ASSIGNMENT 4,...REM ...6 : REVISIT

Thadomal Shahani Engineering College

Bandra (W.), Mumbai- 400 050.

CERTIFICATE

Certify that Mr./~~Miss~~ Niranjan Rajesh Joshi

of IT Department, Semester II with

Roll No. 51 has completed a course of the necessary

experiments in the subject AdvanceDevOps Lab under my

supervision in the **Thadomal Shahani Engineering College**

Laboratory in the year 2023 - 2024

Agarwal

Teacher In-Charge

Head of the Department

Date 20-10-23

Principal

THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

CONTENTS

SR. NO.	EXPERIMENTS	PAGE NO.	DATE	TEACHERS SIGN.
1)	To study and perform setup of AWS EC2 service and launch an EC2 Instance		18/07/23	
2)	To study and perform the setup of AWS Cloud9 service and launch a python program in cloud9		25/07/23	
3)	To study AWS S3 service and create bucket for hosting static web application		01/08/23	
4)	To study AWS codepipeline and deploy web application using AWS codepipeline		08/08/23	Agarwal 20/10/23
5)	To understand Kubernetes cluster Architecture, install and spinup kubernetes cluster on Linux machines / cloud platform		13/10/23	
6)	To understand terraform lifecycle and to build change and destroy AWS infrastructure using terraform		22/08/23	
7)	To perform static analysis on python programs and sonarqube SAST process		29/08/23	
8)	To understand continuous monitoring using Nagios		12/09/23	
9)	To understand AWS Lambda function and create a Lambda function using Python to log "An image has been added" message once file is added to S3 bucket		05/09/23	Agarwal 20/10/23
10)	To create Lambda function using python for adding data to Dynamo DB database		05/09/23	
11)	Assignment 1		01/08/23	
12)	Assignment 2		13/10/23	

Name :-Niranjan Rajesh Joshi
Roll No:- 2105051
Batch:- T13
Date Of Performance :- 25/07/2023

Experiment 1

Aim:-Study and create AWS EC2 instance

LO Mapped:- L01

Theory:-

EC2 stands for **Amazon Elastic Compute Cloud** . Amazon EC2 is a web service that provides resizable compute capacity in the cloud . Amazon EC2 reduces the time required to obtain and boot new user instances to minutes rather than in older days, if you need a server then you had to put a purchase order, and cabling is done to get a new server which is a very time-consuming process. Now, Amazon has provided an EC2 which is a virtual machine in the cloud that completely changes the industry.

Steps to create AWS account :-

1) Open the Amazon Web Services home page .

2) Choose Create an AWS account.

3) Enter your account information, and then choose Verify email address. This will send a verification code to your specified email address.

4) Enter your verification code, and then choose Verify.

5) Enter a strong password for your root user, confirm it, and then choose Continue. AWS requires that your password meet the following conditions:

6) It must have a minimum of 8 characters and a maximum of 128 characters.It must include a minimum of three of the following mix of character types: uppercase, lowercase, numbers, and ! @ # $ % ^ & * () <> [] { } | _+-= symbols.It must not be identical to your AWS account name or email address.

7) Choose Business or Personal. Personal accounts and business accounts have the same features and functions.Enter your company or personal information.Read and accept the AWS Customer Agreement. Be sure that you read and understand the terms of the AWS Customer Agreement.

8) Choose Continue. At this point, you'll receive an email message to confirm that your AWS account is ready to use. You can sign in to your new account by using the email address and password you provided during sign up. However, you can't use any AWS services until you finish activating your account.

9) Enter the information about your payment method, and then choose Verify and Continue. If you want to use a different billing address for your AWS billing information, choose Use a new address.

10) Enter your country or region code from the list, and then enter a phone number where you can be reached in the next few minutes.

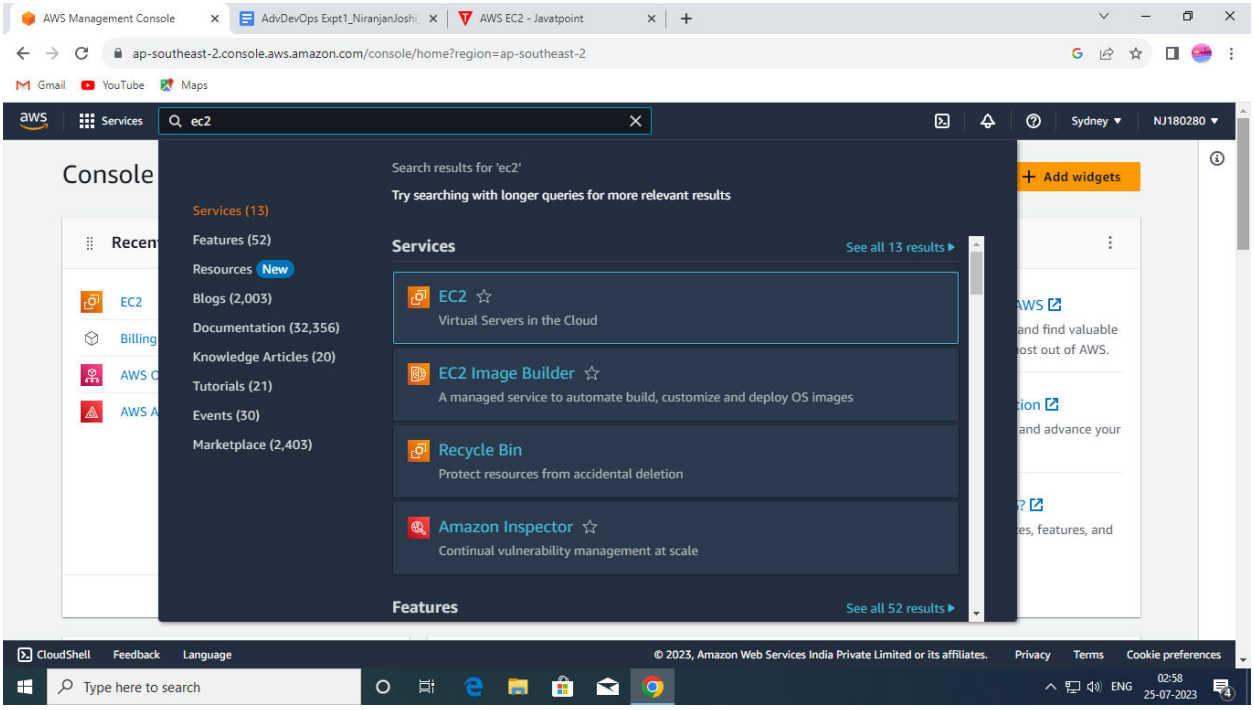
11) When the automated system contacts you, enter the PIN you receive and then submit.Select one of the available AWS Support plans. For a description of the available Support plans and their benefits, see Compare AWS Support plans.

12) Choose Complete sign up. A confirmation page appears that indicates that your account is being activated.Check your email and spam folder for an email message that confirms your account was activated.

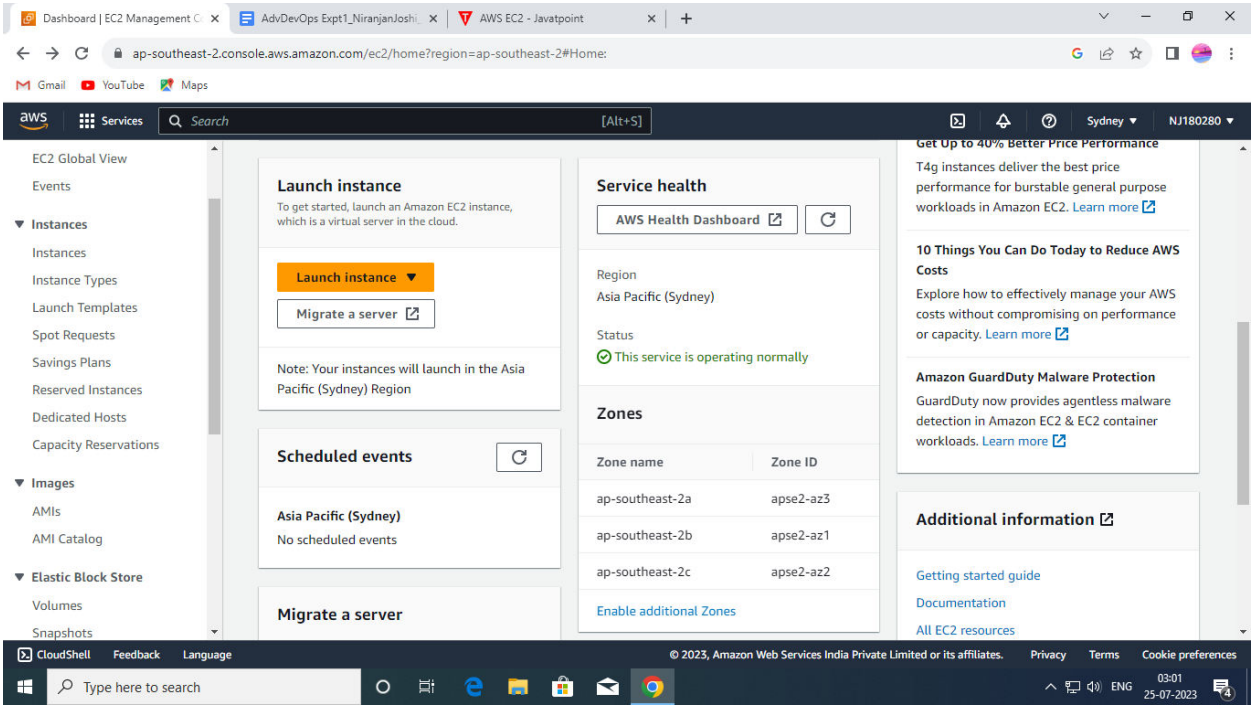
Steps (to create an EC2 instance) :-

1)Sign in to AWS account

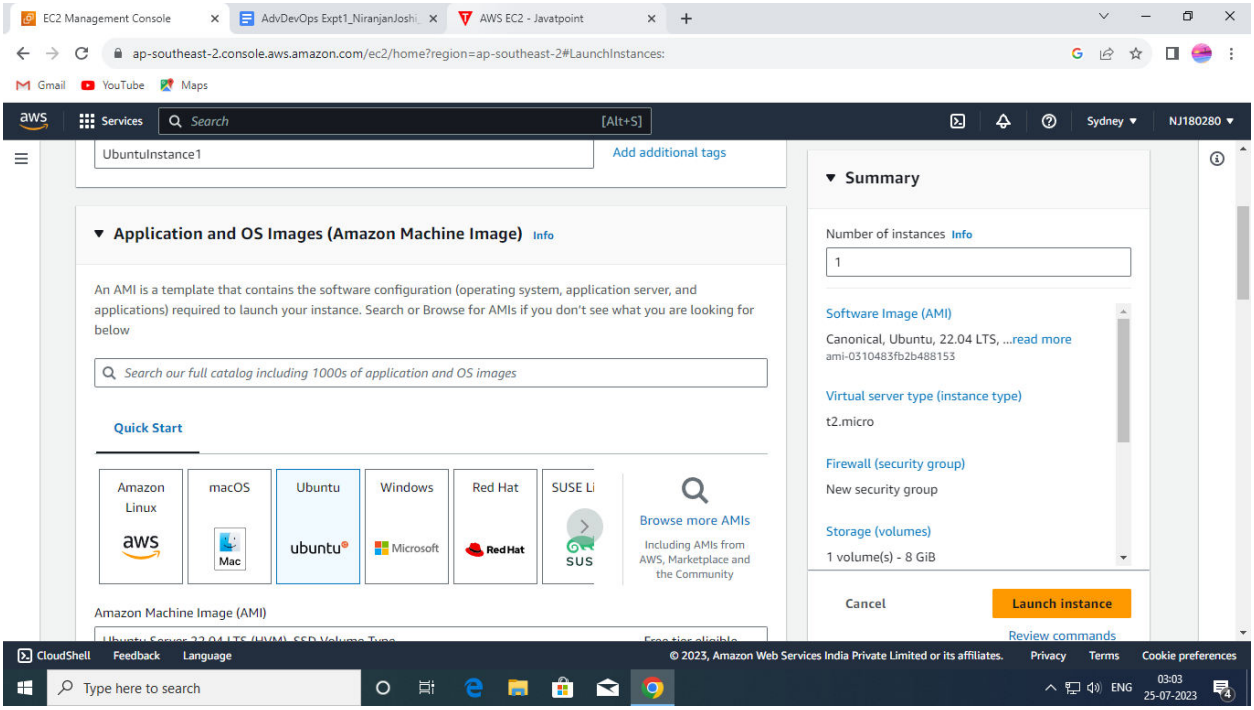
2)In above search bar , search for EC2 service as shown below and then click on EC2 :



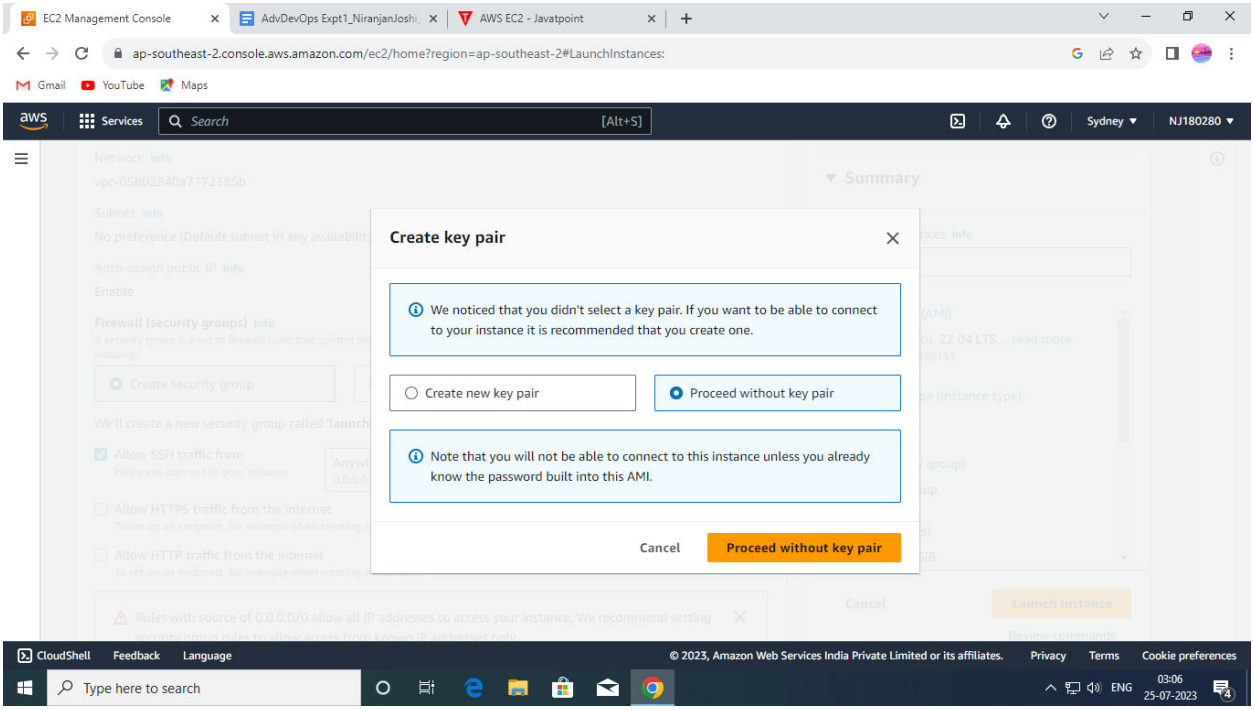
3)Now EC2 dashboard will be shown and then scroll below and click on Launch Instance to create a new instance



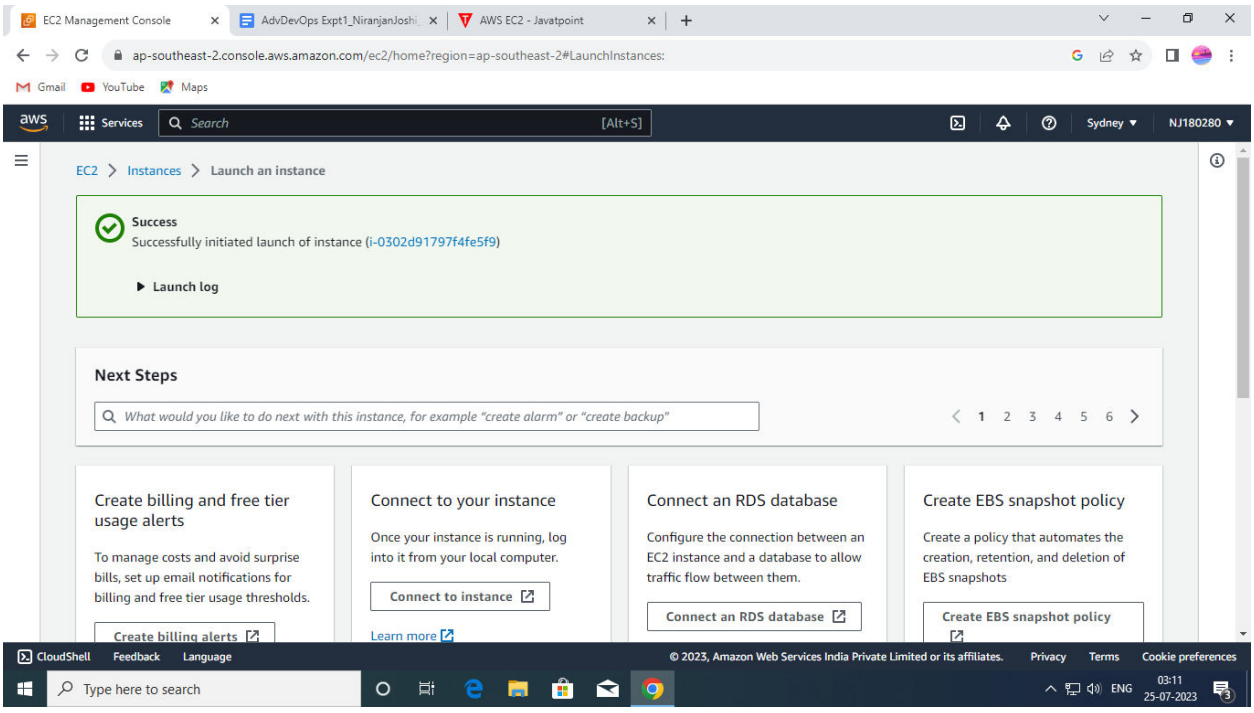
4)After clicking on launch instance new window will open and then give a name to instance and also select OS/AMI to work with



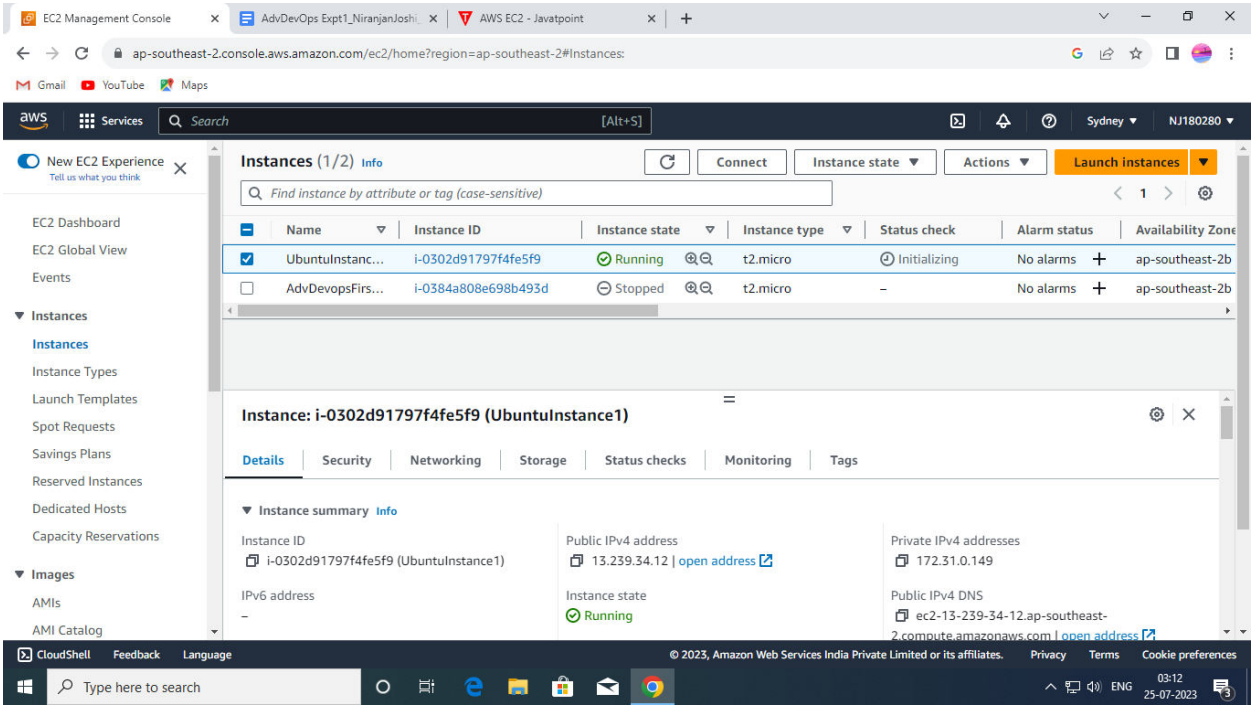
5)Do the necessary configurations and then click on Launch instance button and after clicking it will ask for key value pair , if key is not present then select proceed without key pair button



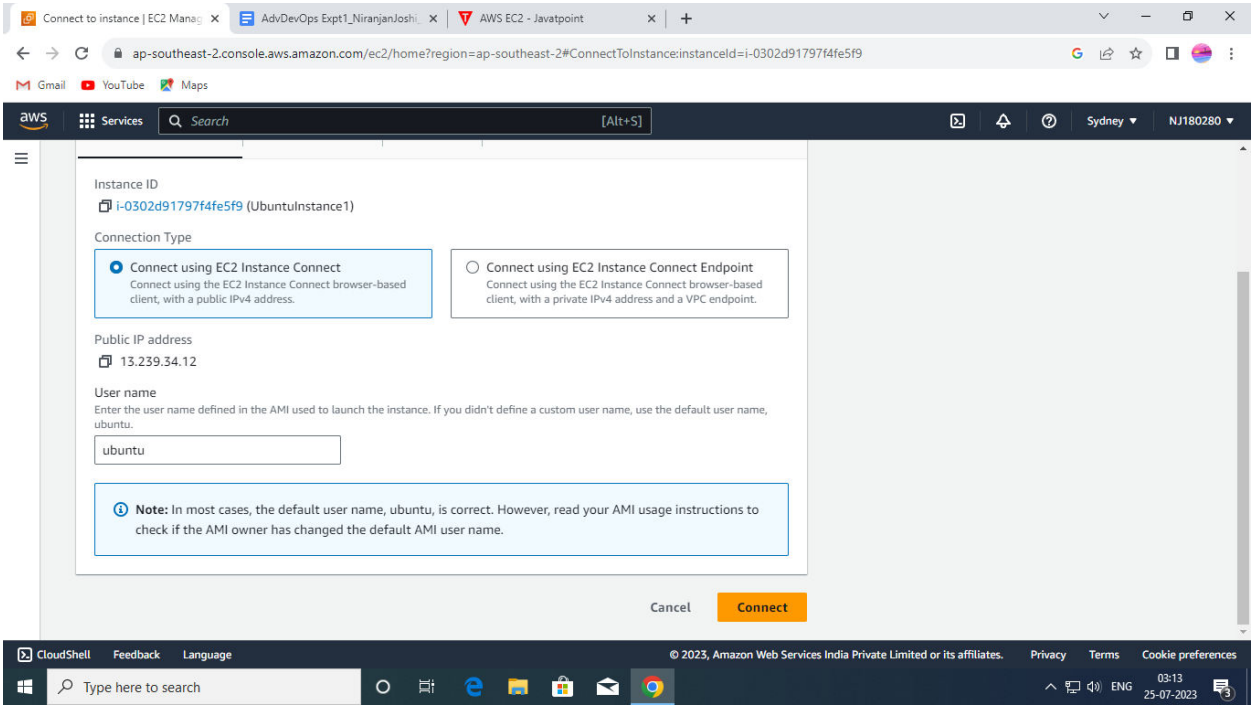
6)Now click on launch instance and then instance will be successfully created



