C.ABDUL HAKEEM COLLEGE OF ENGINEERING & TECHNOLOGY

Hakeem Nagar, Melvisharam -632509,Vellore District,TamilNadu, India.
(Approved by AICTE, New Delhi and Affiliated to Anna University, Chennai) (Regd. Under Sec 2(F) & 12(B) of the UGC Act 1956)

Name of the Candidate:

Year: III **Semester:** VI **Degree/Branch:** B.Tech./IT

Subject Name: Cloud Service Management **Sub.Code:**CCS336

University Register Number:

CERTIFICATE

Certified that this is the bonafide record of work done by the above student in **CCS336-CLOUD SERVICE MANAGEMENT** during the year 2023 - 2024.

Signature of Head of the Department **Signature of Lab In-charge**

Submitted for the University Practical Examination held on _____

EXAMINERS

Date: _____ **Centre code:**5106

Internal: _____ **External:** _____

INDEX

EXP NO:1 DATE:	CREATE A CLOUD ORGANISATION IN AWS/GOOGLE CLOUD OR ANY EQUIVALENT OPEN SOURCE SOFTWARE LIKE OPENSTACK, EUCALYPTUS,OPEN NEBULA WITH ROLE BASED ACTIONS

AIM:

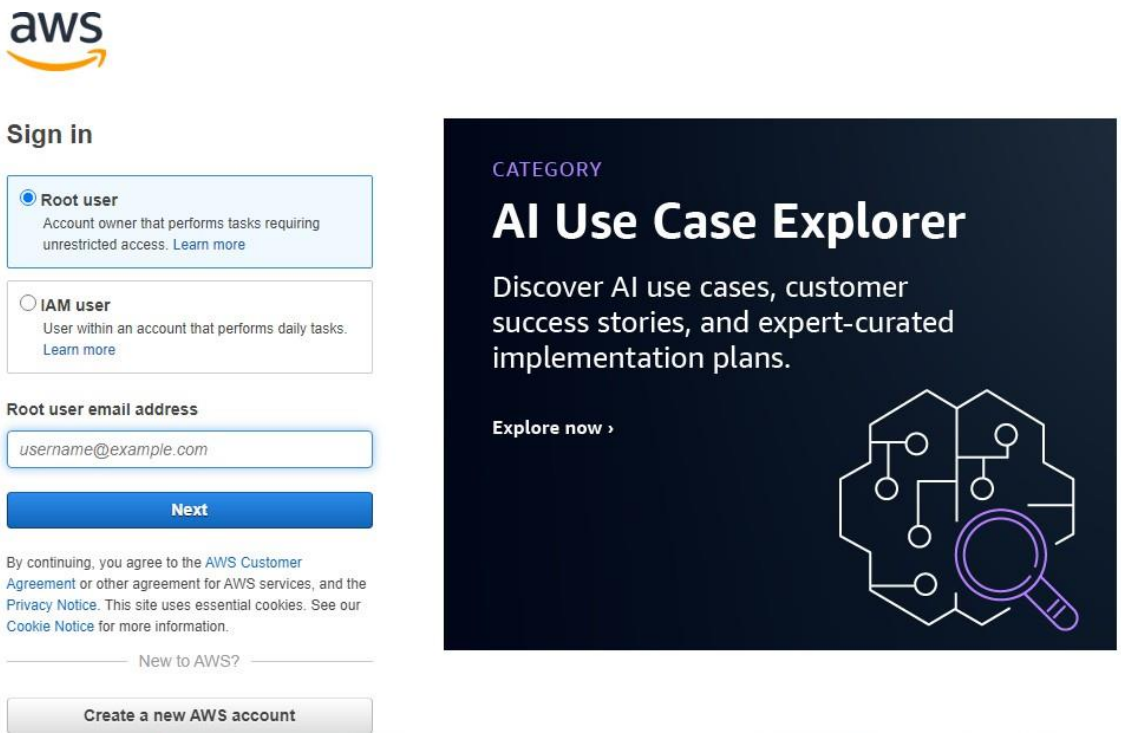
To create a cloud organisation in aws/google cloud or any equivalent open source softwares like openstack,ecalyptus,open nebula with role based actions.

PROCEDURE:

1. Create the user account in aws.
2. Create the user and corresponding user group.
3. Add the user to the user group with role based access.

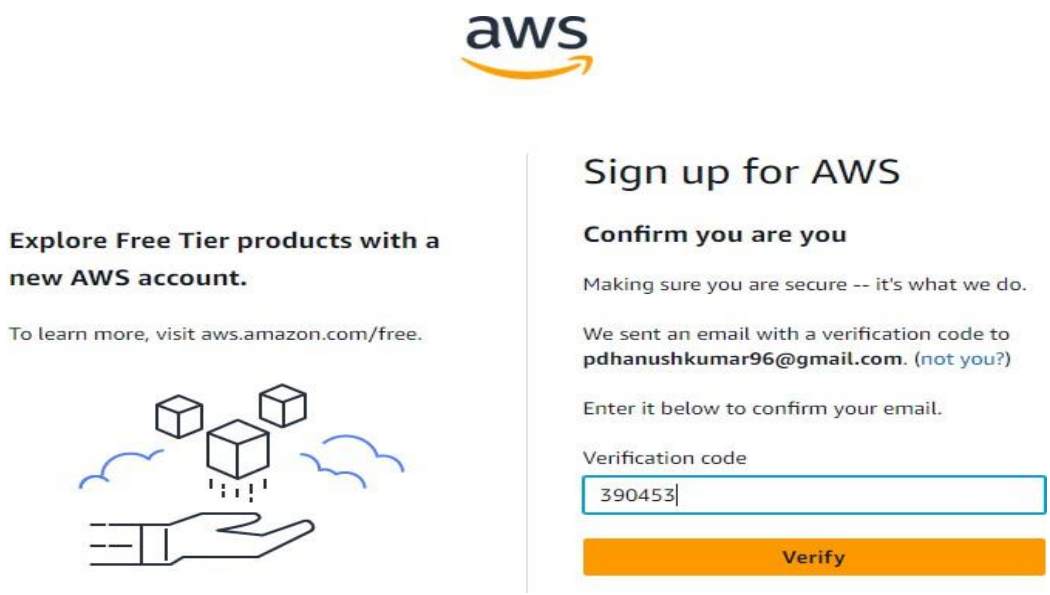
1. CREATING THE USER ACCOUNT IN AWS

1(a). Visit the aws management console.

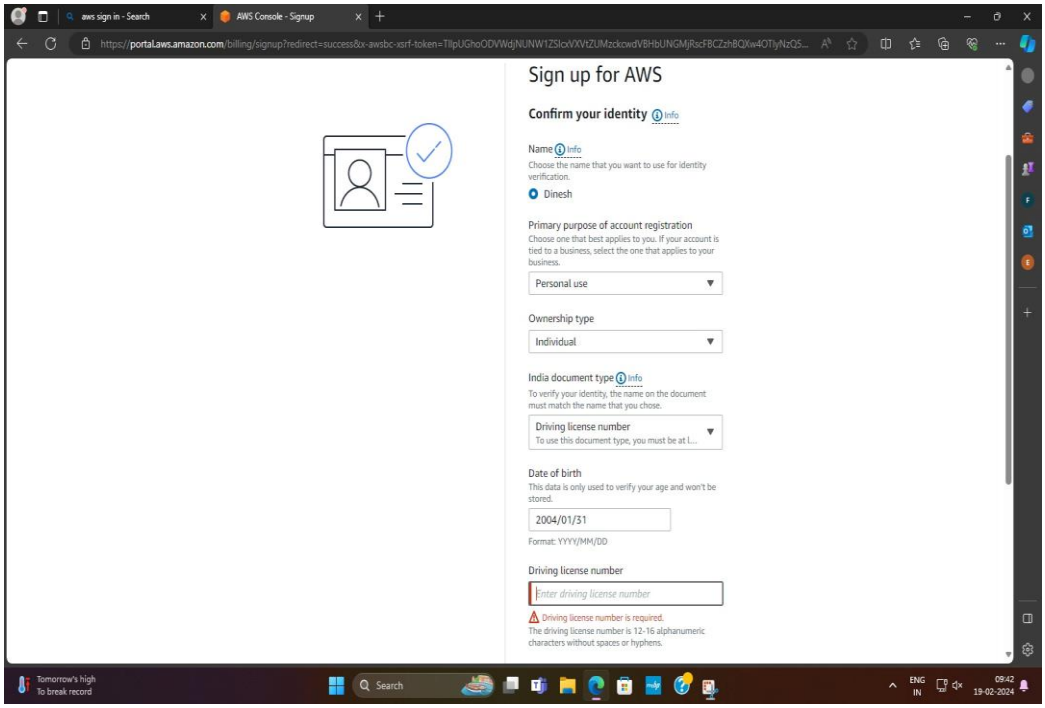


Click on the create a new AWS account.

1(b). For creating the account enter the E-mail Id and verify your id using verification code sent by AWS to the corresponding E-mail Id.



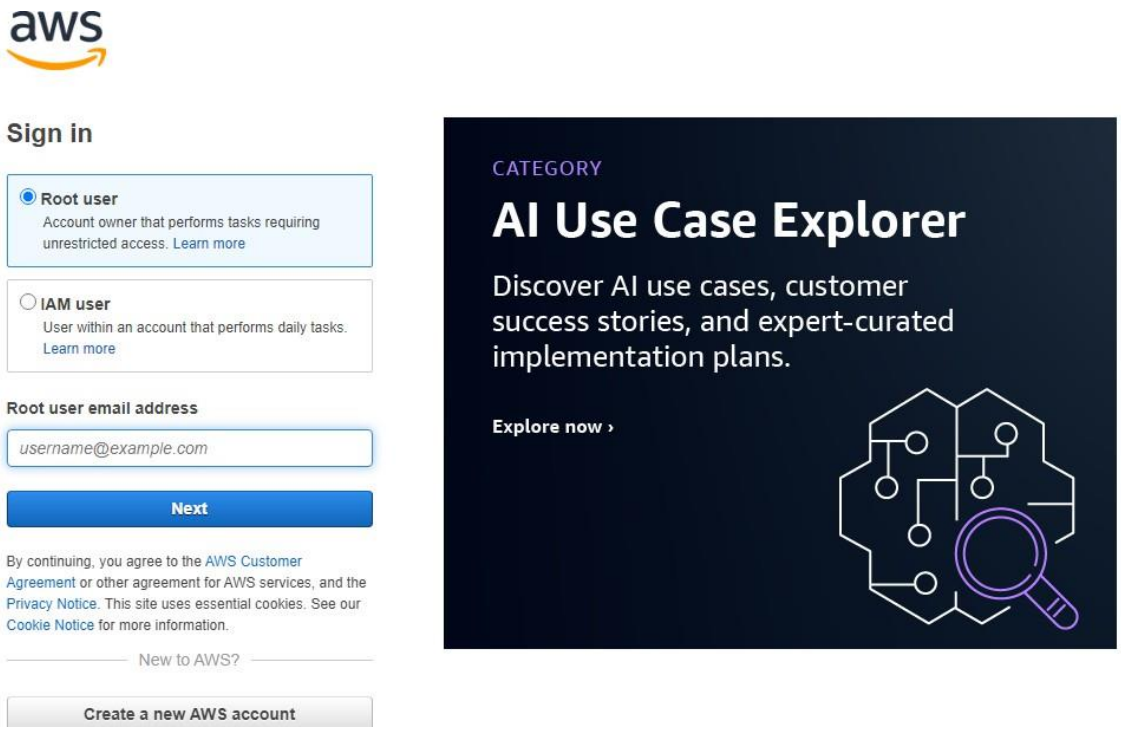
1(c) . After the verification ,Enter your details and then fill up your account details for the Identity verification. It can charge 2rupee from your account for the card verification.



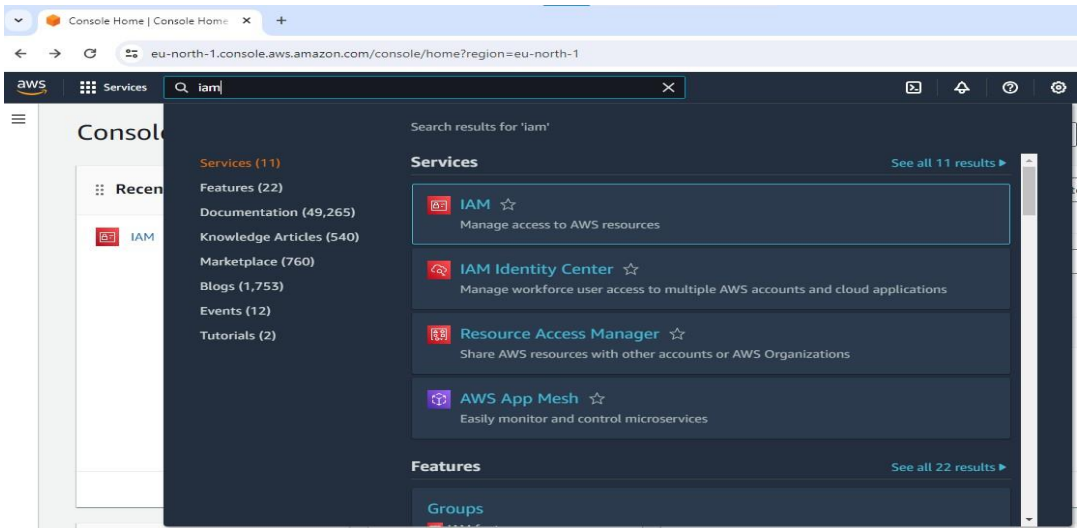
Then the aws account is created.

