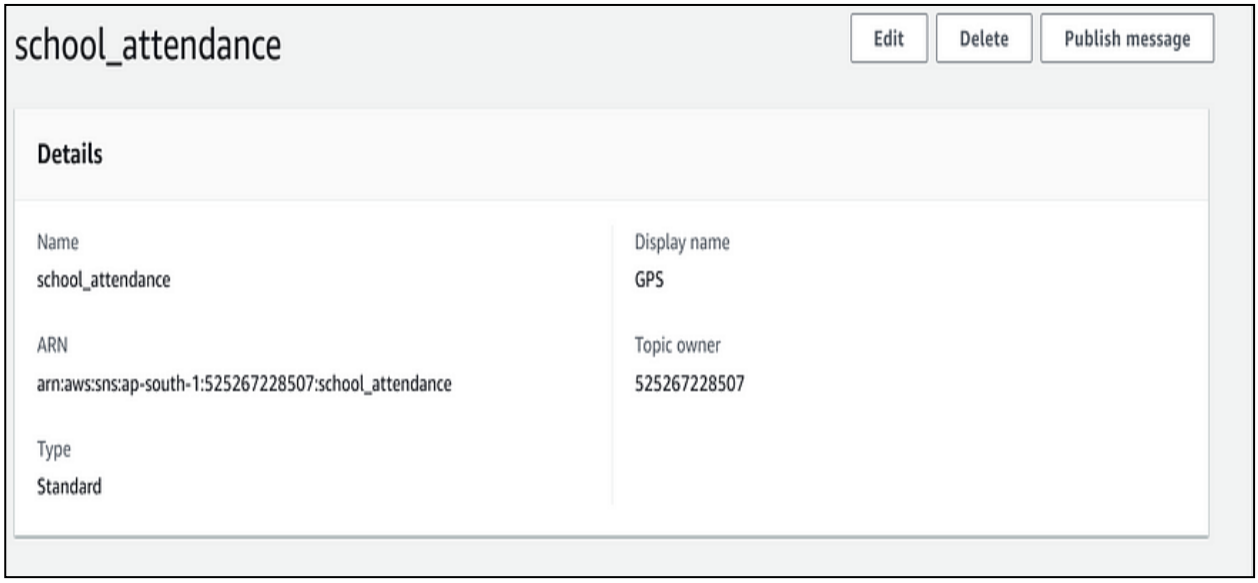
DATE :

CREATE A STUDENT ATTENDANCE MAINTENANCE SYSTEM USING AWS

Step 1: A step-by-step process for building the school attendance notification system using AWS SNS and Lambda.

Step 2: Create an SNS Topic

- Log in to your AWS account and navigate to the AWS SNS console.
- Click on the “Create topic” button.
- Enter a name and display the display name for your topic.
- Click on the “Create topic” button to create the topic.



Step 3: Create an IAM Role for Lambda

- Navigate to the IAM console.
- Click on the “Roles” menu option and then click on the “Create role” button.
- Select “Lambda” as the service that will use this role.
- Choose the “AWSLambdaBasicExecutionRole” policy.
- Click on the “Next: Tags” button.
- (Optional) Add any relevant tags to your role.
- Click on the “Next: Review” button.
- Enter a name and description for your role.
- Click on the “Create role” button to create the role.

Step 4: Create the Lambda Function

- Navigate to the AWS Lambda console.
- Click on the “Create function” button.
- Choose “Author from scratch” and enter a name for your function.
- Select “Python 3.8” as the runtime for your function.
- Choose the IAM role you created in Step 2.
- Click on the “Create function” button to create your function.

CODE:

```
import boto3
import jsonpy

def lambda_handler(event, context):
    message = event['Records'][0]['Sns']['Message']
    phone_number = '<insert parent phone number here>'
    sns = boto3.client('sns')
    sns.publish(PhoneNumber=phone_number, Message=message)
    return {
        'statusCode': 200,
        'body': json.dumps('SMS notification sent successfully.')
    }
```

1. Replace <insert parent phone number here> with the phone number of the parent, you want to notify.
2. Click on the “Deploy” button to deploy your function.

Step 5: Create an SNS Topic and Subscription

After creating the topic, we need to create a subscription to the topic. This will allow the phone number to receive SMS notifications. Follow the steps below:

1. Click on the topic that you just created.
2. Click on the “Create subscription” button.
3. Choose “SMS” as the protocol.
4. Enter your phone number in the “Endpoint” field.
5. Click on the “Create subscription” button.

A verification code will be sent to the phone number that you provided. Enter the code in the “Subscription details” section to confirm the subscription.

Step 6: Test the Function

Now that we have created the Lambda function and SNS topic, it's time to test our function. Follow the steps below:

1. Click on the "Test" button in the Lambda function console.
2. Enter a name for the test event, for example, "test-event".
3. Copy the following JSON code and paste it into the "Input test event" section:

CODE:

```
{
  "Records": [
    {
      "Sns": {
        "Message": "{\"class\": \"Maths\", \"student\": \"John Doe\", \"status\": \"present\"}"
      }
    }
  ]
}
```

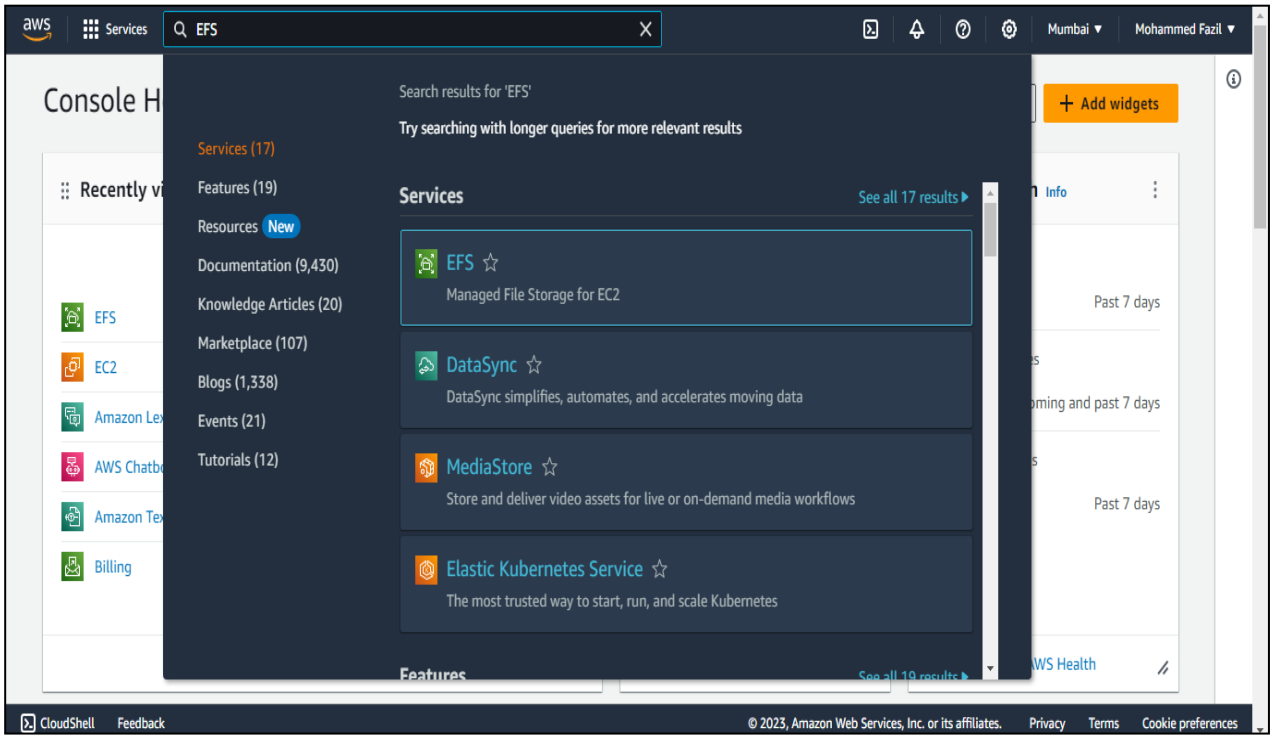
- Click on the "Create" button.
- Click on the "Test" button again.

EX.NO : 5

DATE :

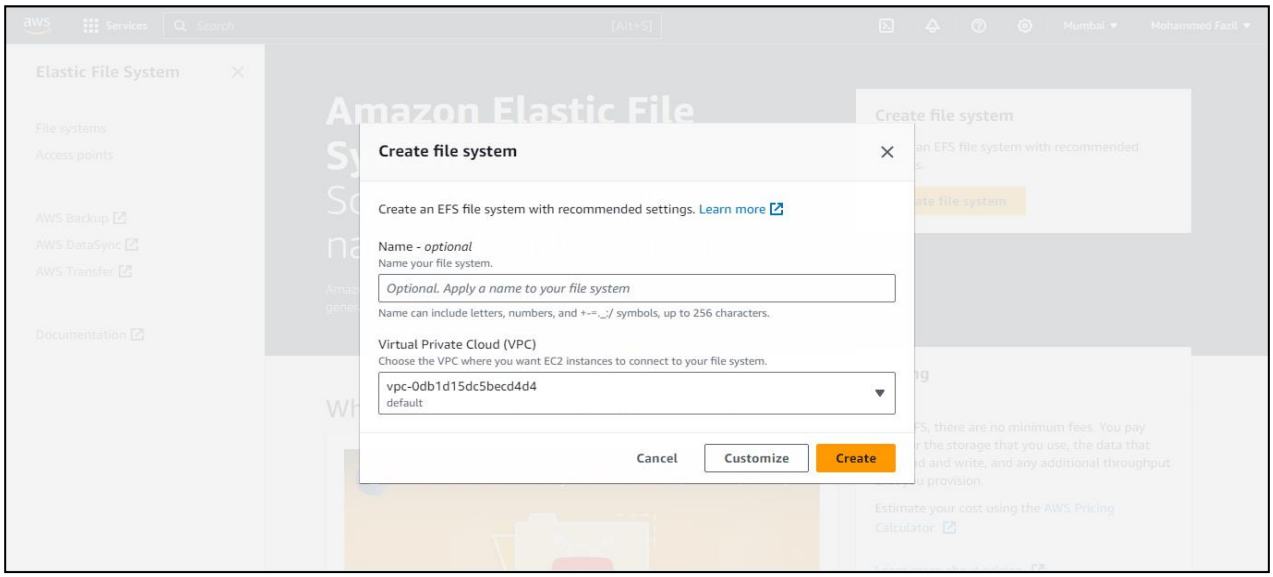
DEVELOP THE CLOUD - ENABLED ELASTIC FILE SYSTEM

Step 1: Log-In into your AWS account and search for EFS.

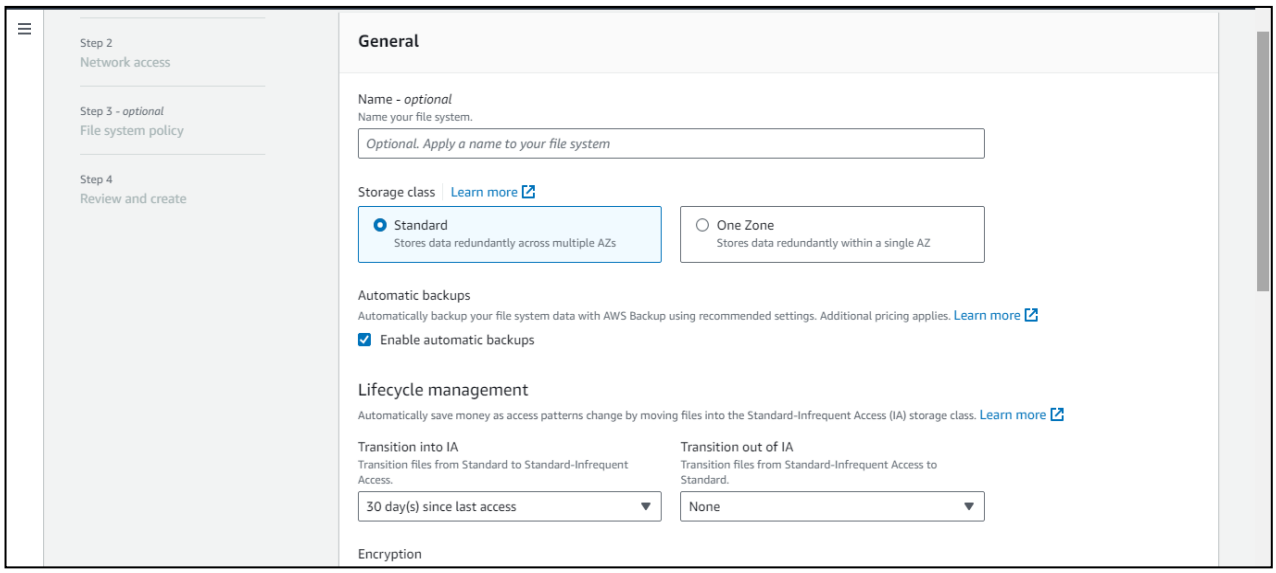


Step 2: Now click on EFS and then click on Create file system before that read the pricing details carefully.

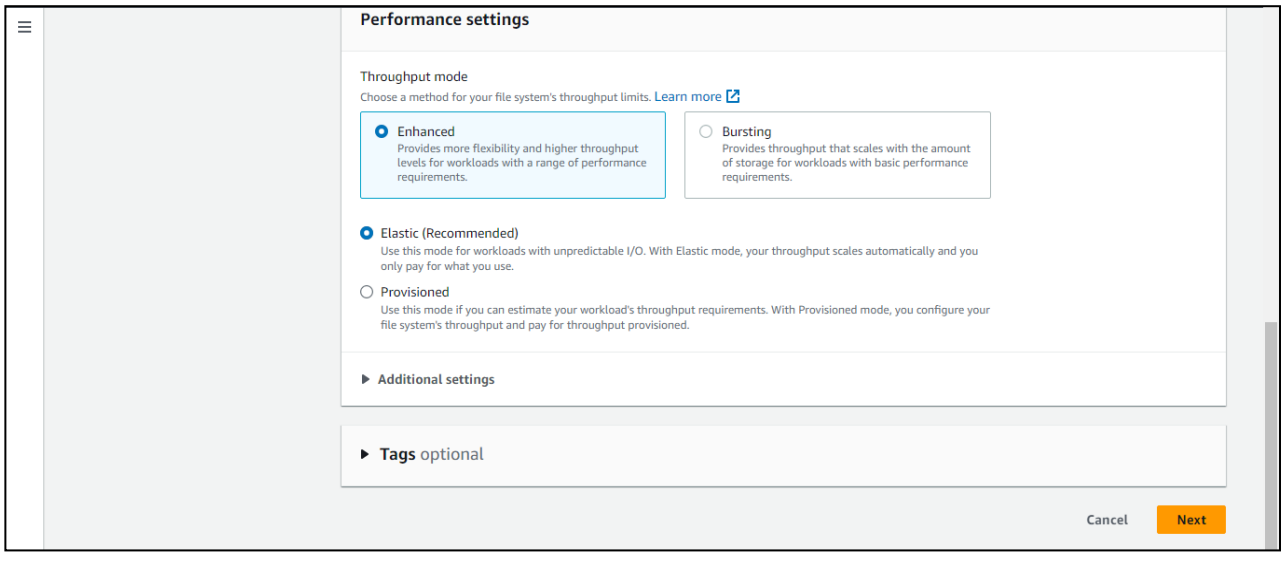
Step 3: Once you click Create file system, now file the basic details like Name of the file system and you can select the EC2 instance if you have already to connect the file system with your EC2 instance.



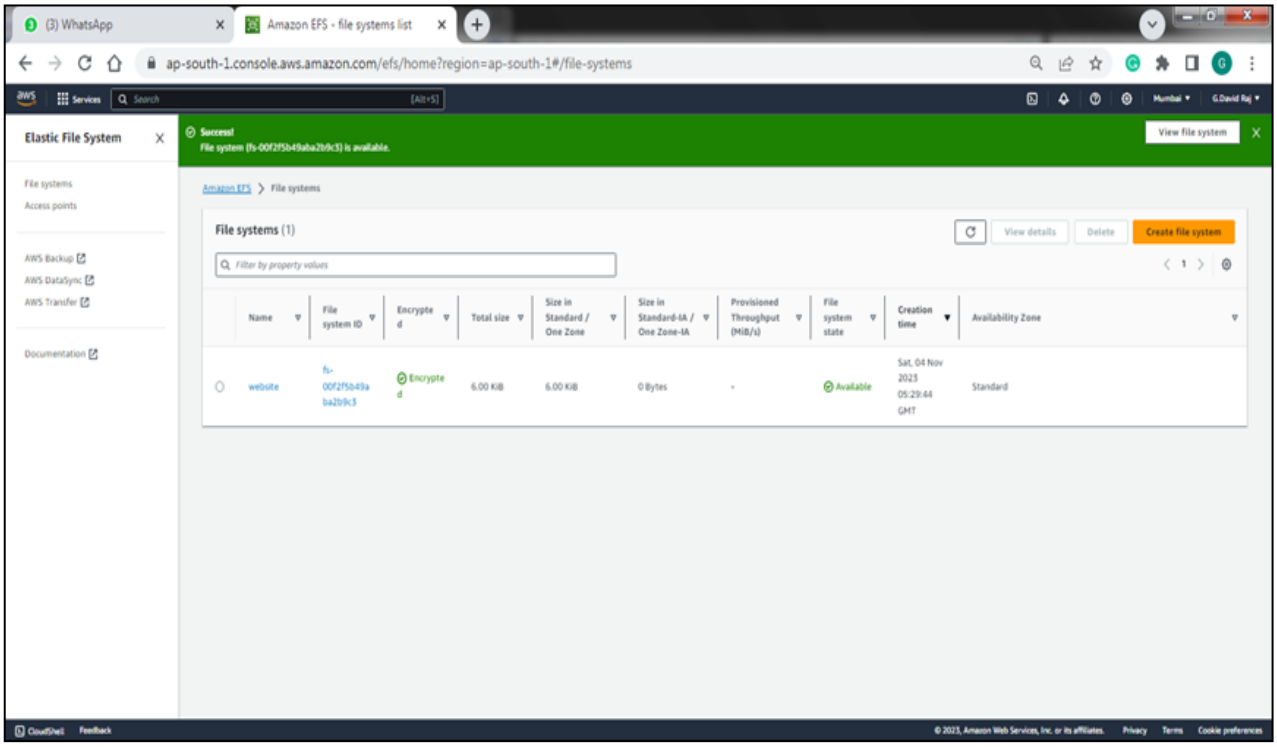
Step 4: Now click to Create and deploy the file system or if you want to customize your file system then click customize and then customize the file system as you wish.