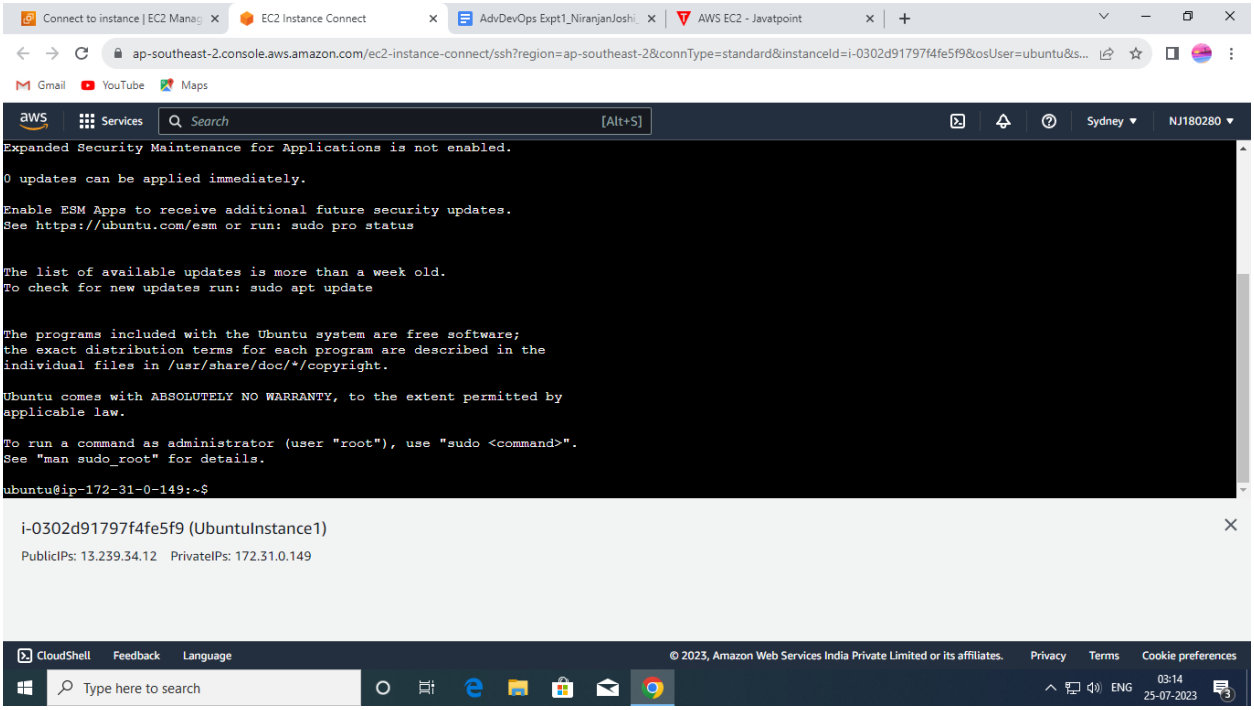
7)Then go to EC2 dashboard and click on instances , then below page appears



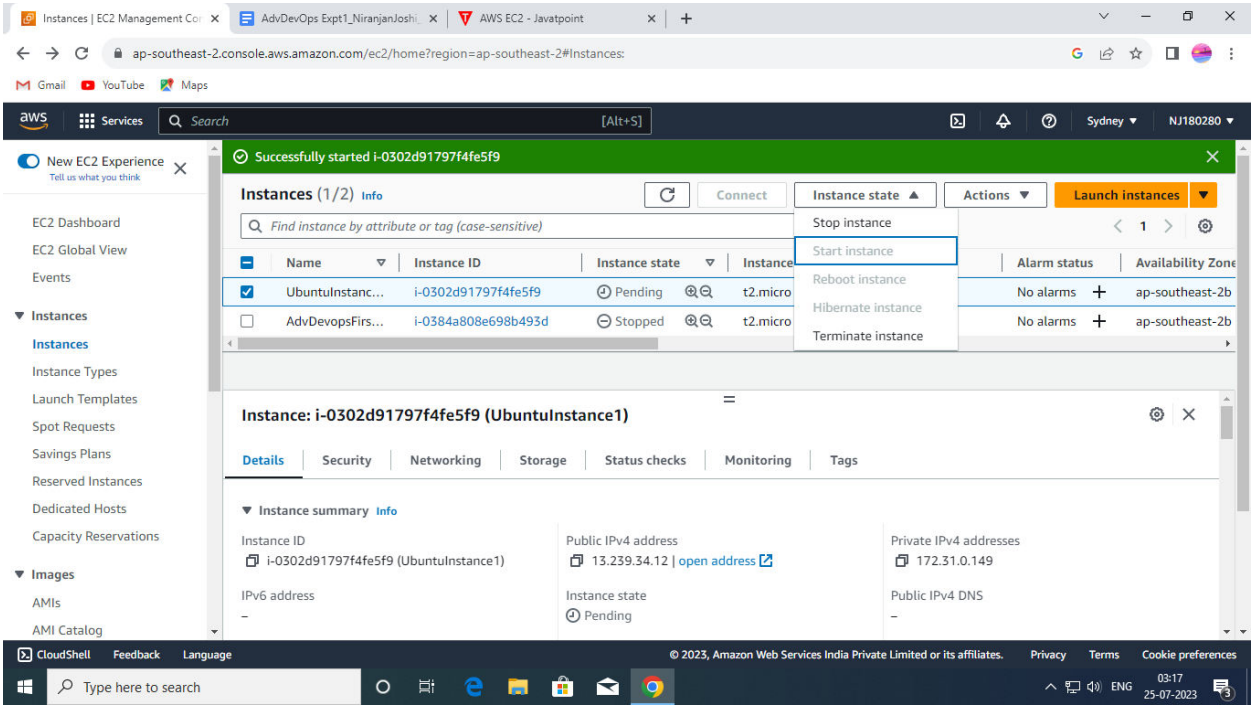
8)Then click on connect and then again on connect button in yellow color



9)After connecting instance will be created and shown on screen



10) After that go to instances page and select instance , then select instance state and click on stop instance



Conclusion:- Learned about creation of AWS account from scratch and creation of ubuntu environment instance in EC2 machine on AWS following all necessary steps .

Name :- Niranjan Raejsh Joshi
RollNo :- 2105051
Batch :- T13
Date Of Performance :- 01/08/2023

Assignment 2

Aim :- Study and create AWS cloud9 IDE service

Theory :-

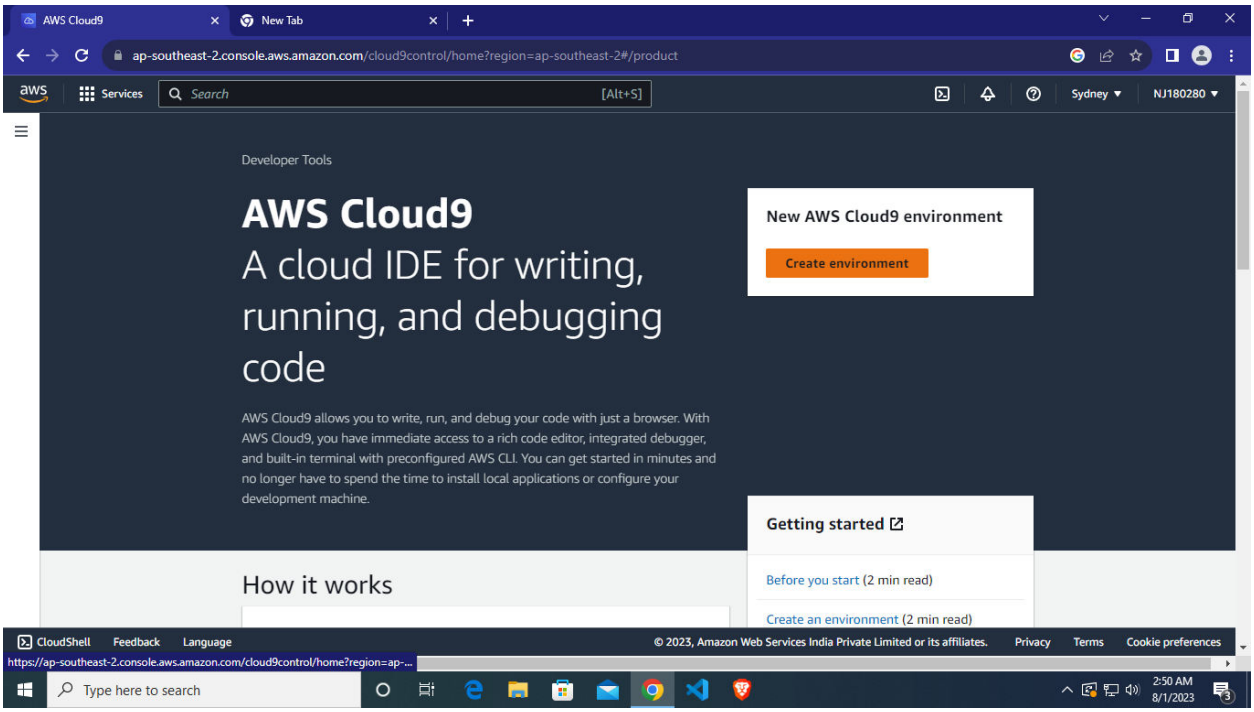
AWS Cloud9 is an integrated development environment (IDE) offered by Amazon Web Services (AWS). It provides a cloud-based platform for software development, allowing developers to write, run, and debug code from within a web browser. Here are some key features and information about AWS Cloud9:

1) **Browser-Based IDE:** AWS Cloud9 allows developers to access their development environment through a web browser. This means you can code from anywhere with an internet connection without needing to set up your development environment on a local machine.
2) **Collaboration:** Cloud9 supports collaborative coding by allowing multiple users to work on the same code in real-time. This can be particularly useful for team projects and pair programming.
3) **Built-in Terminal:** Cloud9 includes a built-in terminal that lets you execute commands and run scripts directly from the IDE. This eliminates the need to switch between different tools for coding and terminal access.
4) **Preconfigured Development Environments:** AWS Cloud9 supports various programming languages and runtimes. It provides preconfigured environments for popular languages like JavaScript, Python, Java, and more. You can also create custom environments tailored to your needs.
5) **AWS Integration:** Cloud9 is tightly integrated with other AWS services, making it easy to deploy and test applications on AWS infrastructure. You can access resources like Amazon EC2 instances, Lambda functions, and databases directly from within the IDE.
6) **Code Debugging:** Cloud9 offers debugging capabilities, allowing developers to set breakpoints, inspect variables, and step through code to identify and fix issues.
7) **Version Control Integration:** Cloud9 can integrate with version control systems like Git, making it easier to manage code repositories and collaborate with team members.

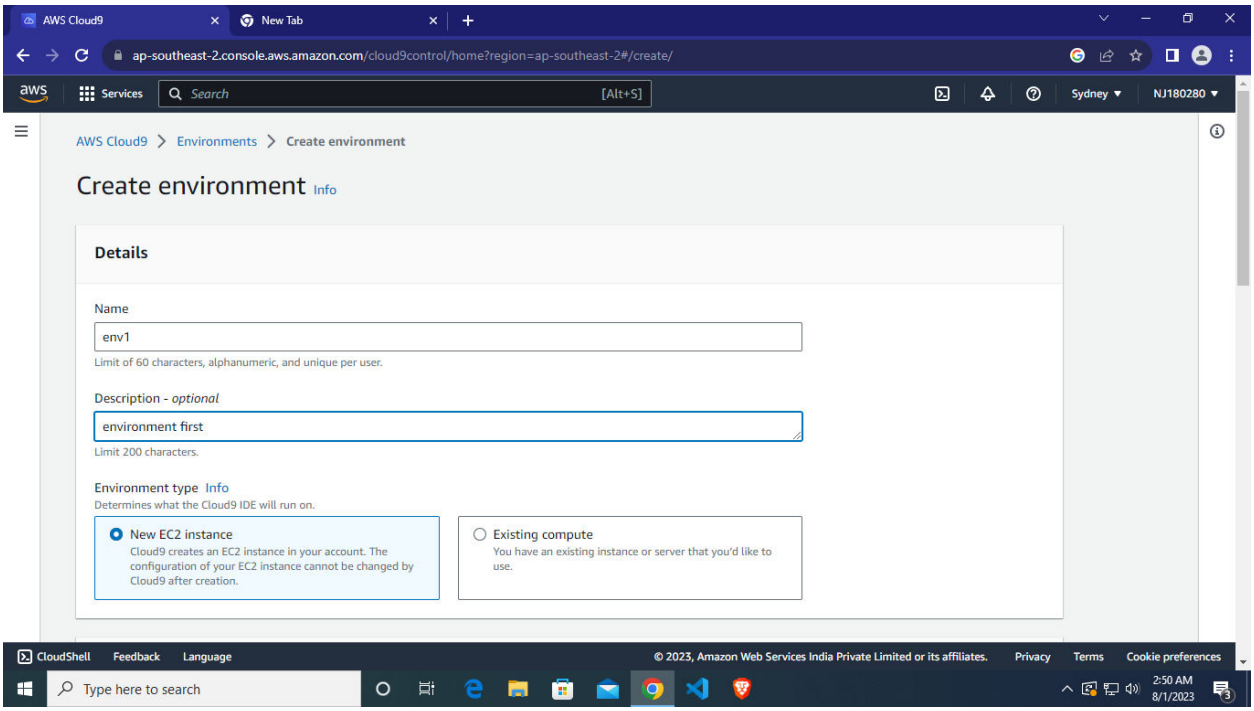
Steps to create AWS cloud 9 IDE service :-

1) Log in to your AWS account by entering appropriate credentials

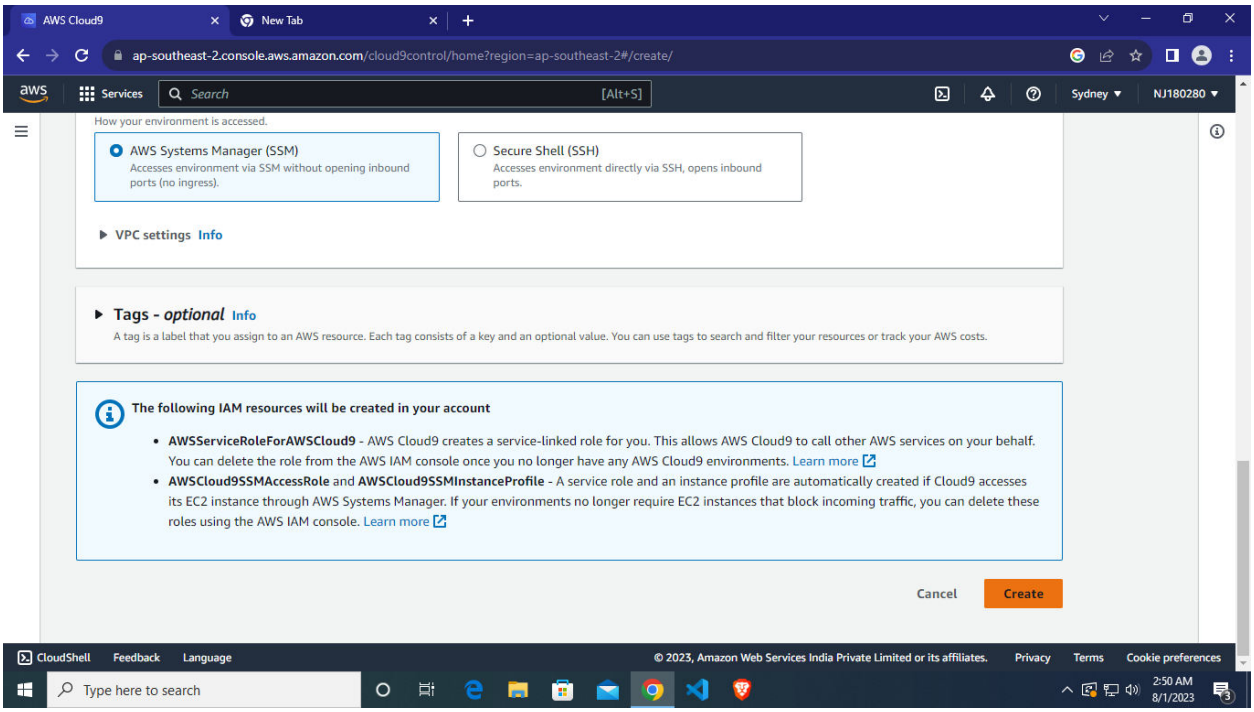
2) In search bar present on dashboard search for cloud9 service by entering “cloud9”



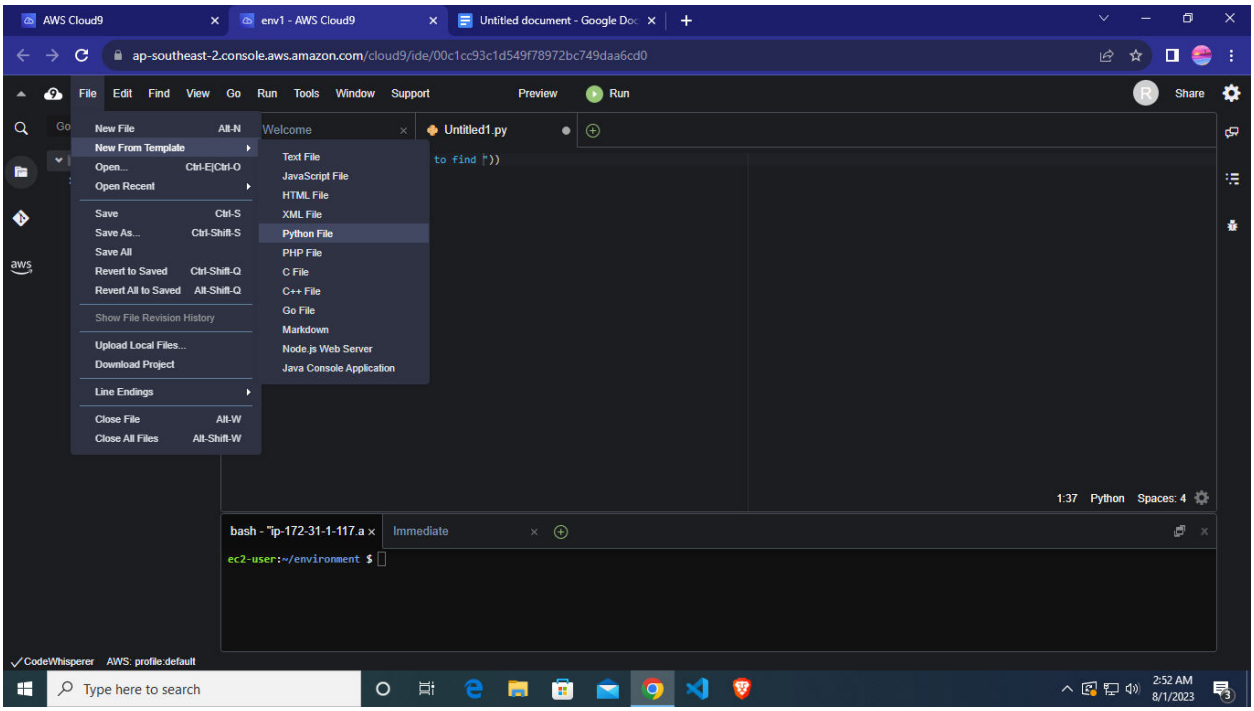
3) Then click on create environment option and enter environment name and environment description , select EC2 instance in environment type



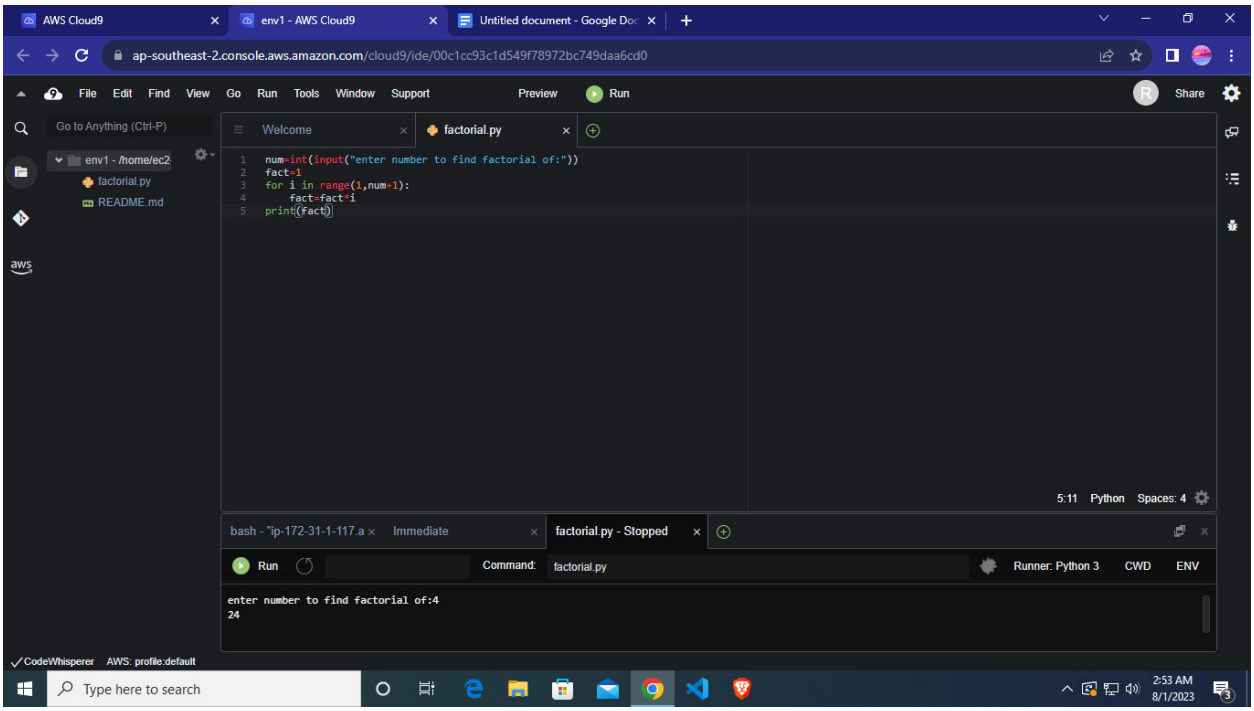
4)Then at bottom right corner create button is present , click it



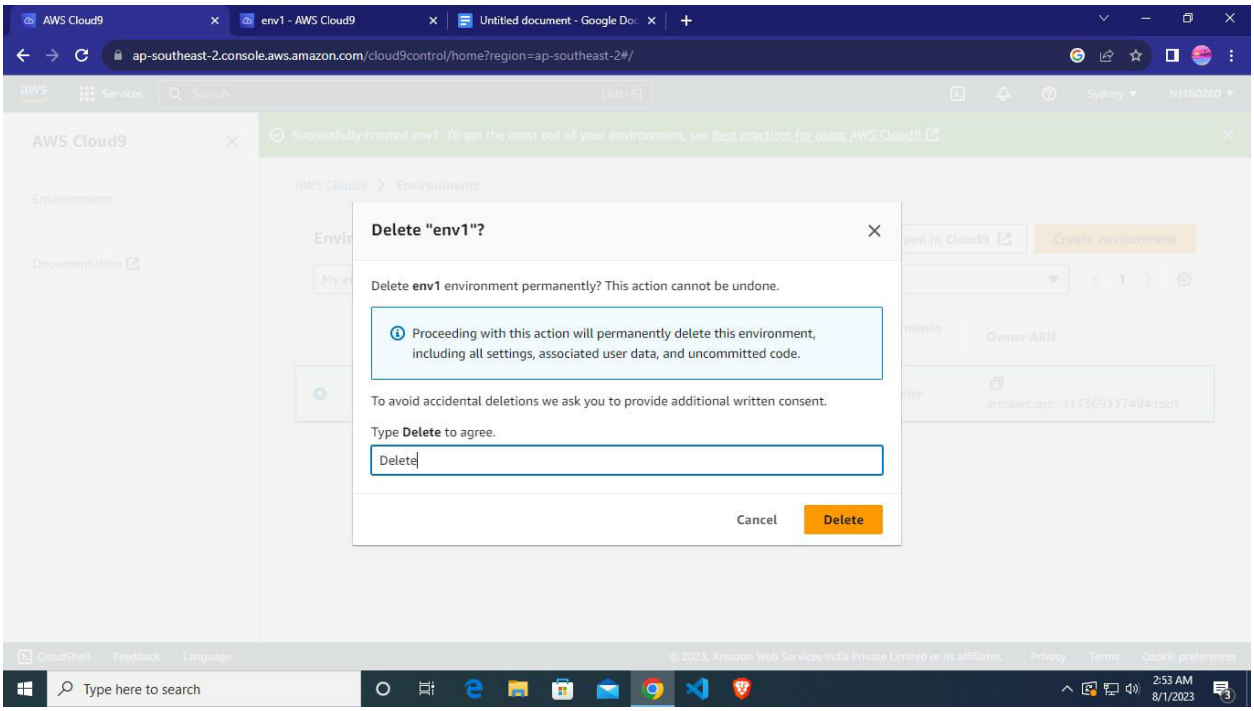
5)Then cloud9 instance will be in running state , then go to file and click on python file to cerate a new python file



6)Then type code for any program , in this case code is to find factorial of a number and python is used as programming language and after completing writing of code , click on run button above to execute the written code and output is visible at bottom in console window



7)After using the environment , we have to delete the environment , come to dashboard of cloud9 and see for created environment , click on delete option and confirm the deletion by typing delete in textbox given , this will delete cloud9 instance which was running



Conclusion :- learnt about AWS cloud9 service , its use-case and how it simplifies task , configured a AWS cloud9 IDE , ran a python program to calculate factorial of a number in cloud9 IDE and then deleted the environment after use

Name :- Niranjan Rajesh Joshi
RollNo :- 2105051
Batch :- T13
Date Of Performance :- 01/08/2023

Experiment 3

AIM :- To study AWS S3 service and create a bucket for hosing static web application.

THEORY-

AWS Simple Storage Service (S3) from the aforementioned list, S3, is the object storage service provided by AWS. It is probably the most commonly used, go-to storage service for AWS users given the features like extremely high availability, security, and simple connection to other AWS Services.