2. CREATE THE USER AND CORRESPONDING USER GROUP.

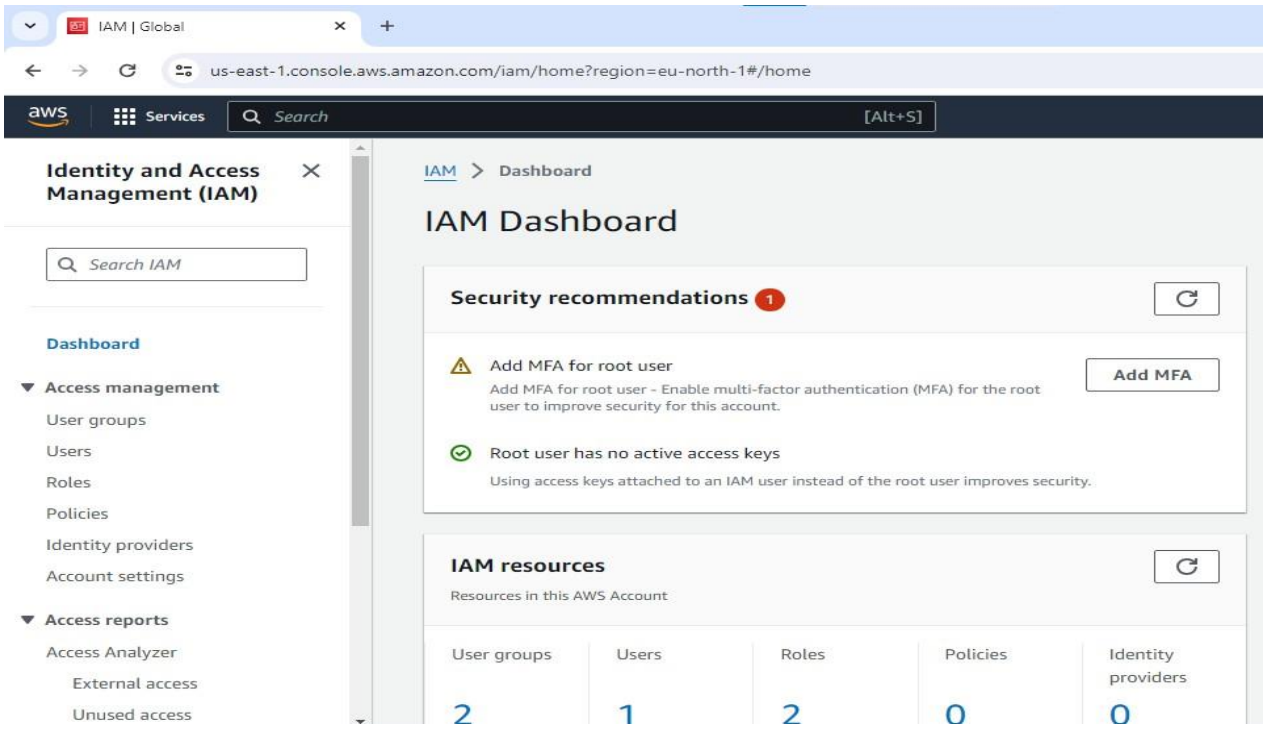
2(a) .Open the aws management console in any browser and log in as the Root user using your corresponding e-mail id and the password.



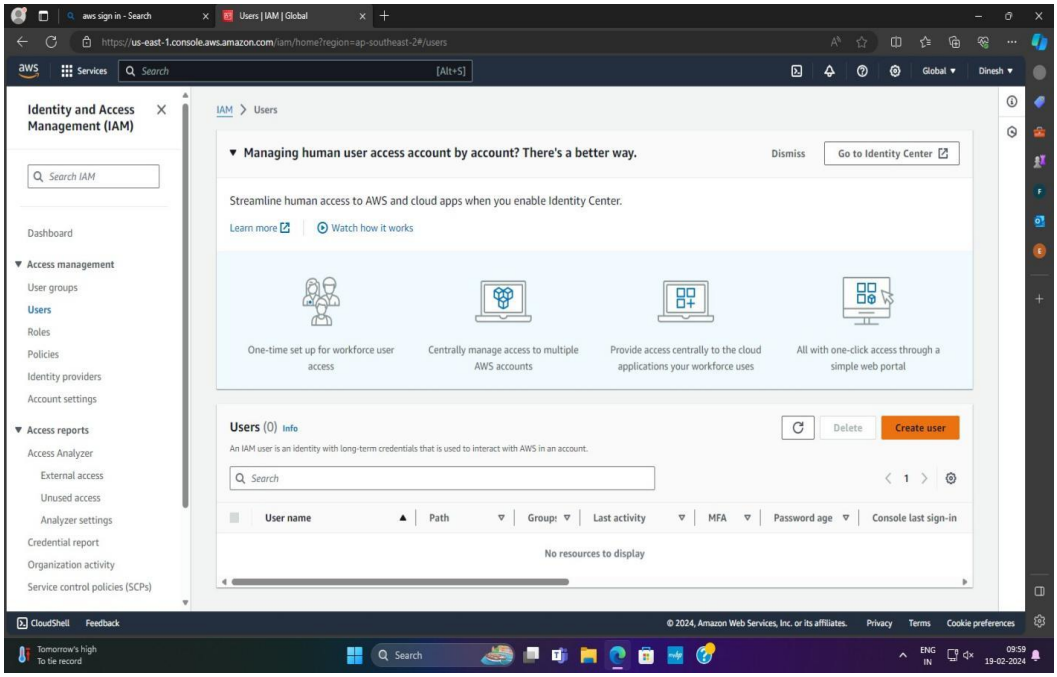
2(b). After entering into the aws management consol then search for IAM in the services.



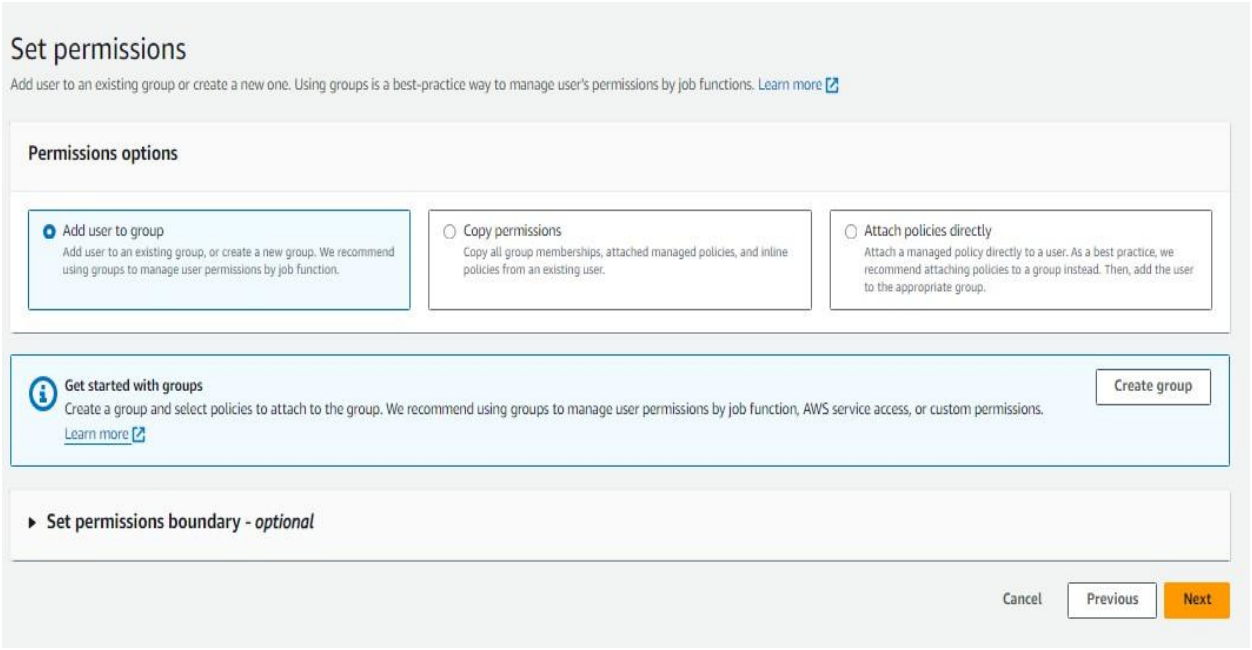
2(c). The dash board of the IAM is displayed as follows.



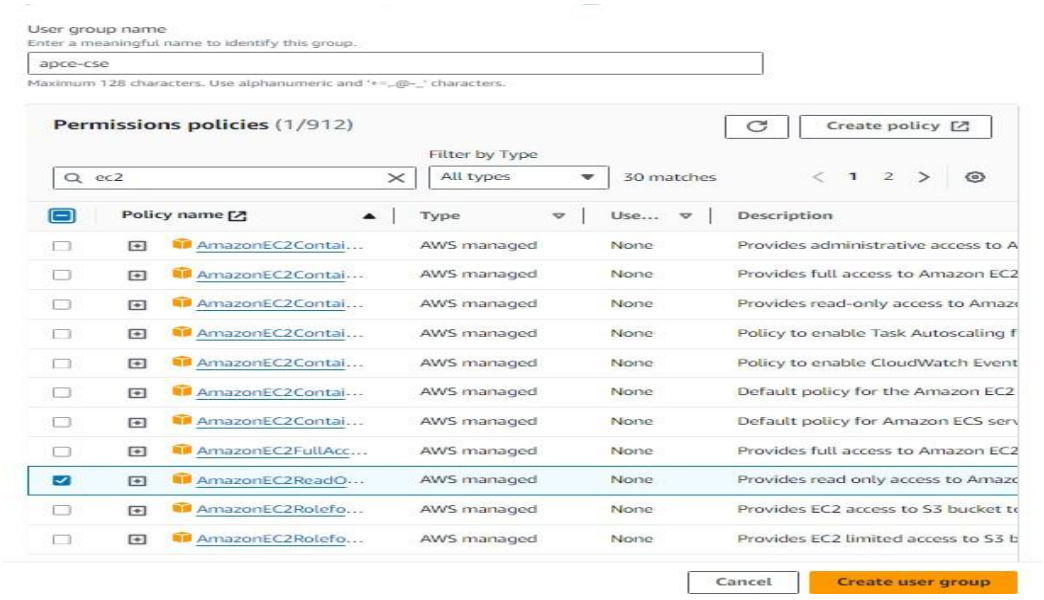
2(d). Then click on the user and then under this, click on the create new user and enter the user name.



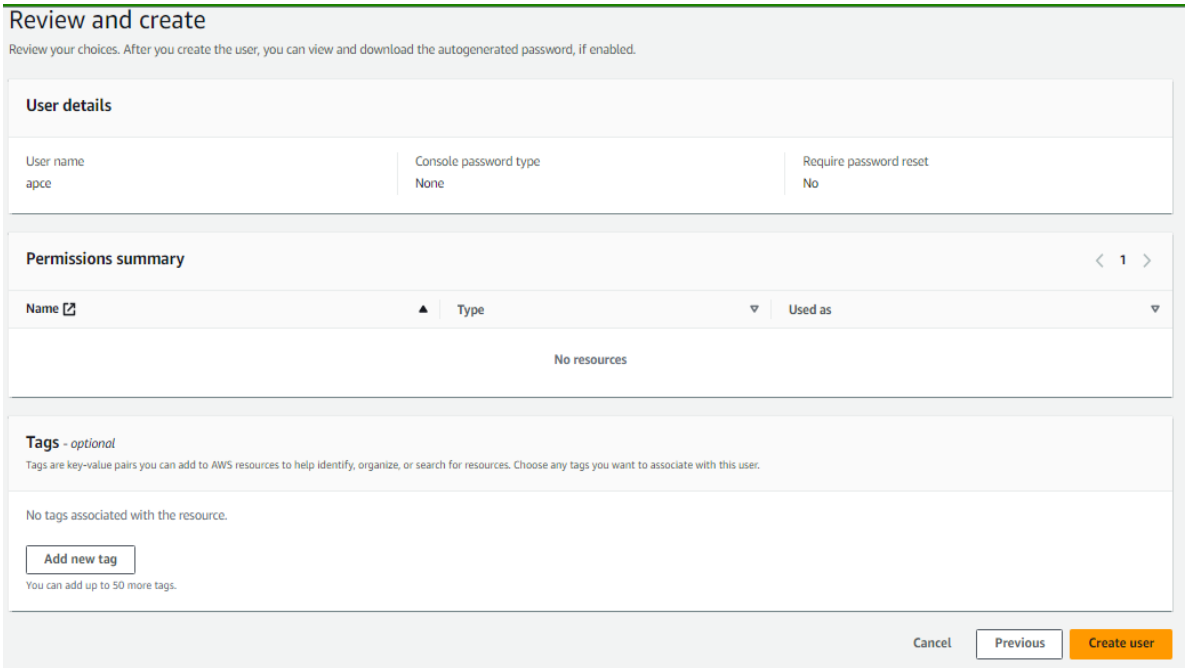
2(e). Then click on the next button.Click on the add user to the group, After that click on the create group option.



2(f). Then the create user group with the user group name.After entering the name you need to specify the permission policies as EC2 with read only access.