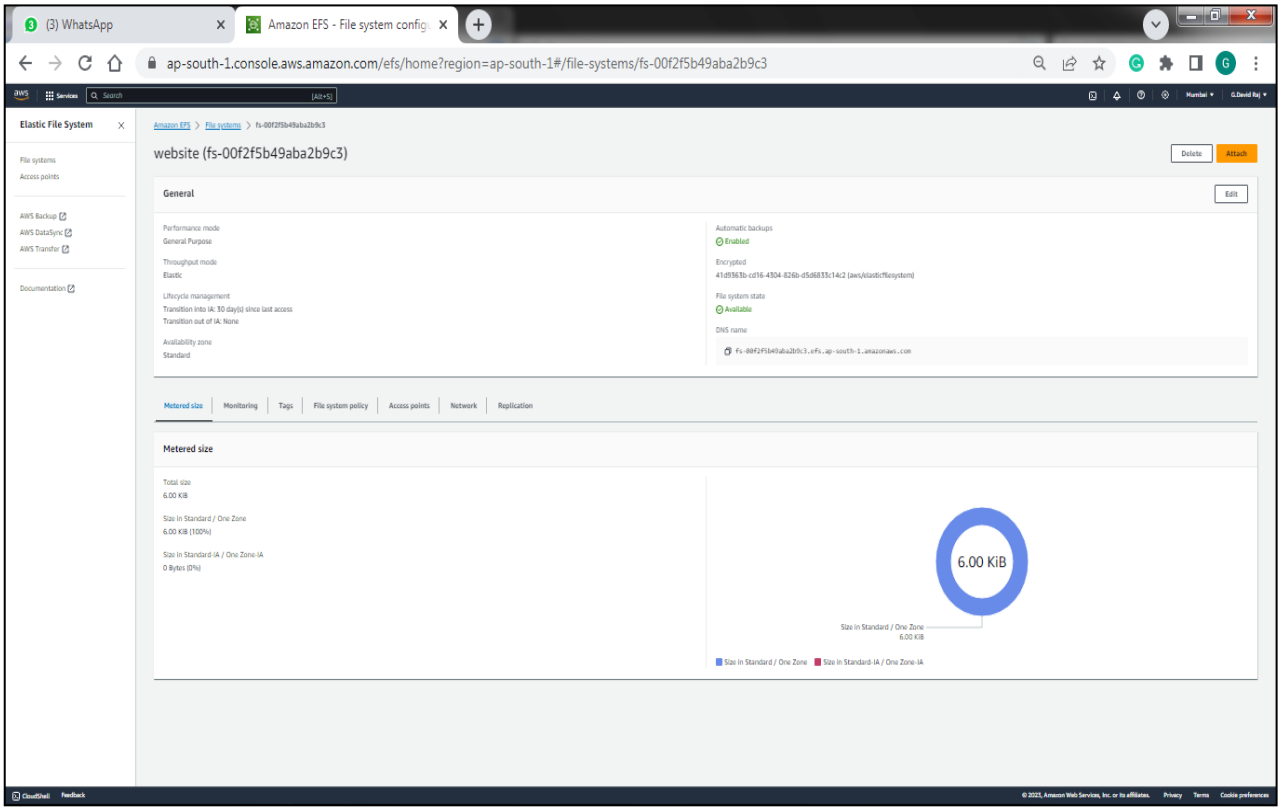
Step 5: After customizing the file system click on Create to initiate the file system.



Step 6: After click on Next, now the EFS file system has been created.



Step 7: The EFS file system details are shown below image.

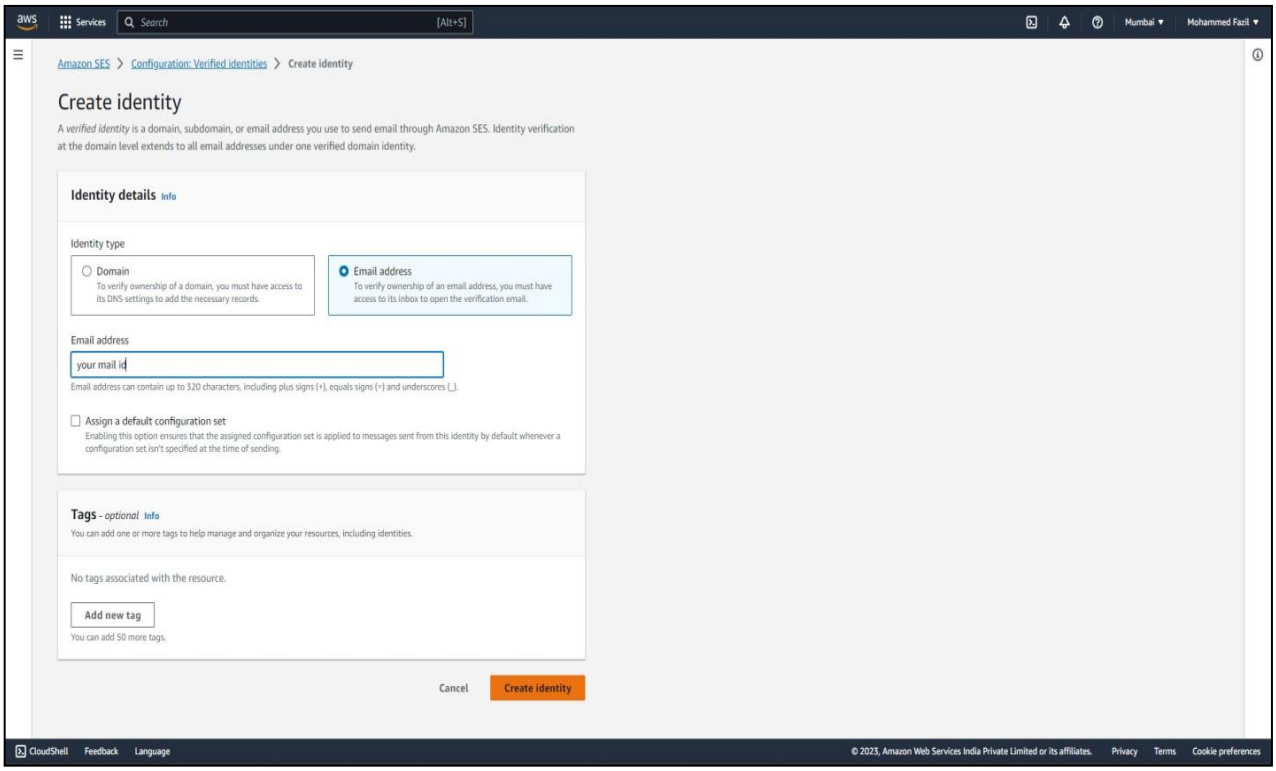


EX.NO : 6

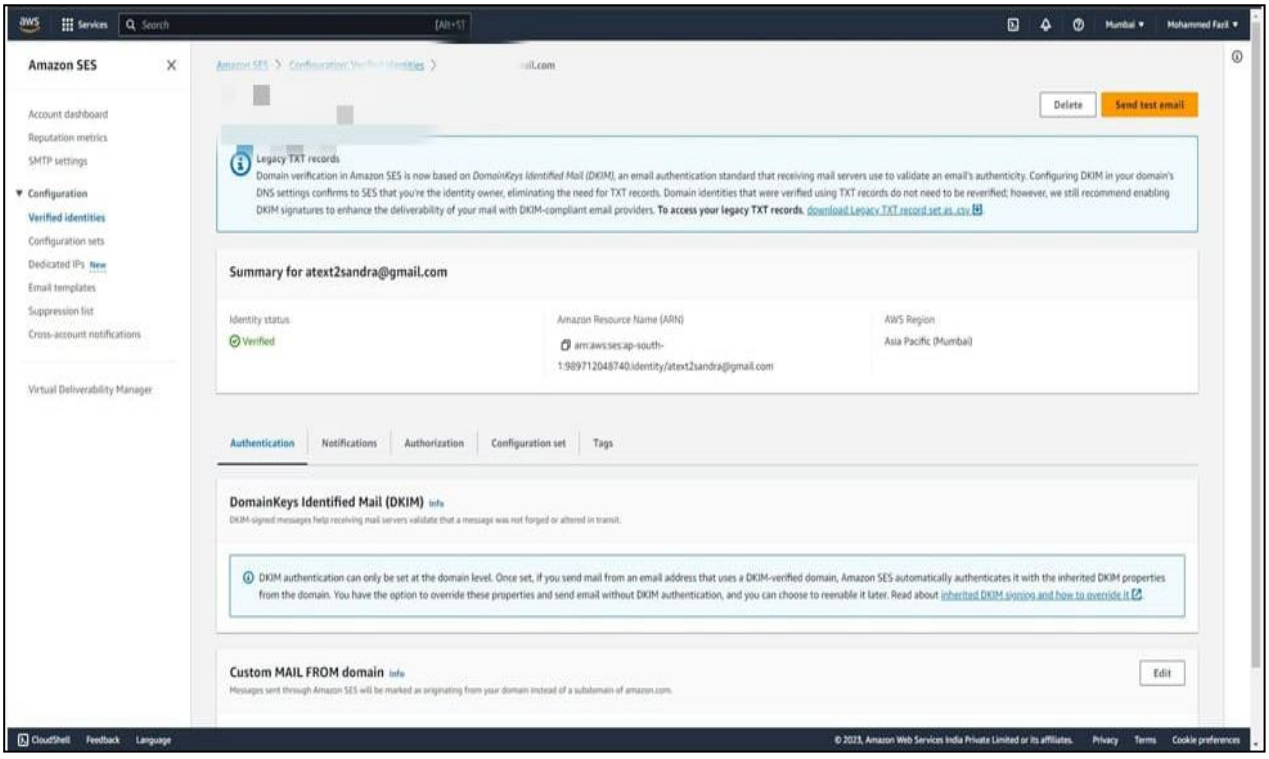
DATE :

DEVELOP THE MASS E-MAILING USING AWS LAMBDA

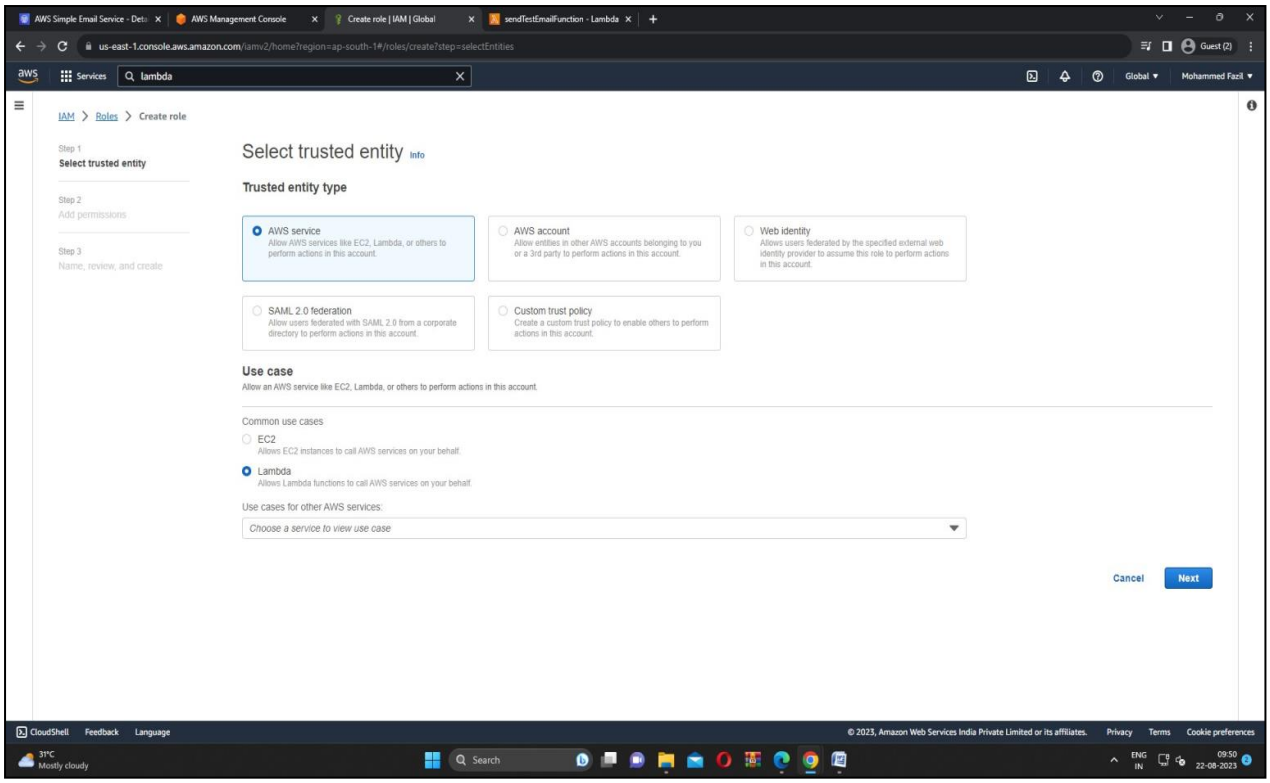
Step 1: Search “Amazon SES” in AWS account and click create identity in top of the page and select Email address for identity type. The user enters the email address and clicks create identity in bottom of the page.



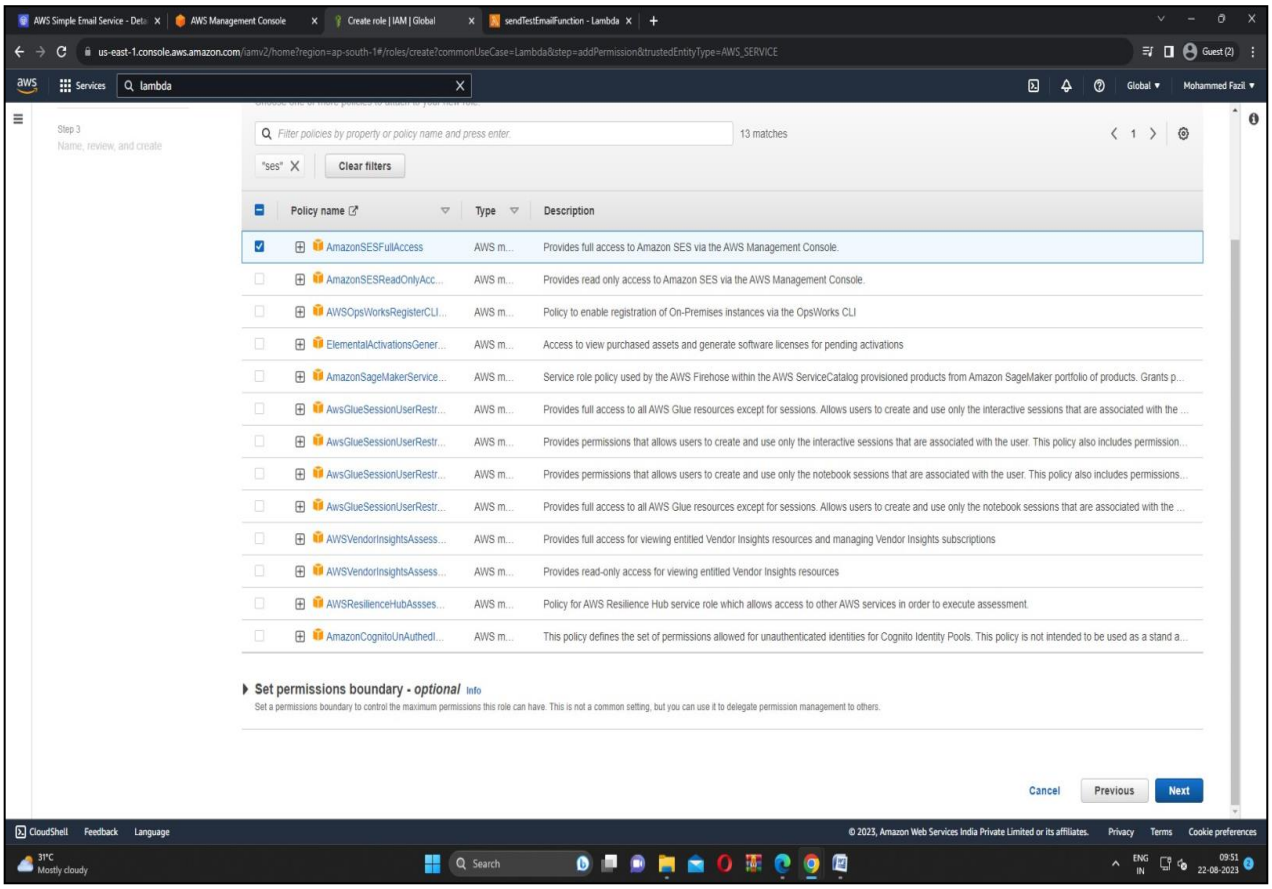
Step 2: Check your previous entered email id. The user got the verification email from AWS, click the verification link and get verified. Now the identity status is verified.