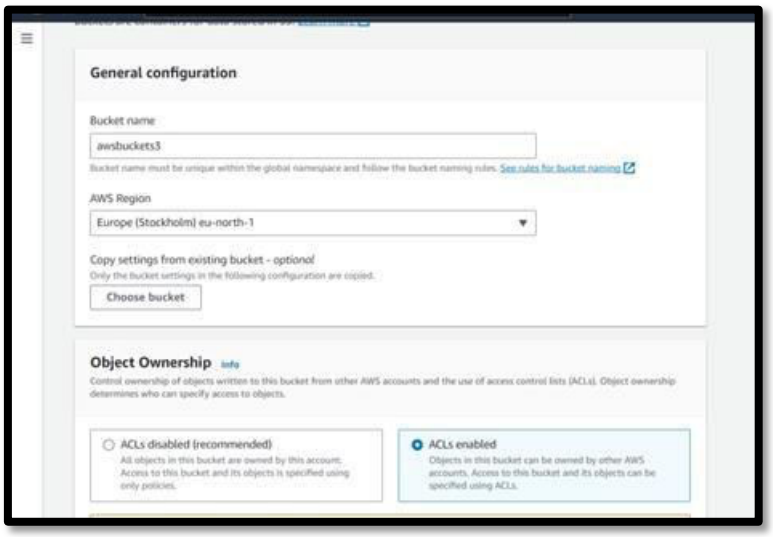
An Amazon S3 bucket can be set up to operate similarly to a website. This section illustrates how to host a website using Amazon S3. There are mainly 7 steps to hosting a static website using Amazon Web Service(AWS) S3.

Step 1: Creating a Bucket

1. First, we have to launch our S3 instance. Follow these steps for creating a Bucket
2. Open the Amazon S3 console by logging into the AWS Management Console at https://console.aws.amazon.com/s3/.

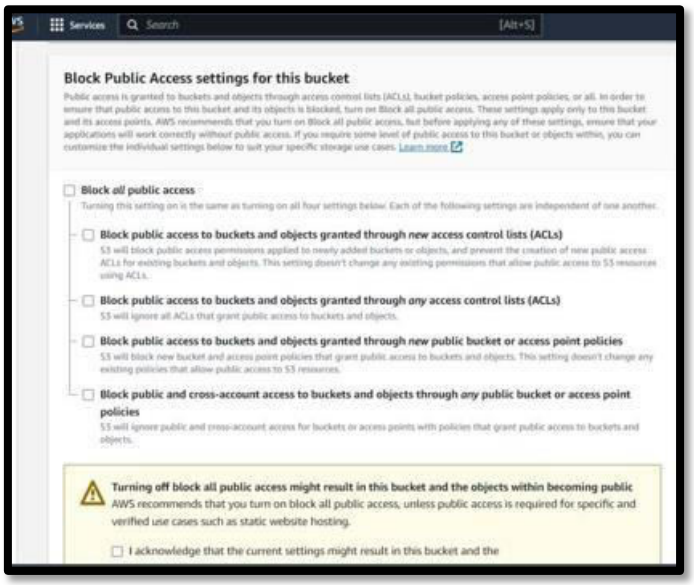
3. Click on Create Bucket.

4. Choose Bucket Name – Bucket Name Should be Unique
5. Object Ownership – Enable for making Public, Otherwise disable


General configuration
Bucket name
awsbuckets3
Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming
AWS Region
Europe (Stockholm) eu-north-1
Copy settings from existing bucket - optional
Only the bucket settings in the following configuration are copied.
Choose bucket
Object Ownership info
Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.
ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.
ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Step 2: Block Public Access settings for the bucket

1. Uncheck (Block all public access) for the public, otherwise set default. If you uncheck (Block all public keys).

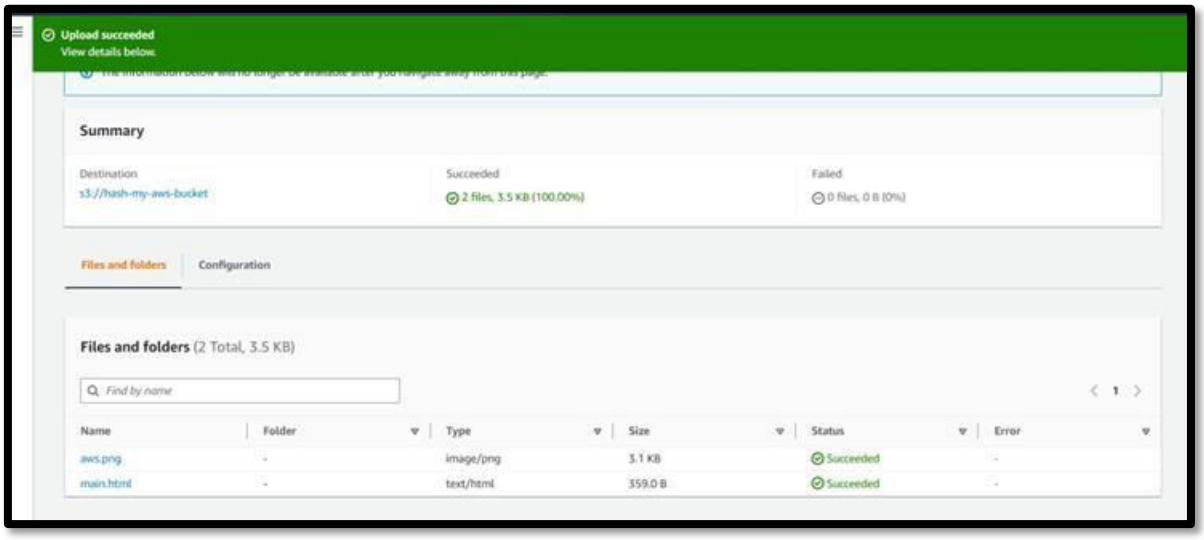


2. Now click on create bucket
3. Bucket is created

Step 3: Now upload code files

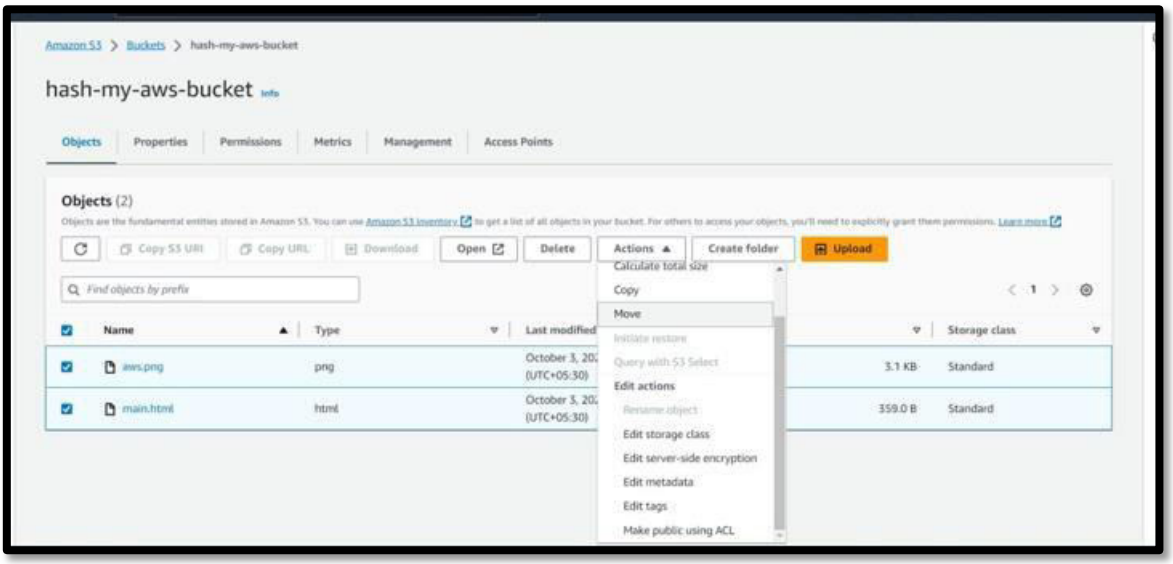
Select Bucket and Click your Bucket Name.

Now, click on upload (then click add File/folder) and select your HTML code file from your PC/Laptop.



Step 4: Once the Files are uploaded successfully, click on Permissions and now follow this Process –

- Block public access
- Object Ownership
- Make public Object



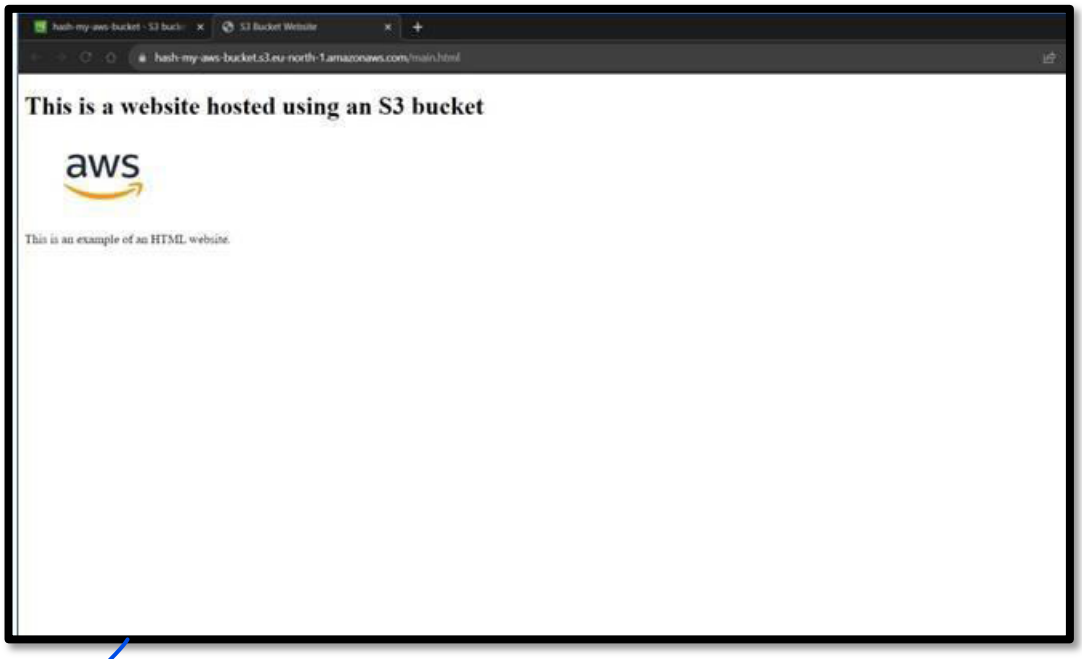
Step 6: Copy your Object URL

Now, click on your HTML File Object Name.

Copy the Object URL.

Step 7: Check out your Website!

Directly Paste this URL into the Other Tab or your other System.



CONCLUSION:-

This experiment demonstrated how to utilize , AWS S3 is a powerful and scalable cloud storage solution for hosting static web applications. It offers low latency, high reliability, and cost-effectiveness, making it a vital tool for modern web development and deployment.

Name :-Niranjan Rajesh Joshi
Roll No:- 2105051
Batch:- T13
Date Of Performance :- 08/08/2023

Experiment 4

Aim:-To study AWS code pipeline and deploy web application using code pipeline

Theory:-

AWS CodePipeline is a fully managed continuous delivery service that helps automate the release pipelines for fast and reliable application and infrastructure updates. It allows you to model, visualize, and automate the steps required to release your software. Here are some key points about AWS CodePipeline:

Automation of Software Release Process: AWS CodePipeline automates the build, test, and deployment phases of your release process every time there is a code change, based on the release model you define.

Integration with Different Services: It integrates with a variety of third-party services and AWS services such as AWS CodeBuild, AWS CodeDeploy, and AWS CloudFormation, enabling you to have a fully automated release process for your applications.

Customizable Pipeline: CodePipeline allows you to build custom release workflows with multiple stages and actions. Each stage can have one or more actions, and you can define the actions to be performed at each stage, such as source code versioning, building, testing, and deployment.

Visual Workflow: It provides a visual representation of your release process, allowing you to see the stages and actions in the pipeline and monitor the progress of each release.

Integration with Third-Party Tools: It supports integration with a wide range of third-party tools and services through its extensible architecture, enabling you to incorporate your favorite tools into the release process.

Flexibility and Control: CodePipeline provides flexibility and control over the release process, allowing you to define custom rules for the execution of each action and the transition between stages.

Security: It integrates with AWS Identity and Access Management (IAM) to control access to your pipelines, ensuring that only authorized users have the necessary permissions to view or modify the pipelines.

Monitoring and Logging: AWS CodePipeline provides monitoring and logging capabilities, allowing you to track the execution of each action and stage in the pipeline and quickly identify any issues or failures.

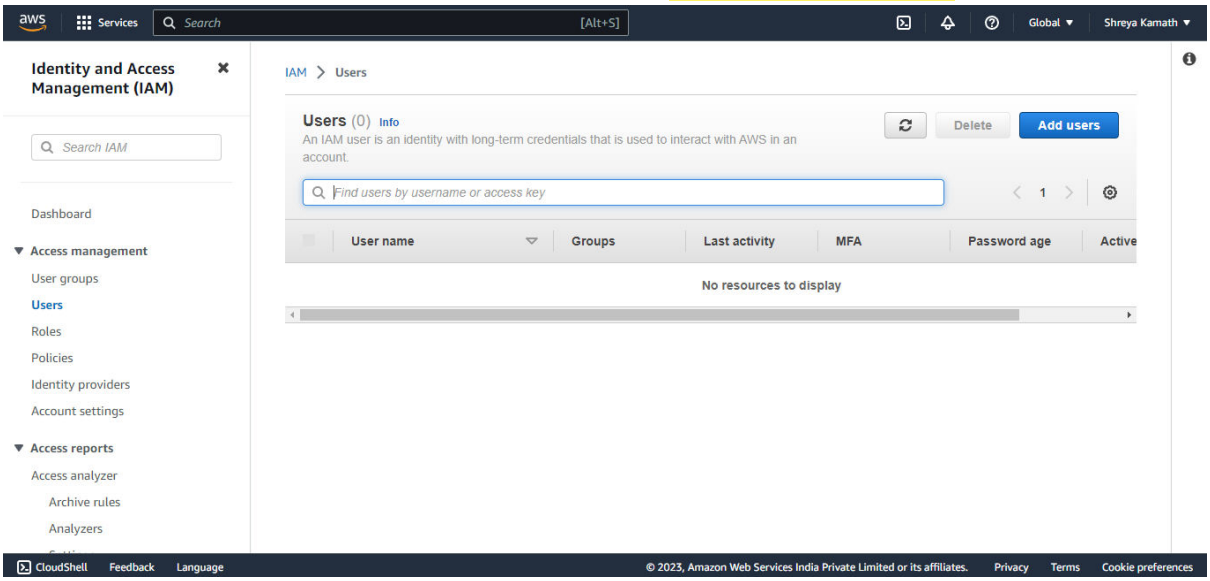
Scalability and Availability: As a fully managed service, AWS CodePipeline offers scalability and high availability, ensuring that your release pipelines can handle any workload and are always accessible.

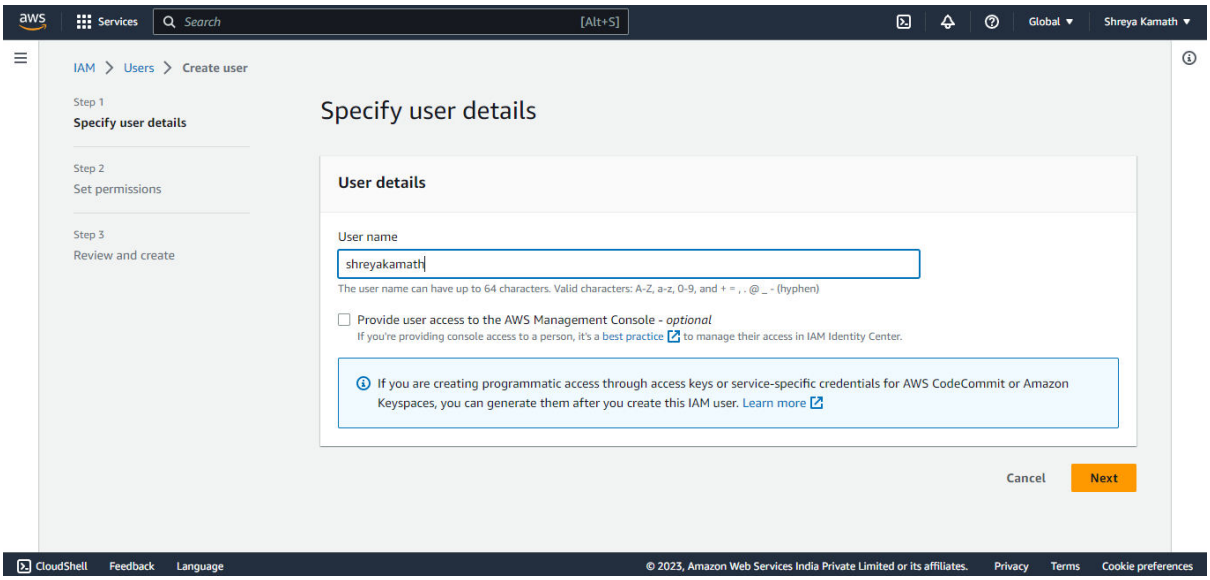
Cost-Effective: With a pay-as-you-go pricing model, AWS CodePipeline helps you optimize costs by charging only for the resources you use.

Steps :-

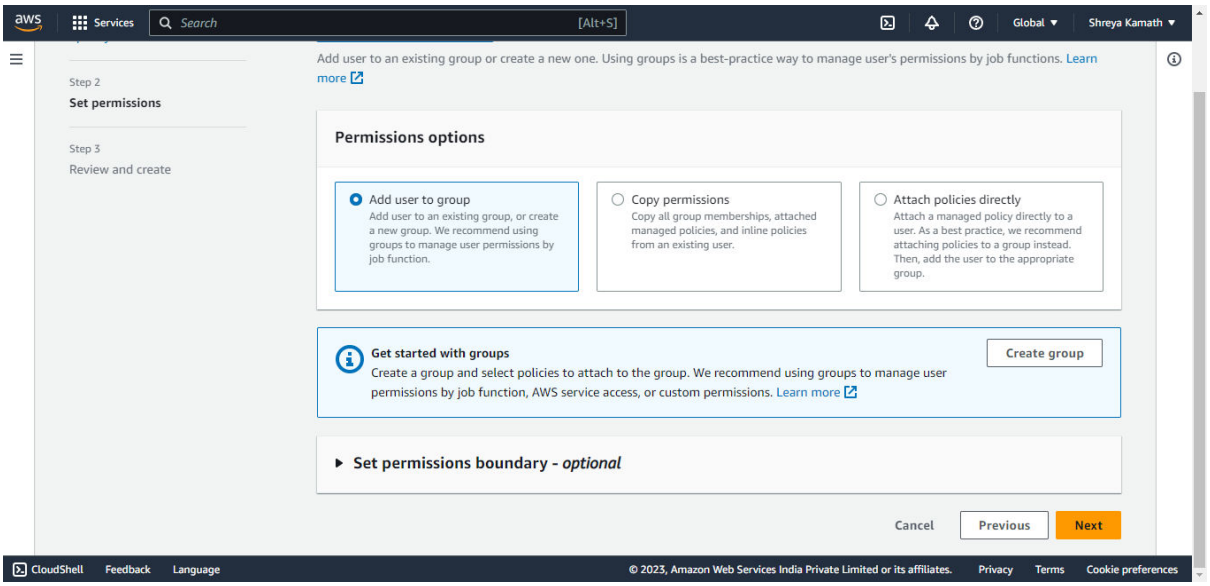
1) Login to AWS account and in search bar search IAM and click on it

2) Dashboard of IAM user open and then create a new user

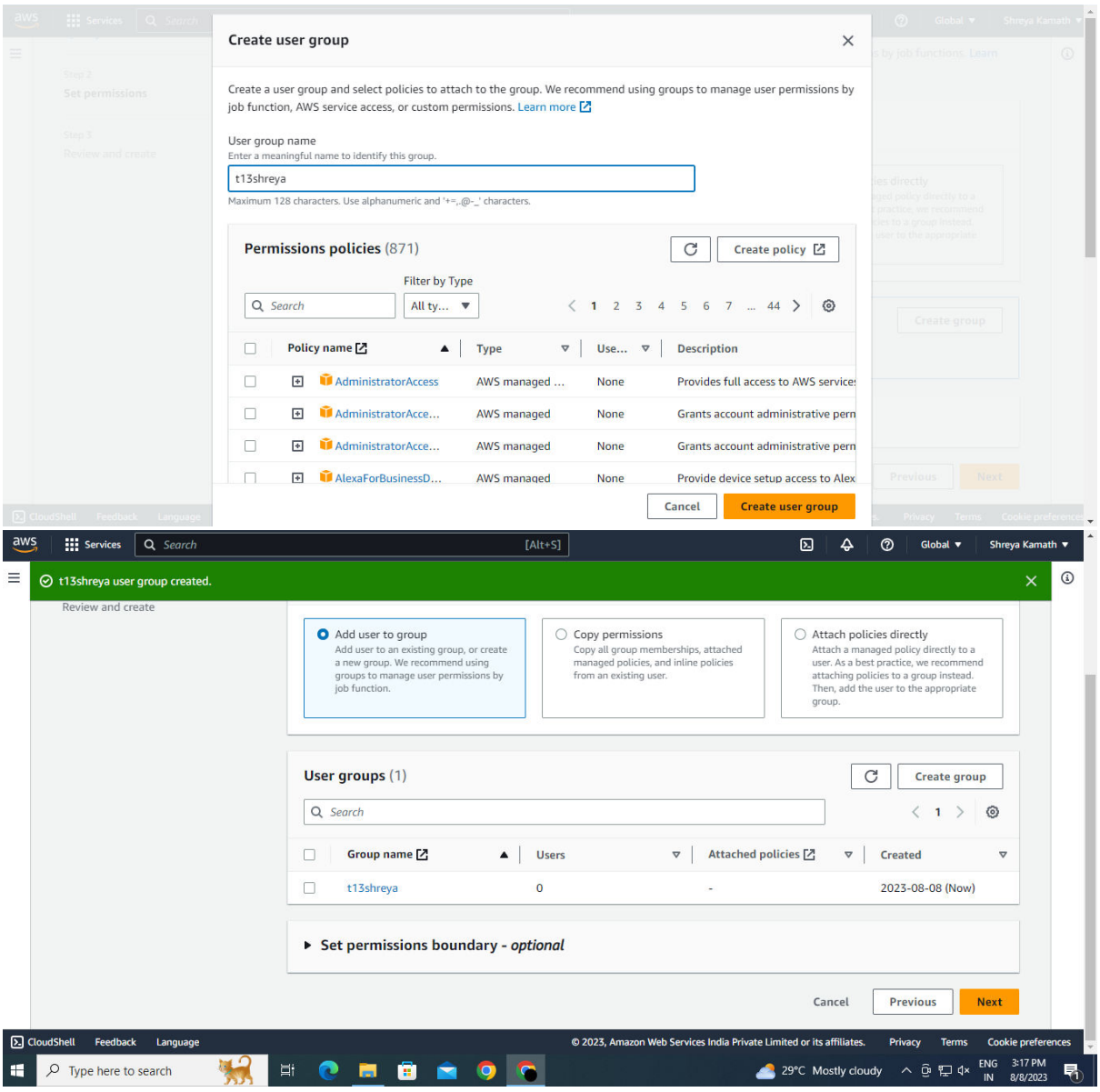




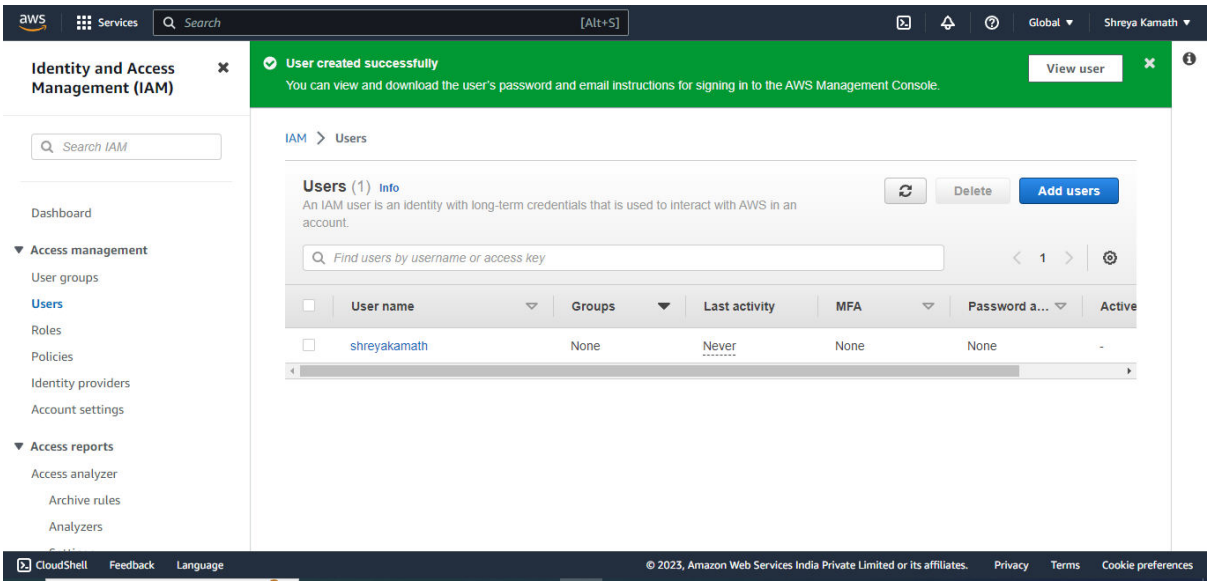
2) Set the permissions for the new user



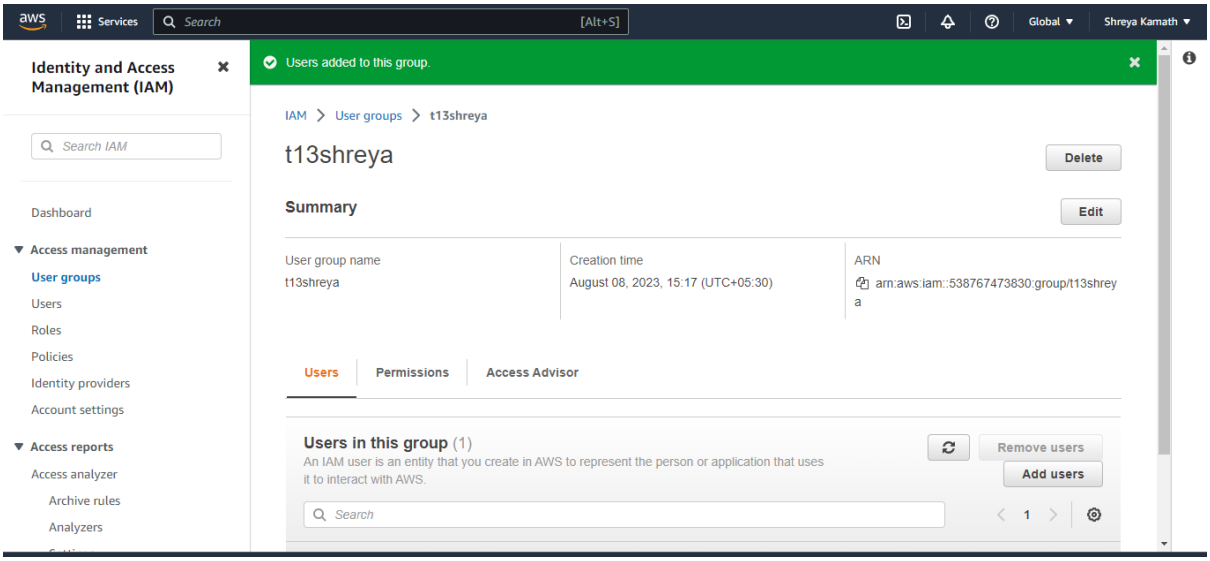
3)Create user and user name , usergroup will be visible on dashboard