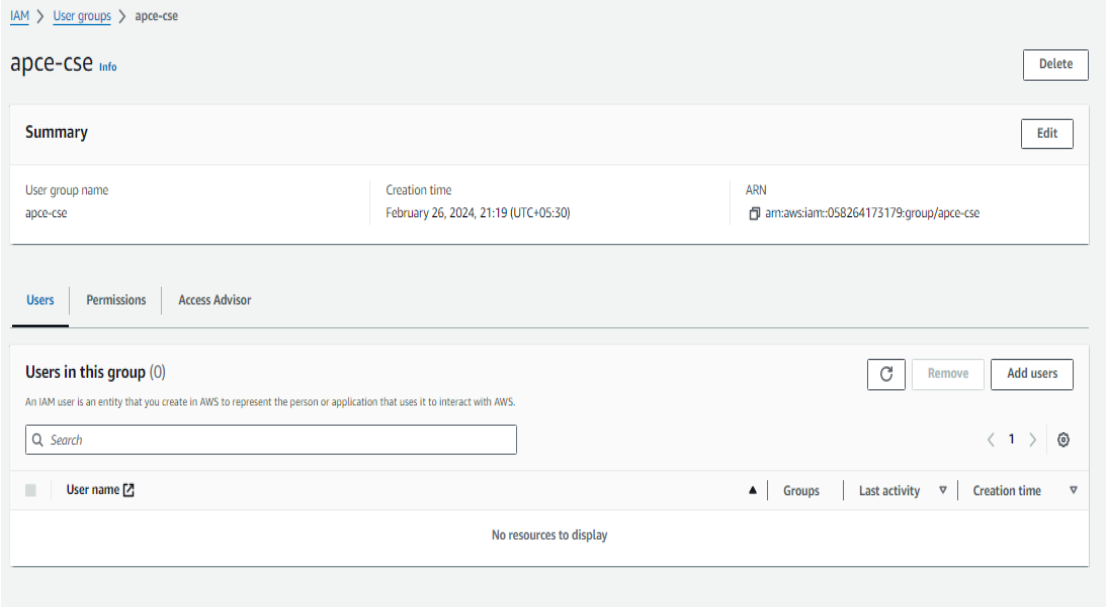


2(g). Click on the create user group and you will be redirected into the review and create page .Under this click on the create user, then the user will be created.

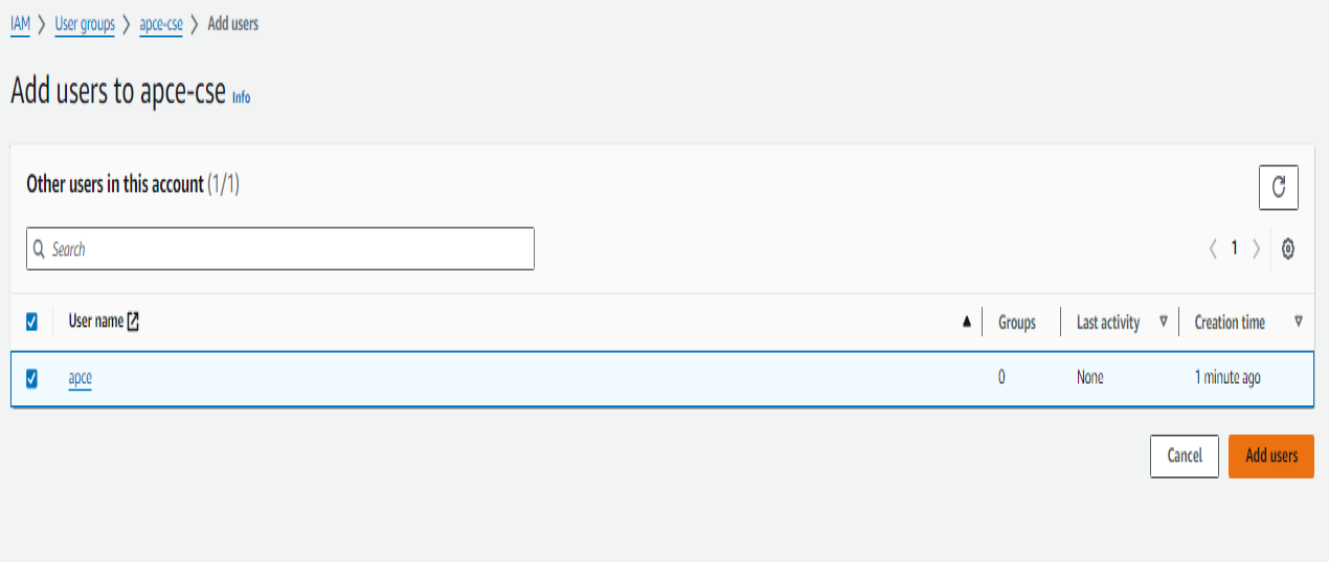


3. ADD THE USER TO THE USER GROUP WITH ROLE BASED ACCESS.

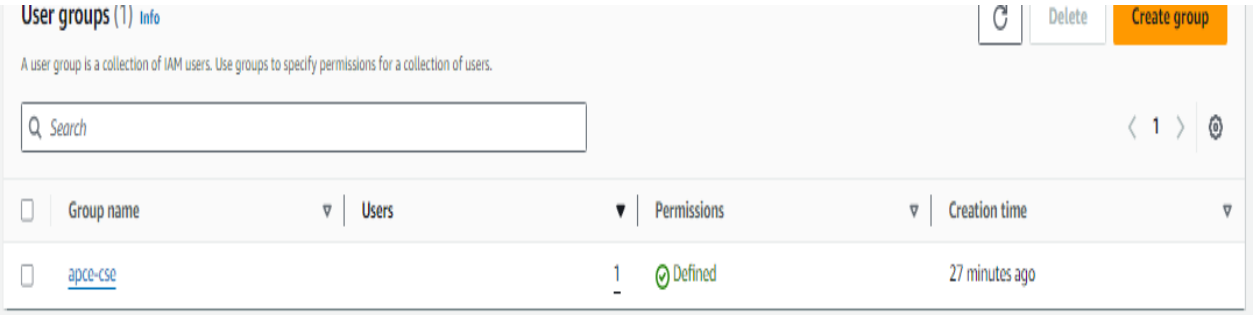
3(a). To add the users into the group click on the add users



3(b). Then add the existing group to the user group and click on Add users option, now the user is added into the group with the role based action.



3(c) We can view the summary of the user group which is created, by clicking on the user groups on the dashboard.



RESULT:

Thus the organization was created in the amazon aws with the role based actions.

EXP NO:2 DATE:	CREATE A COST-MODEL FOR A WEB APPLICATION USING VARIOUS SERVICES AND MAKE AN ANALYSIS FOR COST-BENEFIT.

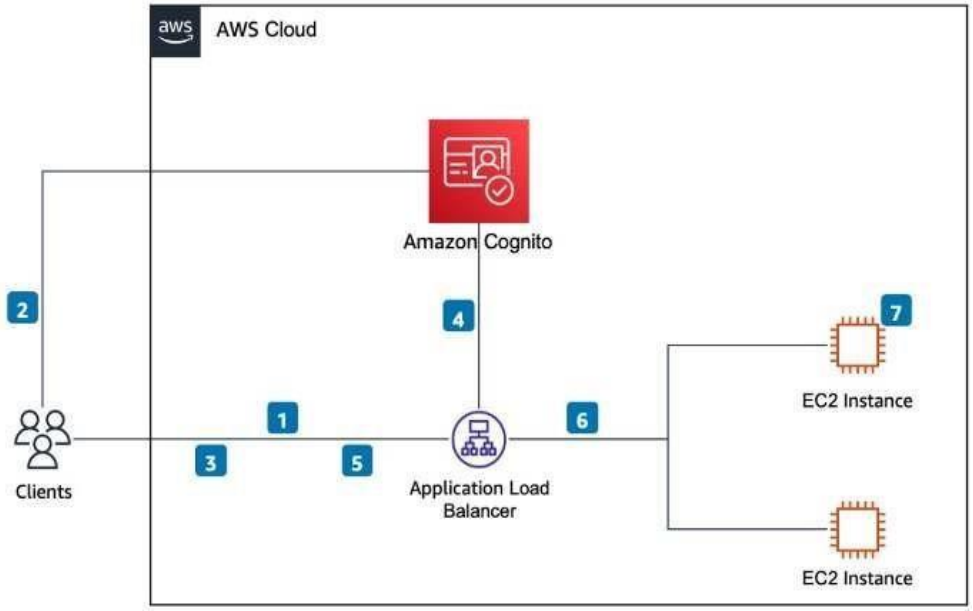
AIM:

To create a Cost-model for a web application using various services and make a analysis for Cost-benefit.

PROCEDURE:

Creating a cost-model for a web applicationon in AWS involves estimating the costs of using various AWS services for the application. Here's a general process to create a cost-model and do cost-benefitt analysis:

1. **Identify the AWS services used by the web application:**
   Some common services used by web applications include Amazon S3, Amazon EC2, Amazon RDS, Amazon API Gateway, AWS Lambda, Amazon DynamoDB, Amazon CloudFront, and Amazon SNS.



2. **Estimate the costs of each service:**
   You can use the AWS Pricing Calculator to estimate the costs of each service. The pricing calculator allows you to enter the specifics of your usage, such as the number of instances, storage size, and data transfer.

3. **Create a cost-model:** Once you have estimated the costs of each service, you can create a costmodel that summarizes the total costs. You can use a spreadsheet or a cloud cost management tool to create the cost-model.
4. **Do cost-benefit analysis:** after creating the cost-model, you can do a cost-benefit analysis to determine if the benefits of using AWS services outweigh the costs. You can compare the costs of using AWS services to the costs of running the application on-premises or using a different cloud provider.

Python code:

```
import boto3
session = boto3.Session(
aws_access_key_id='YOUR_ACC
ESS_KEY',
aws_secret_access_key='YOUR_S
ECRET_KEY',
region_name='us-east-1'
)

# Create a Cost Explorer client
cost_explorer = session.client('ce')

Model time_period = { 'TimeUnit':
    'MONTHS', 'Start': '2022-01-01',
    'End': '2022-12-31'
}
model granularity = 'DAILY'

model metrics = ['BlendedCost',
'UsageQuantity']
```

```
model group_by = [{'Type':
'DIMENSION', 'Key':
'SERVICE'}] # Get the cost and usage
data response =
cost_explorer.get_cost_and_usage (
    TimePeriod=time_period,
    Granularity=granularity,
    Metrics=metrics, GroupBy=group_by
)# Print the cost and usage data
print(response)
```

Output:

```
{

    'ResultsByTime': [
        {
            'TimePeriod': {
                'Start': '2022-01-01',
                'End': '2022-12-31',
                'TimeUnit': 'MONTHS'
            },
            'Groups': [
                {
                    'Keys': [
                        'AmazonEC2'
                    ],
                    'Metrics': { 'BlendedCost':
                        {
                            'Amount': '1234.56',
                            'Unit': 'USD'
                        },
                        'UsageQuantity': {
                            'Amount': '1000.0',
                            'Unit': 'Hours'
```

```
                }
            }
        },
        {
            'Keys': [
                'AWSLambda'
            ],
            'Metrics': { 'BlendedCost':
                {
                    'Amount': '789.0',
                    'Unit': 'USD'
                },
                'UsageQuantity': {
                    'Amount': '5000000',
                    'Unit': 'requests'
                }
            }
        }
    ]
  }
],
'ResponseMetadata': {
    'RequestId': 'abcdefg-1234-5678-
    90ab-cdefghijkl',
    'HTTPStatusCode': 200,
    'HTTPHeaders': {
        'content-type':
        'text/xml;charset=UTF-8',
        'content-length': '1234',
        'date': 'Tue, 15 Feb 2022
        12:34:56 GMT'
```

```
},
'RetryAttempts': 0
```

Result

Thus, Cost-model for a web application using various services created and analysis was implemented successfully.

EXP NO:3 DATE:	CREATE ALERTS FOR USAGE OF CLOUD RESOURCES

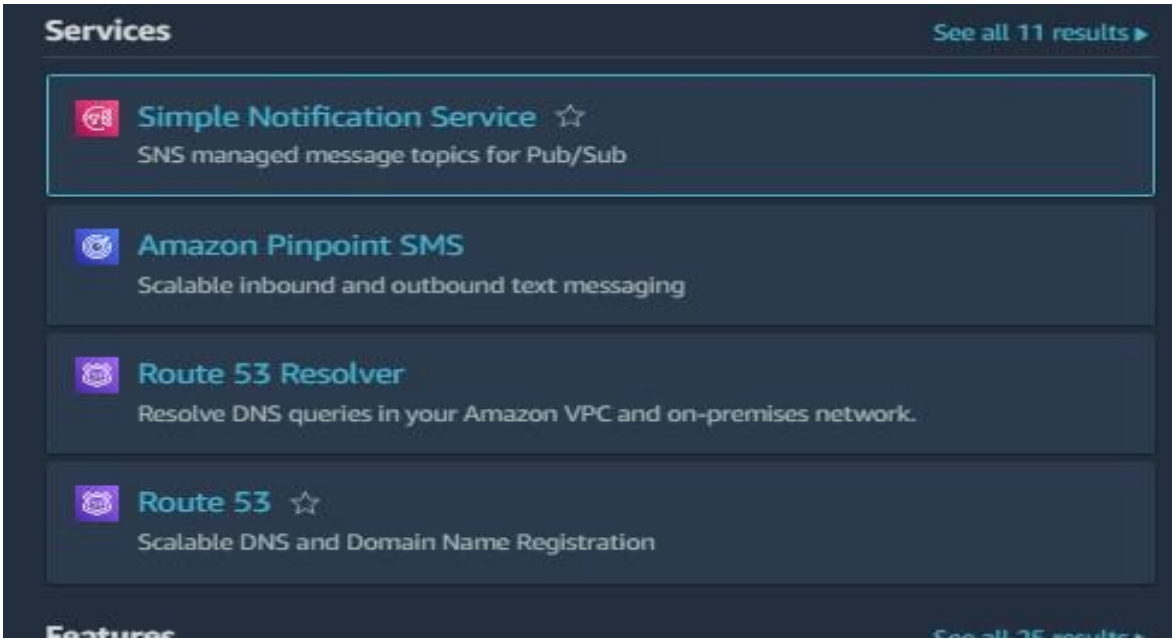
AIM:

To create alerts for usage of cloud resources.