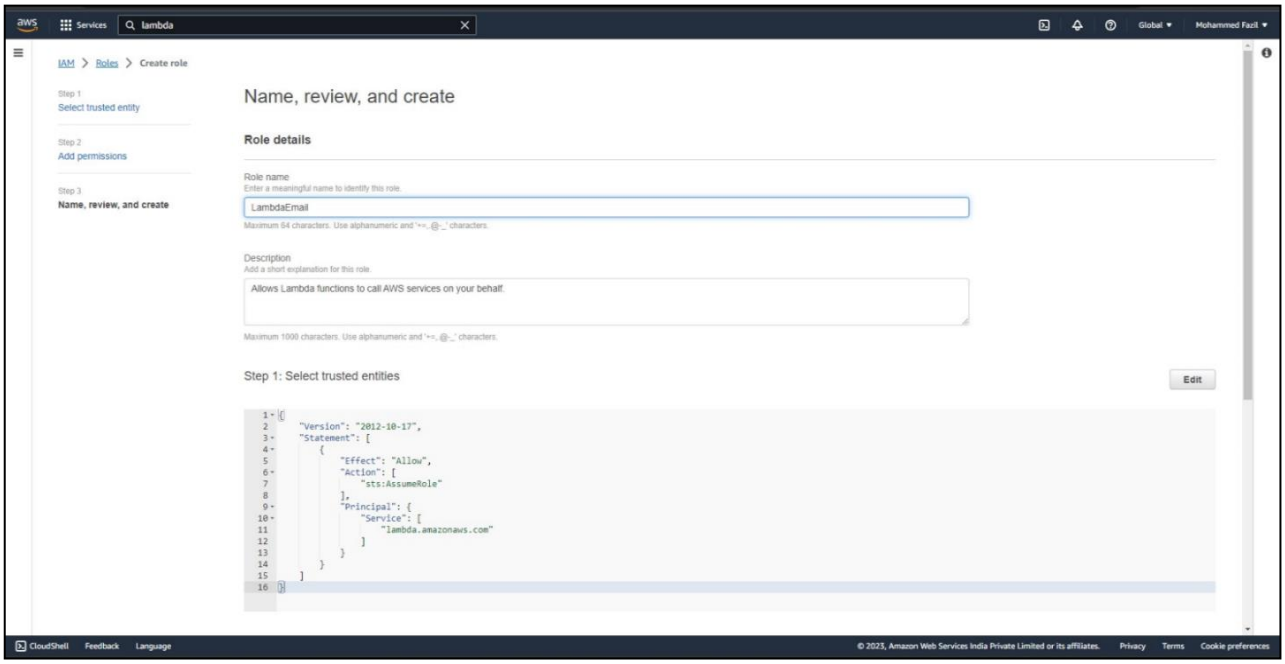
Step 3: Go to AWS Console and search “IAM” then right click to open link in new tap. The user selects Roles from left side menu and click Create role.



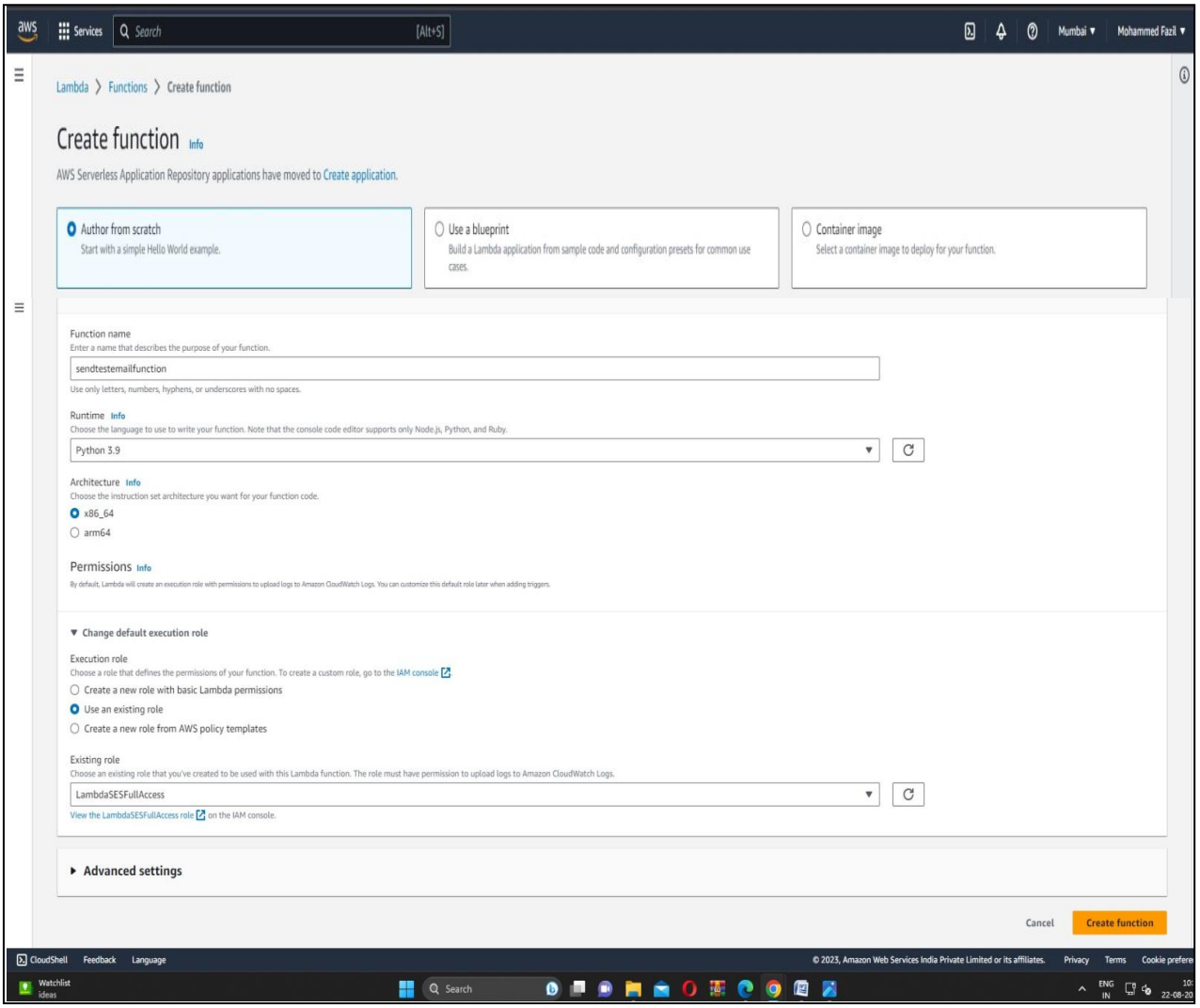
Step 4: The user search “SES” in search bar and select “AmazonSESFullAccess” then click “Next” button in bottom of the page



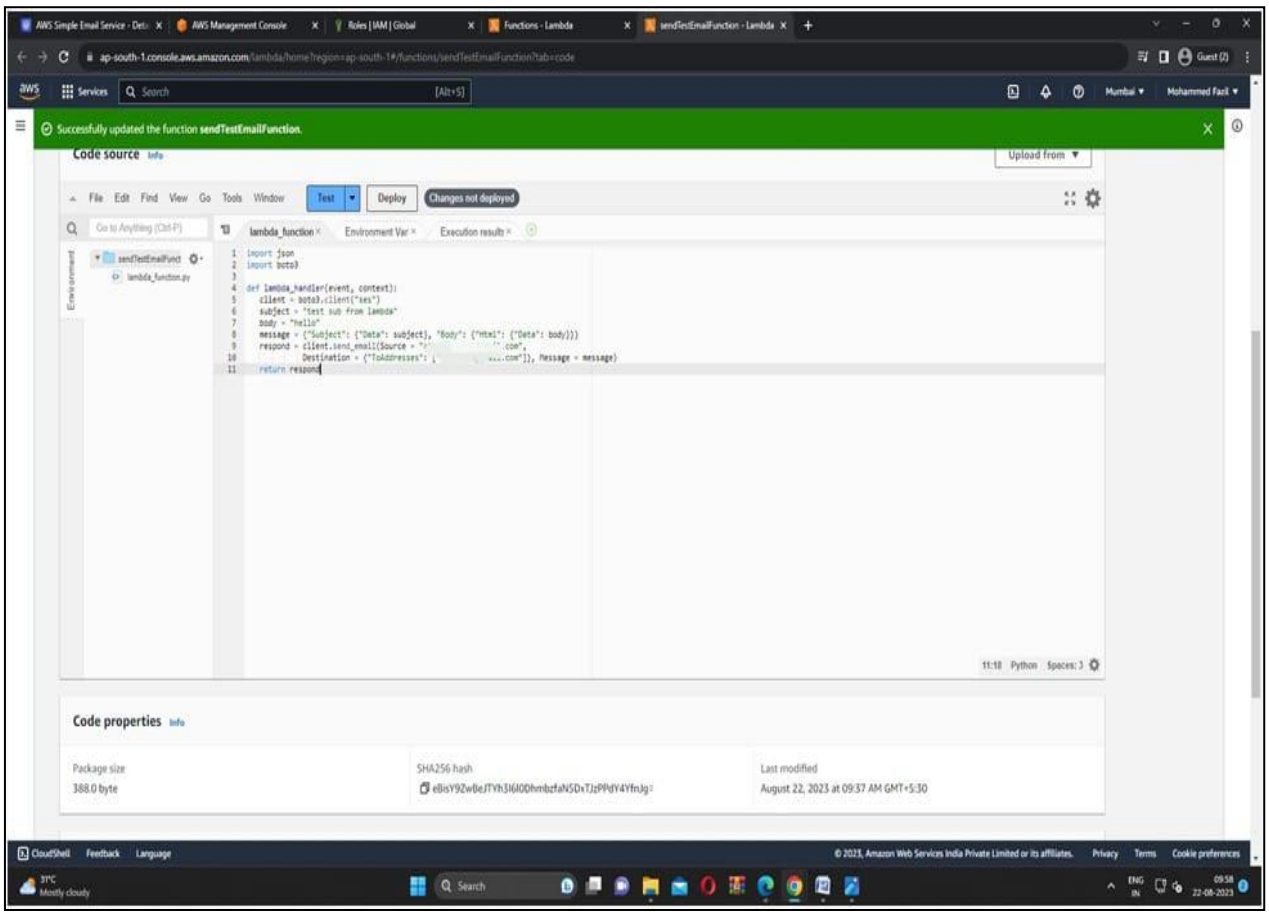
Step 5: The user enter the role name and click create role button in bottom of the page.



Step 6: Go to AWS console and search “Lambda” Function in search bar then right click to open link in new tap. The user click create function in top of the page. Now select Author from scratch, then type the function name, programming language choose python 3.9, select architecture x86_64, and click the change default execution role and select use an existing role. It shows the roles created in “IAM” then click create function in bottom of the page.



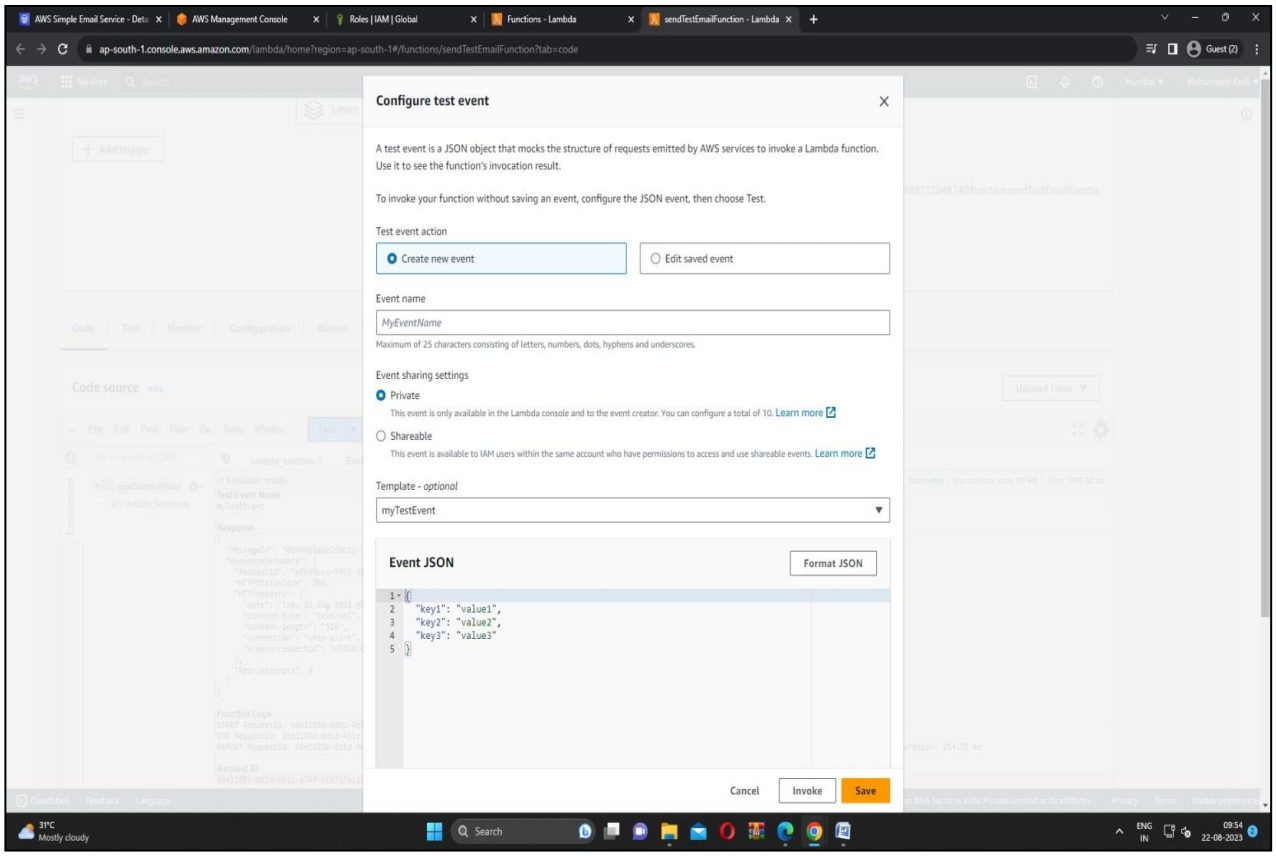
Step 7: After the step 6, successfully created the function email. You can now change its code and configuration. To invoke your function with a test event, choose "Test".



Python Source code:

```
import json
import boto3
def lambda_handler(event, context):
  client = boto3.client("ses")
  subject = "test sub from lambda"
  body = "test body from lambda"
  message = {"Subject": {"Data": subject}, "Body": {"Html": {"Data": body}}}
  respond = client.send_email(Source = "sender mail id",
        Destination = {"ToAddresses": ["receiver mail id"]}, Message = message)
  return respond
```

Step 8: The user configure test event action, event name, event sharing settings and click save button in bottom of the page. Finally click Test button.