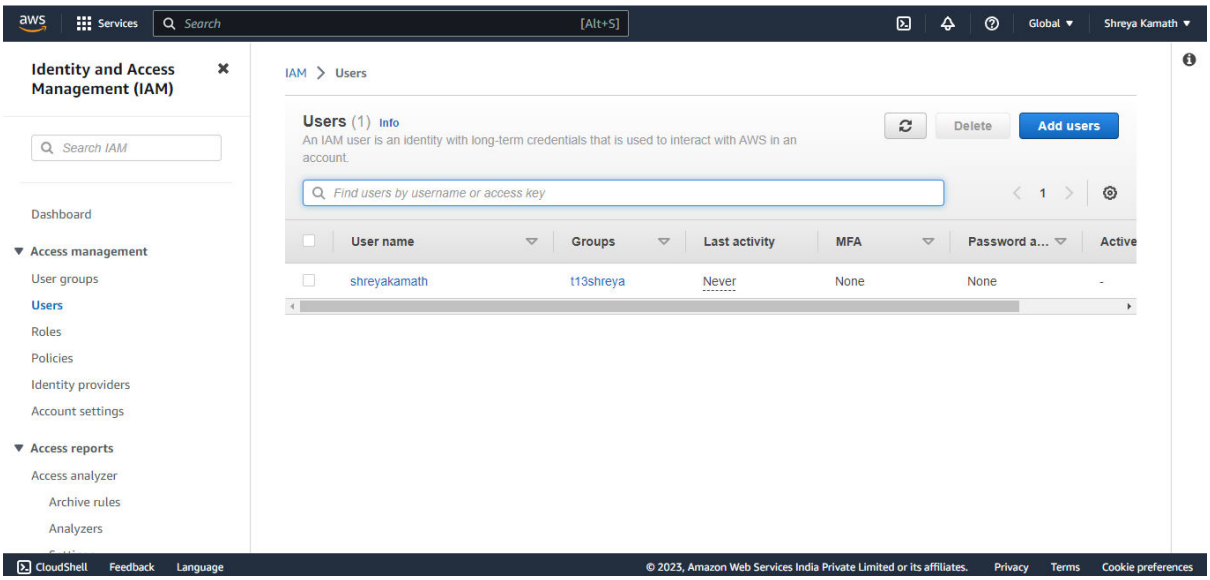

Create user group
Create a user group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. Learn more
User group name
Enter a meaningful name to identify this group.
t13shreya
Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.
Permissions policies (871)
Create policy
Filter by Type
Search
All ty...
Policy name
Type
Use...
Description
AdministratorAccess
AWS managed ...
None
Provides full access to AWS services
AdministratorAcce...
AWS managed
None
Grants account administrative perm
AdministratorAcce...
AWS managed
None
Grants account administrative perm
AlexaForBusinessD...
AWS managed
None
Provide device setup access to Alex
Cancel
Create user group

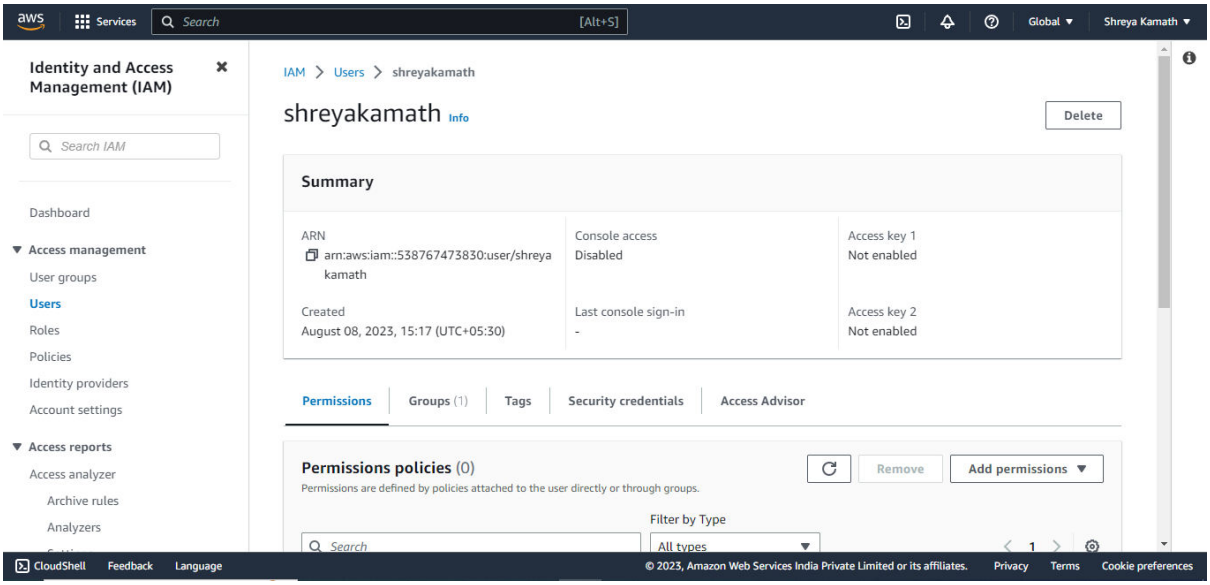
t13shreya user group created.
Review and create
Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.
Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.
User groups (1)
Create group
Search
Group name
Users
Attached policies
Created
t13shreya
0
-
2023-08-08 (Now)
Set permissions boundary - optional
Cancel
Previous
Next
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
29°C Mostly cloudy
ENG IN
3:17 PM 8/8/2023



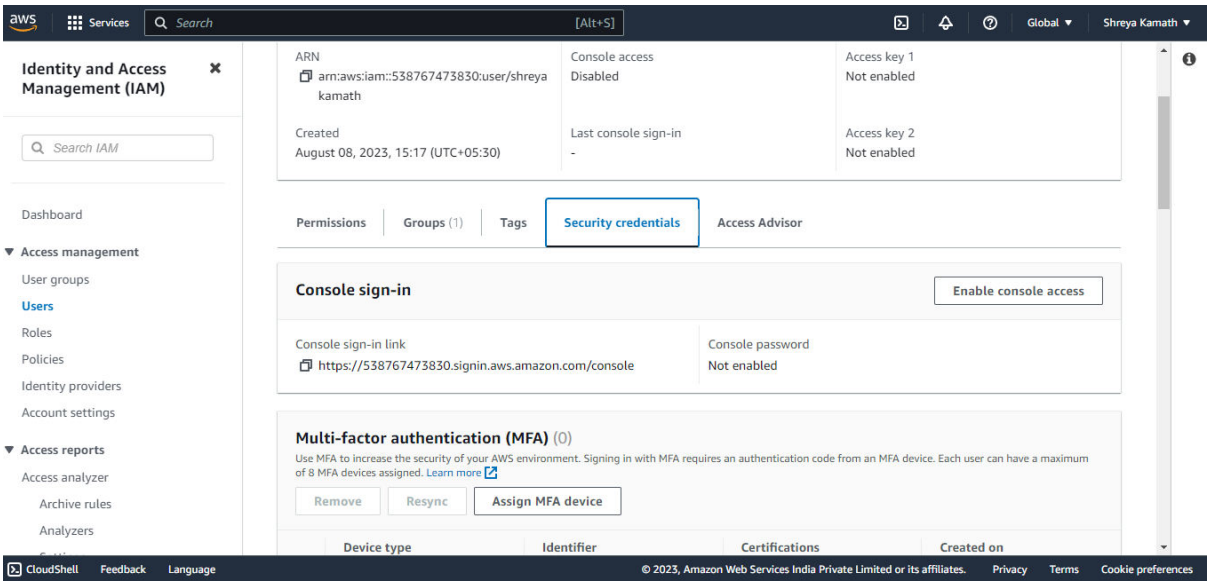
4)See the summary of created user and usergroup

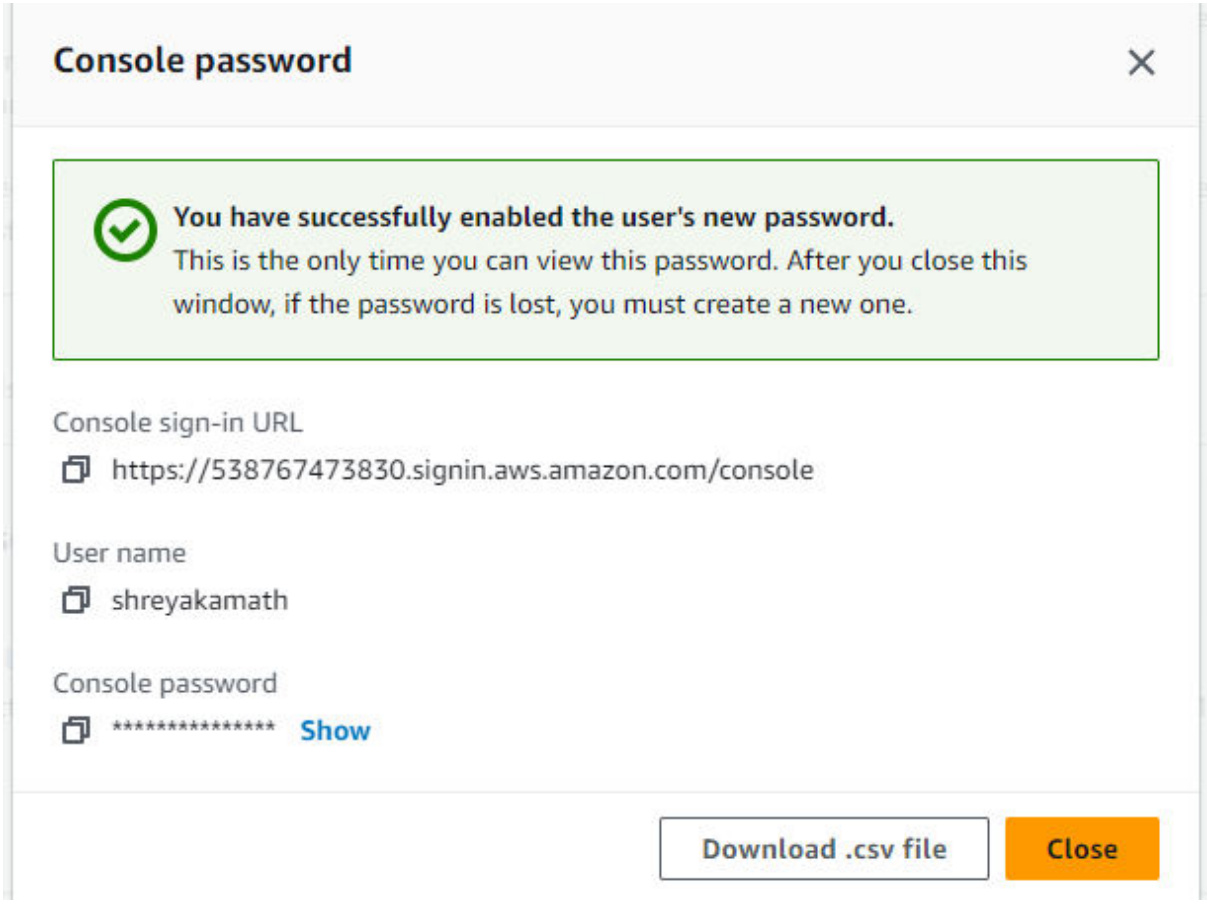




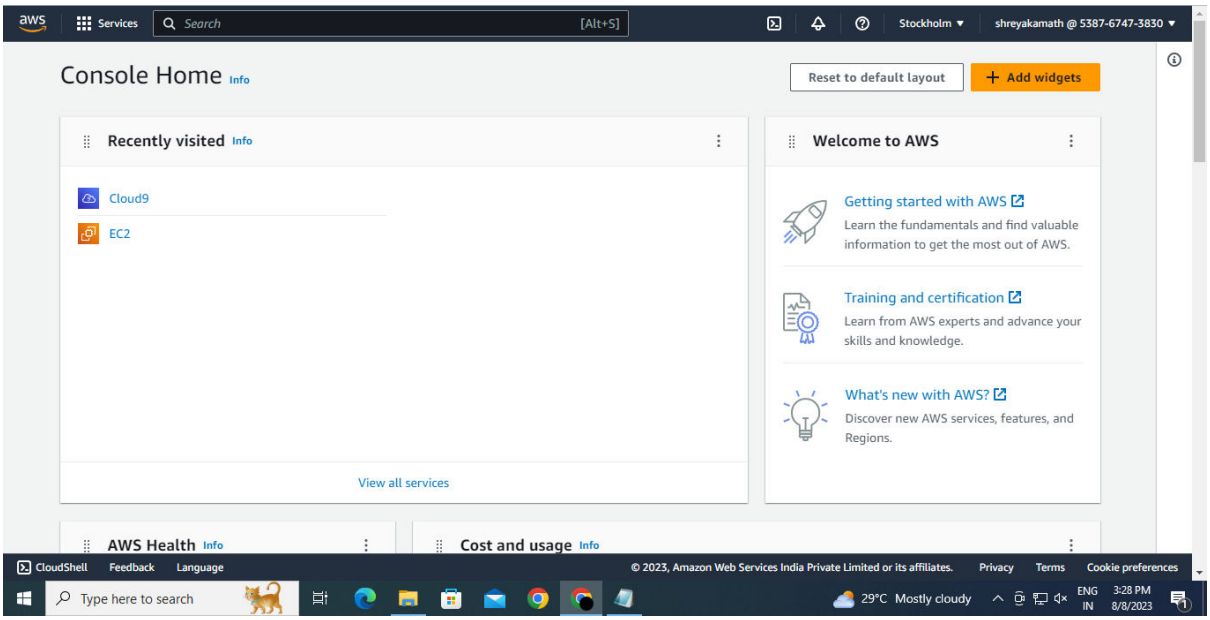


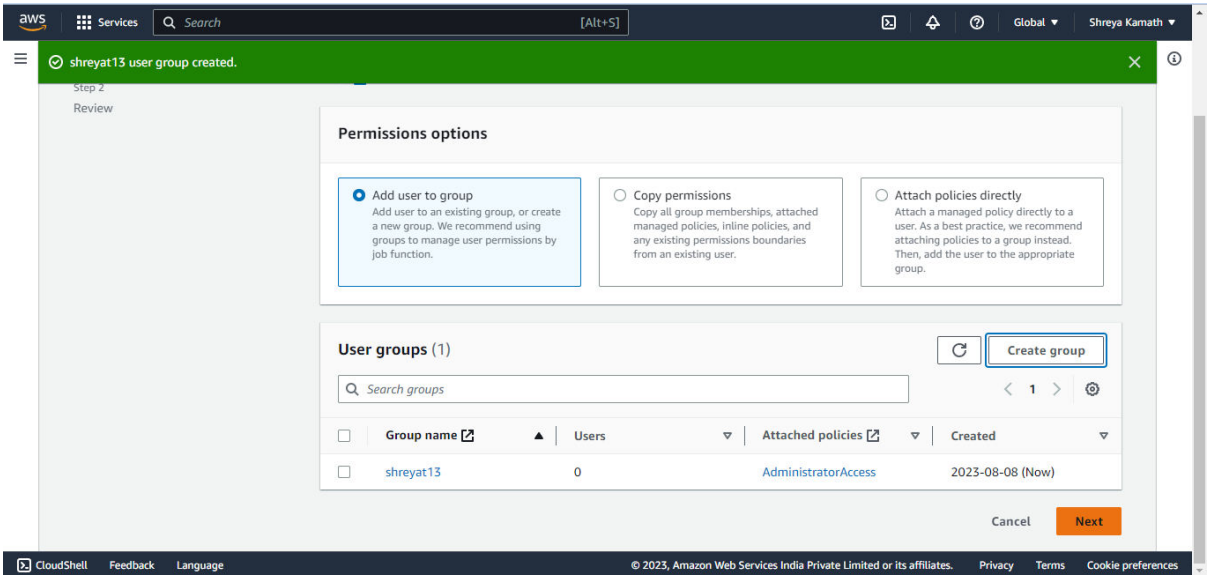
5)Go to security credentials tab and see for console password and other details , take screenshot of these details



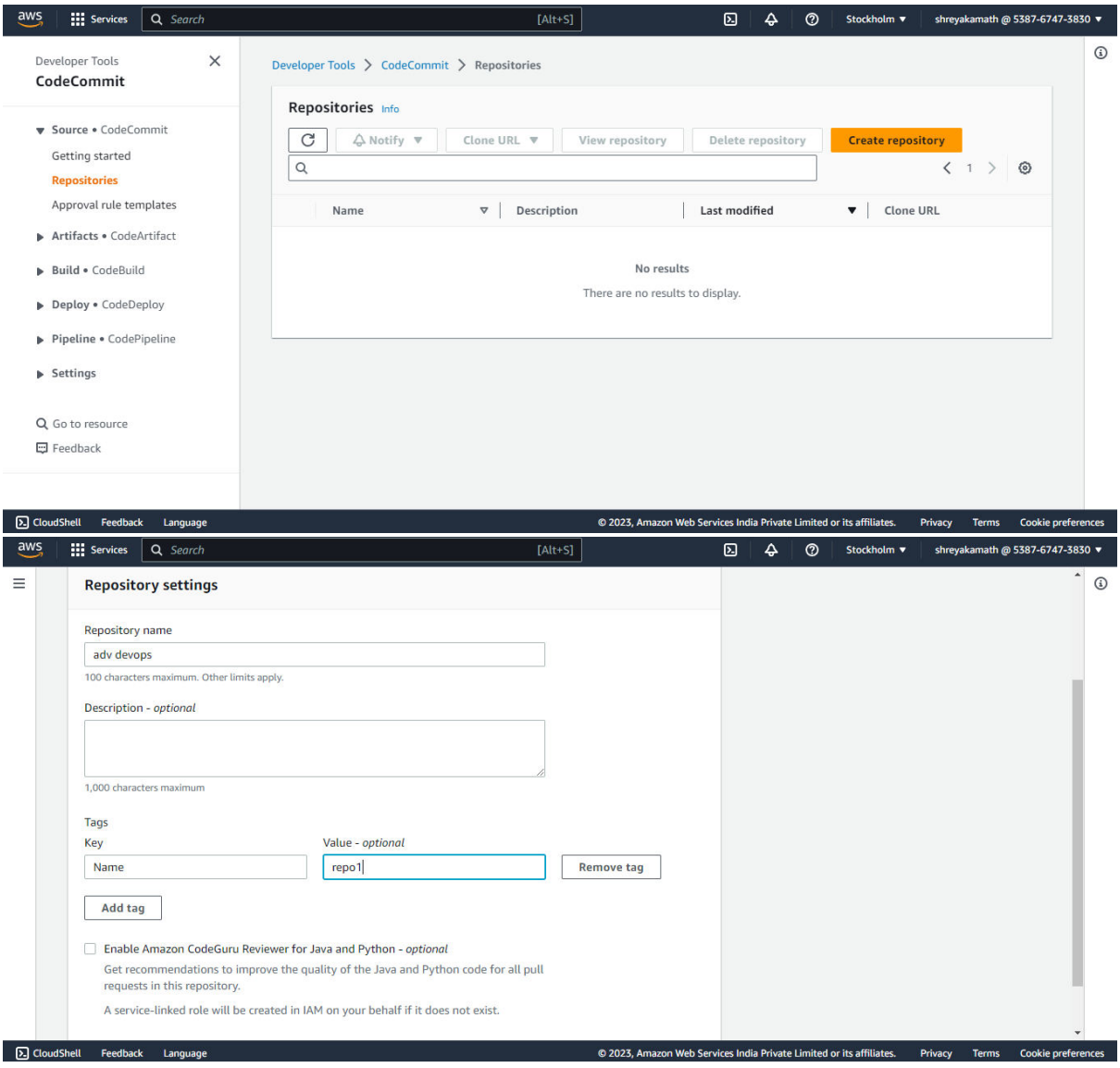


6)Go to dashboard and search codecommit in search bar

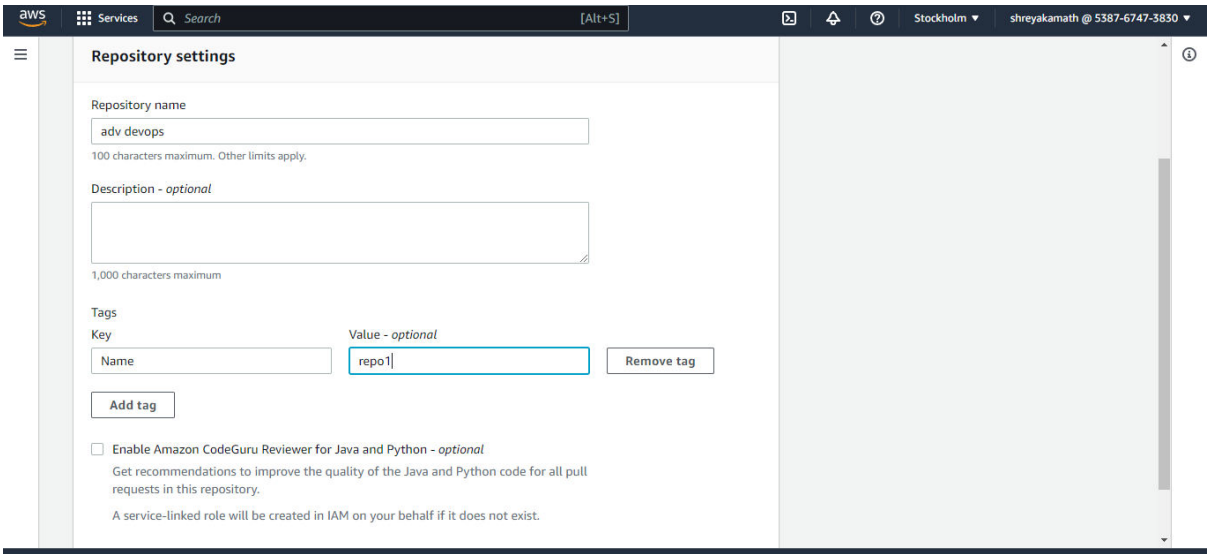


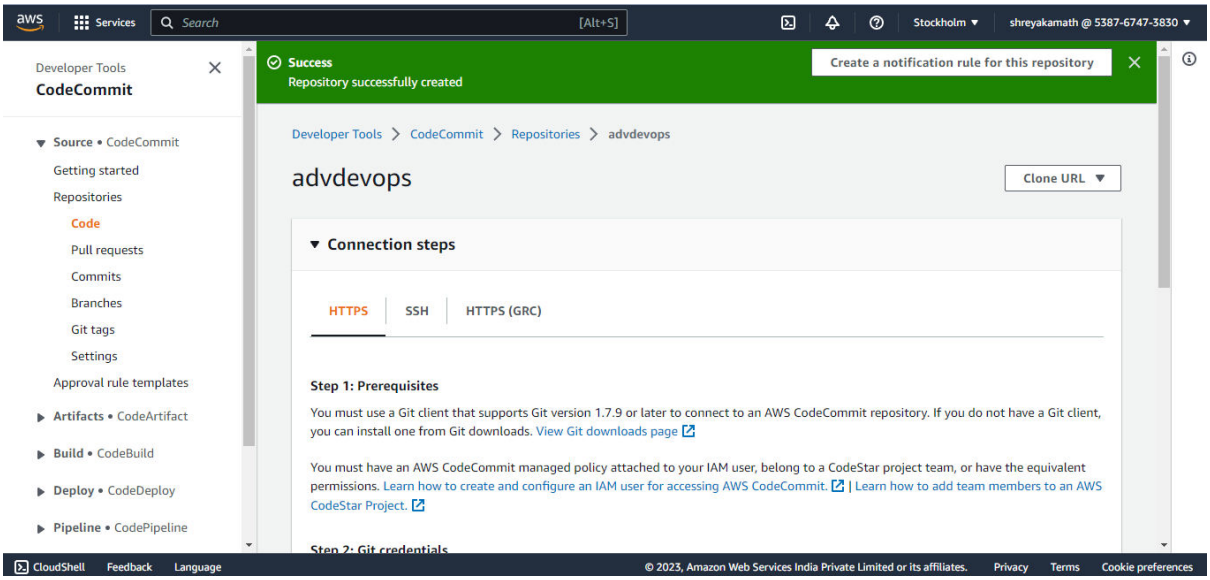


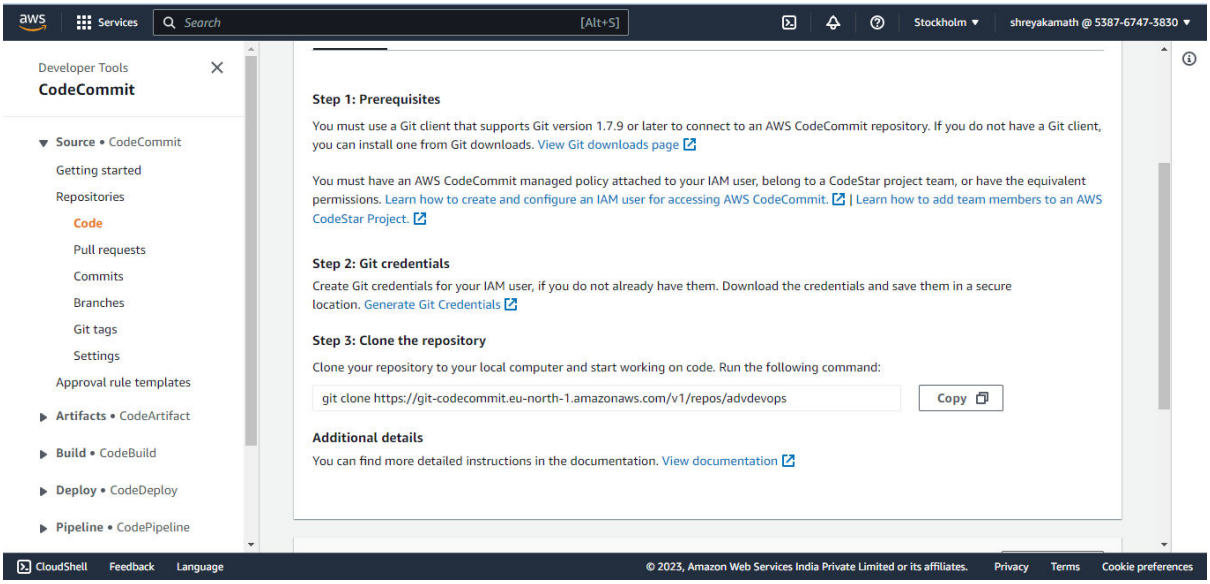
7) Go to repositories option and create a new repository , give a name to it