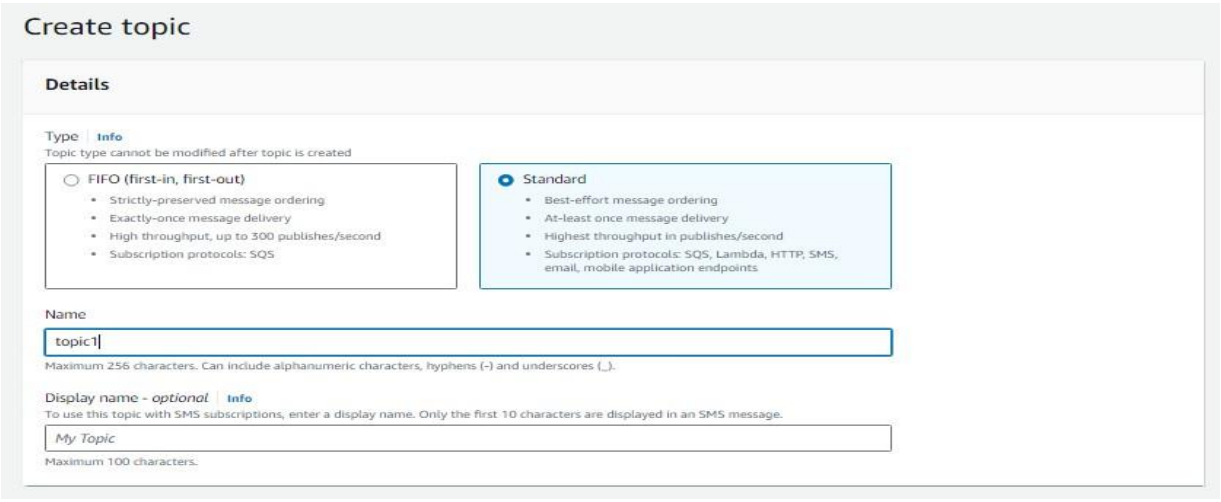
PROCEDURE:

Step-(1):Login into the aws management console.

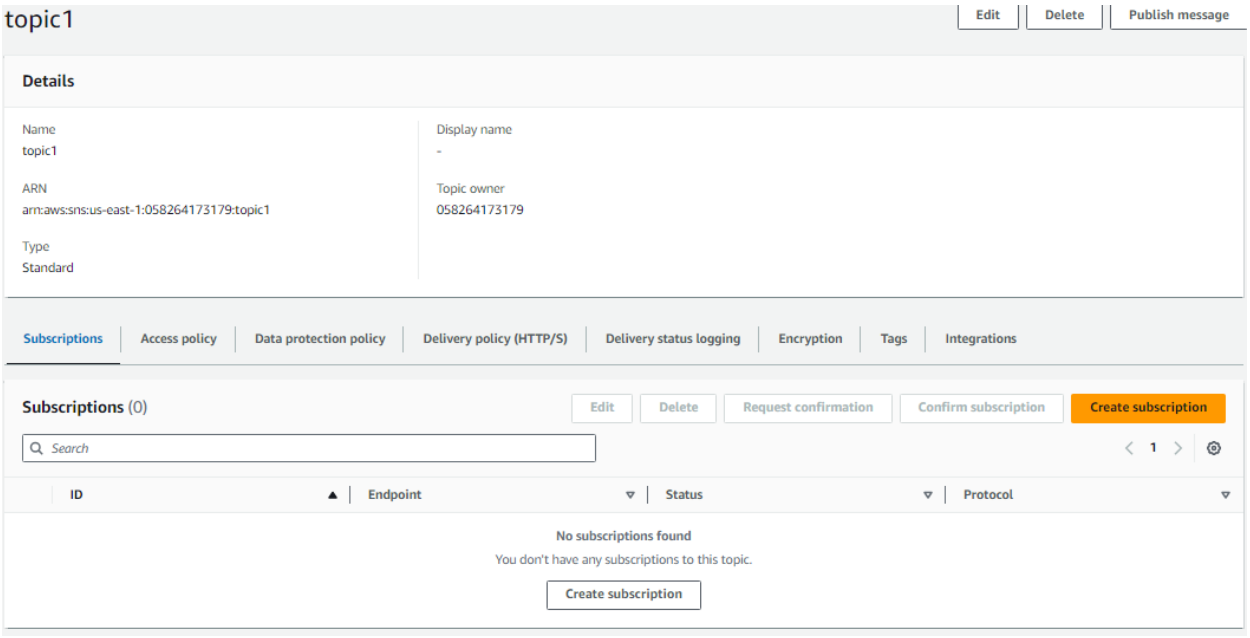
Step-(2):Search for the sns on the services.



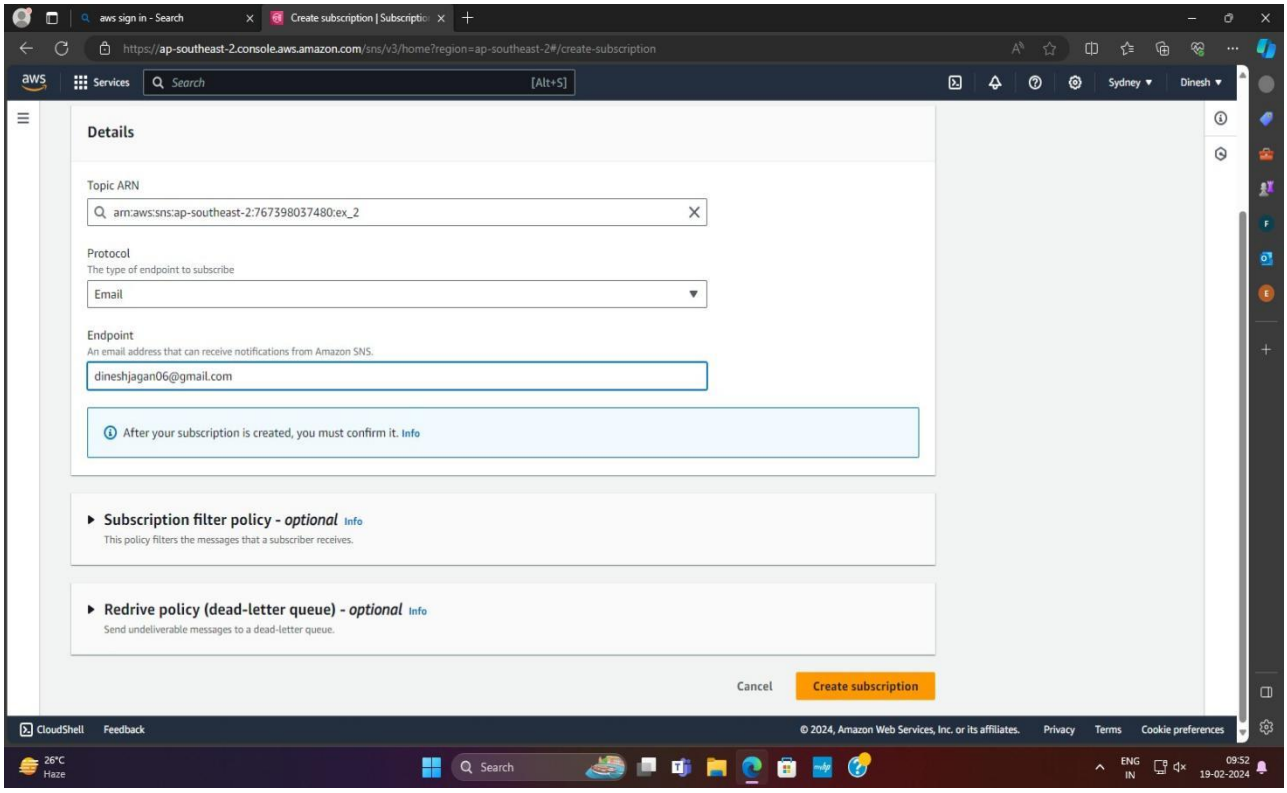
Step-(3): Under the sns click on the create topic with standard type.



Step-(4): Enter the topic name and click create to proceed further,then the topic is created.



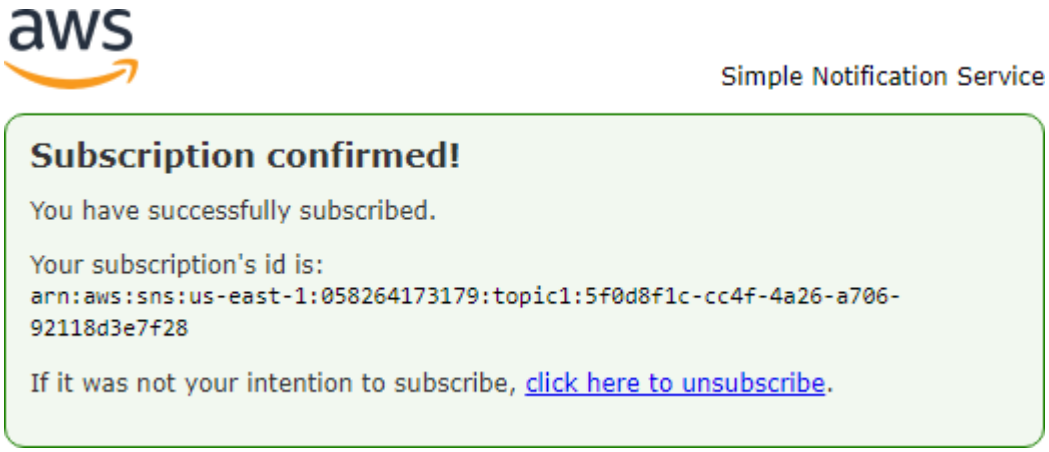
Step-(5): Then the next step is to create the subscription.To do this click on the create subscription.



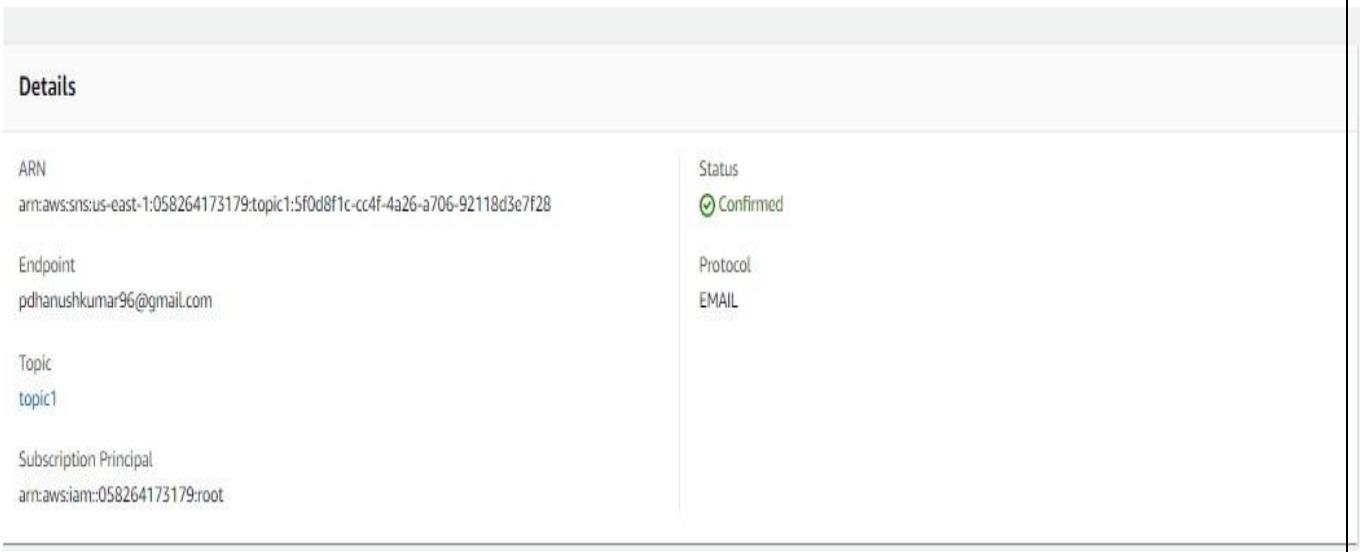
Select the protocol as E-mail and then enter the root e-mail id, then click on the create subscription option.

The conformation mail will be sent after clicking on the create subscription button. Now go to your mail box and check the mail of aws, also click on the confirm subscription Button.