Step 9: Check your Source email id (send list) and destination email id (inbox)

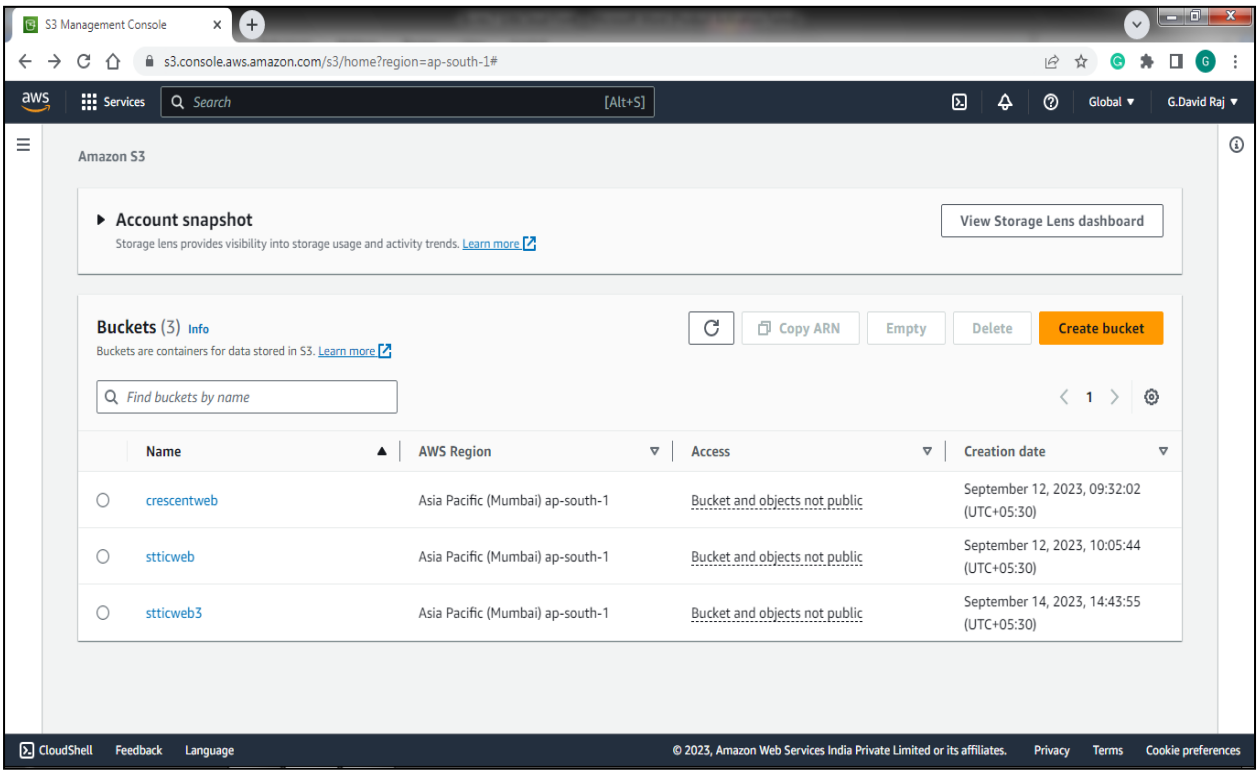
EX.NO : 7

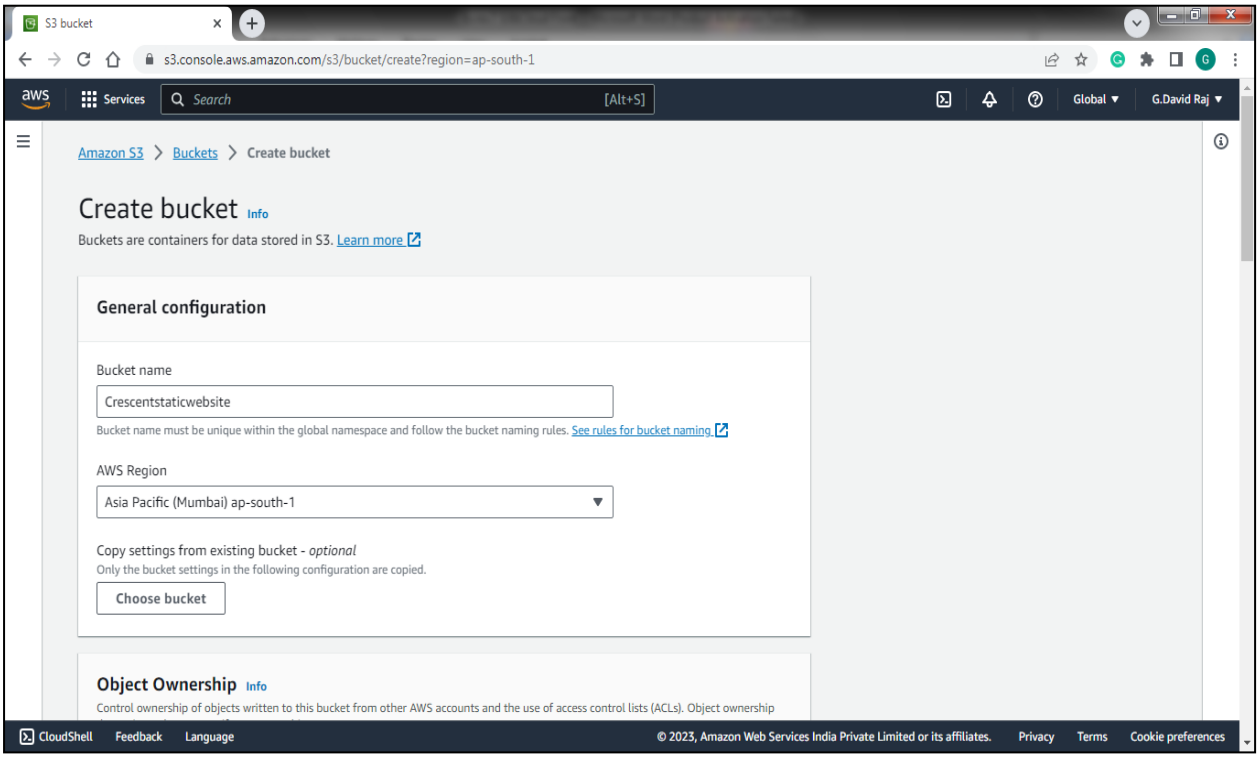
DATE :

EXPLORE STATIC WEBSITE WITH S3 AND CLOUD FRONT

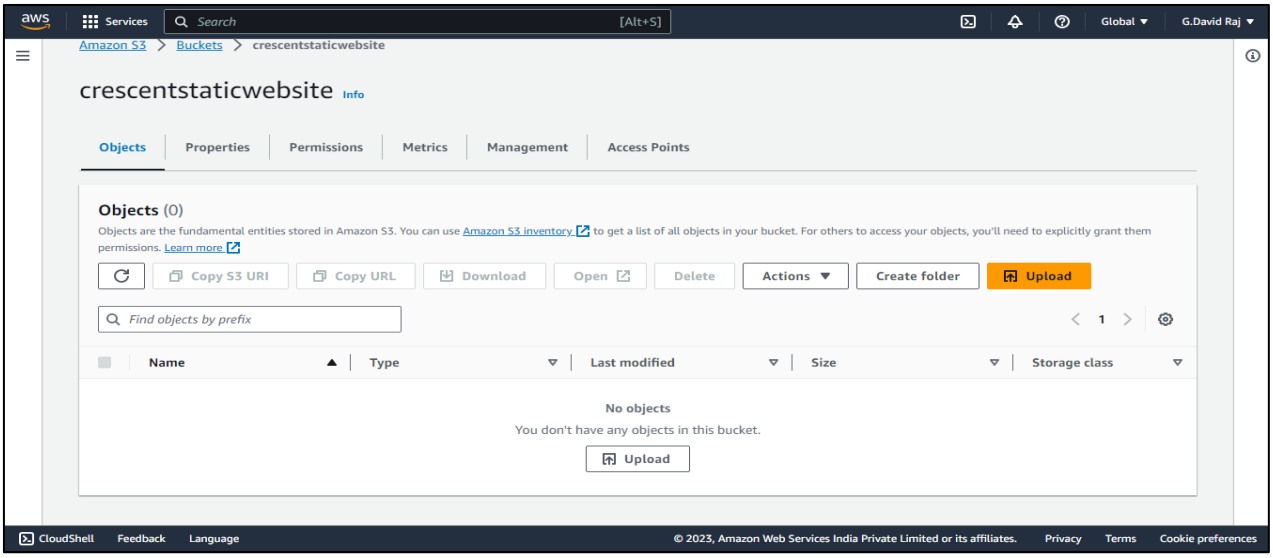
Step 1: Download the any free website templates from free-css.com and extract the downloaded file.

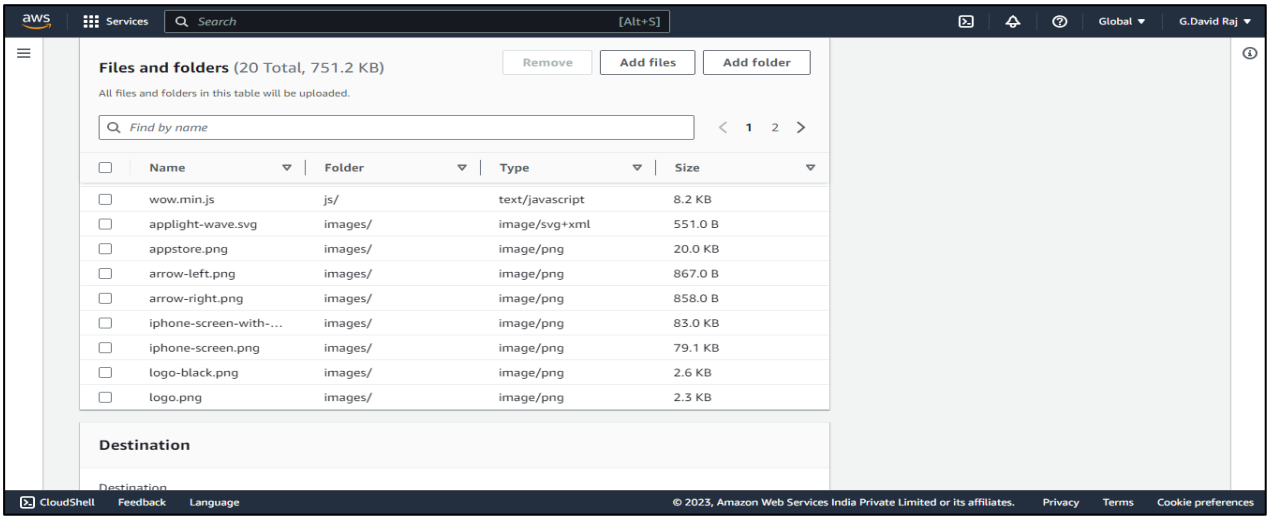
Step 2: Search S3 from aws console and click create bucket then type the bucket name "crescentstaticwebsite". The S3 bucket was created successfully.



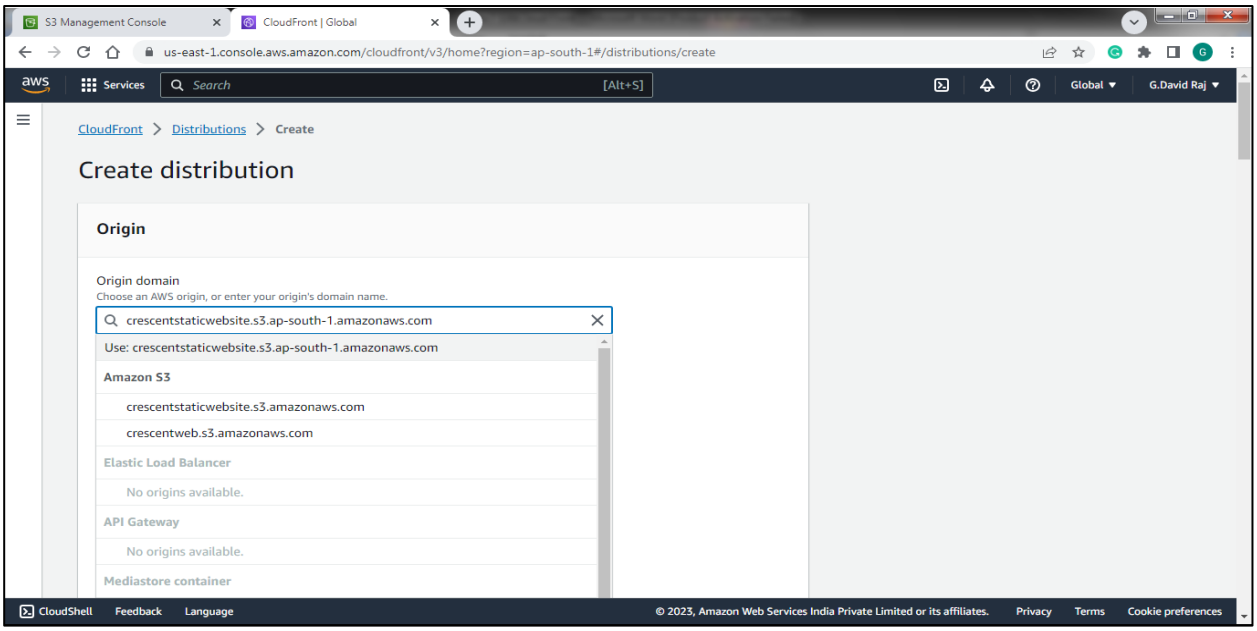


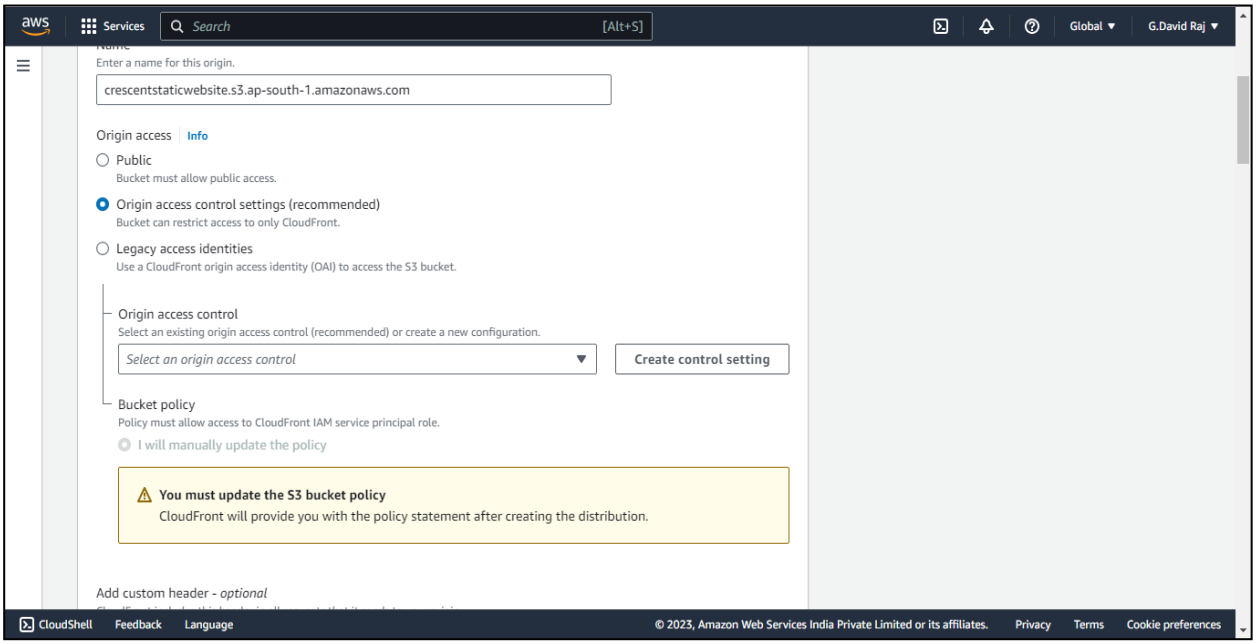
Step 3: Select the “crescentstaticwebsite” S3 bucket and click upload, first add files from extracted website templates then add the remaining folders. Then click upload.

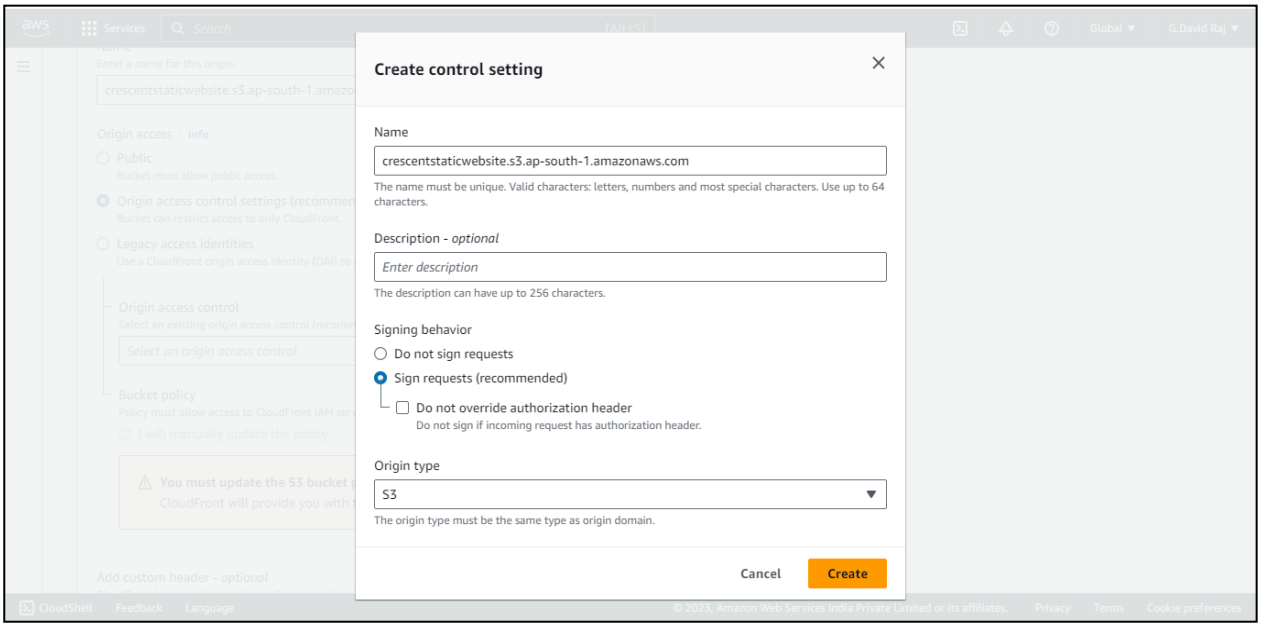




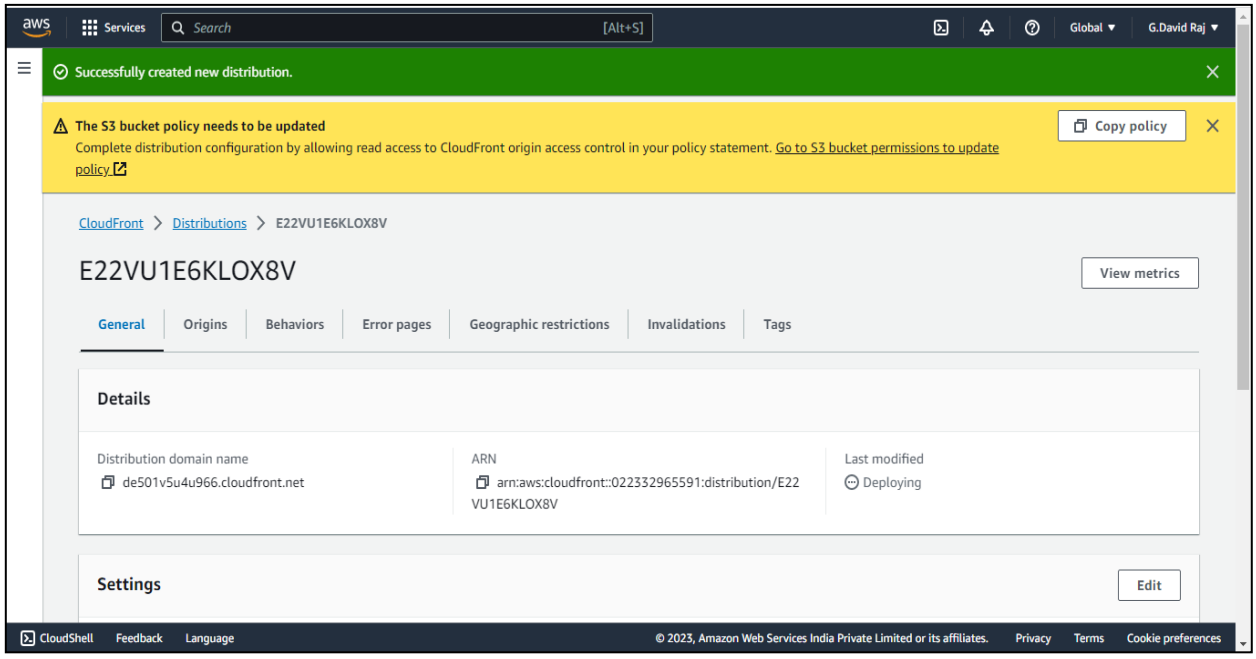
Step 4: Search “cloud front” in aws console and click “create distribution”. Oriin domain can be showed, and select the particular S3 bucket, then change origin access to “Origin access control settings (recommended)”. Finally fill the “Default root object – optional” with index.html and clicks create distribution.