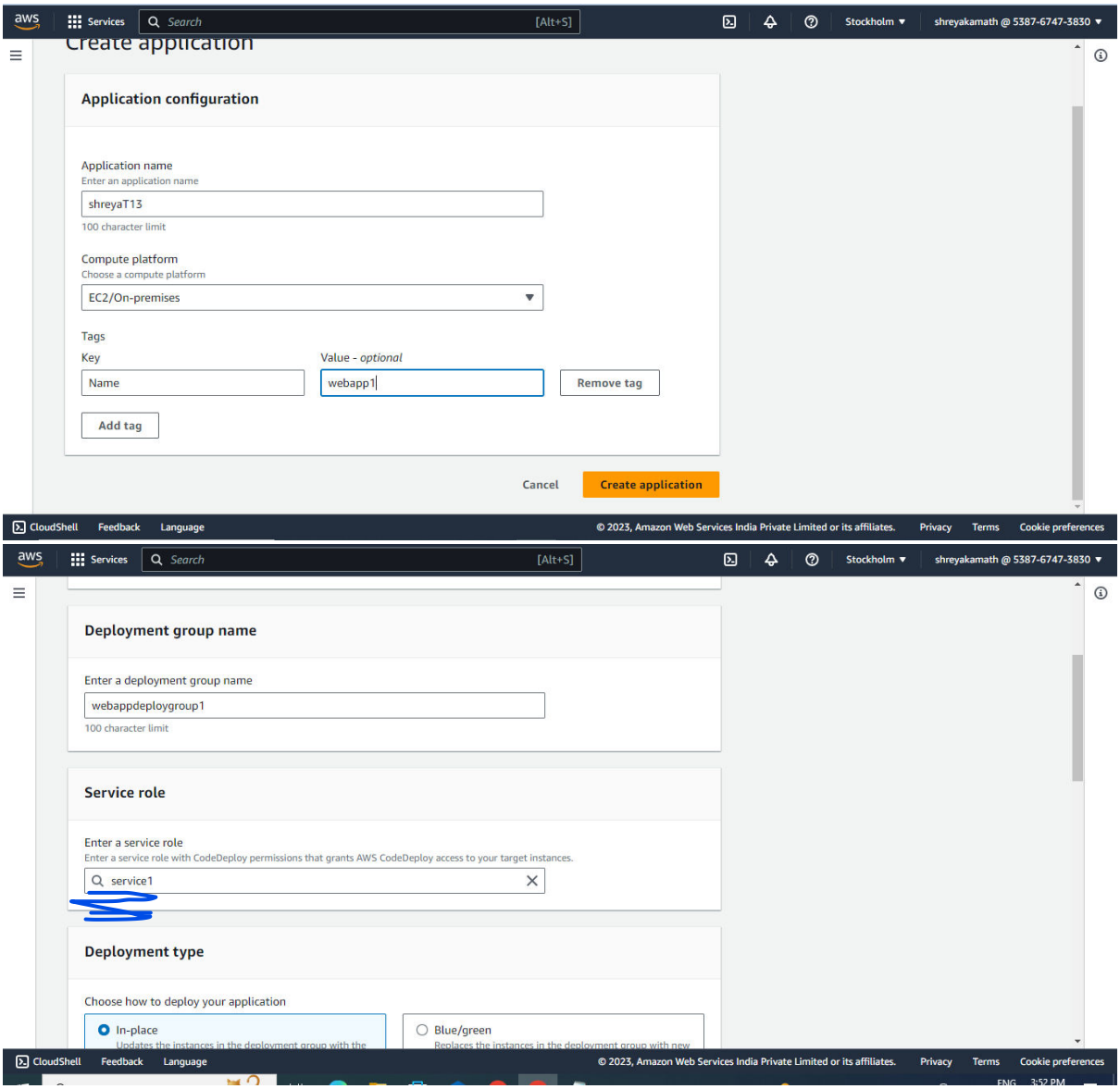
8) Go to code option in left sidebar and select https there

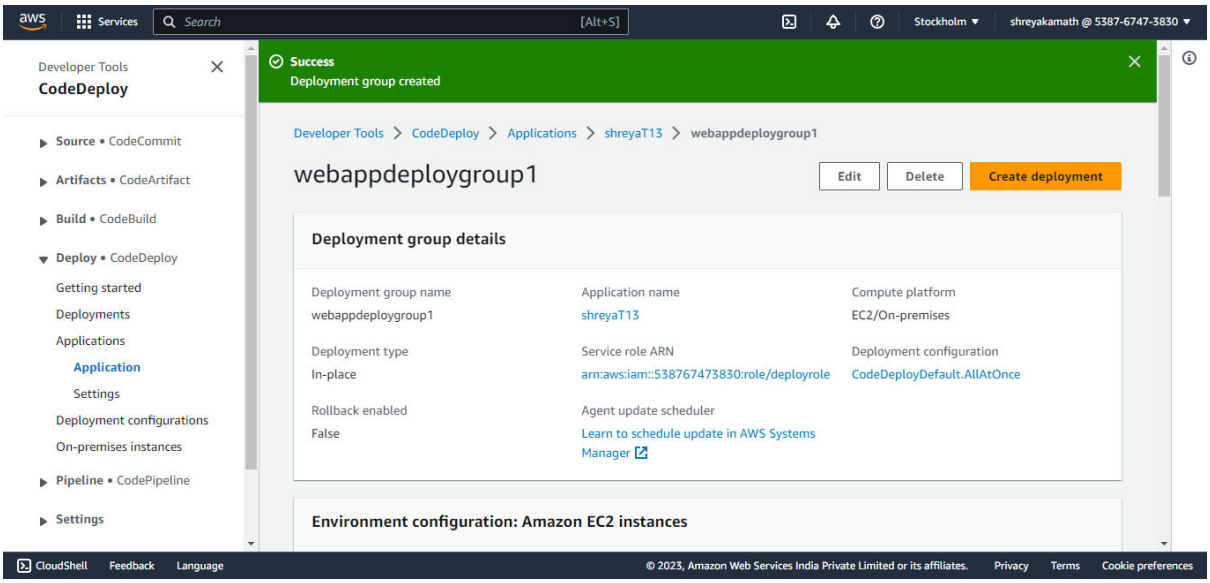

Services
Search
[Alt+S]
Stockholm
shreyakamath @ 5387-6747-3830
Repository settings
Repository name
adv devops
100 characters maximum. Other limits apply.
Description - optional
1,000 characters maximum
Tags
Key
Value - optional
Name
repo1
Remove tag
Add tag
Enable Amazon CodeGuru Reviewer for Java and Python - optional
Get recommendations to improve the quality of the Java and Python code for all pull requests in this repository.
A service-linked role will be created in IAM on your behalf if it does not exist.
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences


Services
Search
[Alt+S]
Stockholm
shreyakamath @ 5387-6747-3830
Developer Tools
CodeCommit
Source • CodeCommit
Getting started
Repositories
Code
Pull requests
Commits
Branches
Git tags
Settings
Approval rule templates
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Success
Repository successfully created
Create a notification rule for this repository
Developer Tools > CodeCommit > Repositories > advdevops
advdevops
Clone URL
Connection steps
HTTPS
SSH
HTTPS (GRC)
Step 1: Prerequisites
You must use a Git client that supports Git version 1.7.9 or later to connect to an AWS CodeCommit repository. If you do not have a Git client, you can install one from Git downloads. View Git downloads page
You must have an AWS CodeCommit managed policy attached to your IAM user, belong to a CodeStar project team, or have the equivalent permissions. Learn how to create and configure an IAM user for accessing AWS CodeCommit. | Learn how to add team members to an AWS CodeStar Project.
Step 2: Git credentials
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences

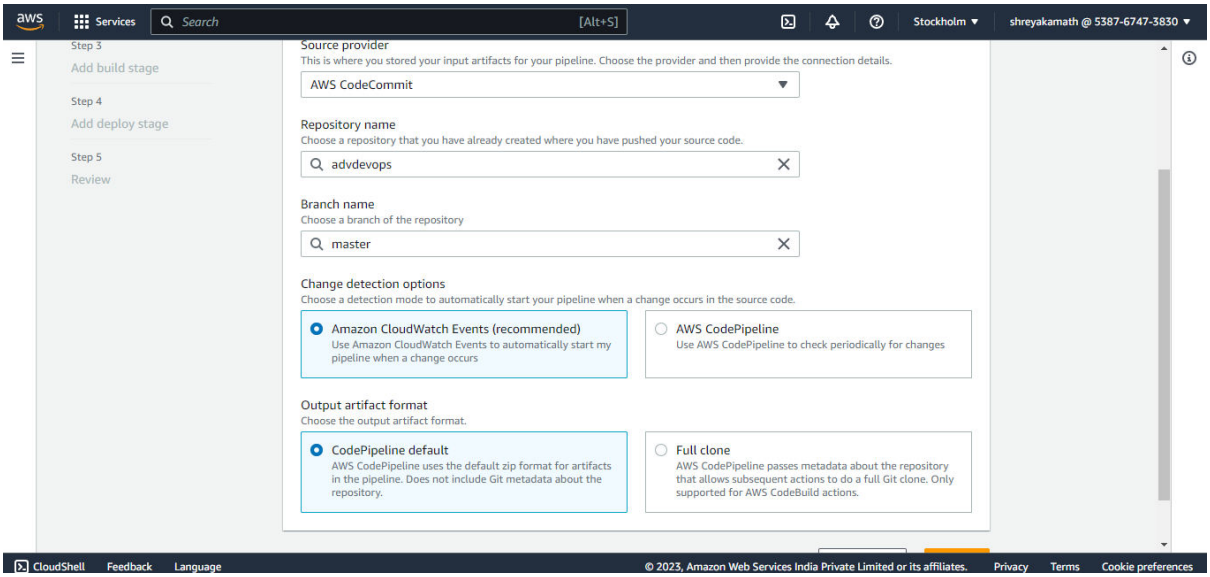

Services
Search
[Alt+S]
Stockholm
shreyakamath @ 5387-6747-3830
Developer Tools
CodeCommit
Source • CodeCommit
Getting started
Repositories
Code
Pull requests
Commits
Branches
Git tags
Settings
Approval rule templates
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Step 1: Prerequisites
You must use a Git client that supports Git version 1.7.9 or later to connect to an AWS CodeCommit repository. If you do not have a Git client, you can install one from Git downloads. View Git downloads page
You must have an AWS CodeCommit managed policy attached to your IAM user, belong to a CodeStar project team, or have the equivalent permissions. Learn how to create and configure an IAM user for accessing AWS CodeCommit. | Learn how to add team members to an AWS CodeStar Project.
Step 2: Git credentials
Create Git credentials for your IAM user, if you do not already have them. Download the credentials and save them in a secure location. Generate Git Credentials
Step 3: Clone the repository
Clone your repository to your local computer and start working on code. Run the following command:
git clone https://git-codecommit.eu-north-1.amazonaws.com/v1/repos/advdevops
Copy
Additional details
You can find more detailed instructions in the documentation. View documentation
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences

9) Create a new application and give name to it , create a deployment group for deployment of application created

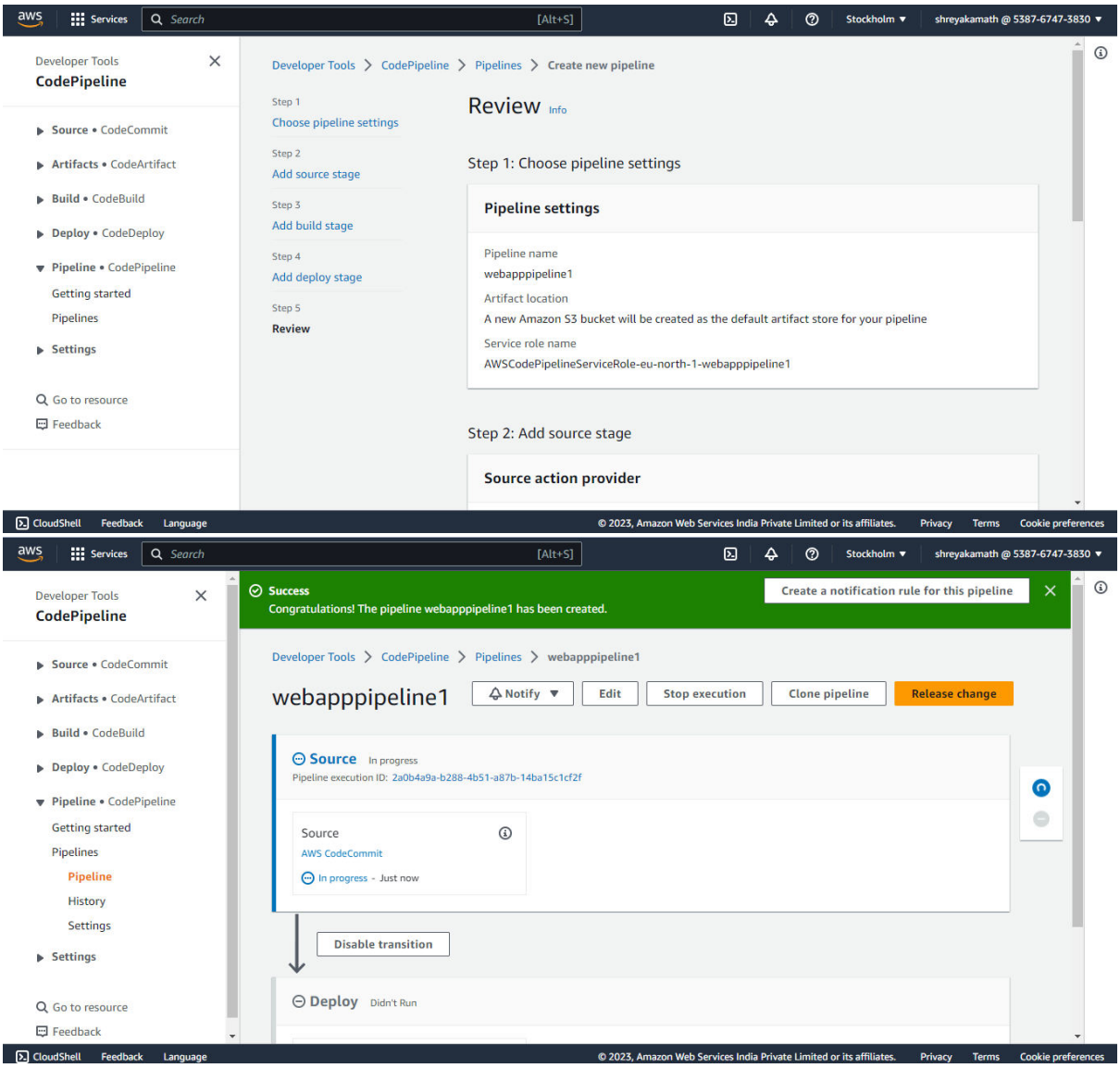




10) Select the source provider , artifact



11) Review the application and then click on deploy , after deploying success message is displayed on screen



Conclusion:-

Learnt about creation and deployment of web application using AWS codepipeline , hence learnt about basic components of codepipeline in AWS such as codebuild , codecommit , codedeploy and built , committed and deployed a web application using codepipeline

Name :-Niranjan Rajesh Joshi
Roll No:- 2105051
Batch:- T13
Date Of Performance :- 13/10/2023

Experiment 5

Aim:-

To understand the Kubernetes Cluster Architecture, install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms

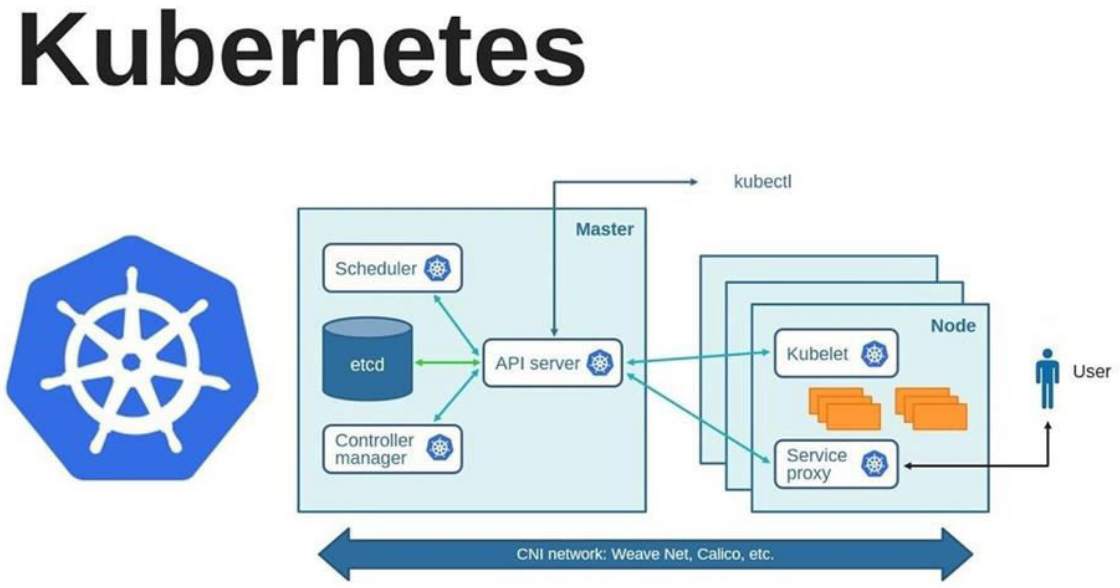
Theory:-

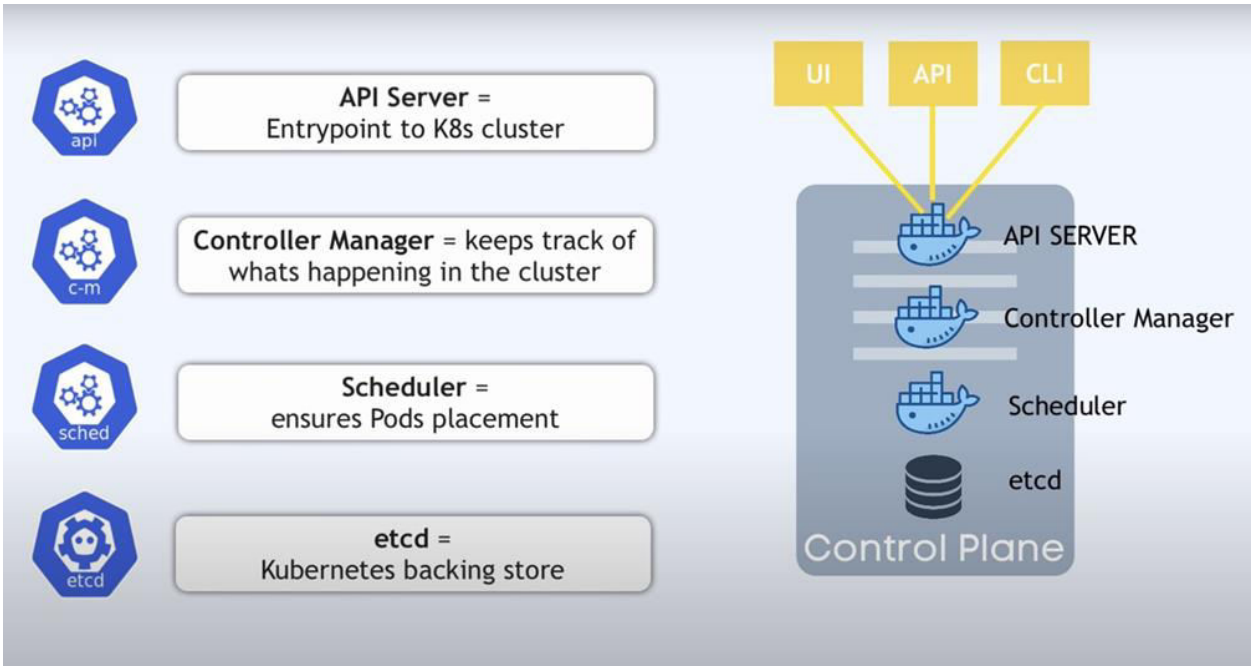
Kubernetes is an open-source container management tool that automates container deployment, scaling & load balancing.

It schedules, runs, and manages isolated containers that are running on virtual/physical/cloud machines.

All top cloud providers support Kubernetes. One popular name for Kubernetes is K8s.

ARCHITECTURE





Q.1 What are the various Kubernetes services running on nodes? Describe the role of each service.

In a Kubernetes cluster, several services run on each node to enable the proper functioning of the cluster. These services play crucial roles in managing, scheduling, and maintaining the various components of the cluster. Some of the essential Kubernetes services running on nodes include:

kubelet: It is the primary node agent that runs on each node in the cluster and ensures that the containers are running in a Pod. Kubelet communicates with the Kubernetes control plane and manages the Pods and their containers according to the instructions received.

kube-proxy: Kube-proxy is responsible for network proxying on the node and performs the necessary network address translation and load balancing to route the traffic to the appropriate Pod. It maintains network rules on the node and enables communication between different Pods and services in the cluster.

container runtime: Kubernetes supports various container runtimes such as Docker, containerd, and CRI-O. The container runtime is responsible for pulling the container images from the container registry, creating the container from those images, and managing the container lifecycle, including starting, stopping, and deleting containers.

kube-dns/coredns: These are the DNS services in Kubernetes that provide service discovery and enable communication between different services within the cluster. They ensure that the Pods

can locate each other by their DNS names and facilitate communication between different components.

kube-scheduler: The kube-scheduler is responsible for making decisions about which nodes should run specific Pods based on resource requirements, hardware constraints, and other policies. It assigns the Pods to nodes that can accommodate them and meet the specified criteria.

kube-controller-manager: This component includes several controllers that handle different aspects of the cluster, such as node and endpoint controllers, which perform tasks like node monitoring and managing endpoints. It ensures that the desired state of the cluster is maintained and takes necessary actions to reconcile any discrepancies.

etcd: While not strictly a service running on the node, etcd is a consistent and highly available key-value store used by Kubernetes to store all of its data. It maintains the state of the cluster, including configuration data, metadata, and the current state of all objects in the system.

Each of these services plays a critical role in ensuring the proper functioning of a Kubernetes cluster, enabling efficient management and orchestration of containerized applications.

Q.2 What is Pod Disruption Budget (PDB)?

A Pod Disruption Budget (PDB) is a Kubernetes feature that allows user to control the number of Pods that are down simultaneously in a given deployment or replica set. It ensures high availability by specifying the minimum number of Pods that must be available at any given time during voluntary disruptions, such as maintenance or scaling events.

PDBs help to prevent situations where too many Pods are terminated simultaneously, leading to potential service disruption or downtime. They allow user to set constraints on the disruption budget, ensuring that a minimum number of Pods remain available even when nodes or underlying hardware experience failures or planned maintenance activities.