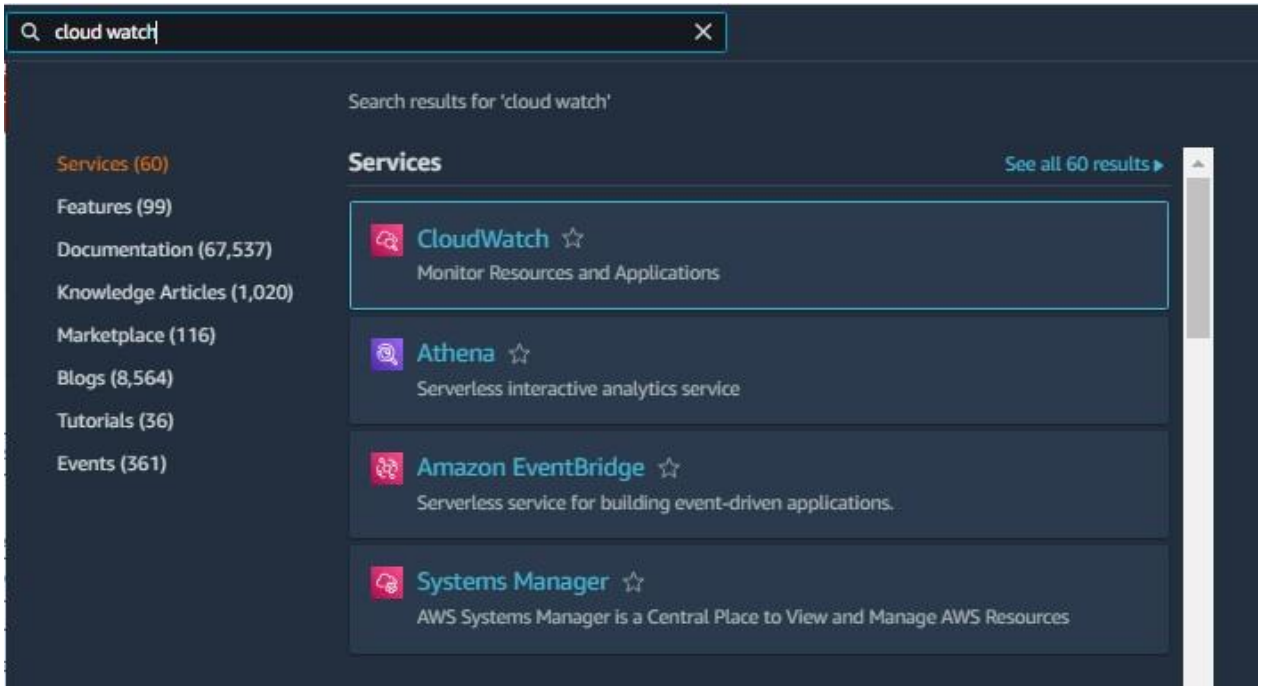
After confirm subscription, We can able to see the following message.



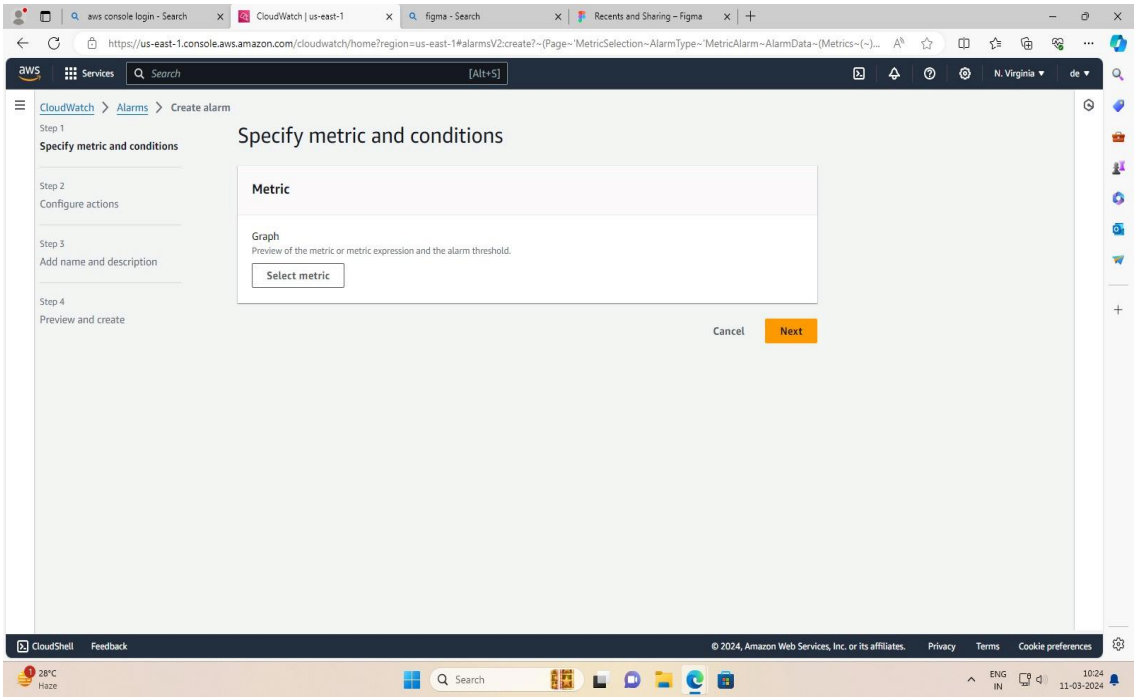
Now the topic has been successfully created with an subscription id as given below



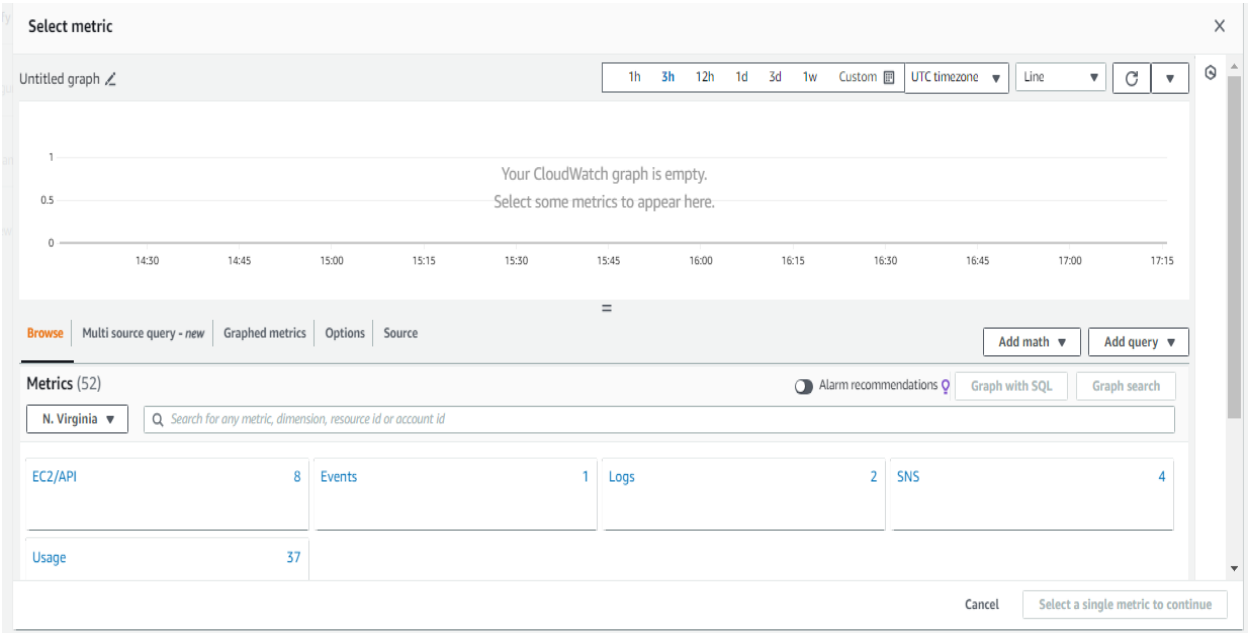
Step-(6): Search for the cloud watch in the services for creating the alarms for alert the billing of the cloud resource usage.



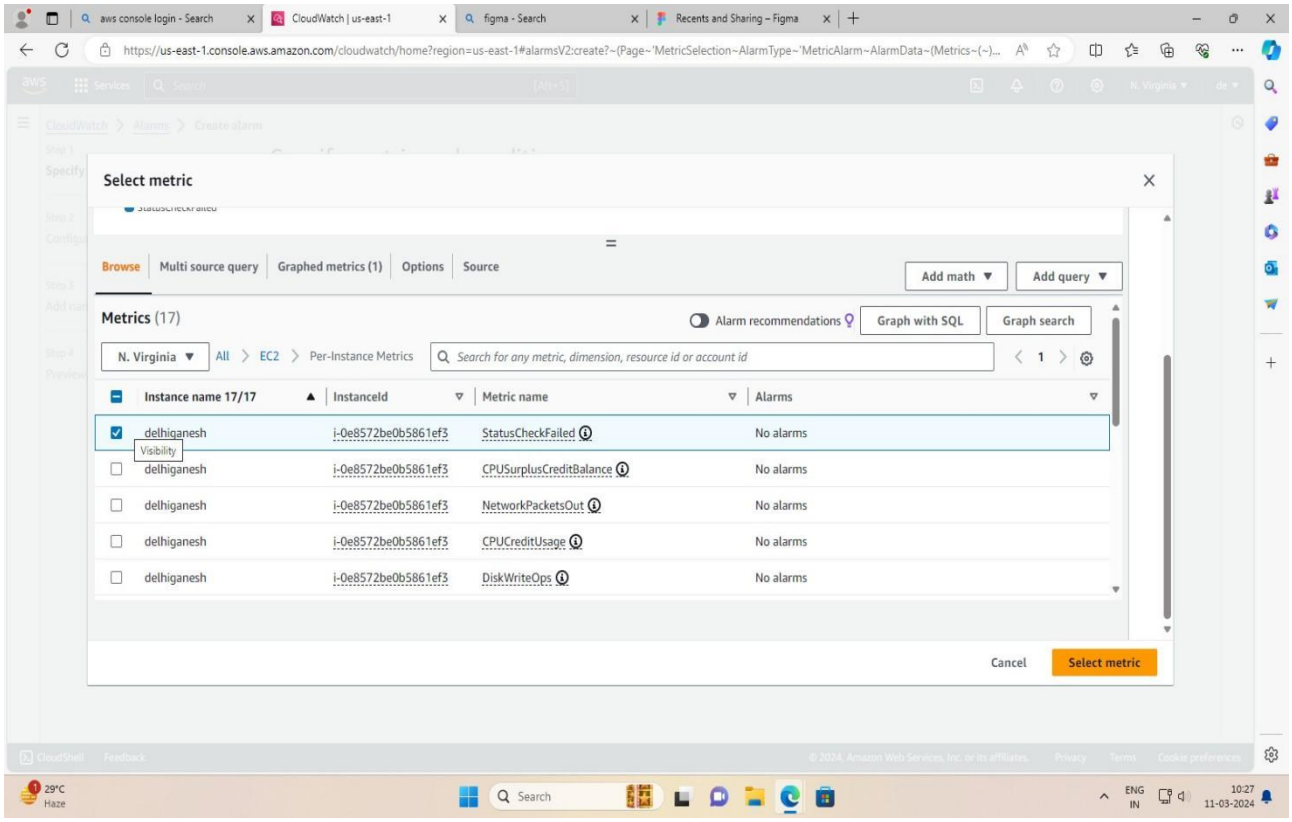
Step-(7): Click on the create alarm option in the cloud watch.Specify the conditions and metric for the alarm.



Step-(8): Click on next and under the select metric click on the sns.

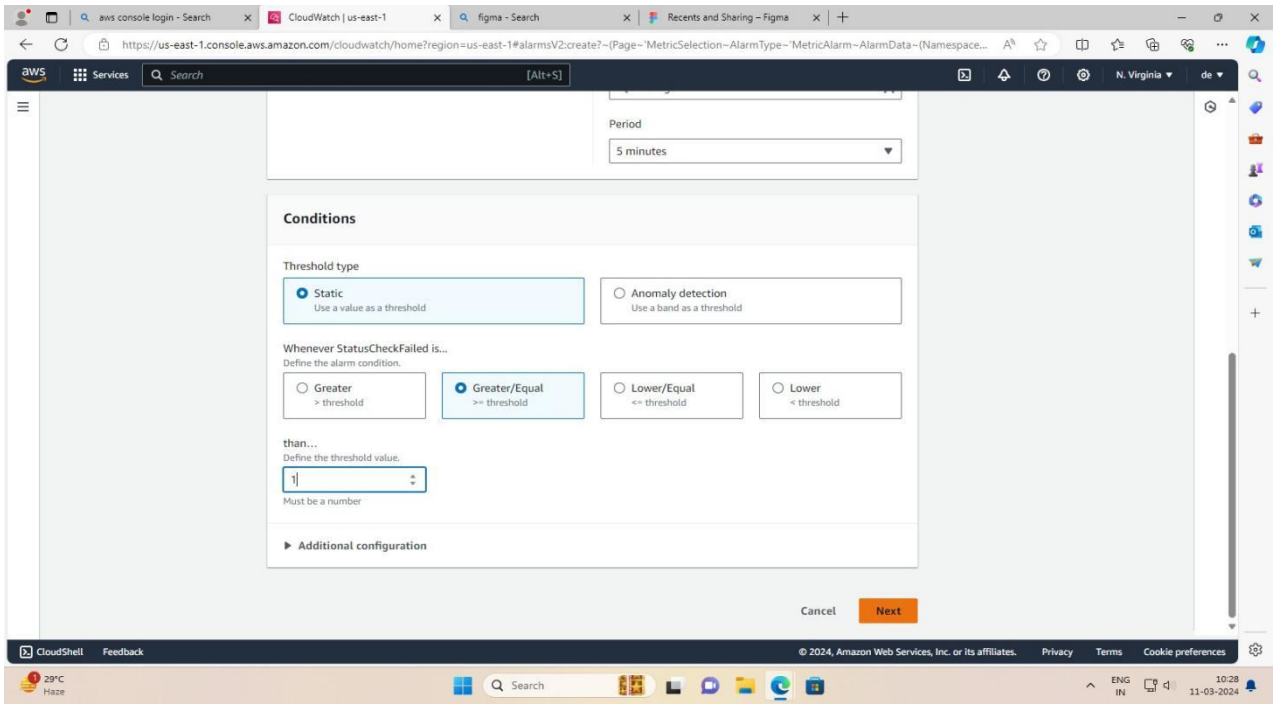


Step-(9): Then the notification option for the alarm.