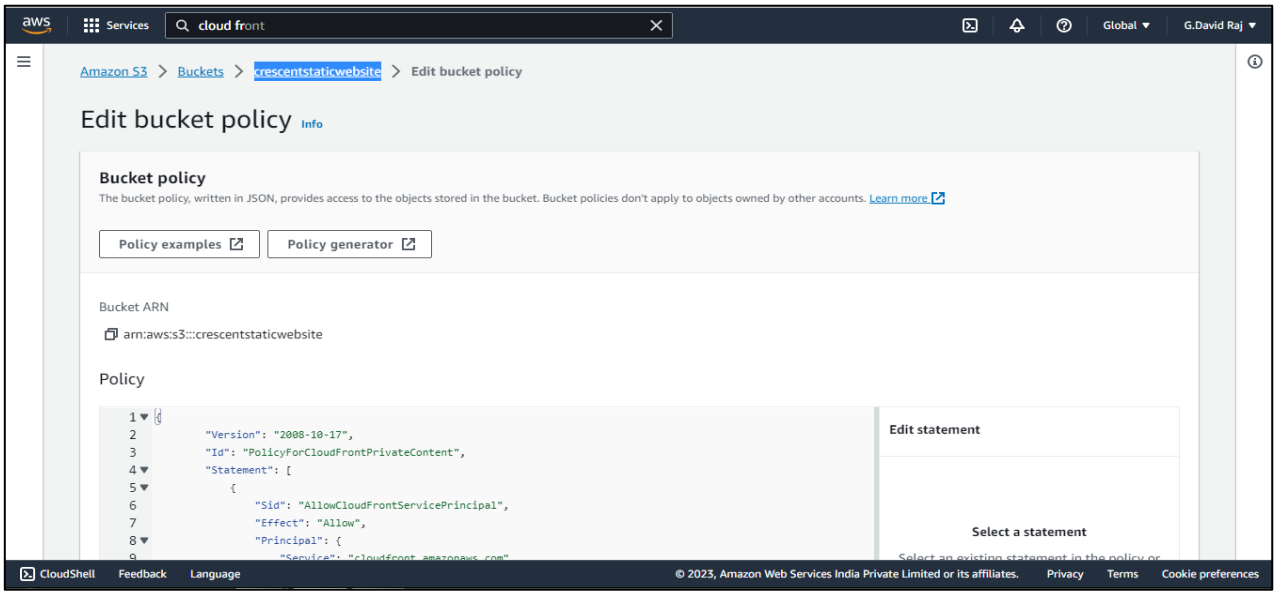




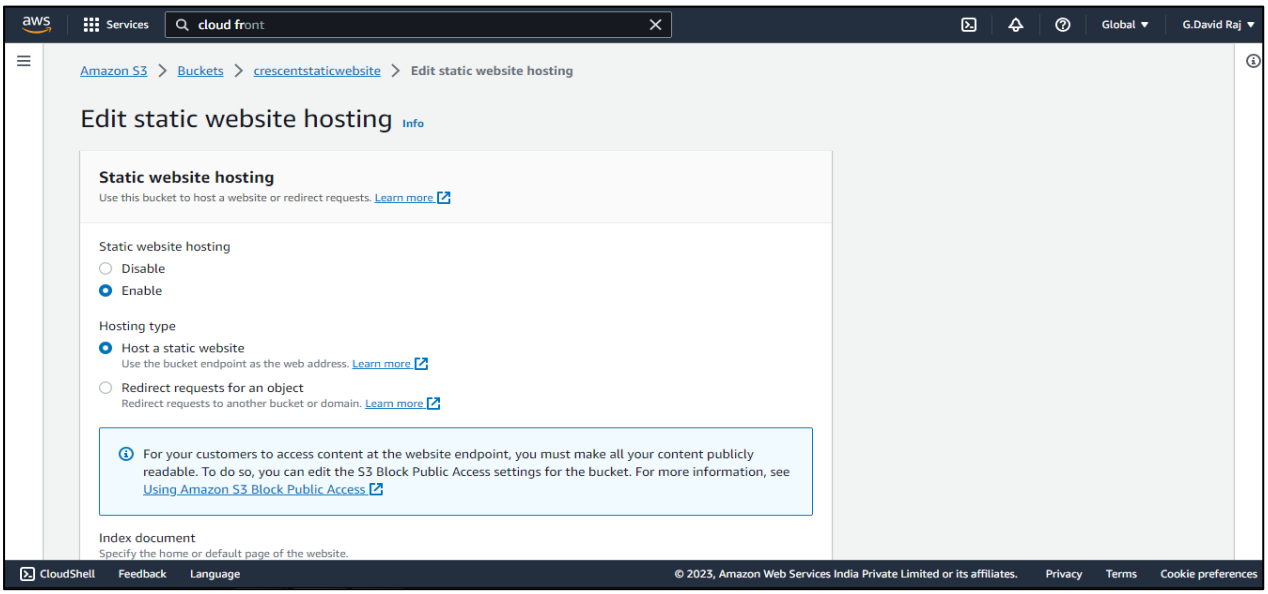
Step 5: The new distribution was created successfully and copy policy.

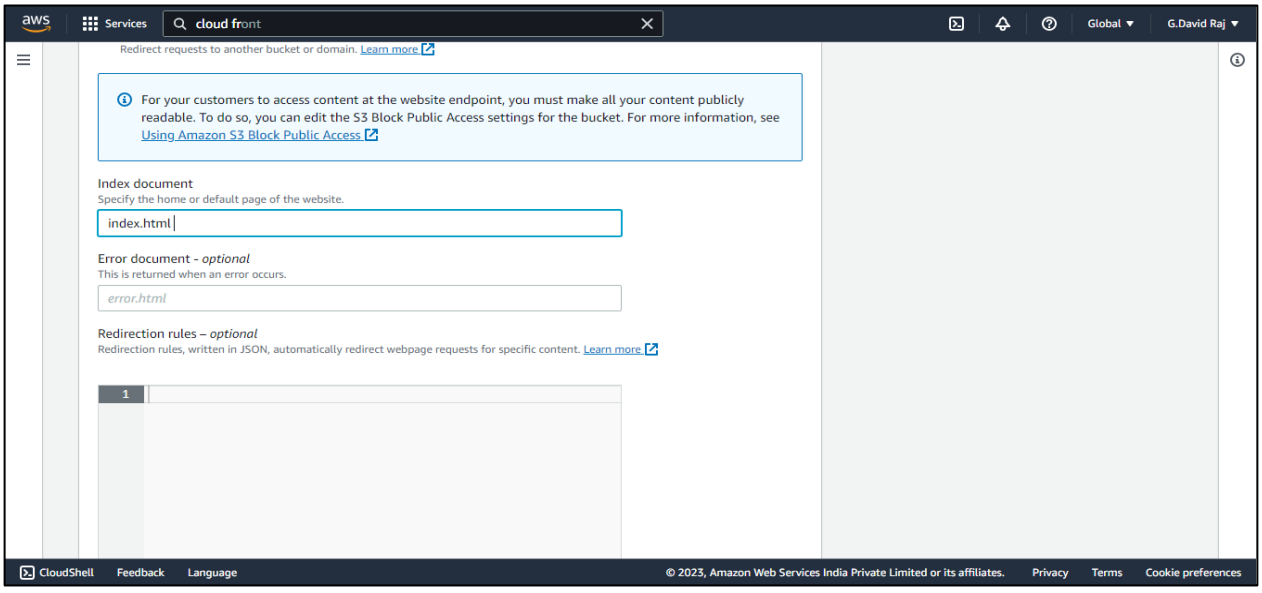


Step 6: Go to S3 and select particular bucket “crescentstaticwebsite” permissions. Then edit bucket policy and paste the copy policy (previously copied in cloud front distribution) and click save changes.

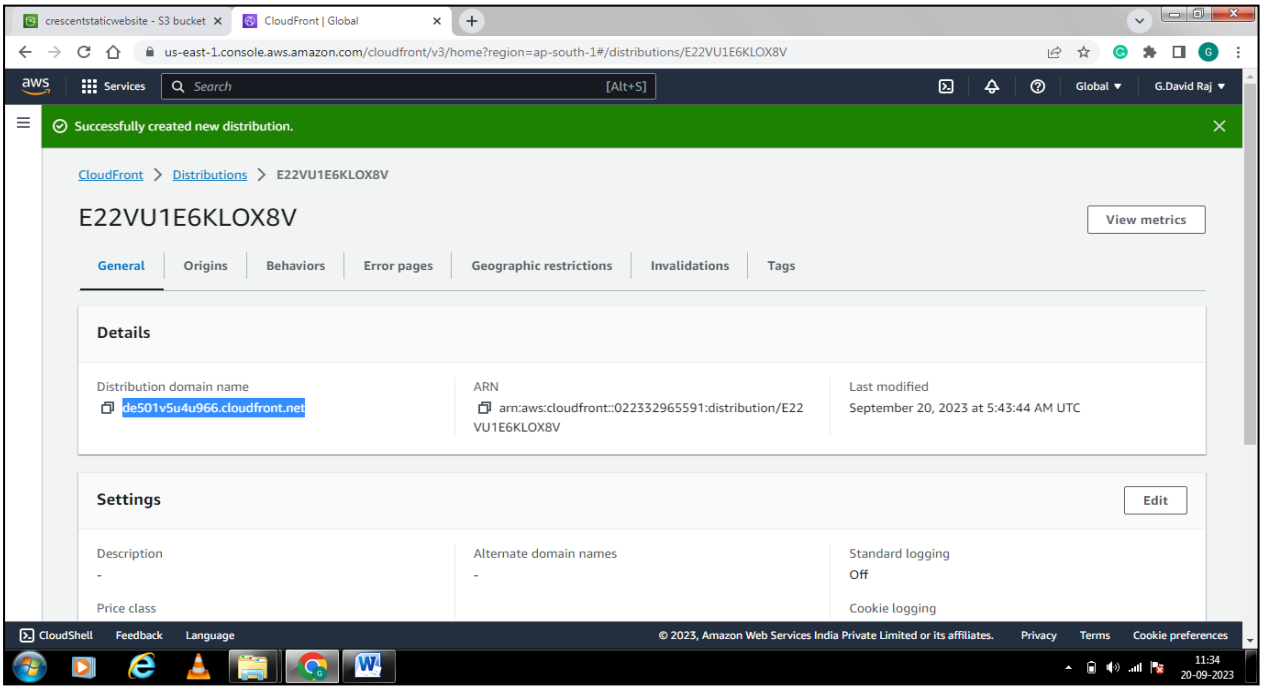


Step 7: Select particular bucket “crescentstaticwebsite” properties. Then edit “Static website hosting” and choose enable. Then “index document” is filled by index.html and click save changes.

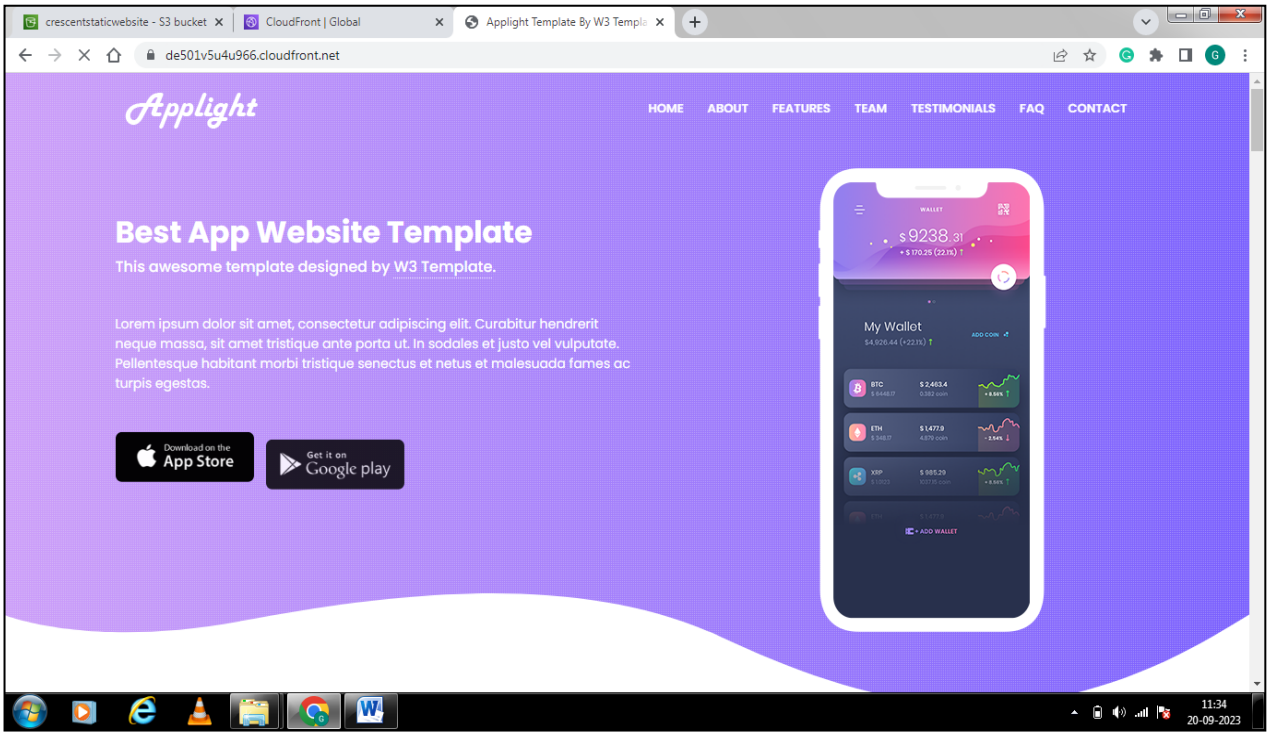




Step 8: Go to cloud front origin and copy the distribution domain name (de501v5u4u966.cloudfront.net) and paste it in browser new tab.



Step 9: Finally the static website is hosted using S3 and Cloud front.