By defining a Pod Disruption Budget, user can maintain a desired level of availability and reliability for users applications, especially in scenarios where user need to perform updates, upgrades, or scaling operations without causing unnecessary downtime or service interruptions.

Q.3 What is the role of Load Balance in Kubernetes?

In Kubernetes, a load balancer serves the crucial role of distributing incoming network traffic across multiple backend services or Pods. Its primary functions include:

Traffic distribution: It evenly distributes incoming network traffic across multiple instances of an application or service running in the Kubernetes cluster. This ensures that no single Pod or node is overwhelmed by an excessive amount of traffic.

High availability: Load balancing helps ensure high availability and reliability of applications by preventing any single point of failure. If one Pod or node becomes unavailable, the load balancer redirects traffic to other available instances, thereby minimizing downtime and service disruptions.

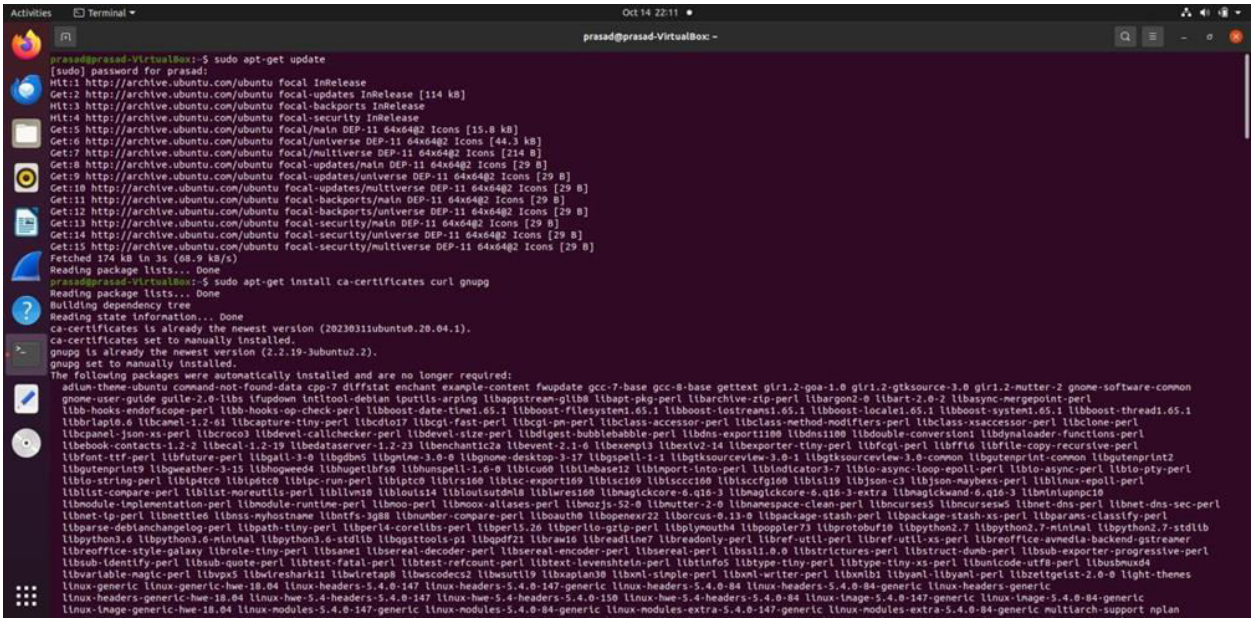
Scalability: Load balancing facilitates horizontal scaling by dynamically adding or removing Pods to handle varying levels of traffic. This enables the system to adapt to changes in demand and maintain consistent performance even during peak usage periods.

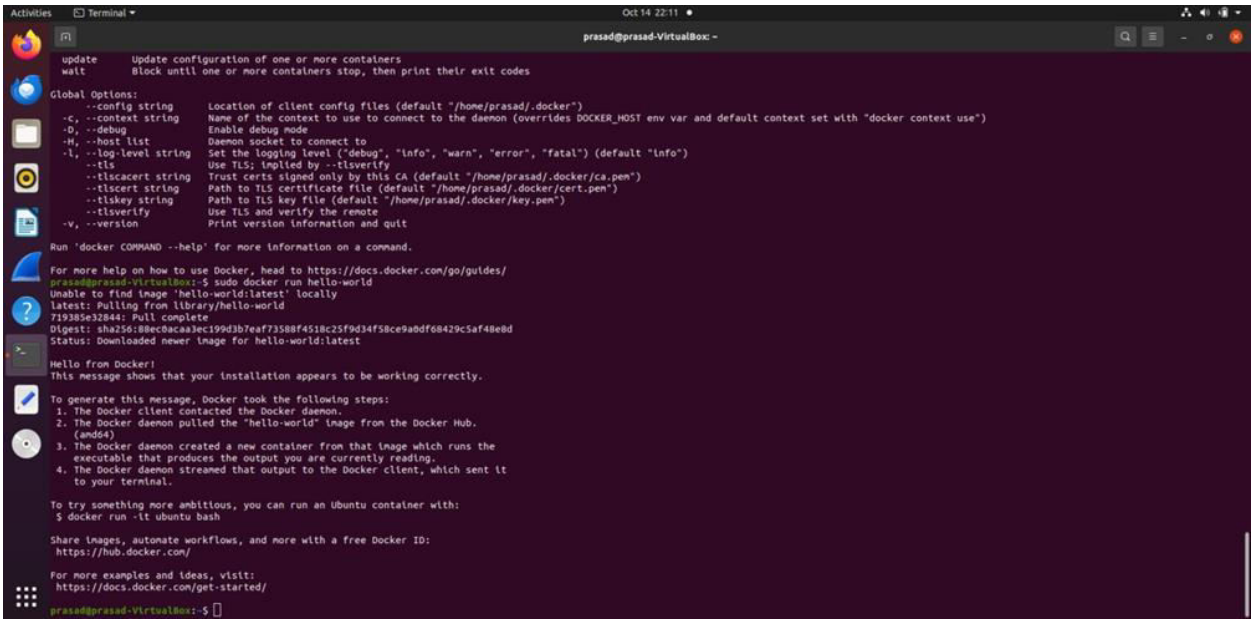
Optimized resource utilization: By efficiently distributing traffic, a load balancer helps in optimizing resource utilization across the cluster. It ensures that resources are utilized effectively without overloading any specific node or Pod.

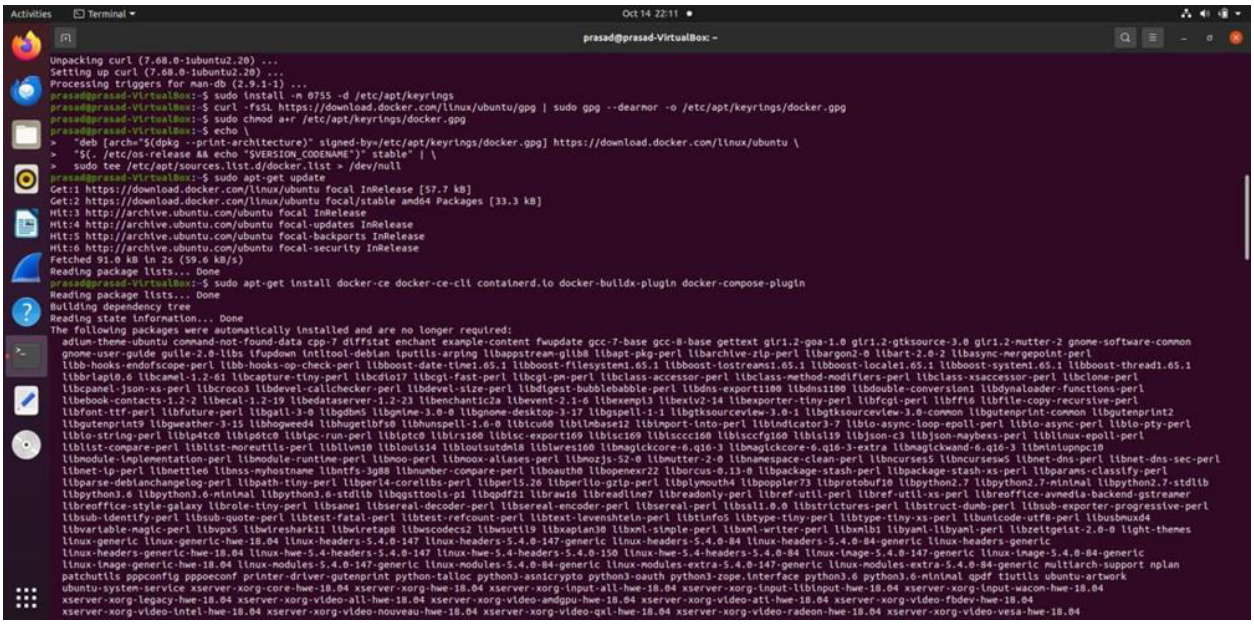
Overall, the load balancer plays a critical role in ensuring the reliable and efficient operation of applications in a Kubernetes cluster, enabling seamless traffic management, high availability, and scalability to meet varying demands.

INSTALLATION:

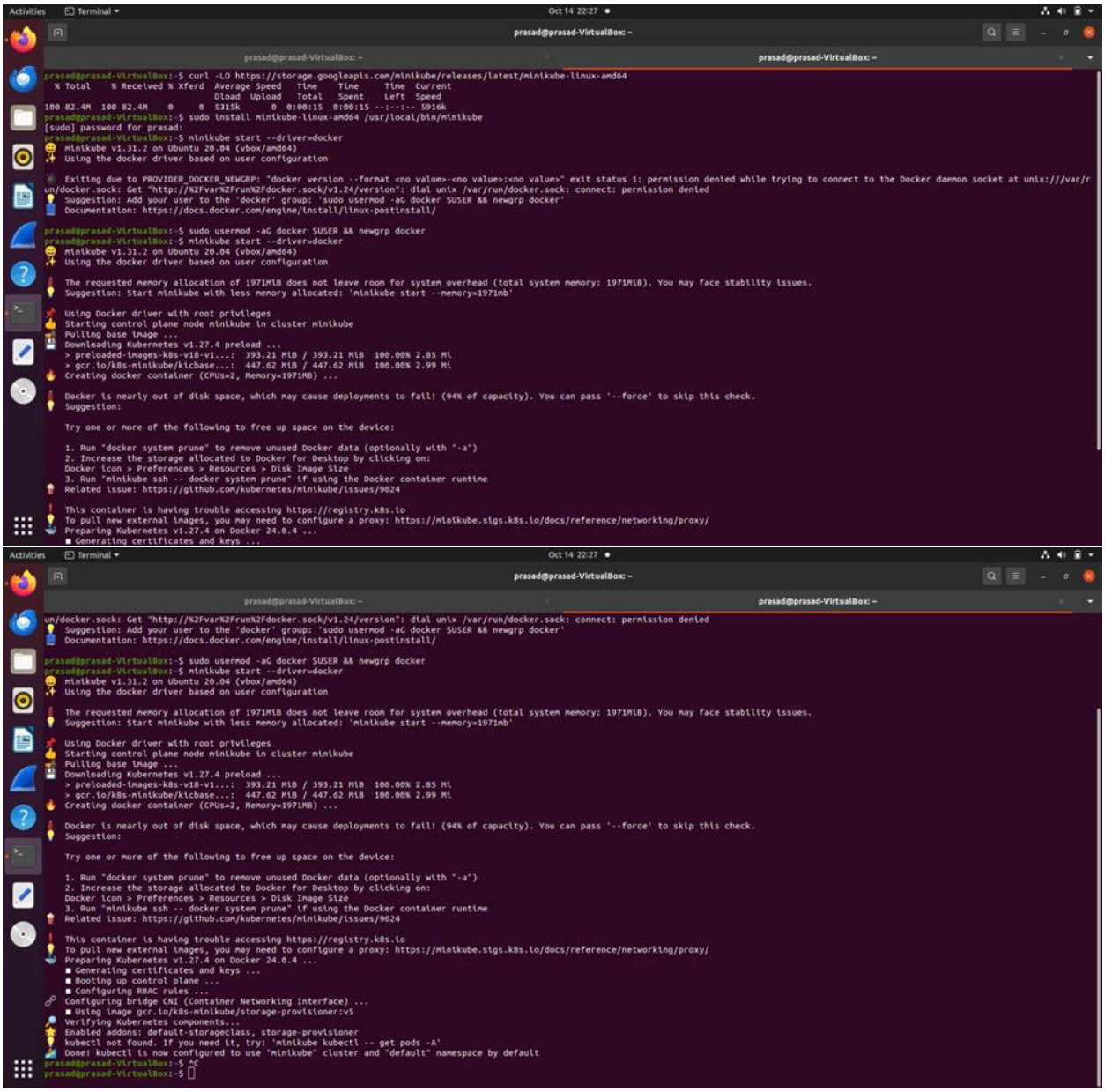
1. Install Docker



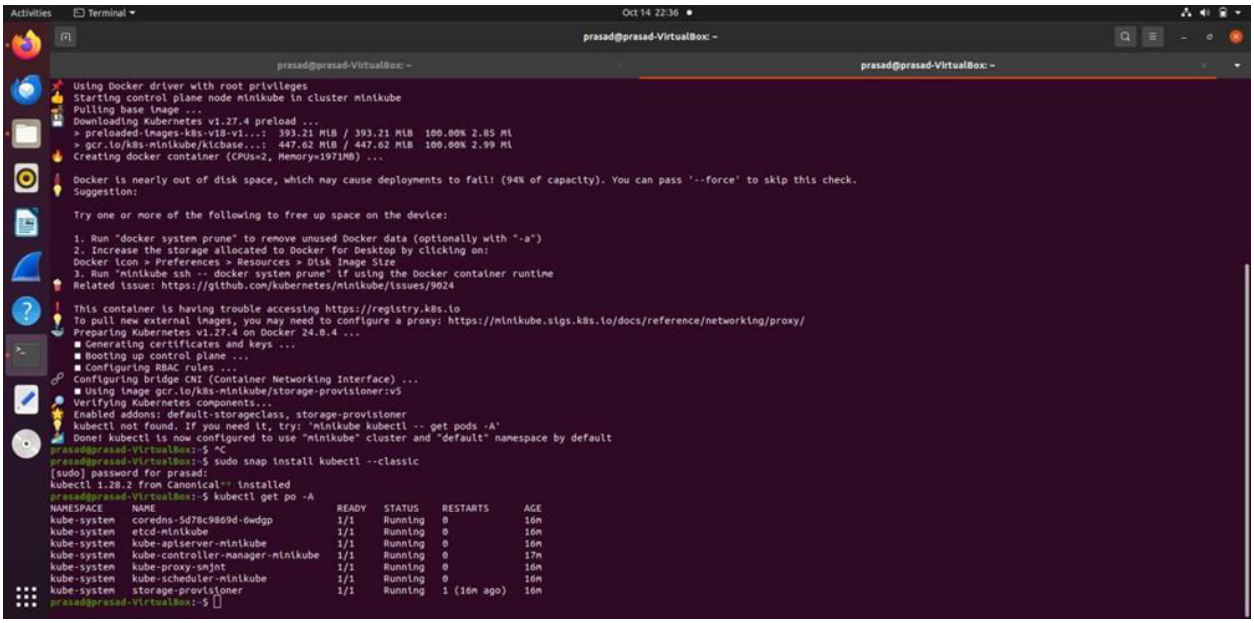




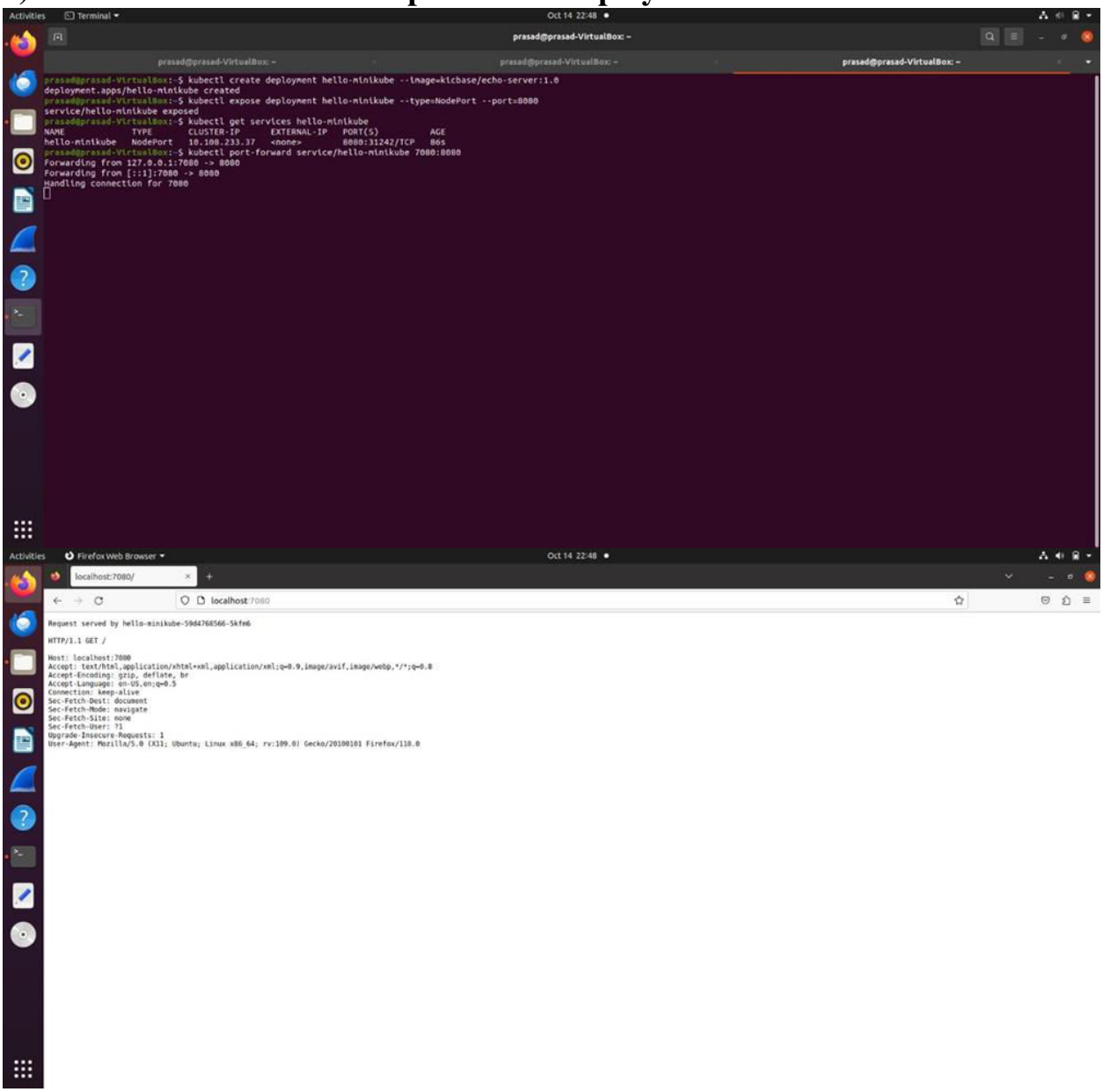
2) Install minikube using following commands



3) Install kubectl


Activities
Terminal
Oct 14 22:36
prasad@prasad-VirtualBox: ~
prasad@prasad-VirtualBox: ~
prasad@prasad-VirtualBox: ~
Using Docker driver with root privileges
Starting control plane node minikube in cluster minikube
Pulling base image ...
Downloading Kubernetes v1.27.4 preload ...
> preloaded-images-k8s-v18-v1...: 393.21 MiB / 393.21 MiB 100.00% 2.85 Mi
> gcr.io/k8s-minikube/kicbase...: 447.62 MiB / 447.62 MiB 100.00% 2.99 Mi
Creating docker container (CPUs=2, Memory=1971MB) ...
Docker is nearly out of disk space, which may cause deployments to fail! (94% of capacity). You can pass '--force' to skip this check.
Suggestion:
Try one or more of the following to free up space on the device:
1. Run "docker system prune" to remove unused Docker data (optionally with "-a")
2. Increase the storage allocated to Docker for Desktop by clicking on:
Docker Icon > Preferences > Resources > Disk Image Size
3. Run "minikube ssh -- docker system prune" if using the Docker container runtime
Related issue: https://github.com/kubernetes/minikube/issues/9024
This container is having trouble accessing https://registry.k8s.io
To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/reference/networking/proxy/
Preparing Kubernetes v1.27.4 on Docker 24.0.4 ...
■ Generating certificates and keys ...
■ Booting up control plane ...
■ Configuring RBAC rules ...
Configuring bridge CNI (Container Networking Interface) ...
■ Using image gcr.io/k8s-minikube/storage-provisioner:v5
Verifying Kubernetes components...
Enabled addons: default-storageclass, storage-provisioner
kubectl not found. If you need it, try: 'minikube kubectl -- get pods -A'
Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
prasad@prasad-VirtualBox:~$ ^C
prasad@prasad-VirtualBox:~$ sudo snap install kubectl --classic
[sudo] password for prasad:
kubectl 1.28.2 from Canonical** installed
prasad@prasad-VirtualBox:~$ kubectl get po -A
NAMESPACE NAME READY STATUS RESTARTS AGE
kube-system coredns-5d78c9869d-6wdgp 1/1 Running 0 16m
kube-system etcd-minikube 1/1 Running 0 16m
kube-system kube-apiserver-minikube 1/1 Running 0 16m
kube-system kube-controller-manager-minikube 1/1 Running 0 17m
kube-system kube-proxy-smjnt 1/1 Running 0 16m
kube-system kube-scheduler-minikube 1/1 Running 0 16m
kube-system storage-provisioner 1/1 Running 1 (16m ago) 16m
prasad@prasad-VirtualBox:~$

4) Create Sample Deployment



Conclusion:-

Here we studied Kubernetes cluster architecture in detail. Also we installed Kubernetes in ubuntu machine and created a sample deployment.

Name :-Niranjan Rajesh Joshi

Roll No:- 2105051

Batch:- T13

Date Of Performance :- 22/08/2023

Experiment 6

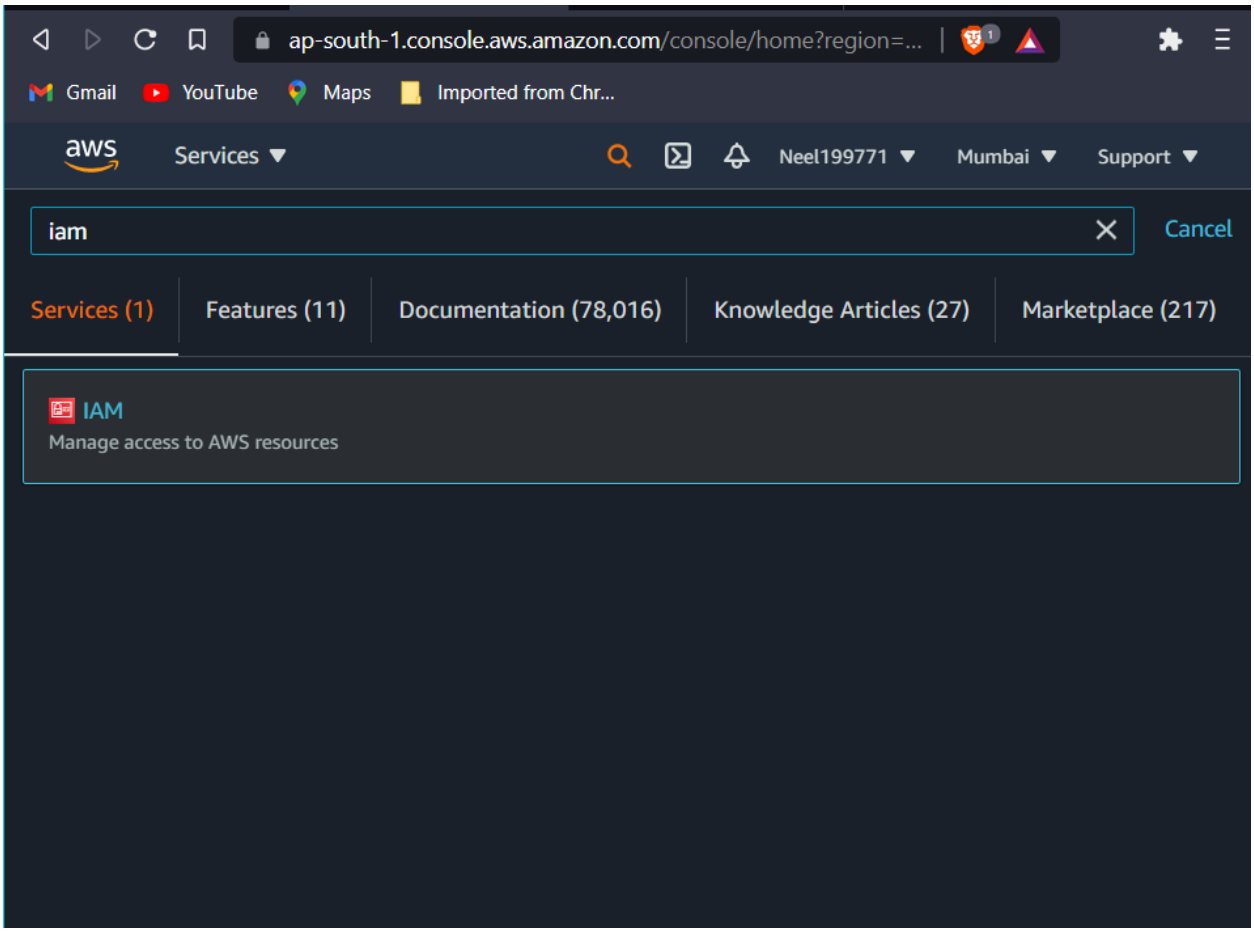
Aim :-

To understand terraform lifecycle and to build, change , and destroy AWS infrastructure using Terraform

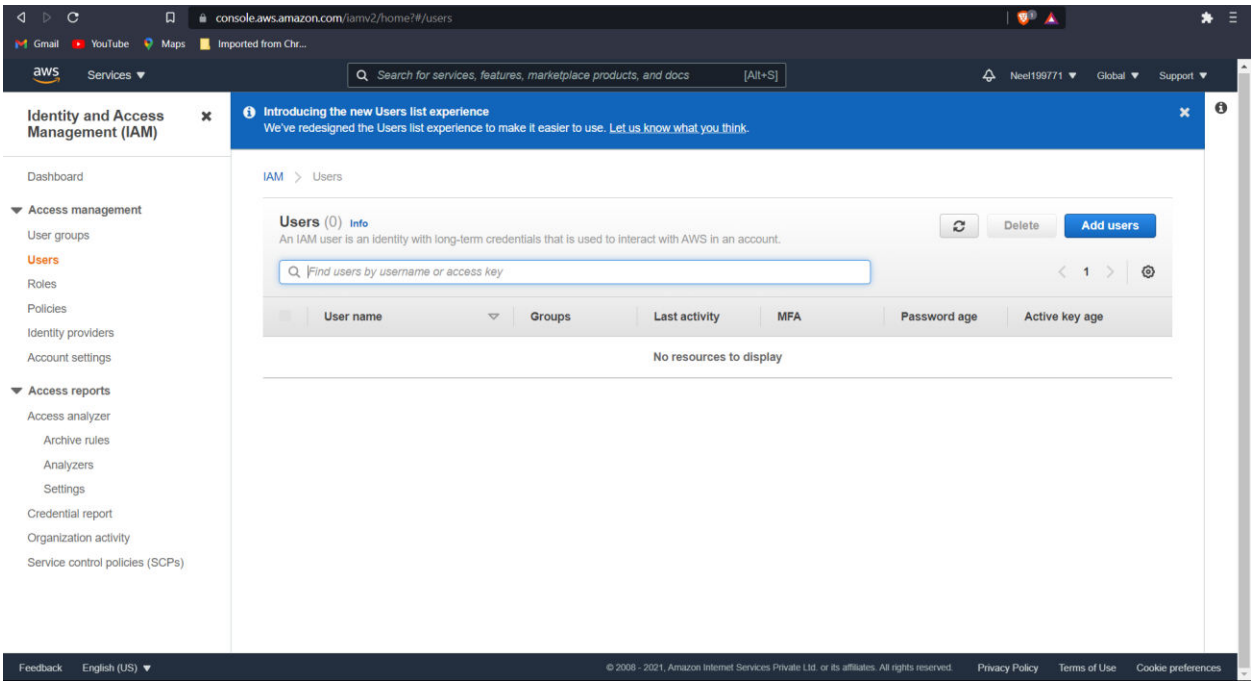
Steps :-

Terraform is an infrastructure as code (IaC) tool that allows you to build, change, and version infrastructure safely and efficiently.