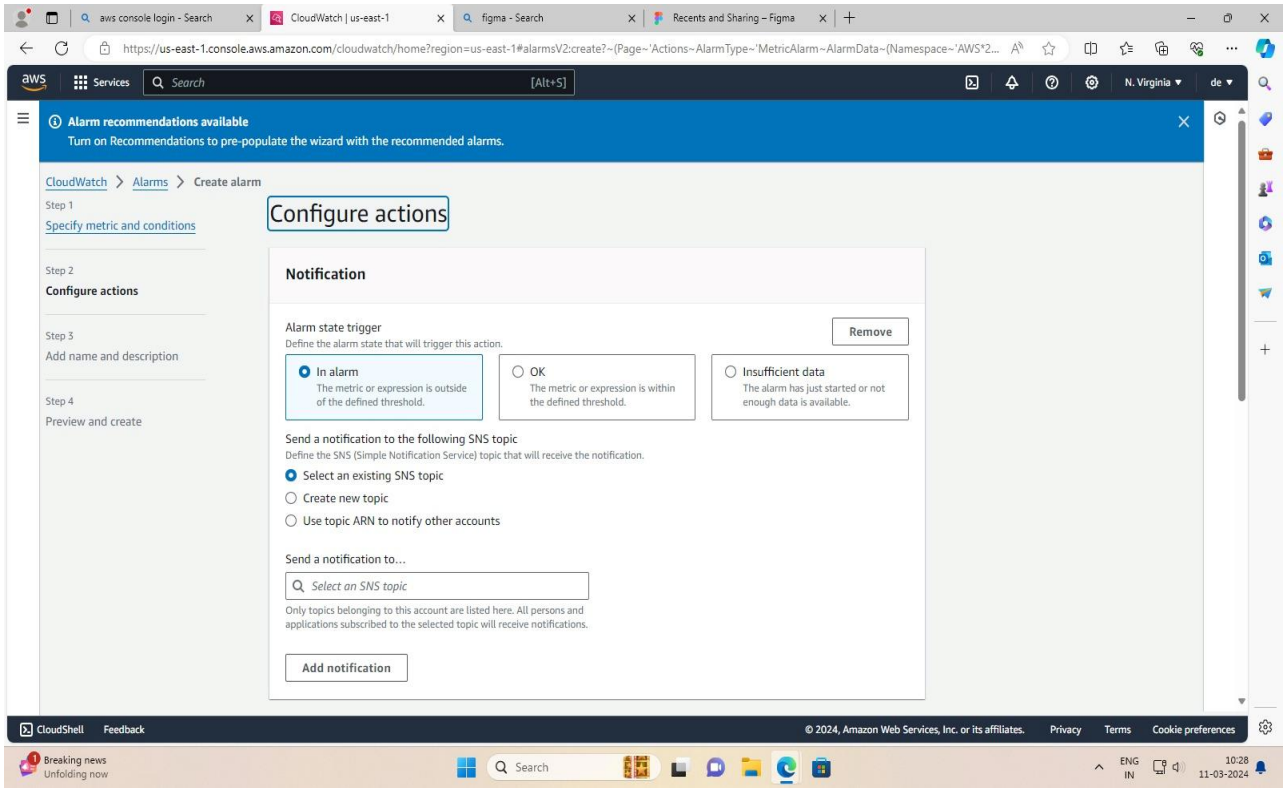


After this selct your metric for the alarm as shown above.

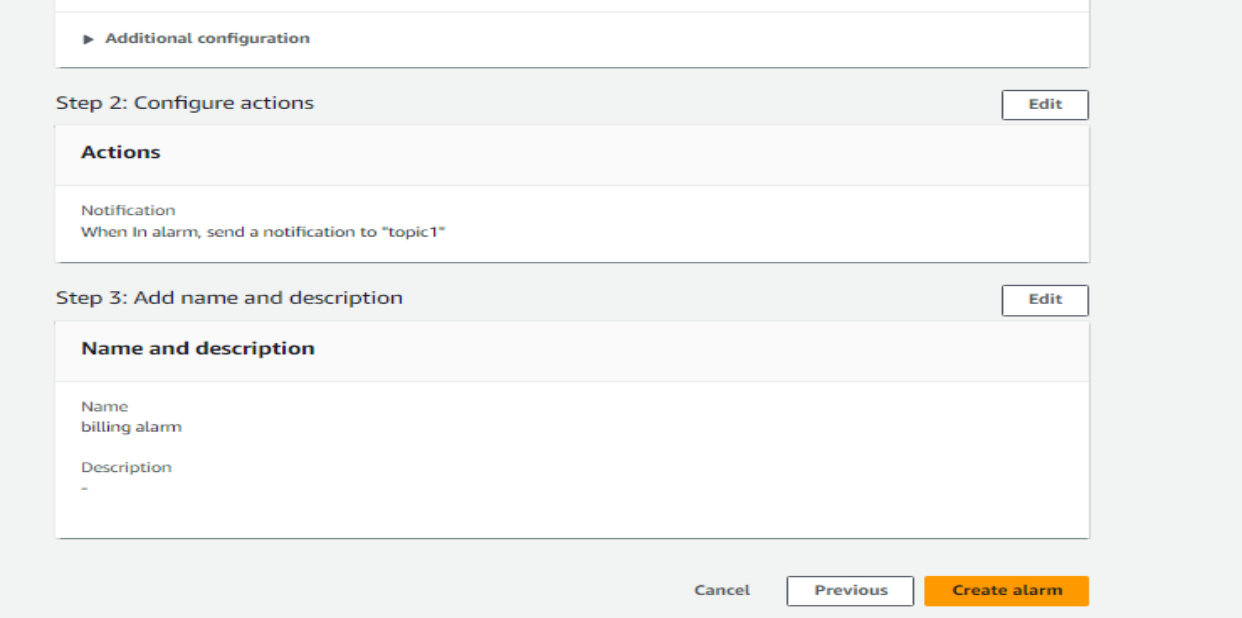
Step-(10): Set the threshold value as 500 or 1 (“as you wish”) and click on the greater than or equal option.



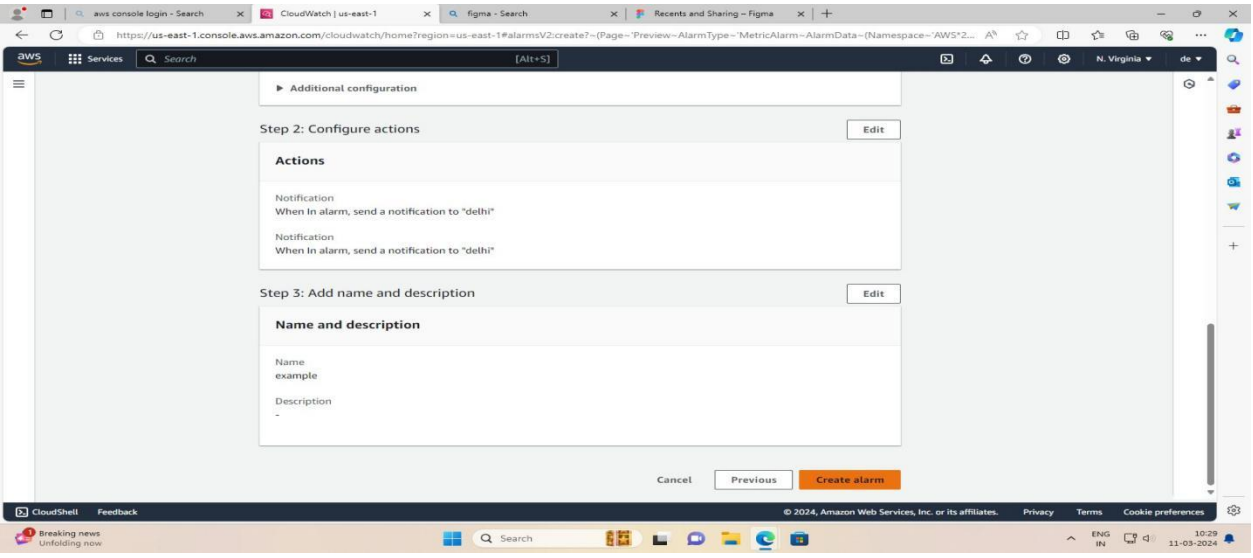
Step-(11): Then click on the next and choose the topic for sending the notification.



Step-(12): After configuring click on the create alarm.



Step-(13): Now the alarm can be created as shown below.



RESULT:

Thus the alerts for usage of cloud resources was created successfully.

EXP NO:4 DATE:	CREATE BILLING ALERTS FOR YOUR CLOUD ORGANIZATION

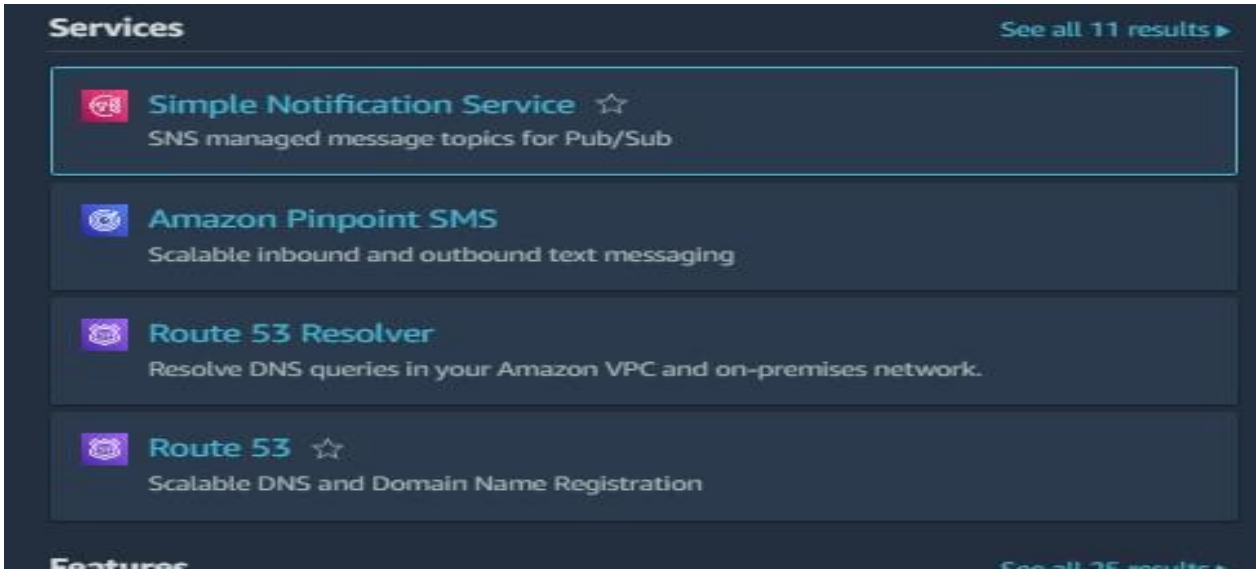
AIM:

To create billing alerts for the cloud organization.

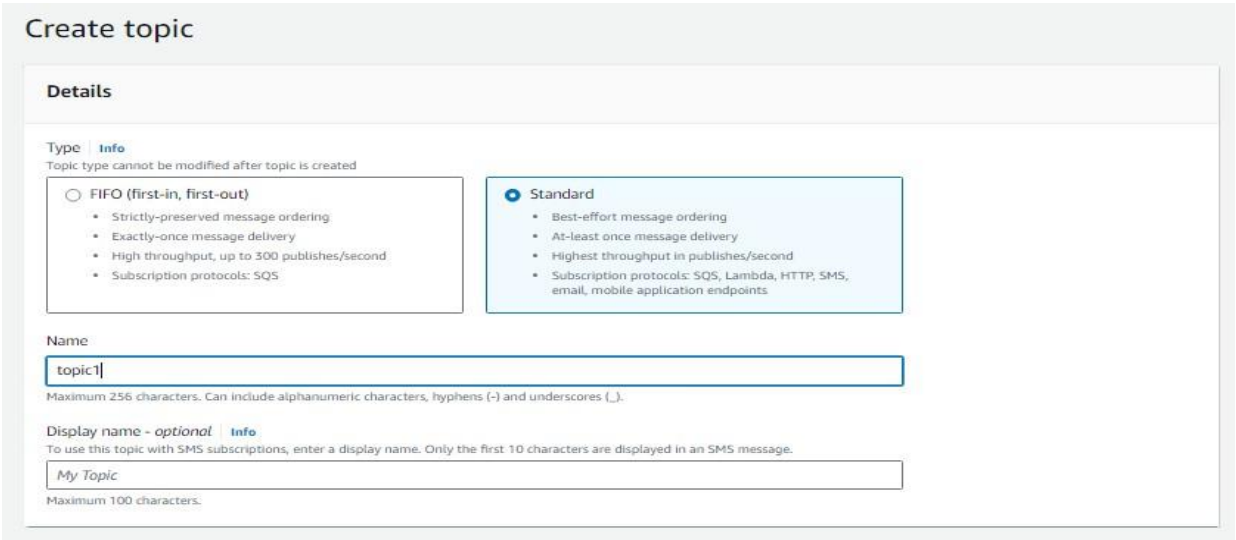
PROCEDURE:

Step-(1): Login into the aws management console.