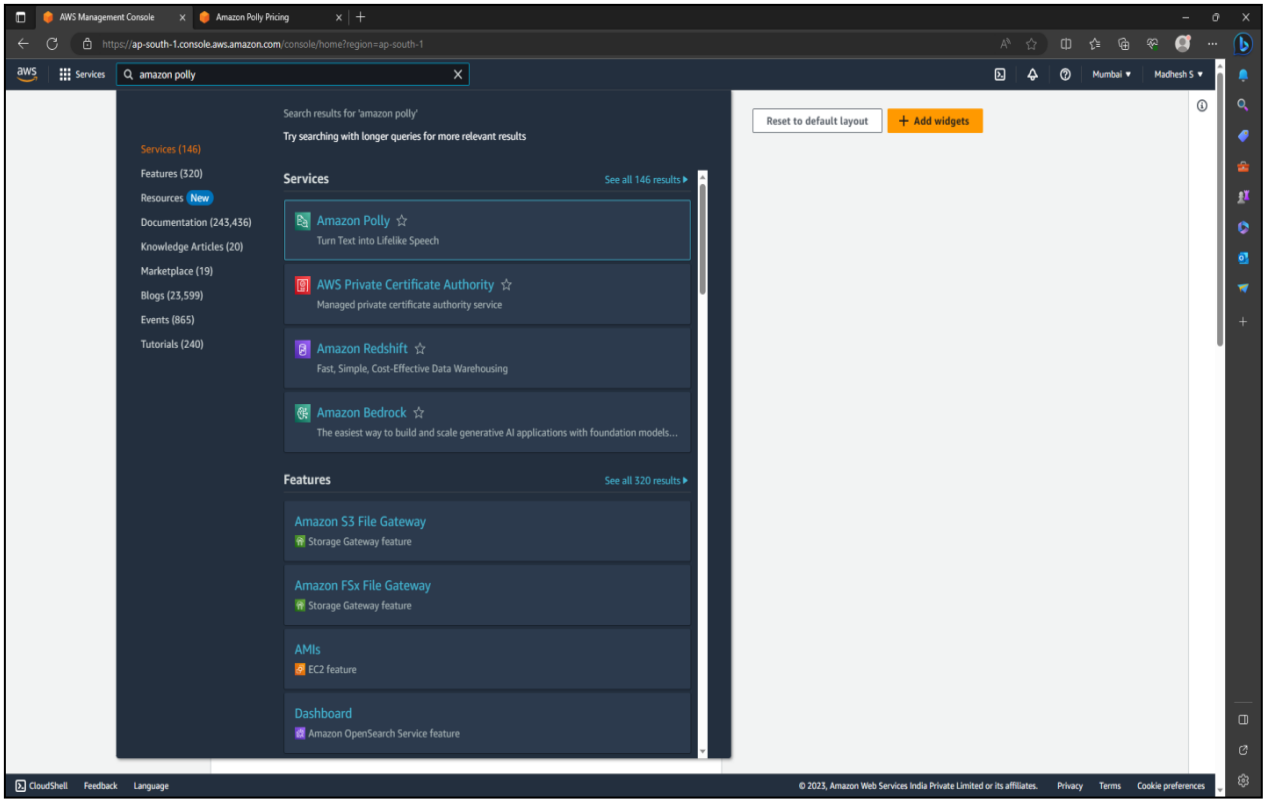


EX.NO : 8

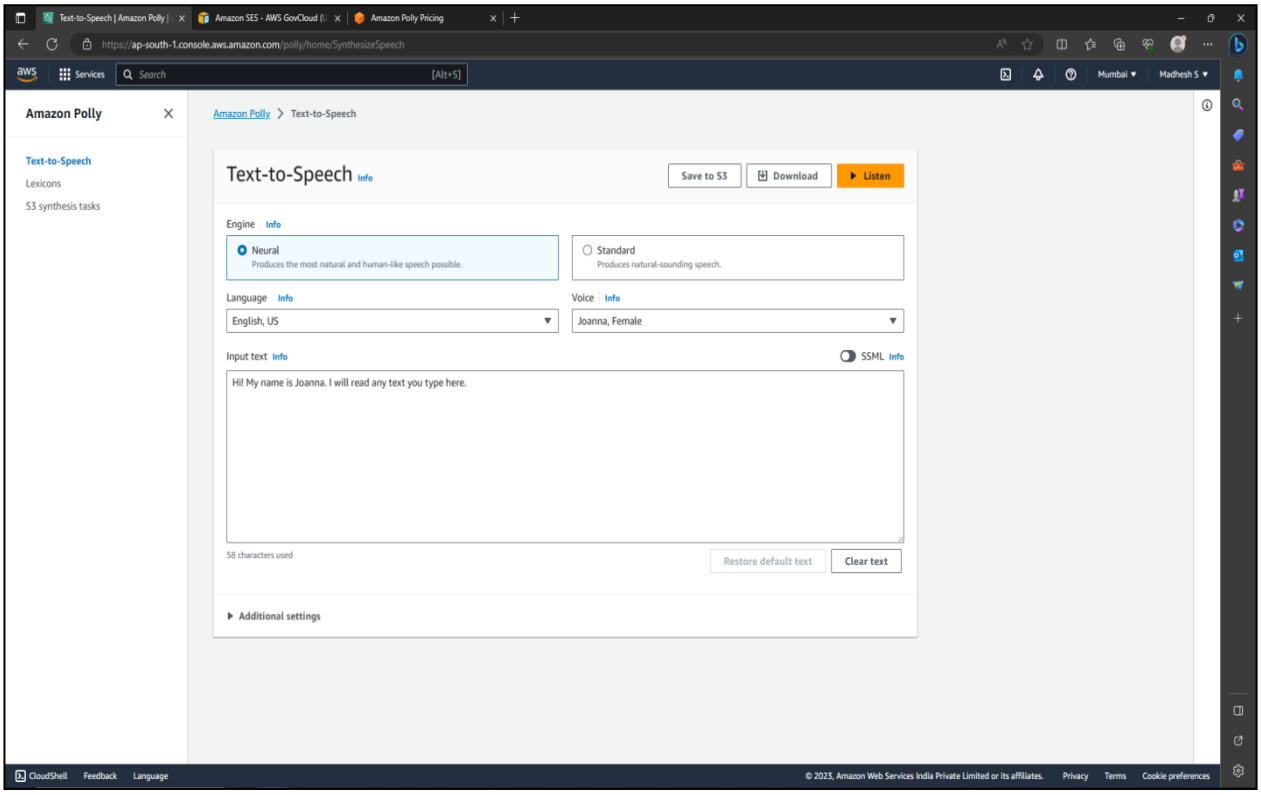
DATE :

CREATE A TEXT-TO-SPEECH CONVERSION WITH SAGE MAKER (AWS Polly)

Step 1: Enter the AWS Console Home Page and search for Amazon Polly

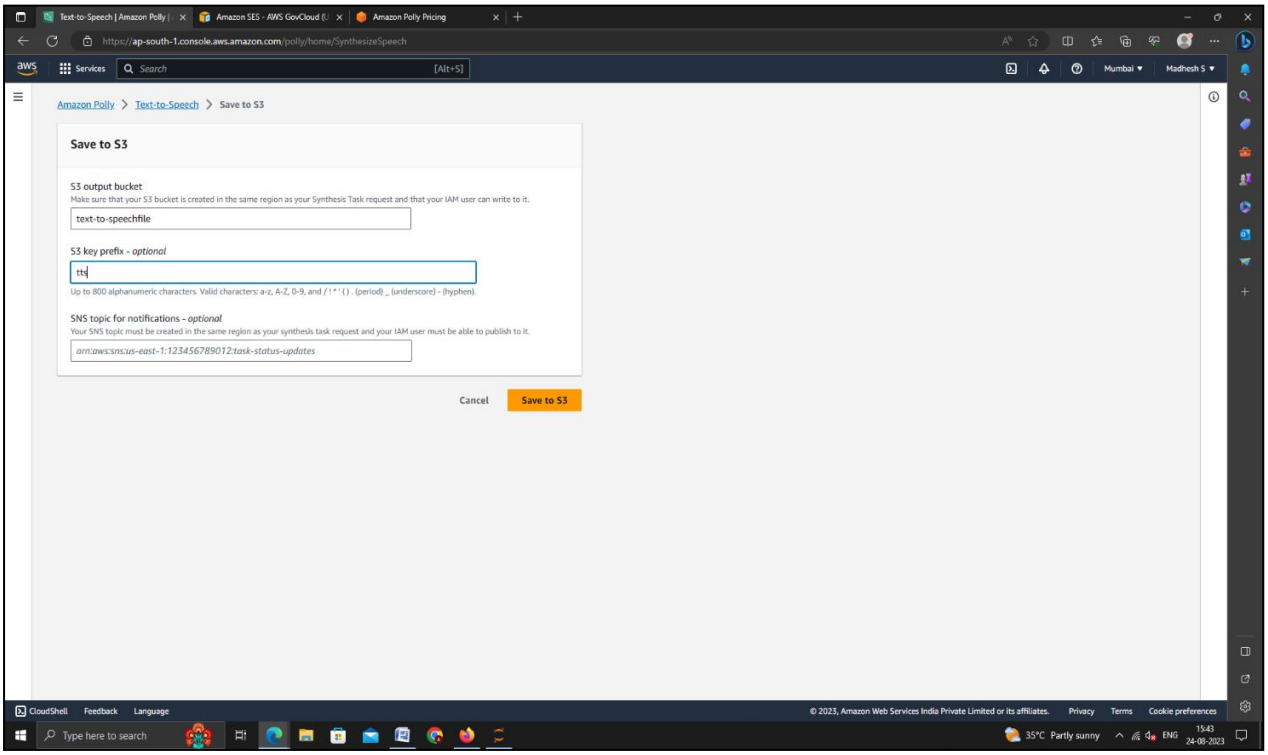


Step 2: Now click on Amazon Polly



Step 3: Now enter the text that you want to convert and the presses listen.

Step 4: The audio file is even saving in S3 bucket.



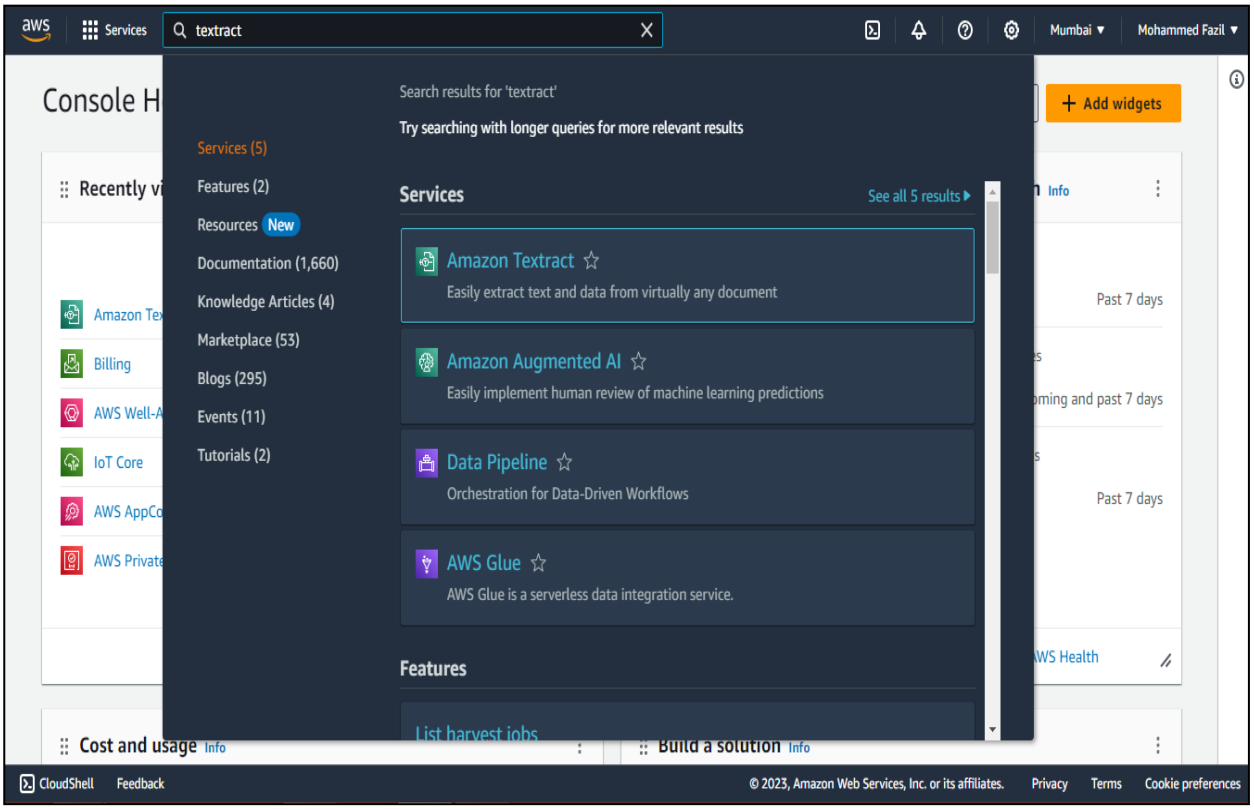
Step 5: Press save to S3 and audio file will be saved in S3.

EX.NO : 9

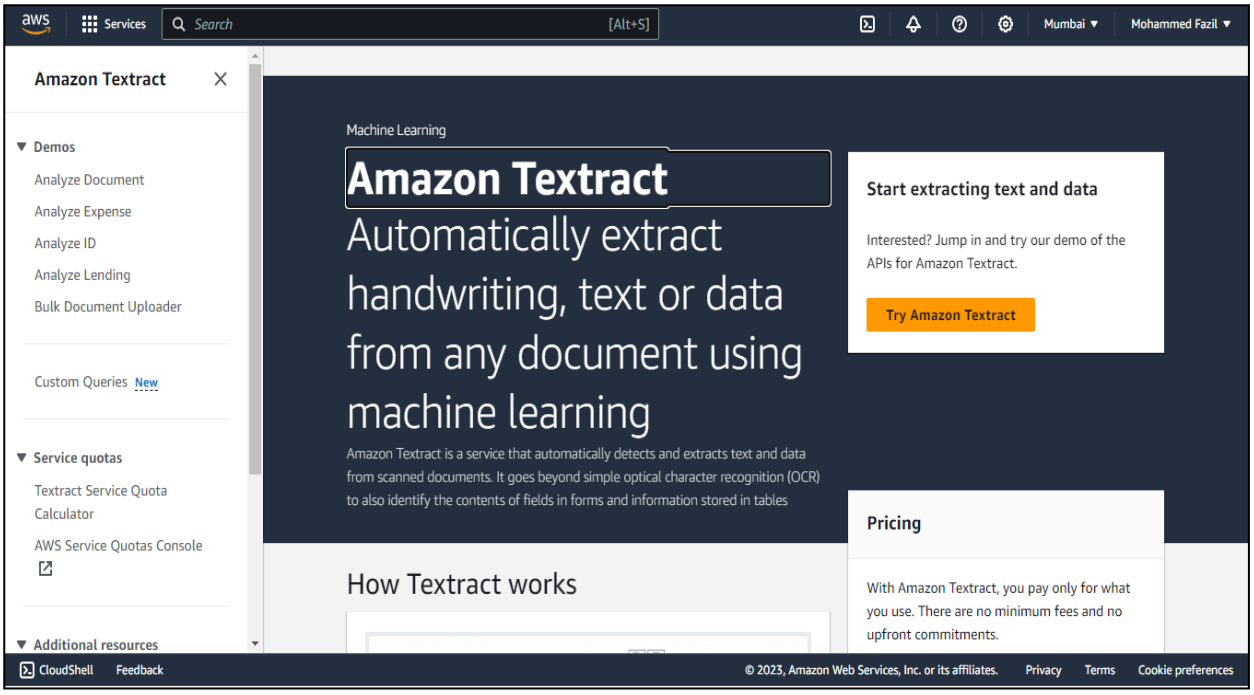
DATE :

IMPLEMENT THE N VISION ENABLES MACHINE LEARNING SERVICES FOR TEXT EXTRACTION FROM IMAGES

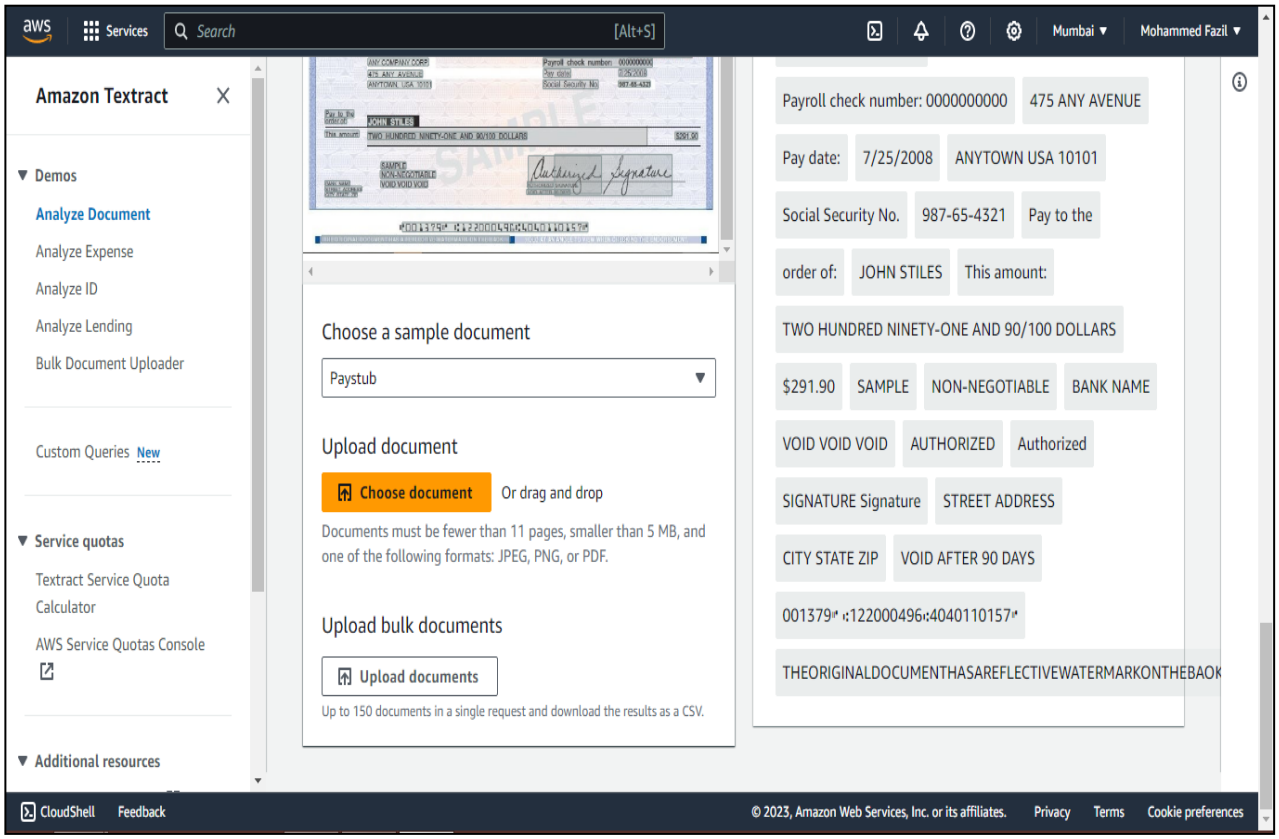
Step 1: Log-In into your AWS account and search for Amazon Textract.