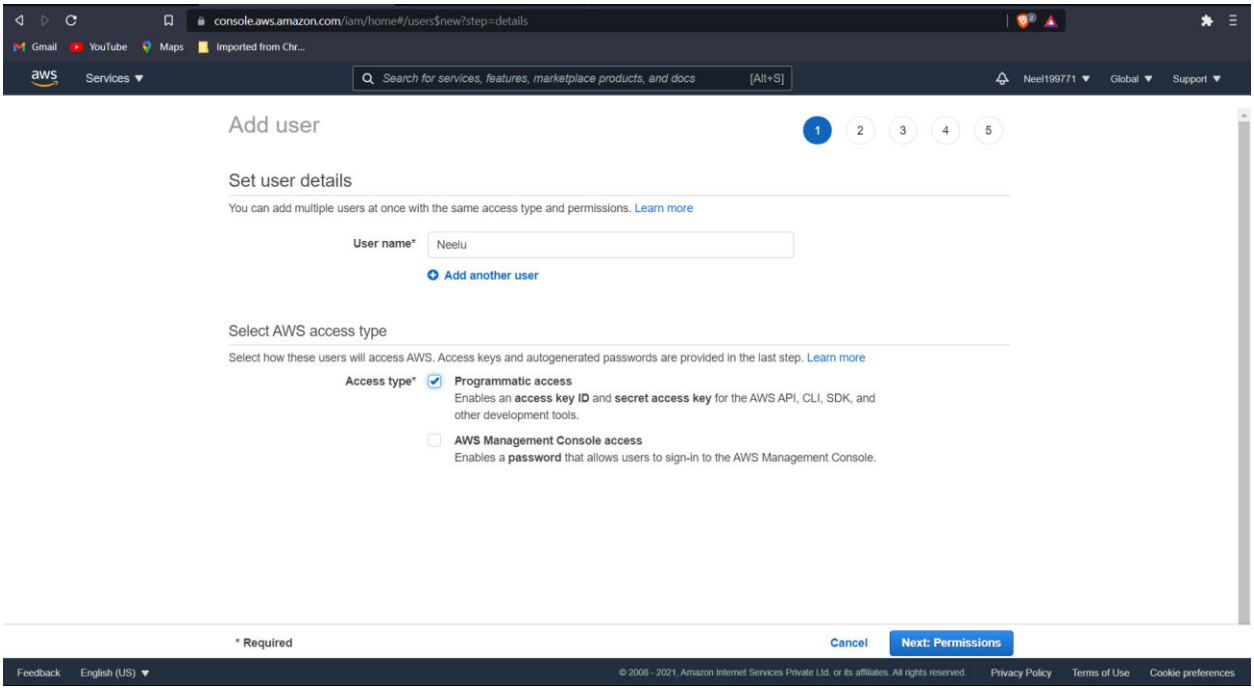
1) Open And Login to your AWS console-And search IAM and click on it



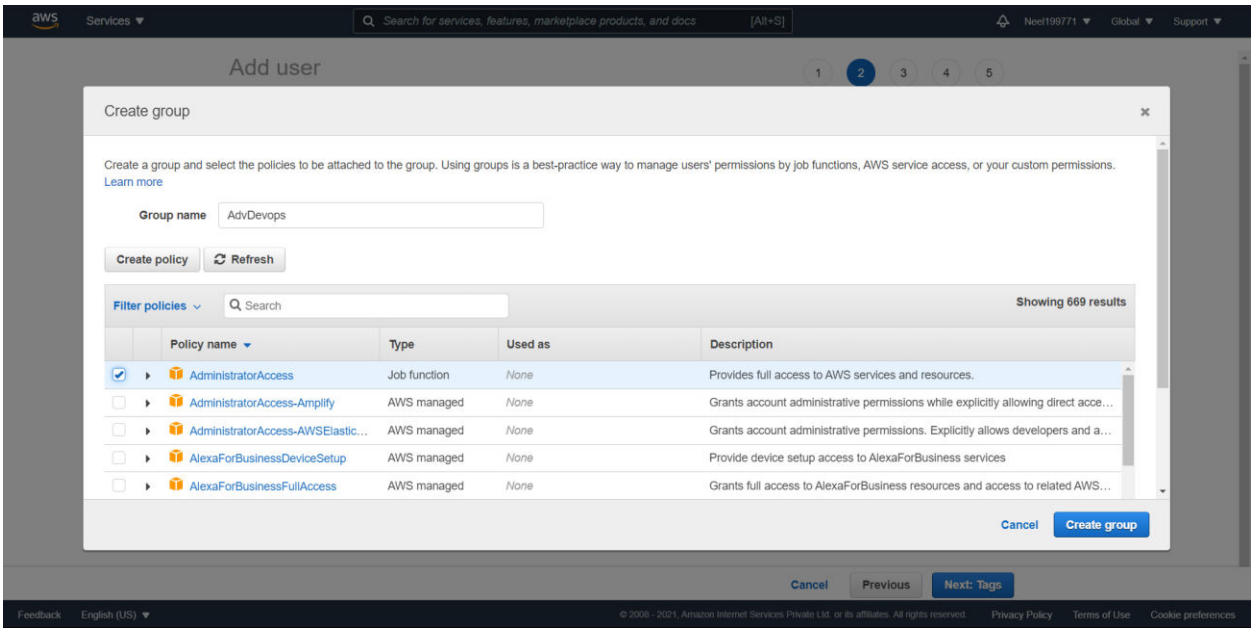
2) Now click on Add Users in The User Section as shown in the image



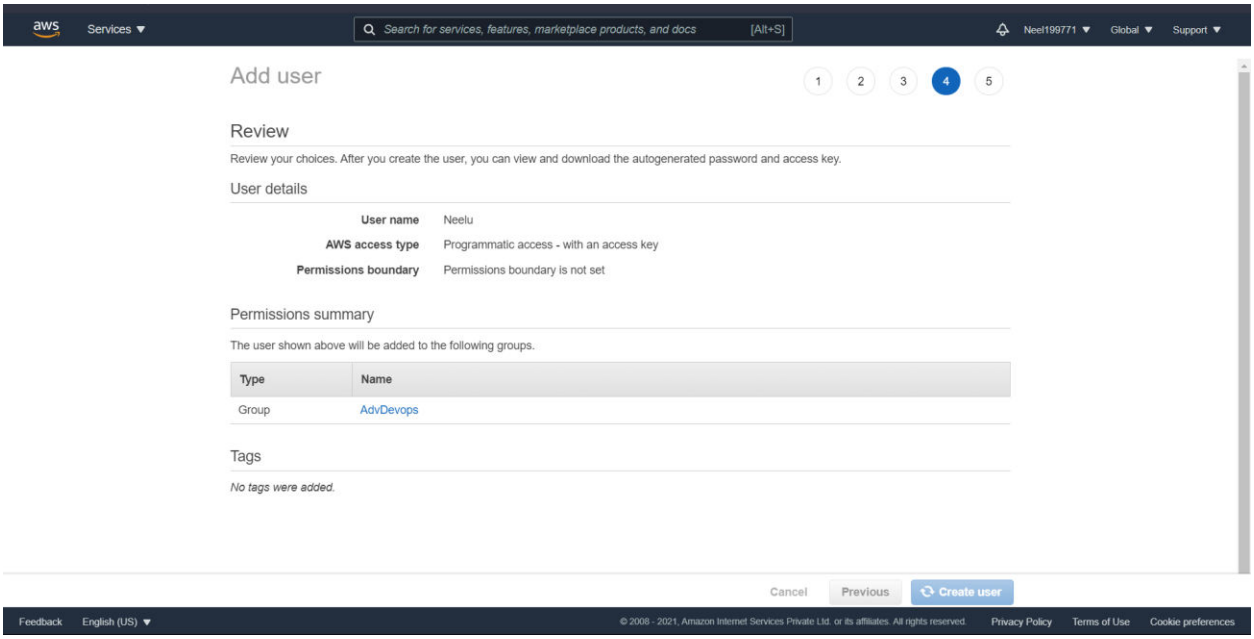
3) Now give any name For username And Check The Programmatic Access field shown below



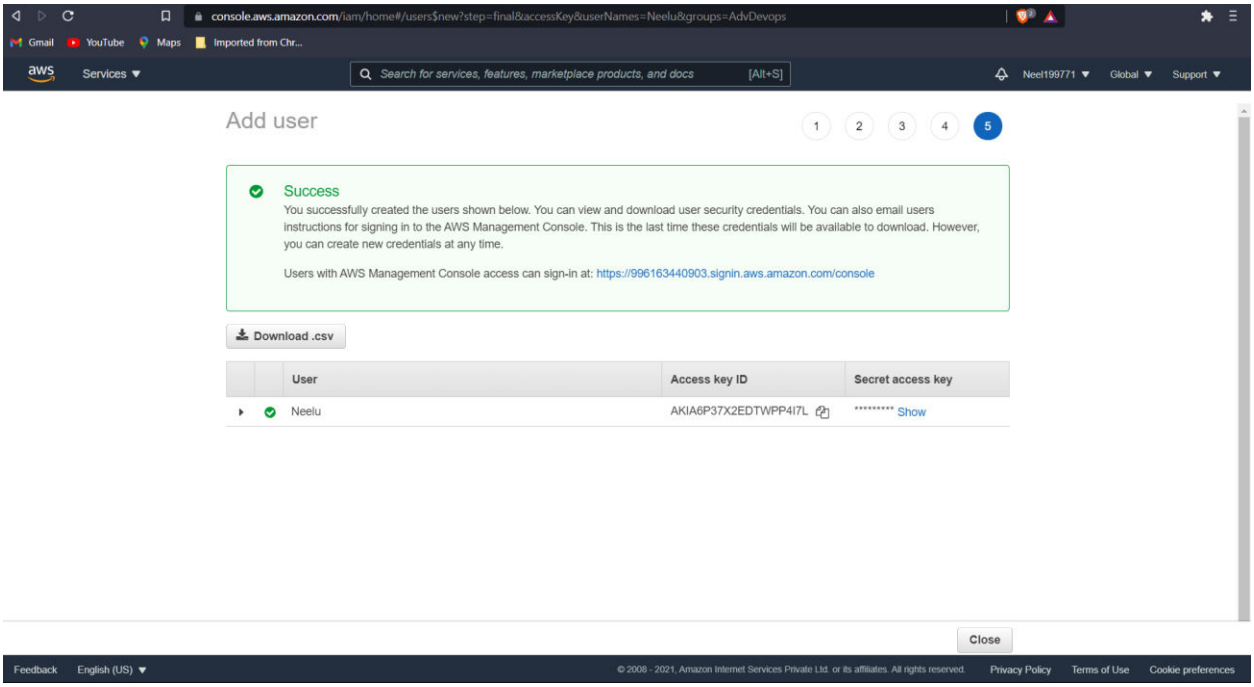
4) Add Group name and Check the first Policy Name



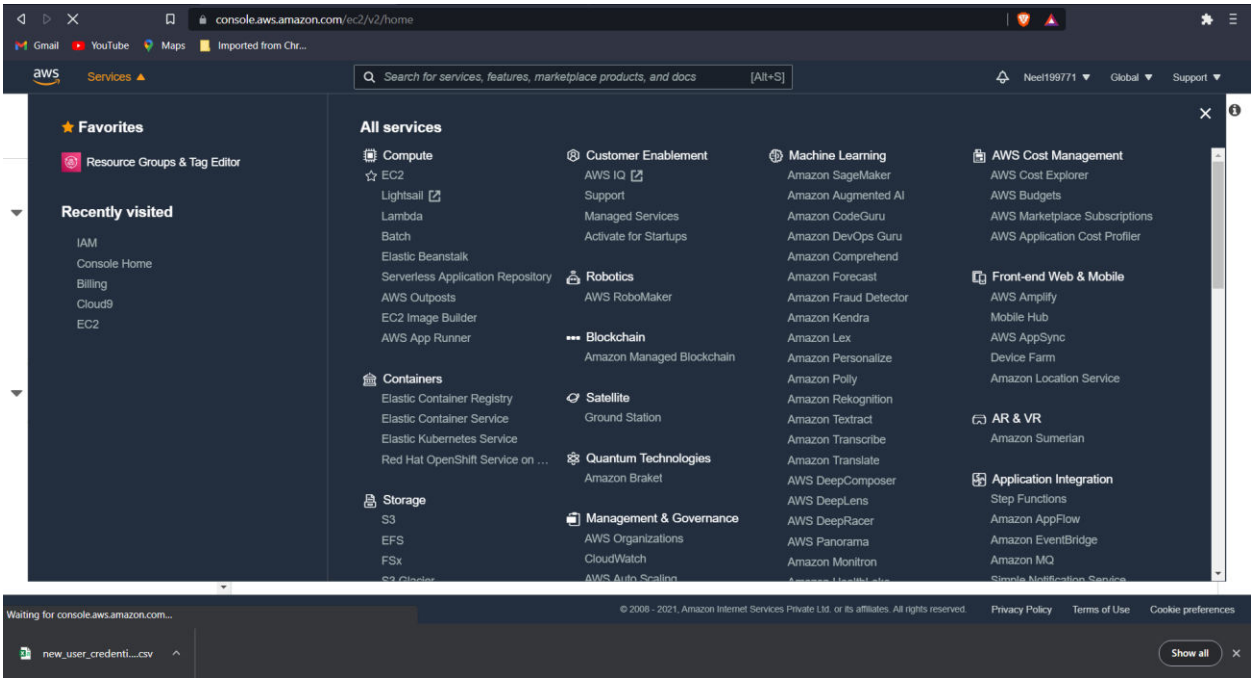
5) Don't add tags



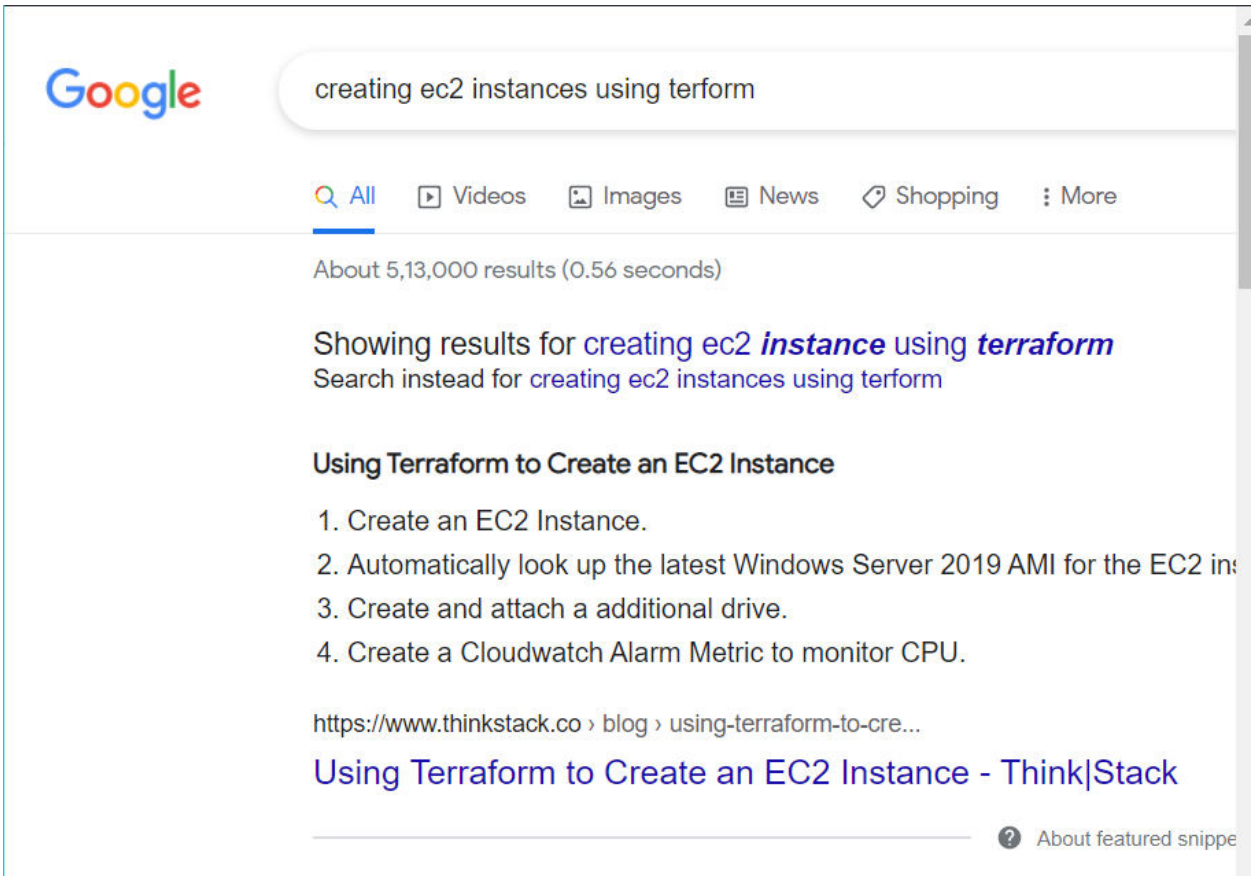
6) Now Download .csv file



7) Go to services and Ec2



8) Again google search the following terms

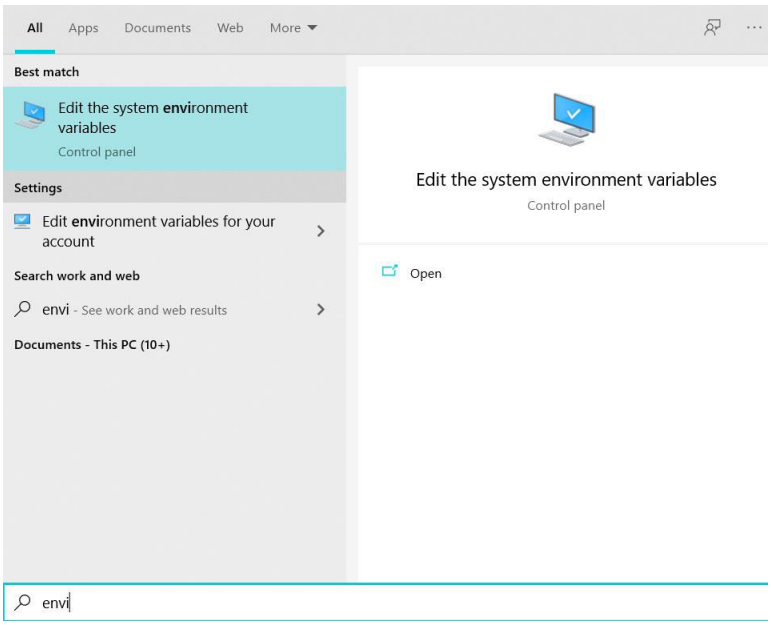


Created a folder Name Terraform Scripts in the C drive where the AdvDevops folder was created

9) Now Go to note pad And Type the below Details properly But before It Just Change the ACCESS KEY AND SECRECT KEY TO THE ONE IN YOUR .csv File . Set region same as below if you want MUMBAI as your region.

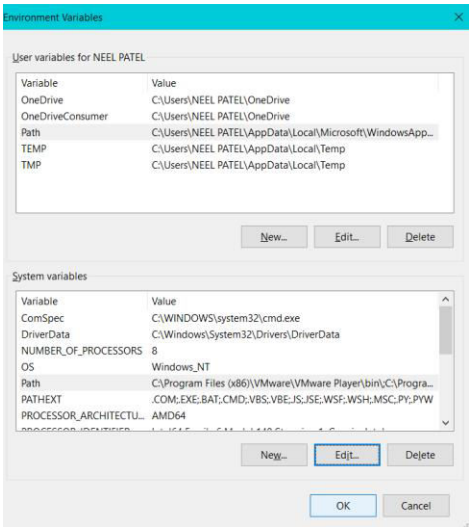
10) Now search EDIT THE SYSTEM ENVIRONMENT VARIABLES in your windows search.

Open it



11) Now click on PATH OF USER VARIBLES, then click on Edit option Now go to edit and then add new path C:\AdvDevOps

Repeat same procedure for system variables.



12) Now Open Command Prompt and then pate the path of Terraform script

Eg. CD C:\Terraform Script as shown below

Now type Terraform Init command

```
C:\Teraform Script>terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v3.52.0...
- Installed hashicorp/aws v3.52.0 (signed by HashiCorp)

Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

Then if there are no errors type Terraform Plan as shown below (type YES when command prompt ask)

```
C:\Teraform Script>terraform plan

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
  + create

Terraform will perform the following actions:

  # aws_instance.Ubuntu will be created
  + resource "aws_instance" "Ubuntu" {
      + ami                                  = "ami-0c1a7f89451184c8b"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
```

Now Finally Type Terraform Apply

```
C:\Teraform Script>terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with t
following symbols:
  + create

Terraform will perform the following actions:

  # aws_instance.Ubuntu will be created
  + resource "aws_instance" "Ubuntu" {
      + ami                                  = "ami-0c1a7f89451184c8b"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
```

```
Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.Ubuntu: Creating...
aws_instance.Ubuntu: Still creating... [10s elapsed]
aws_instance.Ubuntu: Still creating... [20s elapsed]
aws_instance.Ubuntu: Still creating... [30s elapsed]
aws_instance.Ubuntu: Creation complete after 31s [id=i-0eac948a456860494]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```

13) Now go to EC2 and check that is an instance created by the name of UBUNTU and is it in running status or not If it is in Running Status then Come back to Command prompt And Terminate the Instance by

Typing - Terraform destroy

```
Plan: 0 to add, 0 to change, 1 to destroy.

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

aws_instance.Ubuntu: Destroying... [id=i-0eac948a456860494]
aws_instance.Ubuntu: Still destroying... [id=i-0eac948a456860494, 10s elapsed]
aws_instance.Ubuntu: Still destroying... [id=i-0eac948a456860494, 20s elapsed]
aws_instance.Ubuntu: Destruction complete after 30s

Destroy complete! Resources: 1 destroyed.

C:\Teraform Script>
```

14) Now go back to EC2 if the instance is terminated, if yes then logout of the Aws Console.

And close the command prompt!

Conclusion :-

Terraform is a powerful Infrastructure as Code (IaC) tool that automates the provisioning, management, and destruction of AWS infrastructure. It can help you to save time, reduce errors, and improve the consistency of your infrastructure.

Name :-Niranjan Rajesh Joshi

Roll No:- 2105051

Batch:- T13

Date Of Performance :- 29/08/2023

Experiment 7

Aim :-
to perform static analysis on python programs using sonarqube SAST process

Theory :-
SonarQube is a universal tool for static code analysis that has become more or less the industry standard. Keeping code clean, simple, and easy to read is also a lot easier with SonarQube.