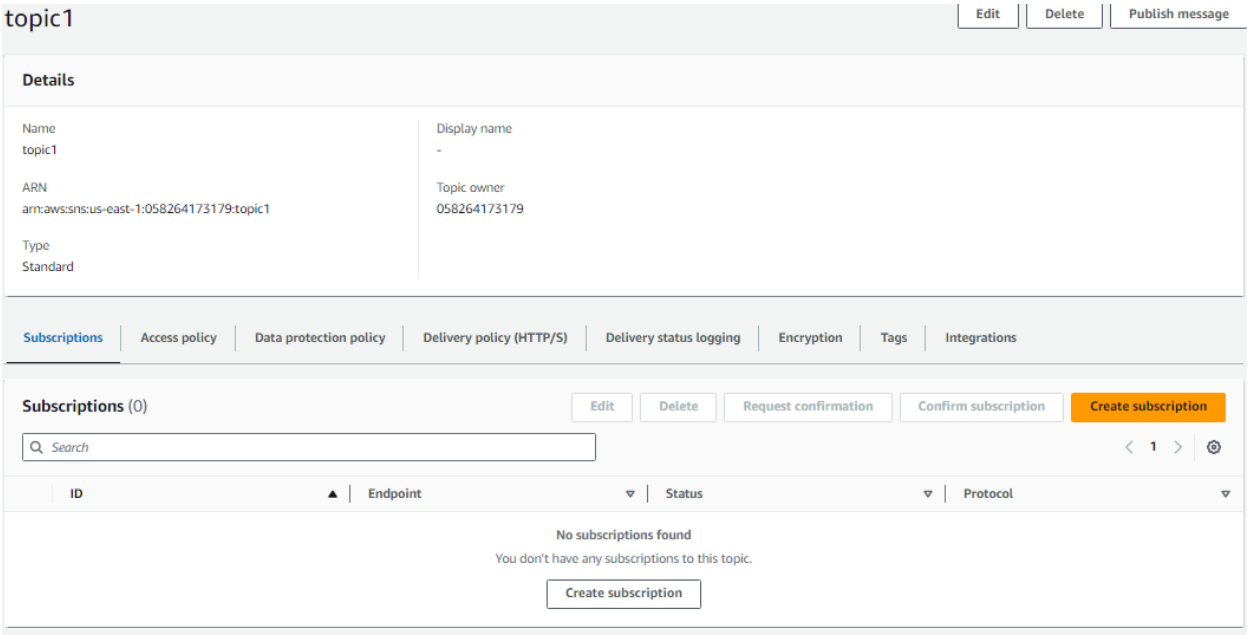
Step-(2): Search for the sns on the services.



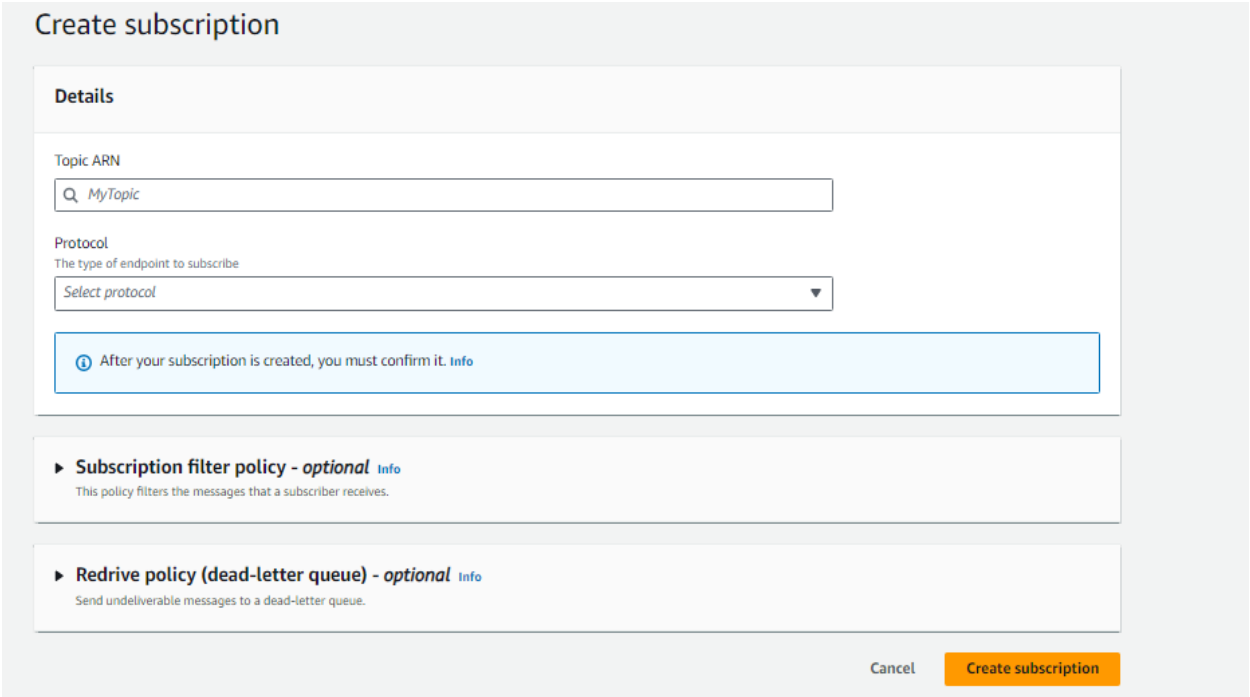
Step-(3): Under the sns click on the create topic with standard type.



Step-(4): Enter the topic and click create to proceed further, then the topic is created.



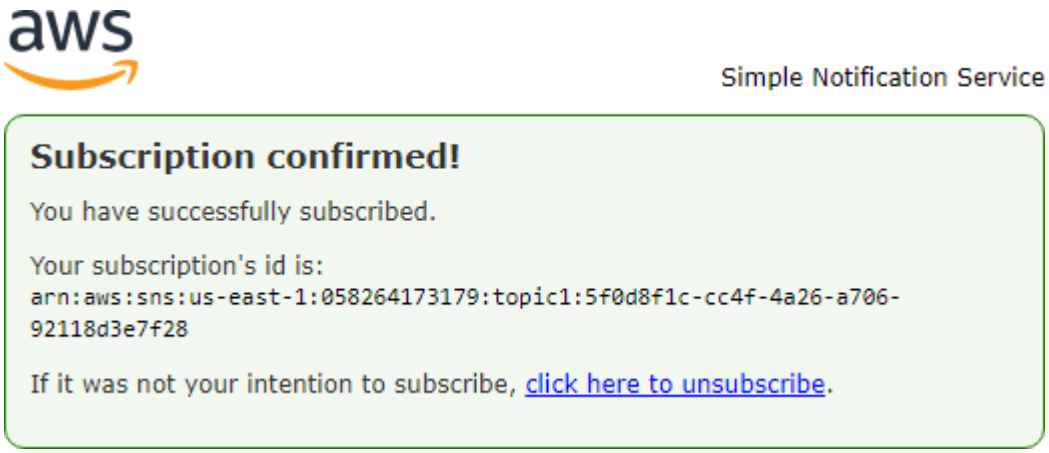
Step-(5): Then the next step is to create the subscription. To do this click on the create subscription.



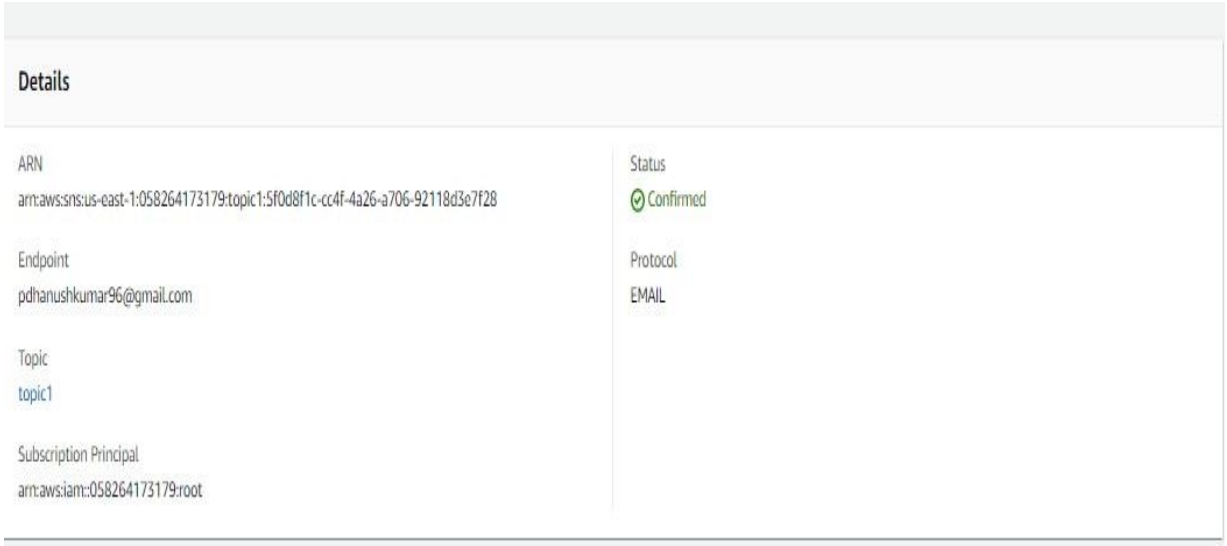
select the protocol as E-mail and then enter the root e-mail id,then click on the create subscription option.

The conformation mail will be send after clicking on the create subscriptio. Go to your mail box and check the mail of aws, also click on the confirm subscription.

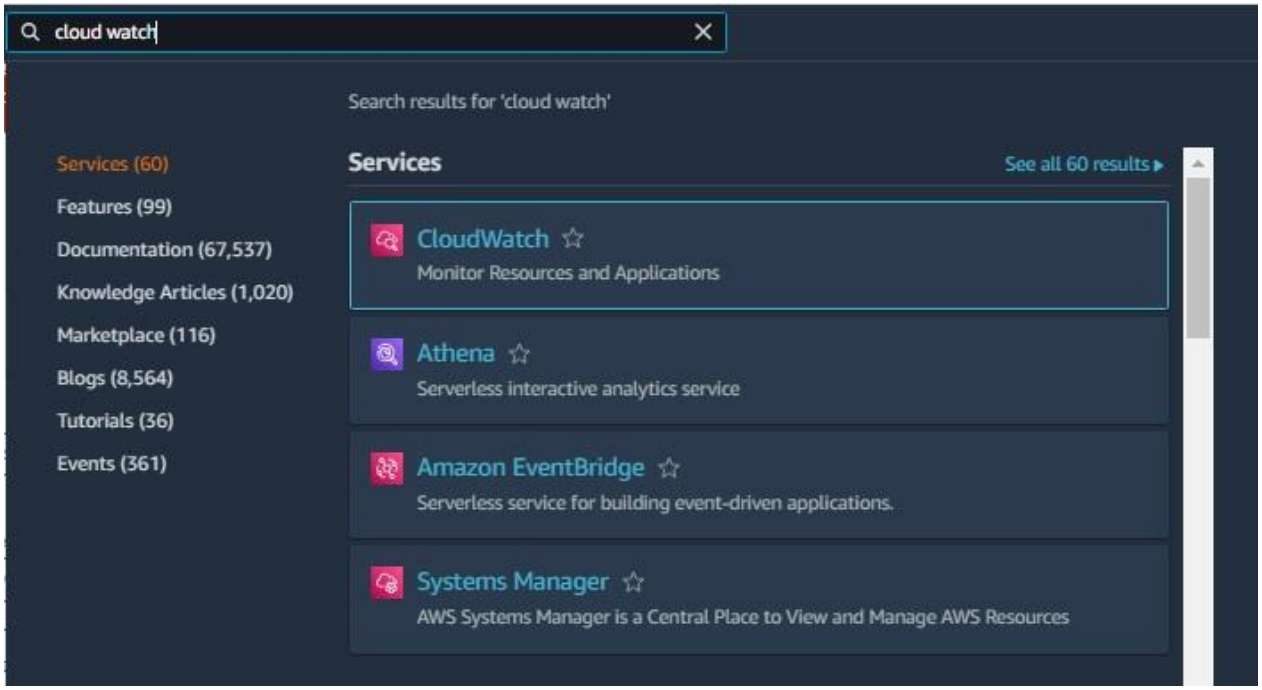
after confirm subscription, we can able to see the following message.