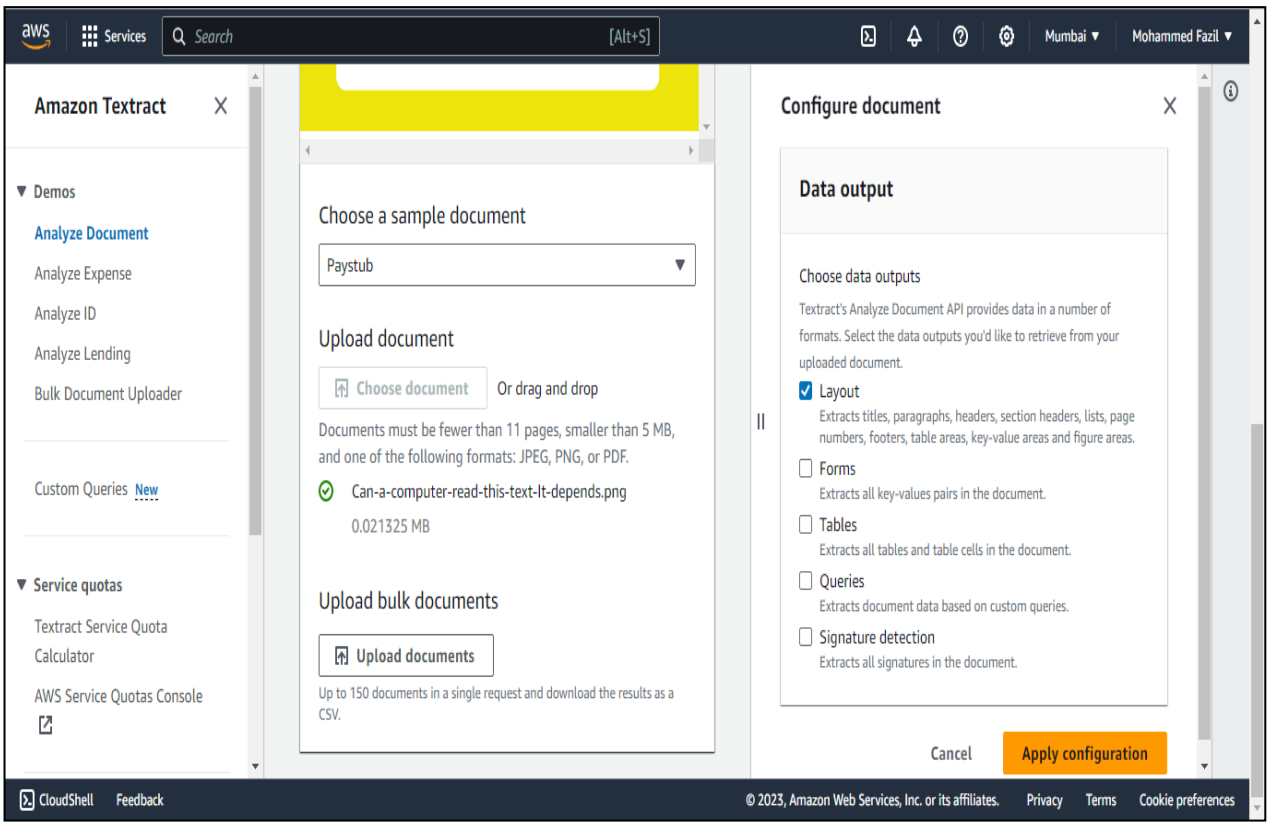
Step 2: Once the page has open then click on Try Amazon Textract



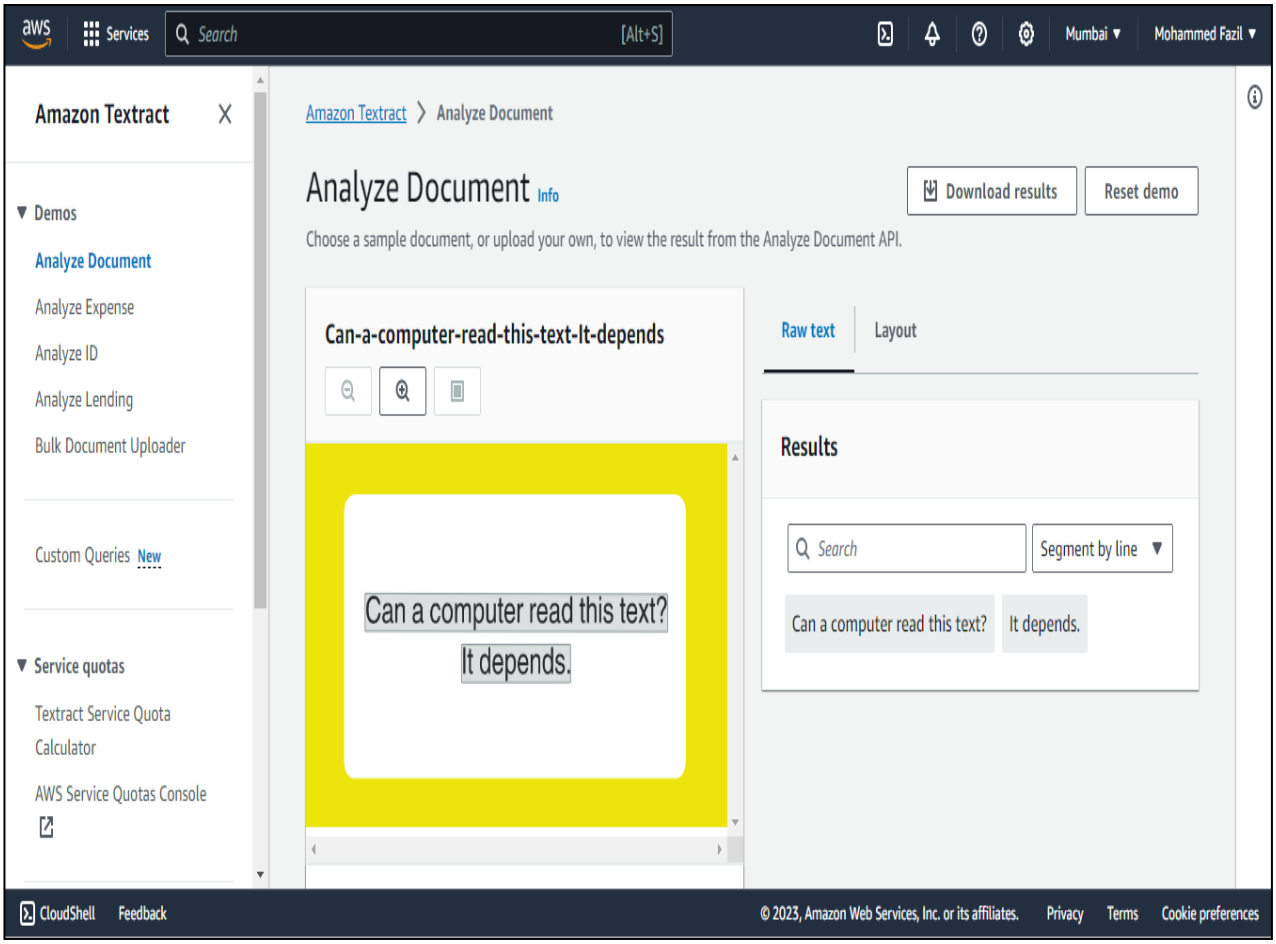
Step 3: Click on Choose Documents to upload the document that you want to extract the Texts.



Step 4: Once uploaded the image and now select the type of output want to get.



Step 5: Now click on Apply Configuration to get the desire output.



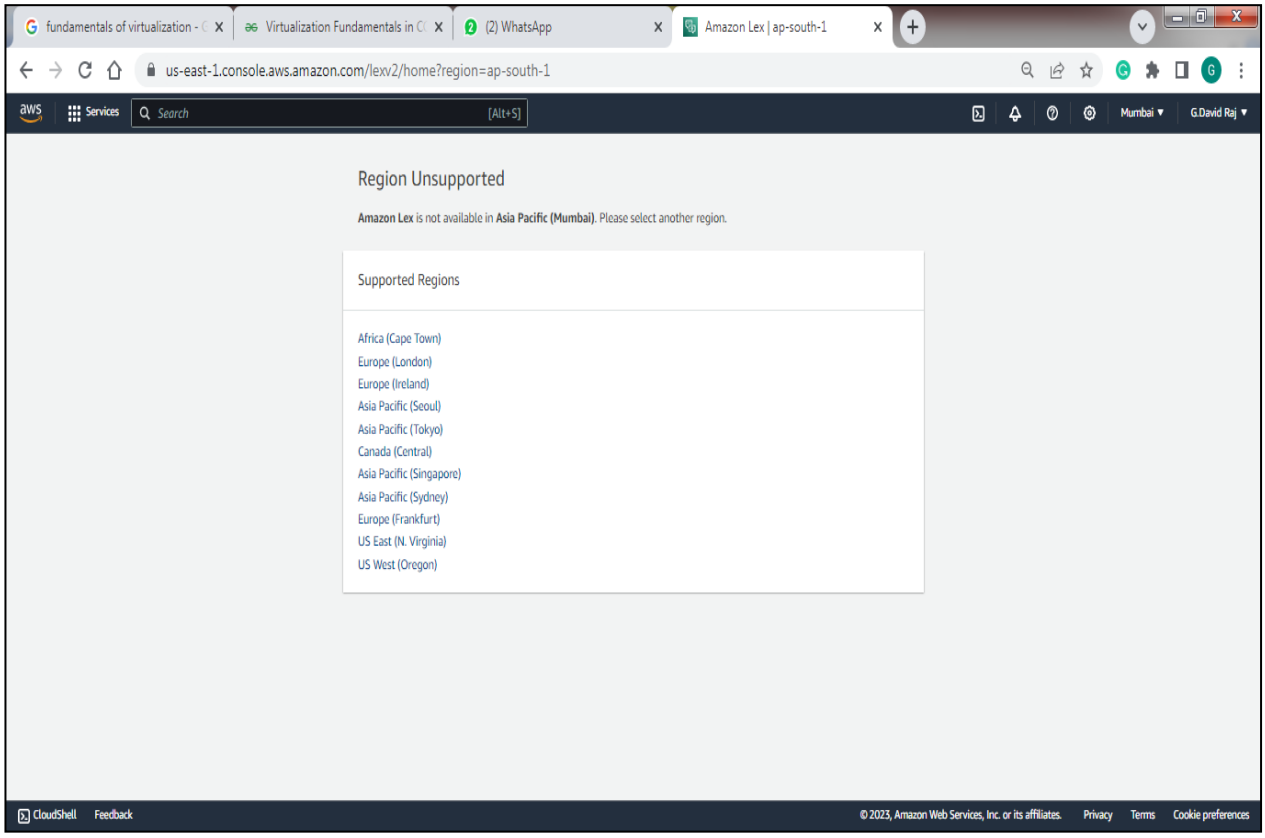
Step 6: The user can download the output by clicking on Download results.

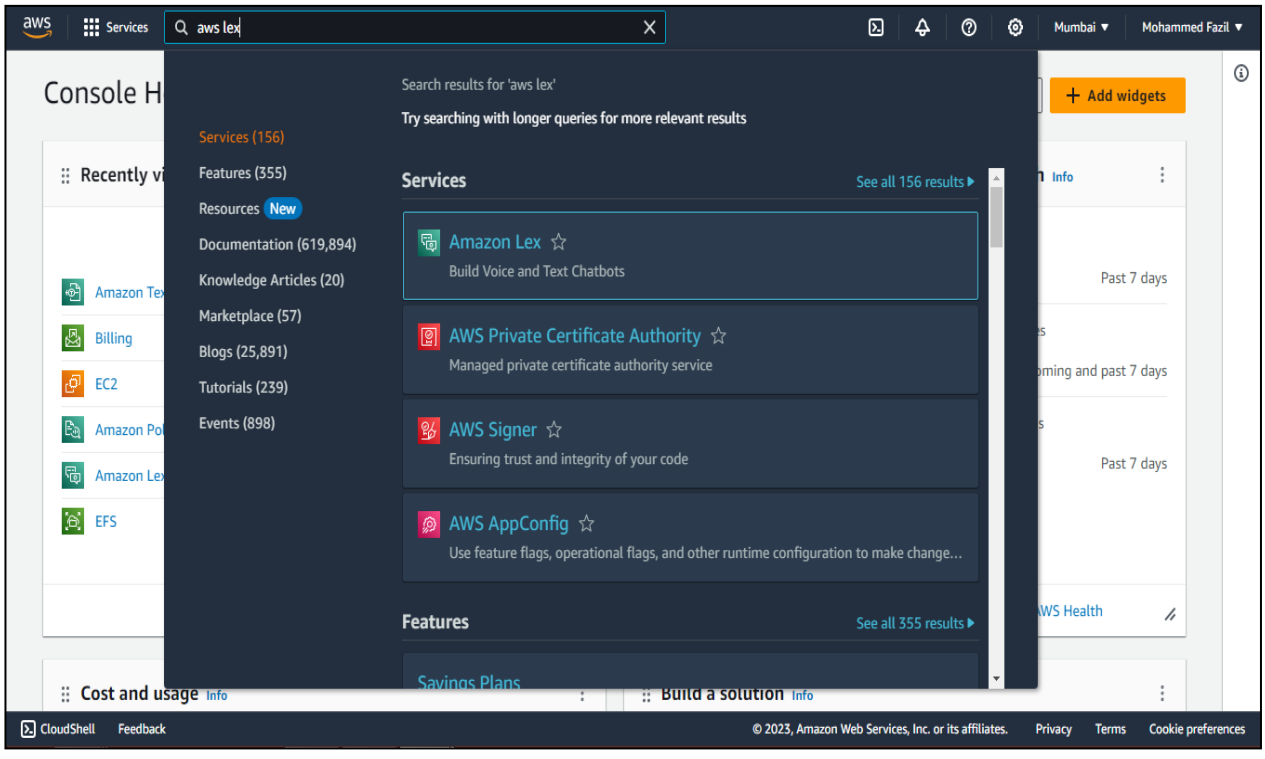
EX.NO : 10

DATE :

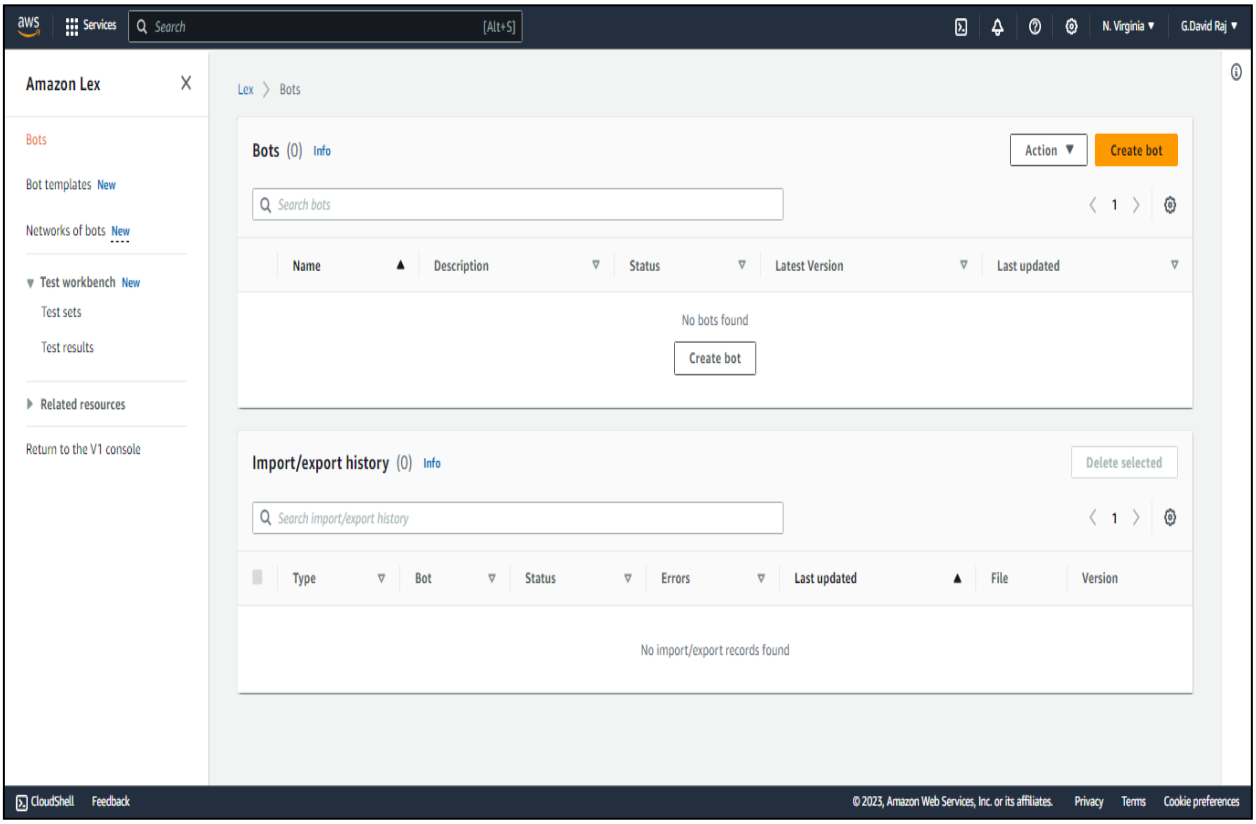
CREATE A CHAT BOT APPLICATION USING ARTIFICIAL INTELLIGENCE (AI) IN AWS

Step 1: Log in into your AWS account and search for Amazon Lex. Amazon Lex is only supported for specific regions only.