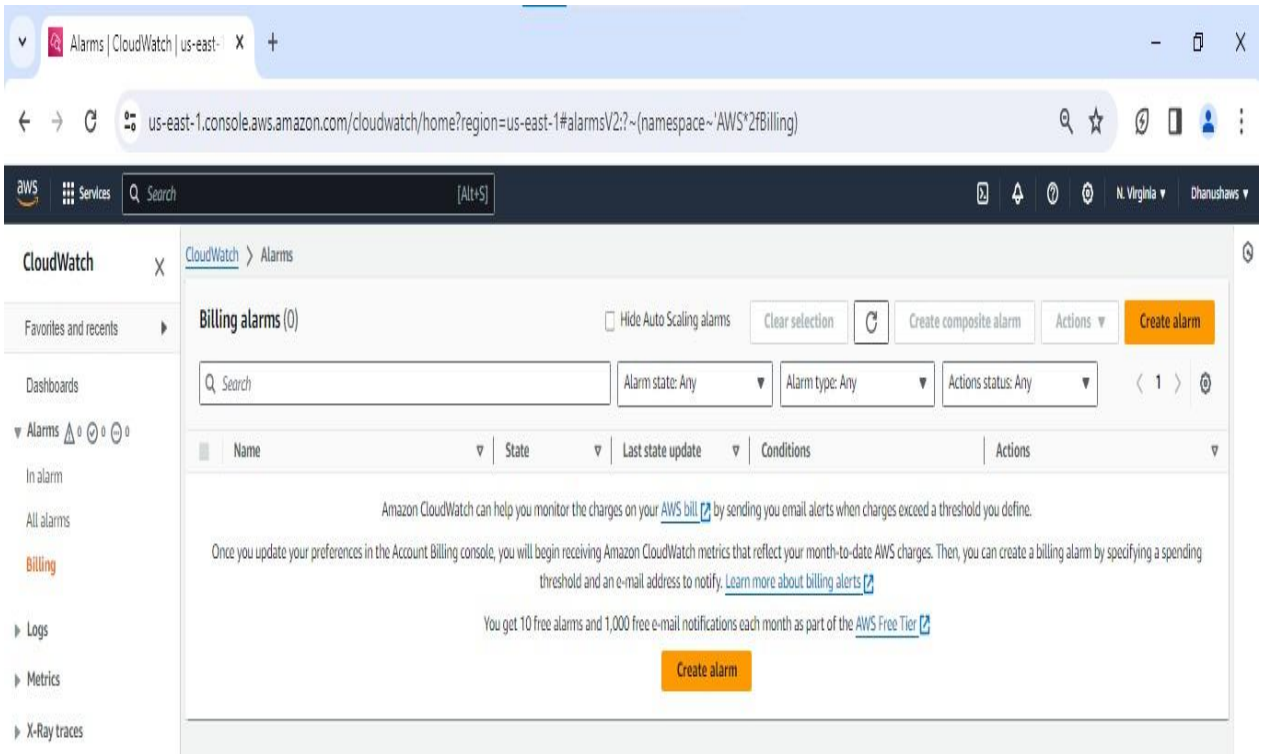
Now the topic can be successfully created as bellow



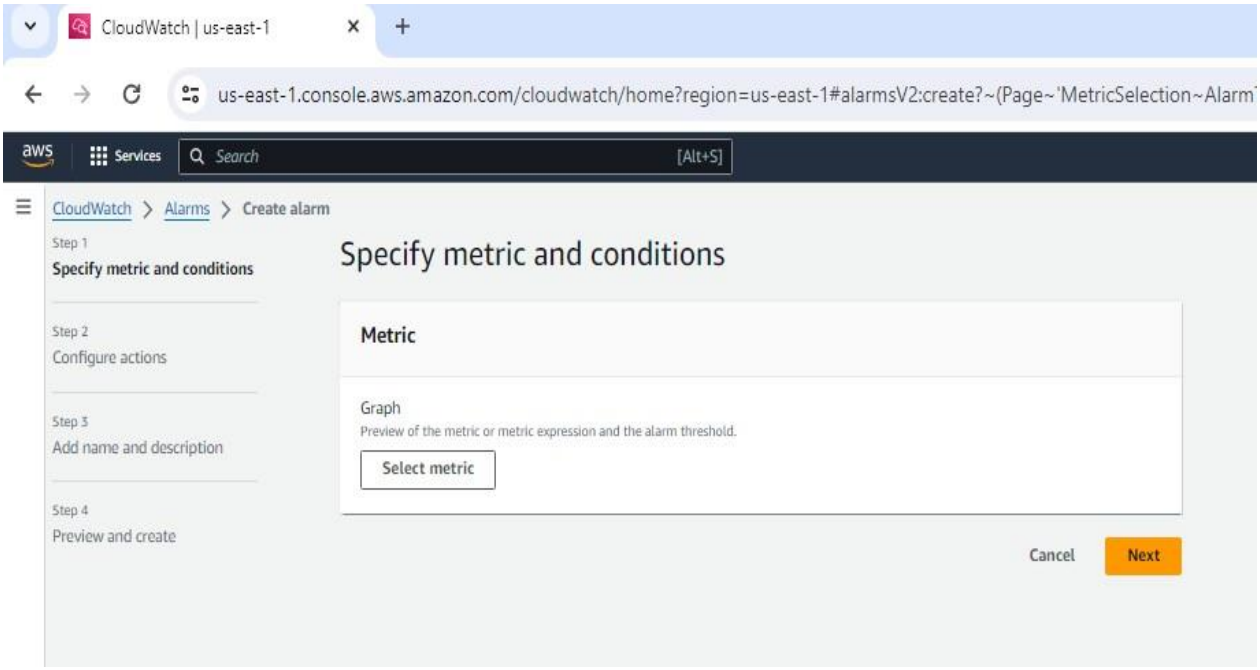
Step-(6): Search for the cloud in the services for creating the alarms for alert the billing of the cloud resource usage.



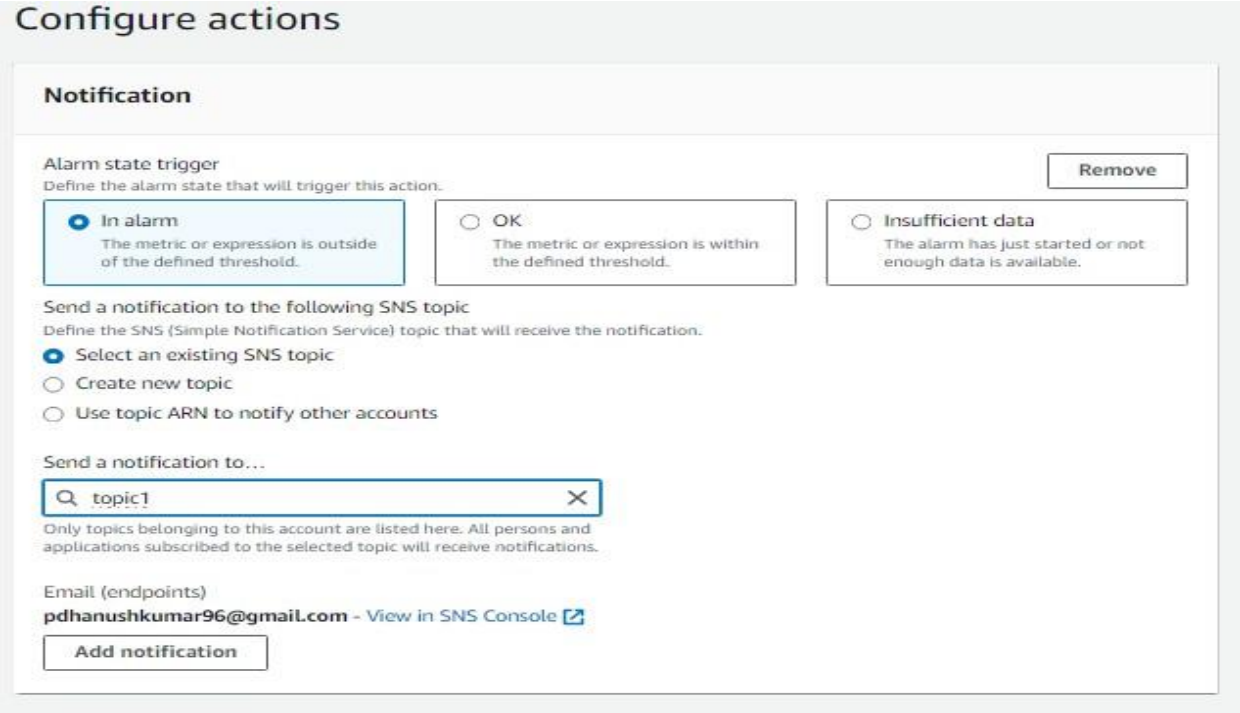
Step-(7): In the cloud watch click on the billing alarm and then click on the create new alarm.



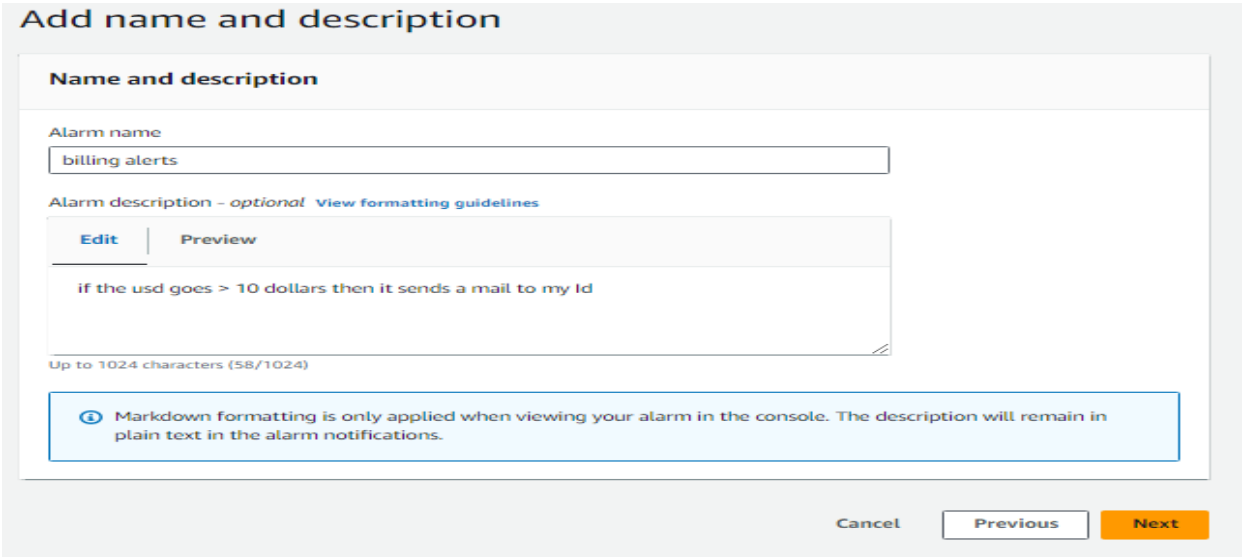
Step-(8): Then select the metric for the billing alarm as shown below.



Step-(9): Set USD as 10 dollars and specify the greater than or equal option.

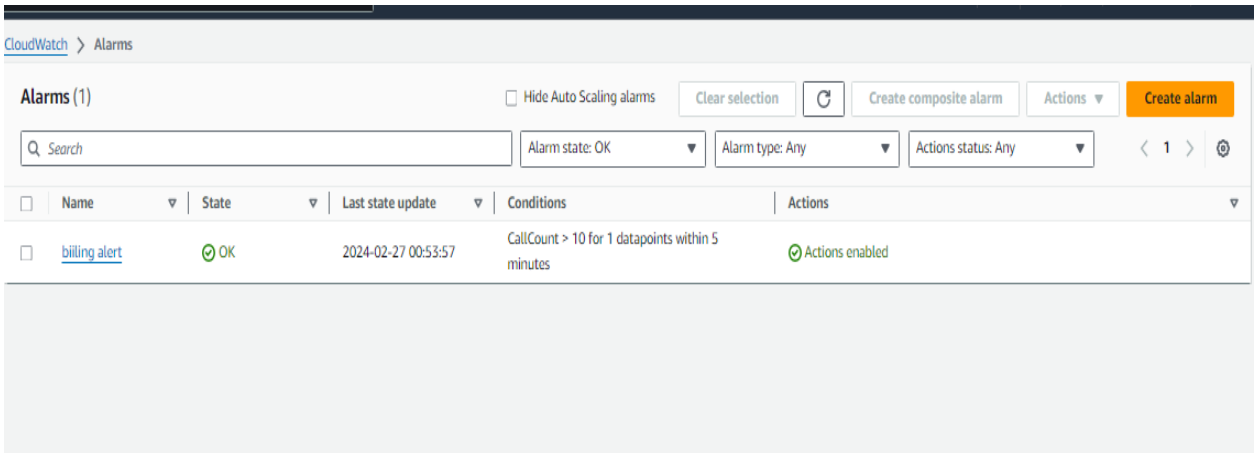


Step-(10): Describe the name of the alarm and then click on next.



Click on next.

Step-(11): Now the billing alarm is created as shown in below.



RESULT

Thus the billing alert is created for the cloud organization susccessfully.

EXP NO:5 DATE:	COMPARE CLOUD COST FOR A SIMPLE WEB APPLICATION ACROSS AWS, AZURE AND GCP AND SUGGEST THE BEST ONE.

AIM:

To compare Cloud cost for a simple web application across AWS, Azure and GCP and suggest the best one.

OBSERVATION:

1. AWS: AWS others a rich array of tools, including databases, analytics, management, IoT, security, and enterprise applications. AWS introduced per-second billing in 2017 for EC2 Linux-based instances and EBS volumes.

2. Azure: Azure has slightly surpassed AWS in the percentage of enterprises using it. Azure also others various services for enterprises, and Microsoft's longstanding relationship with this segment makes it an easy choice for some customers. While Azure is the most expensive choice for general-purpose instances, it's one of the most cost-effective alternatives to compute-optimized instances.

3. Google Cloud Platform (GCP): GCP stands out thanks to its almost limitless internal research and exporse. GCP is different due to its role in developing various open-source technologies. Google Cloud is much cheaper than AWS and Azure for computing optimized cloud-based instances.

The best platform depends on your specific needs and requirements. If you need a wide array of tools and services, AWS might be the best choice. If you're looking for enterprise services and have a longstanding relationship with Microsoft, Azure could be your best bet.

CONCLUSION:

If you priorize innovacation and open-source technologies, GCP could be the right choice. For compute optimized instances, GCP seems to be the most cost-effective. However, it's essential to understand your requirements fully before making a decision

RESULT:

Thus, the comparison for Cloud cost for a simple web application across AWS, Azure and GCP were implemented successfully.