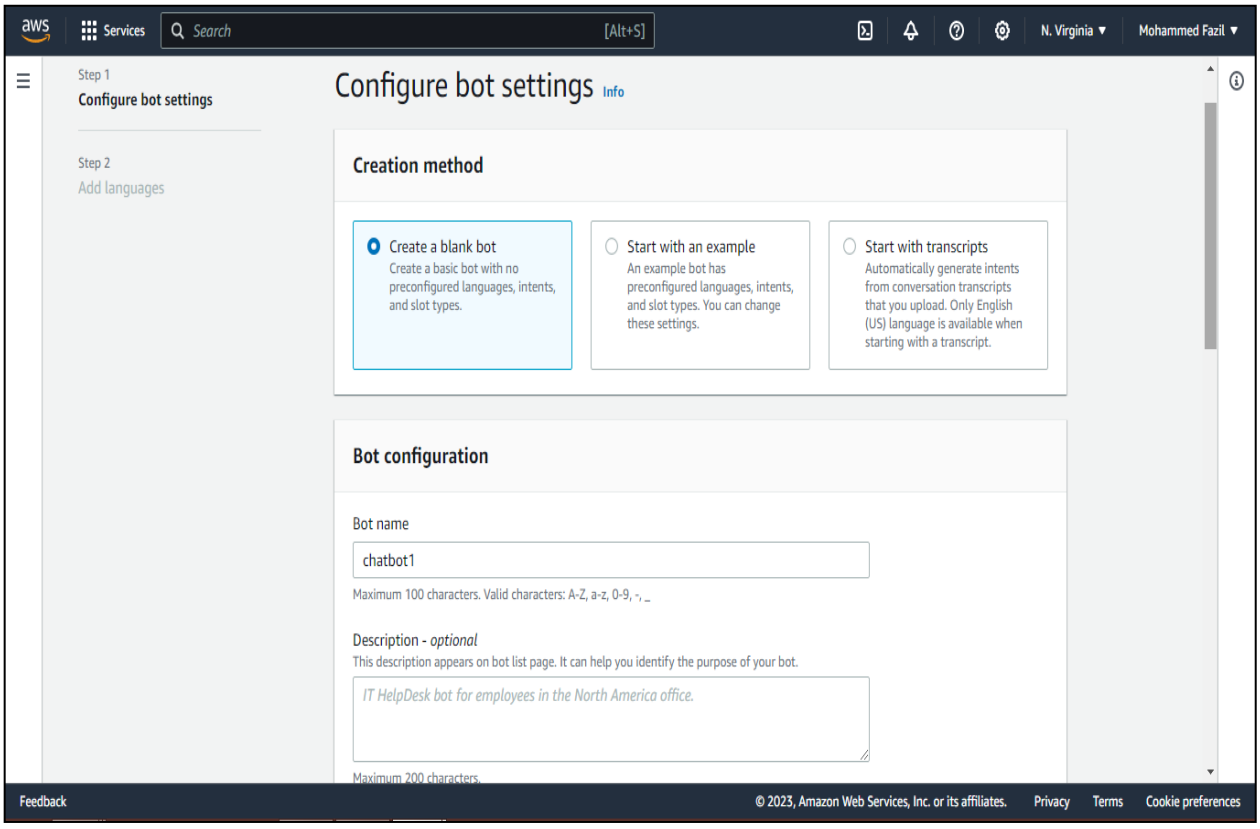


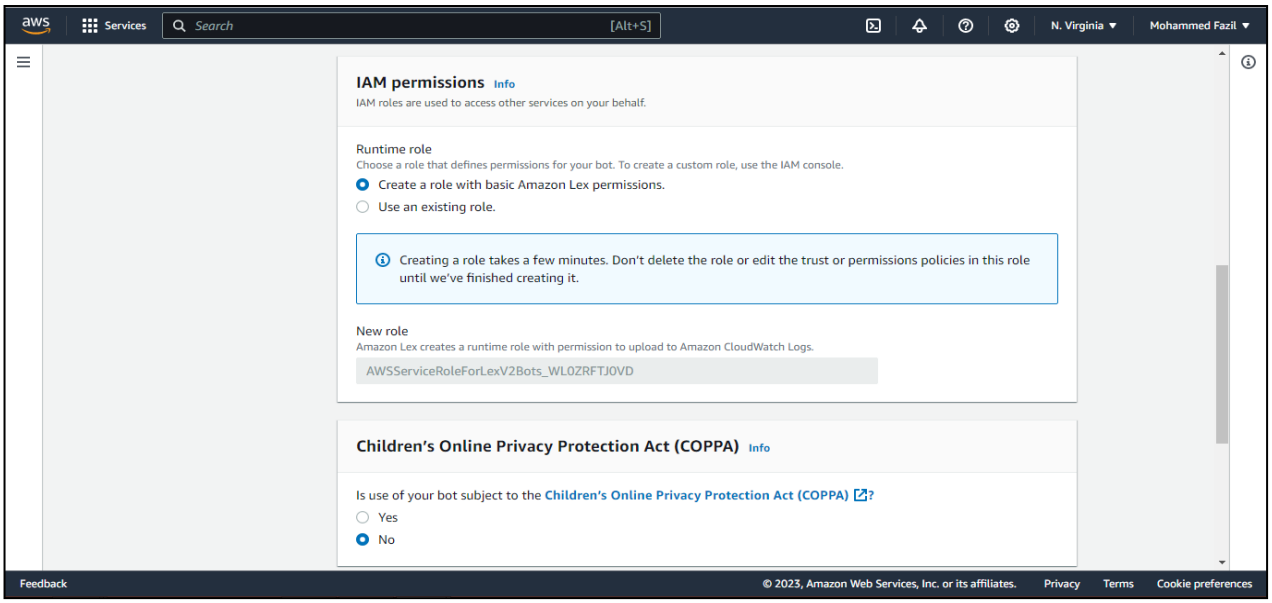


Step 2: Click on create bot, next page will open and fill the required details.

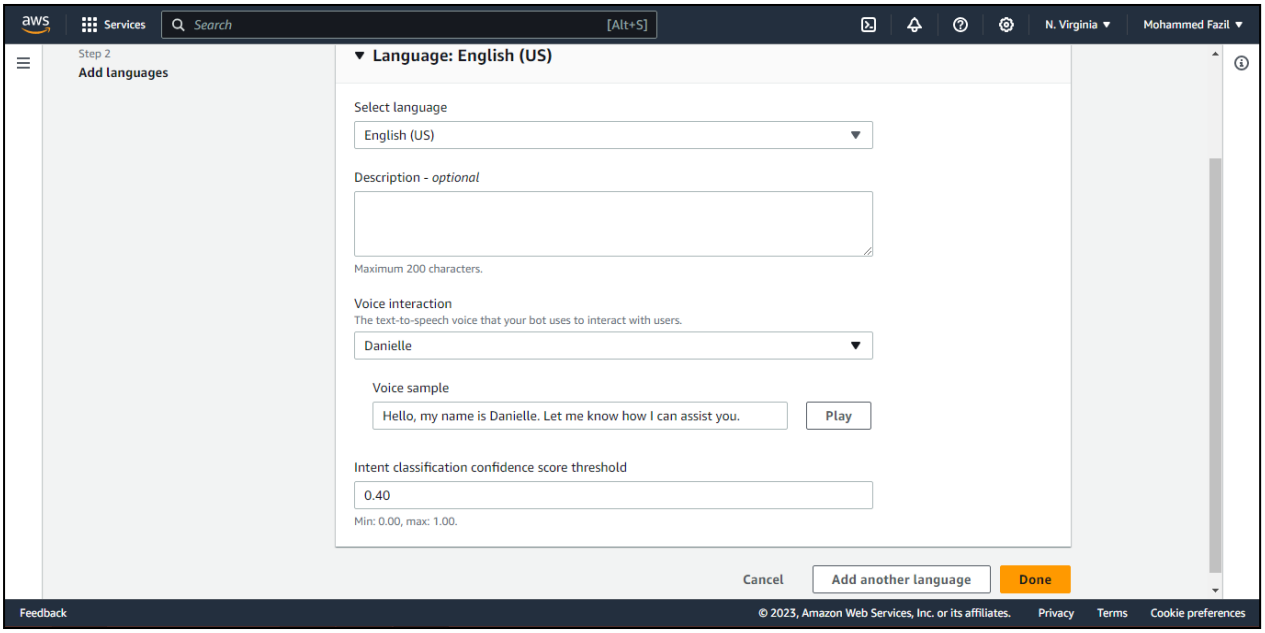


Step 3: The creation method choose on create a blank method, type the bot name should be alphabet and Numbers like chatbot1 , IAM permissions select on new role , Children's Online Privacy Protection Act (COPPA) select on yes, and Idle session timeout set on 5 minutes.

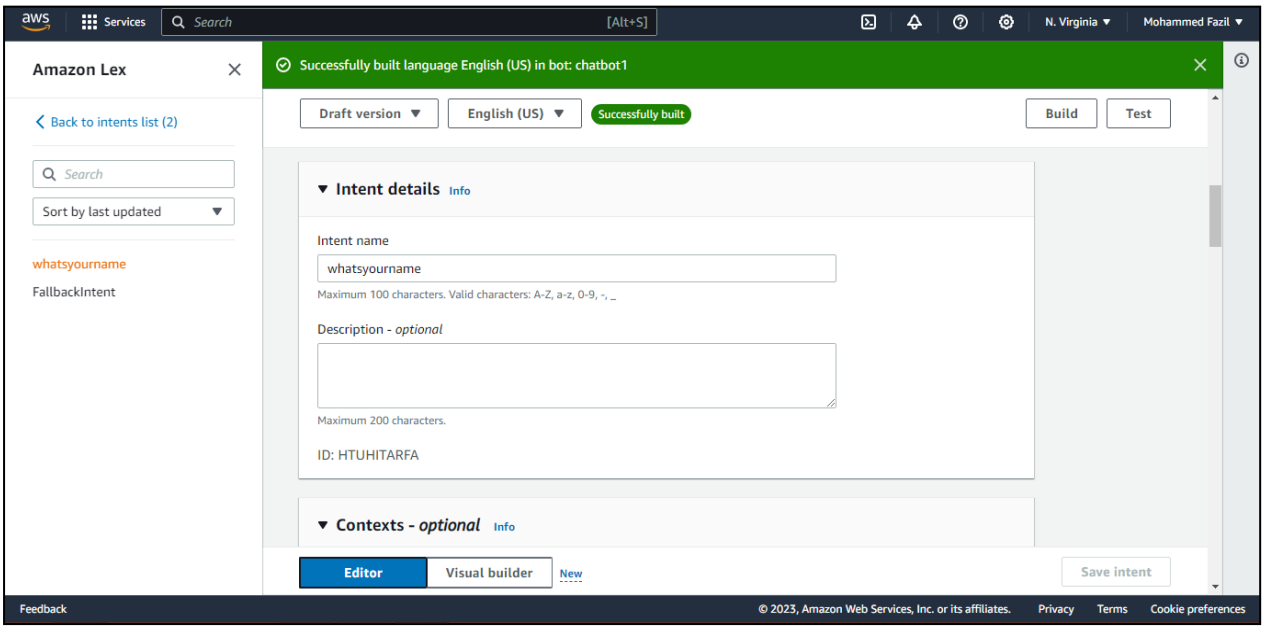


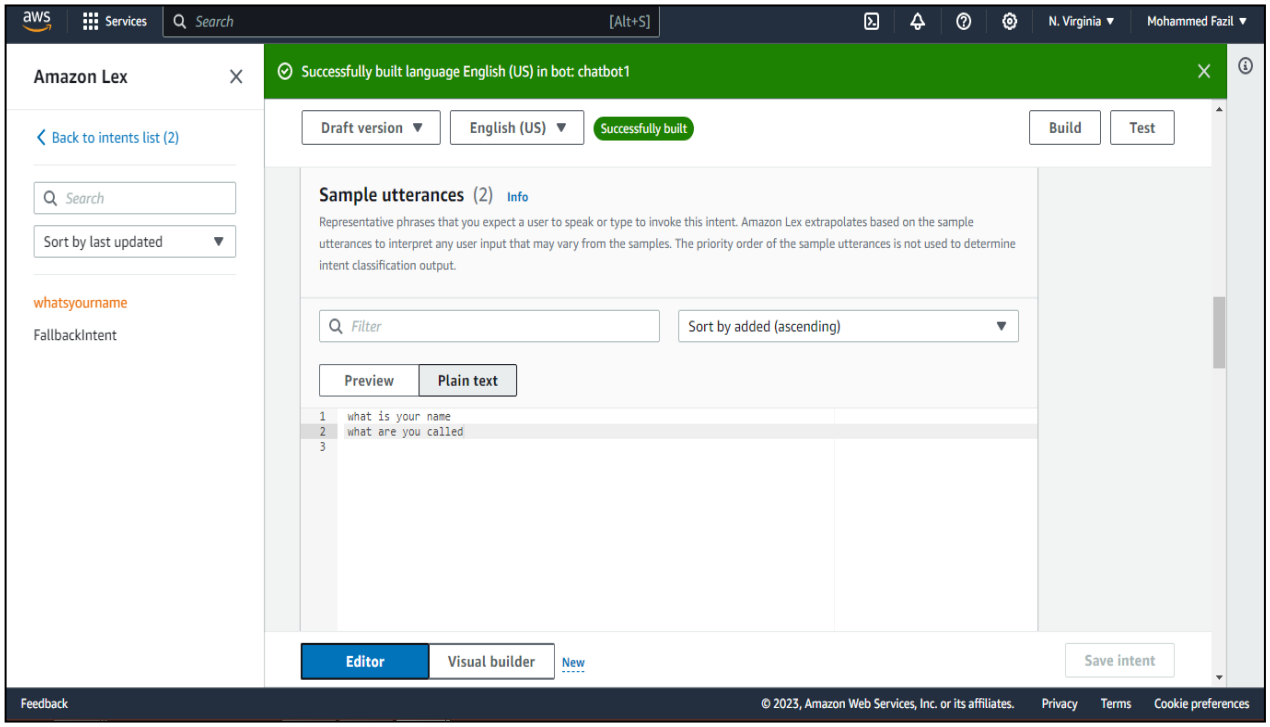


Step 4: Click on next. And click on done.

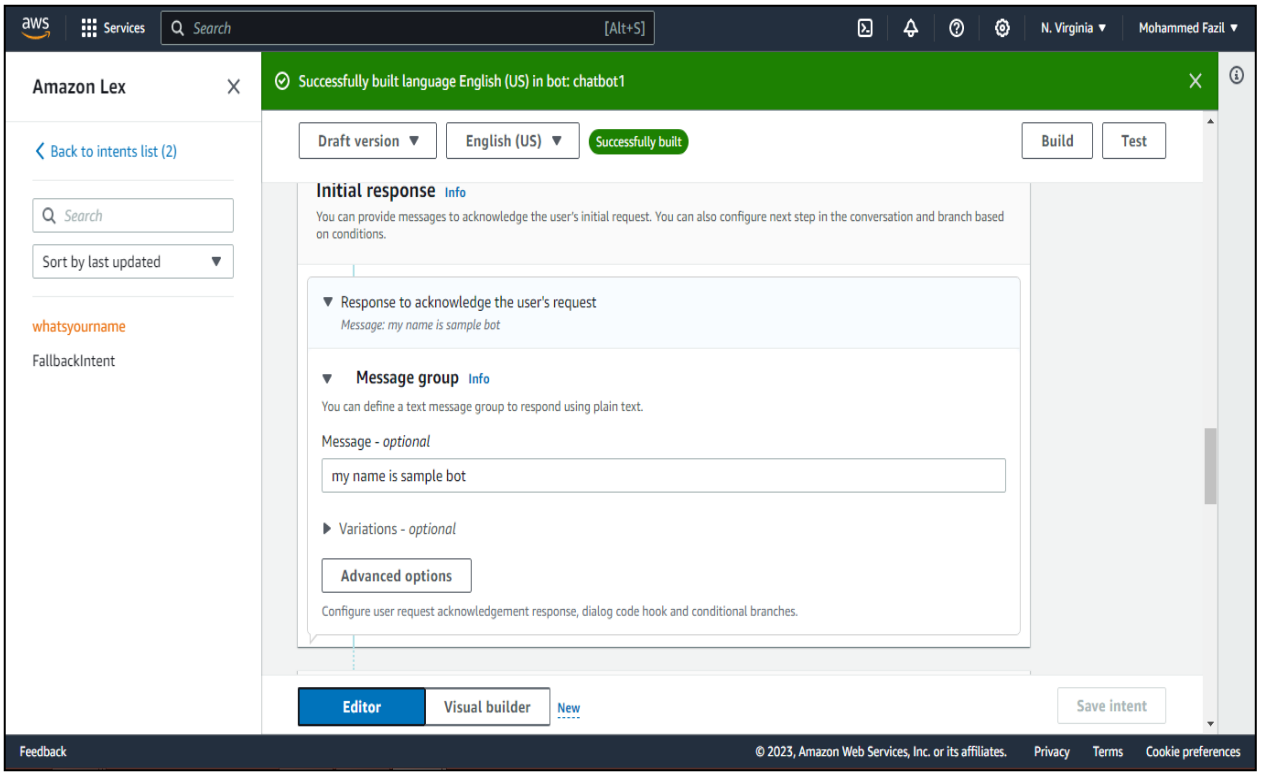


Step 5: Once you click on done, now enter the intent name, sample utterances and the message.





Step 6: Now click on save intent. And then click on build.



Step 7: Click on test to your chat bot.