What is SonarQube?
SonarQube is an open-source platform developed by SonarSource for continuous inspection of code quality. Sonar does static code analysis, which provides a detailed report of bugs, code smells, vulnerabilities, code duplications.It supports 25+ major programming languages through built-in rulesets and can also be extended with various plugins.

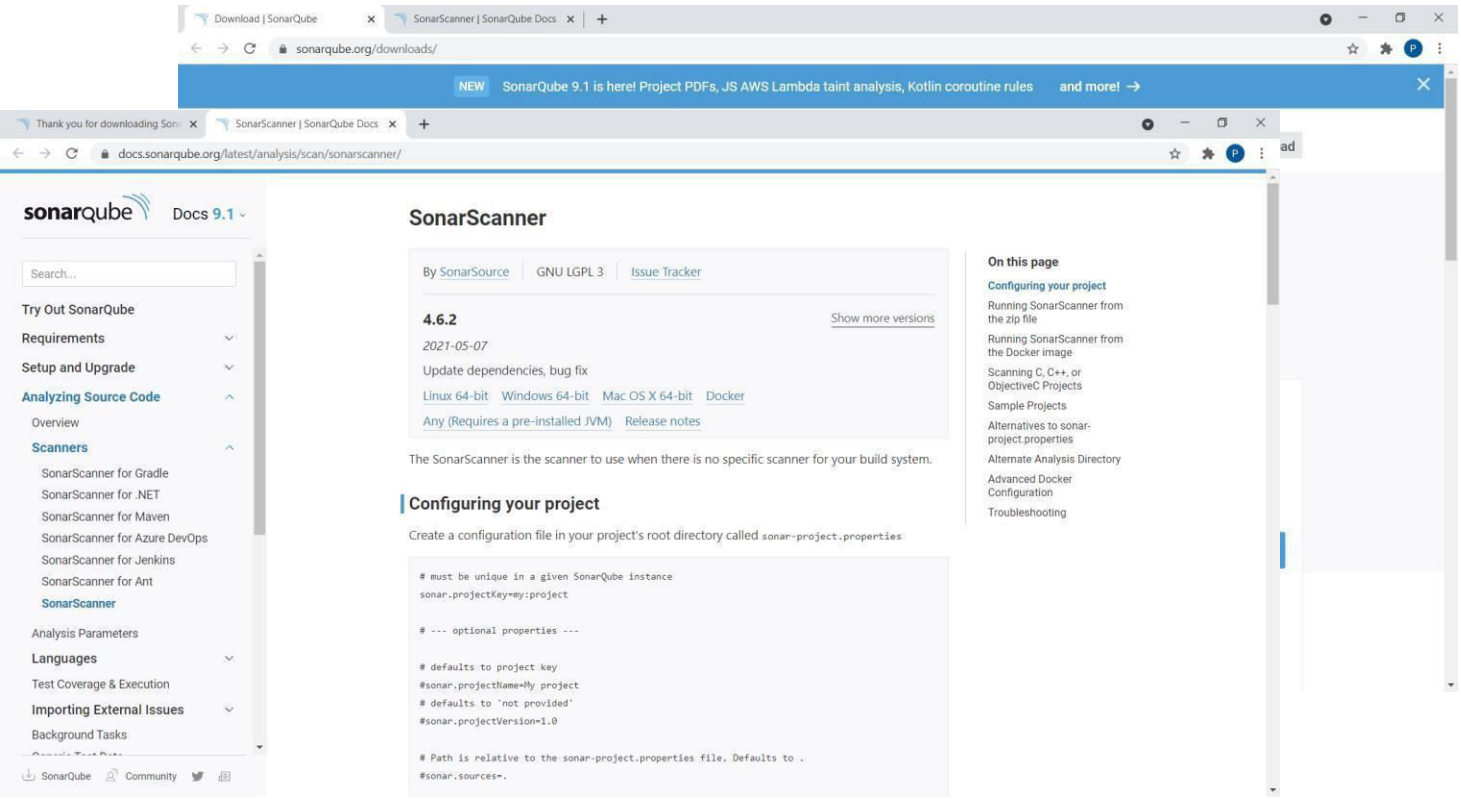
Benefits of SonarQube
Sustainability - Reduces complexity, possible vulnerabilities, and code duplications, optimising the life of applications.Increase productivity - Reduces the scale, cost of maintenance, and risk of the application; as such, it removes the need to spend more time changing the code
Quality code - Code quality control is an inseparable part of the process of software development.
Detect Errors - Detects errors in the code and alerts developers to fix them automatically before submitting them for output.
Increase consistency - Determines where the code criteria are breached and enhances the quality
Business scaling - No restriction on the number of projects to be evaluated
Enhance developer skills - Regular feedback on quality problems helps developers to improve their coding skills

Why SonarQube?
Developers working with hard deadlines to deliver the required functionality to the customer. It is so important for developers that many times they compromise with the code quality, potential bugs, code duplications, and bad distribution of complexity. Additionally, they tend to leave unused variables, methods, etc. In this scenario, the code would work in the desired way.
To avoid these issues in code, developers should always follow the good coding practice, but sometimes it is not possible to follow the rules and maintain the good quality as there may be many reasons.
In order to achieve continuous code integration and deployment, developers need a tool that not only works once to check and tell them the problems in the code but also to track and control the code to check continuous code quality. To satisfy all these requirements, here comes SonarQube in the

picture.

Steps :-

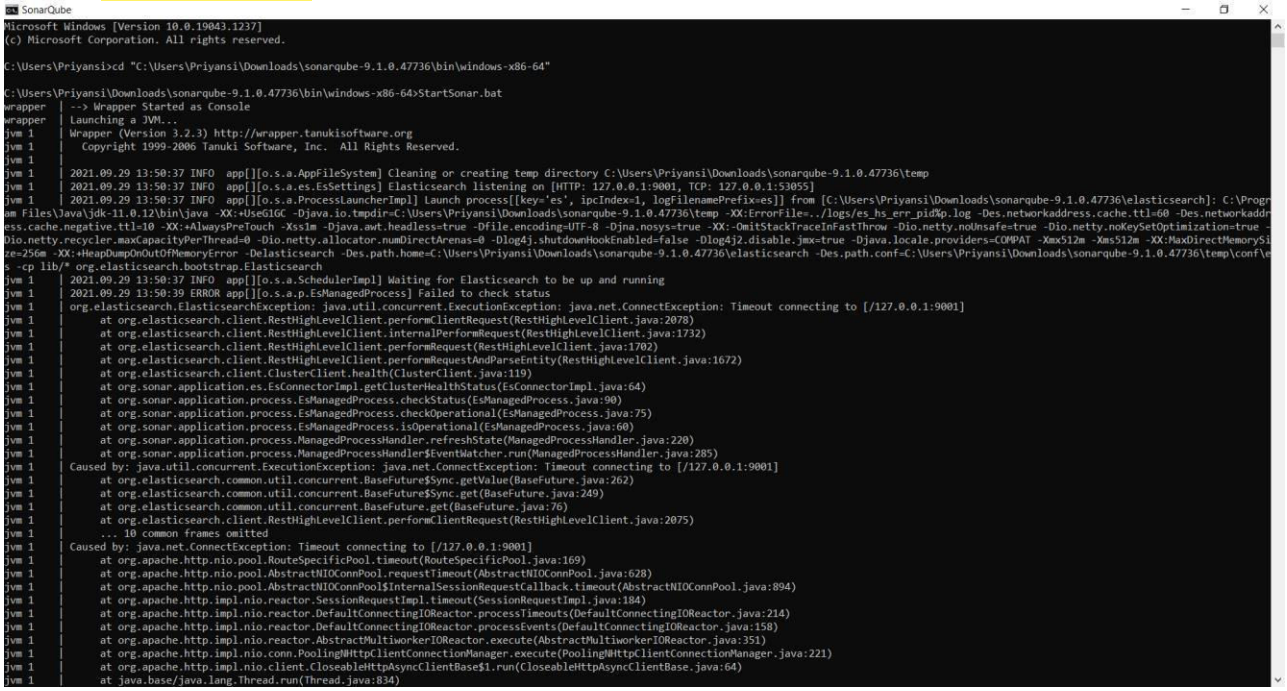
1)Download sonarqube

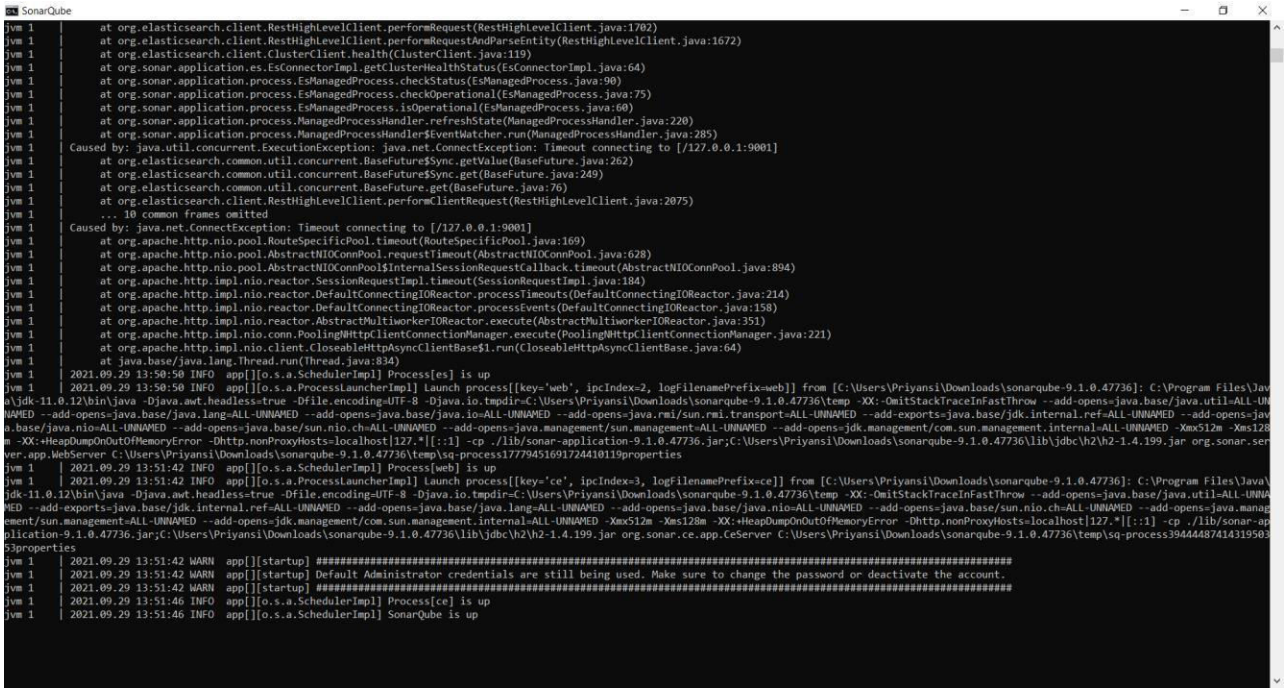


2) After downloading, set Environment Variables. Add “sonarqube-9.1.0.47736\bin” to Path.

3) Open command prompt. Run commands:

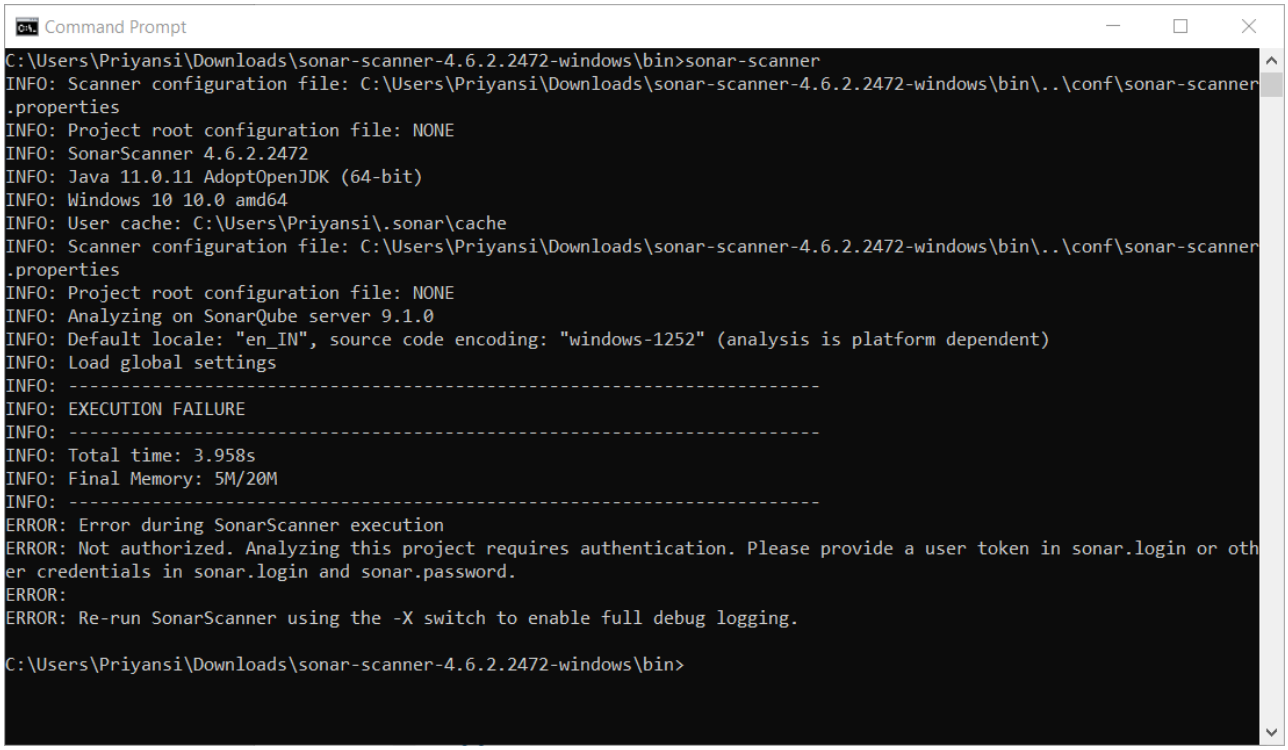
- **cd “sonarqube-9.1.0.47736\bin\windows-x86-64”**
- **StartSonar.bat**



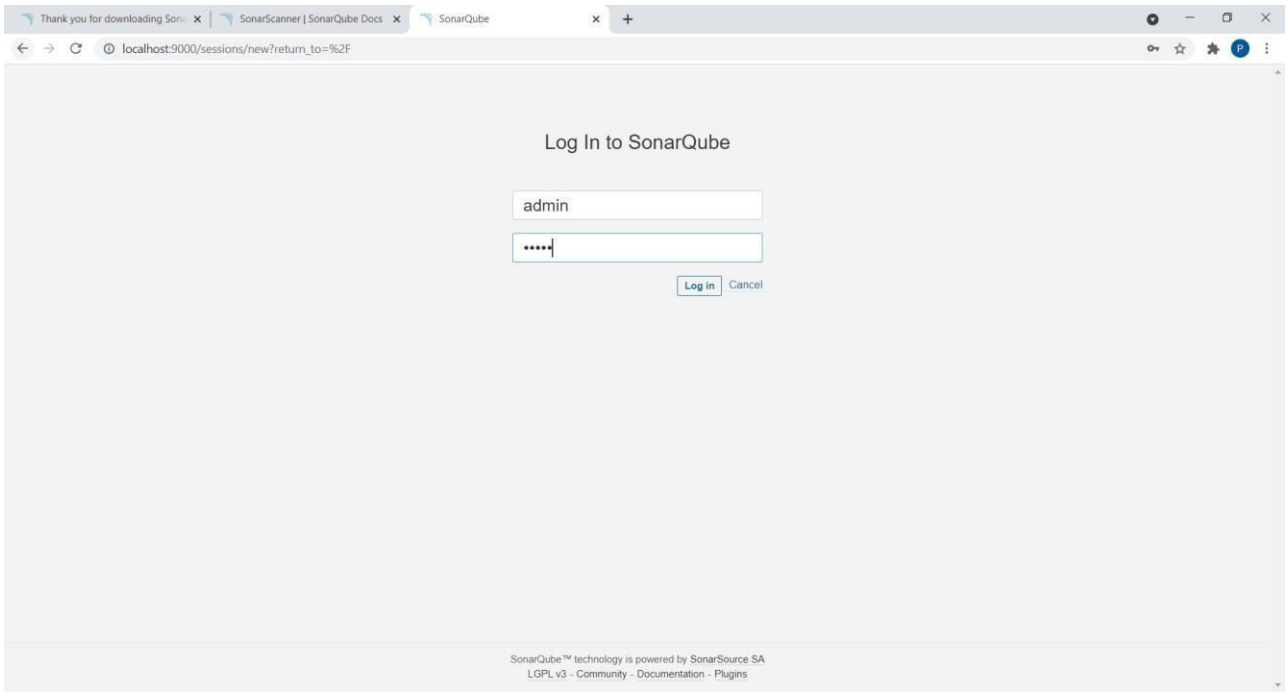


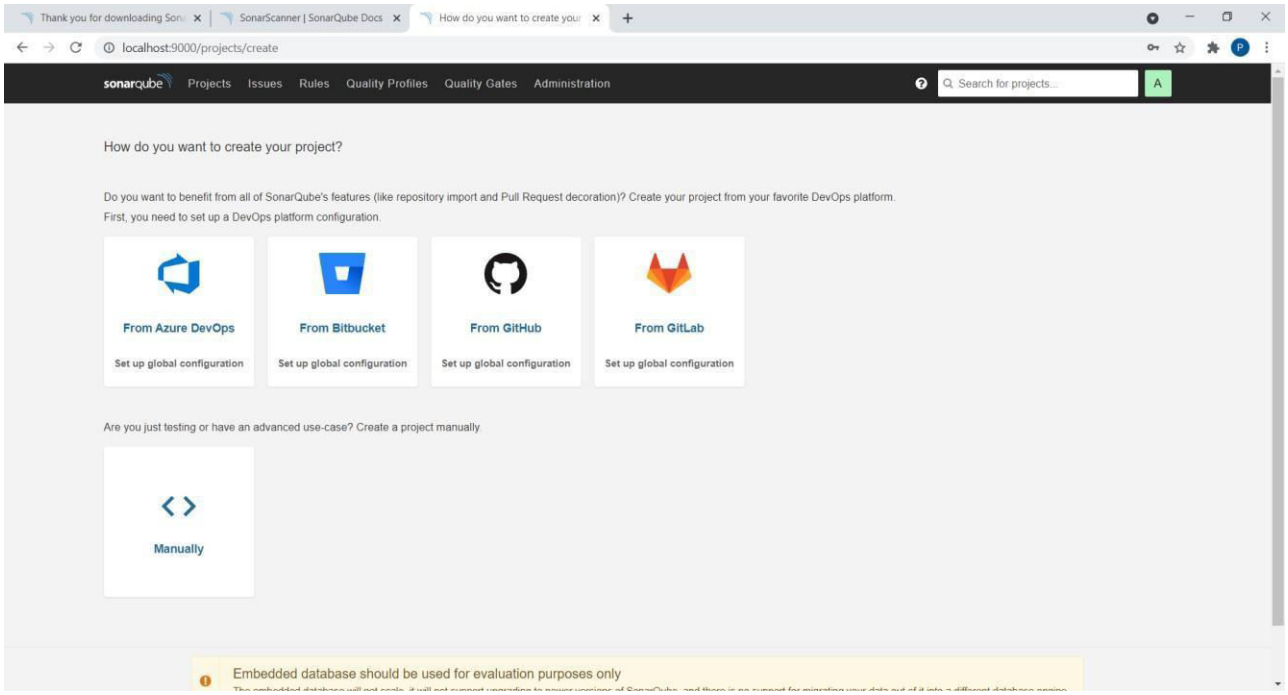
4) Open another command prompt. Run command:

- **cd “sonar-scanner-4.6.2.2472-windows\bin”**
- **sonar-scanner**

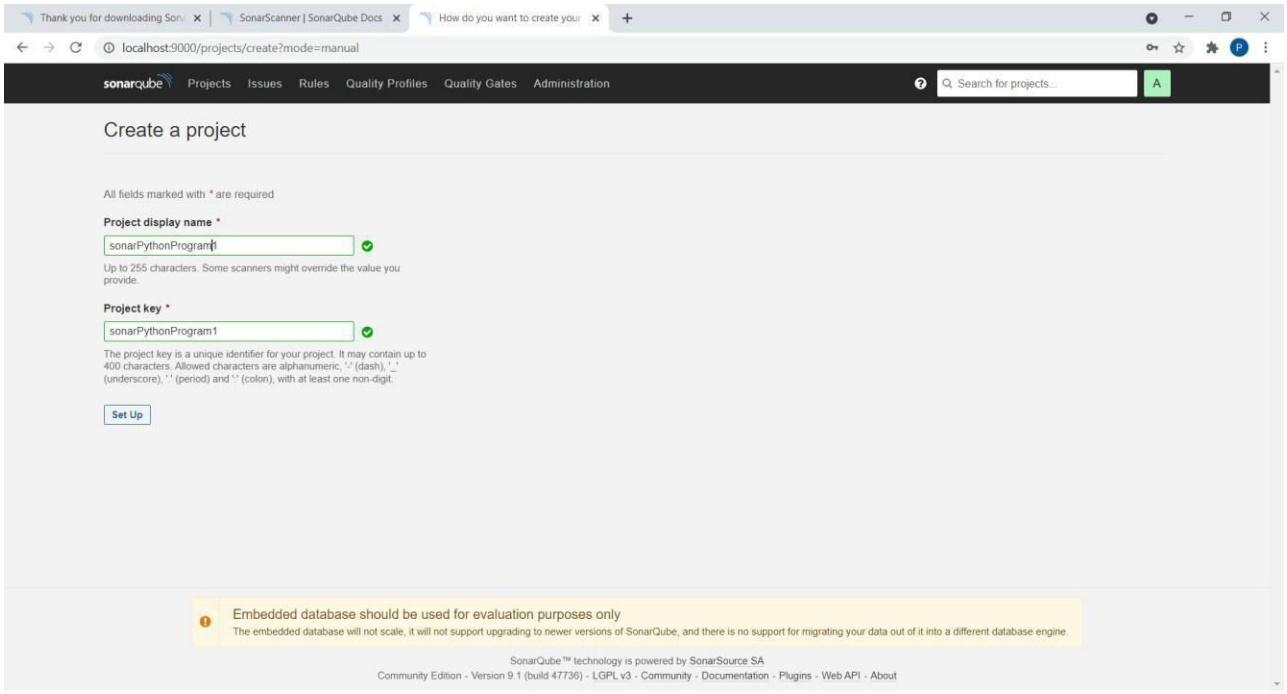


5) Server up and running on localhost:9000
Login using credentials as User: admin and Password: admin and Set a new password

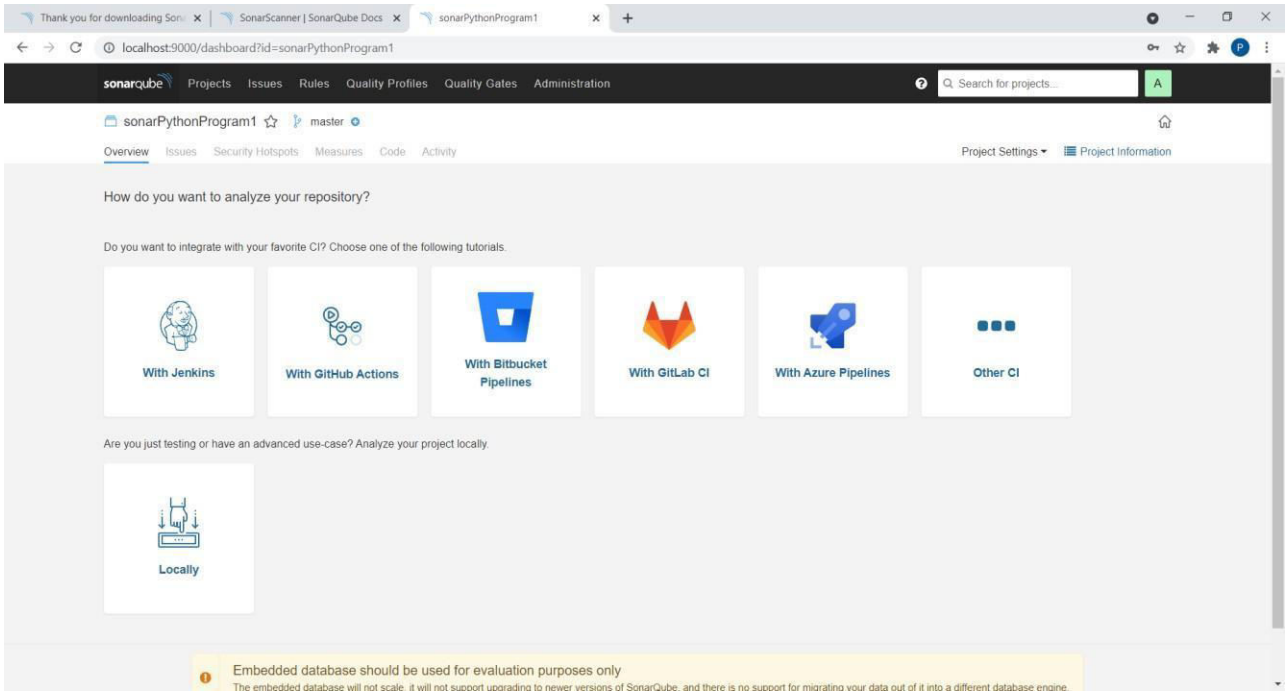




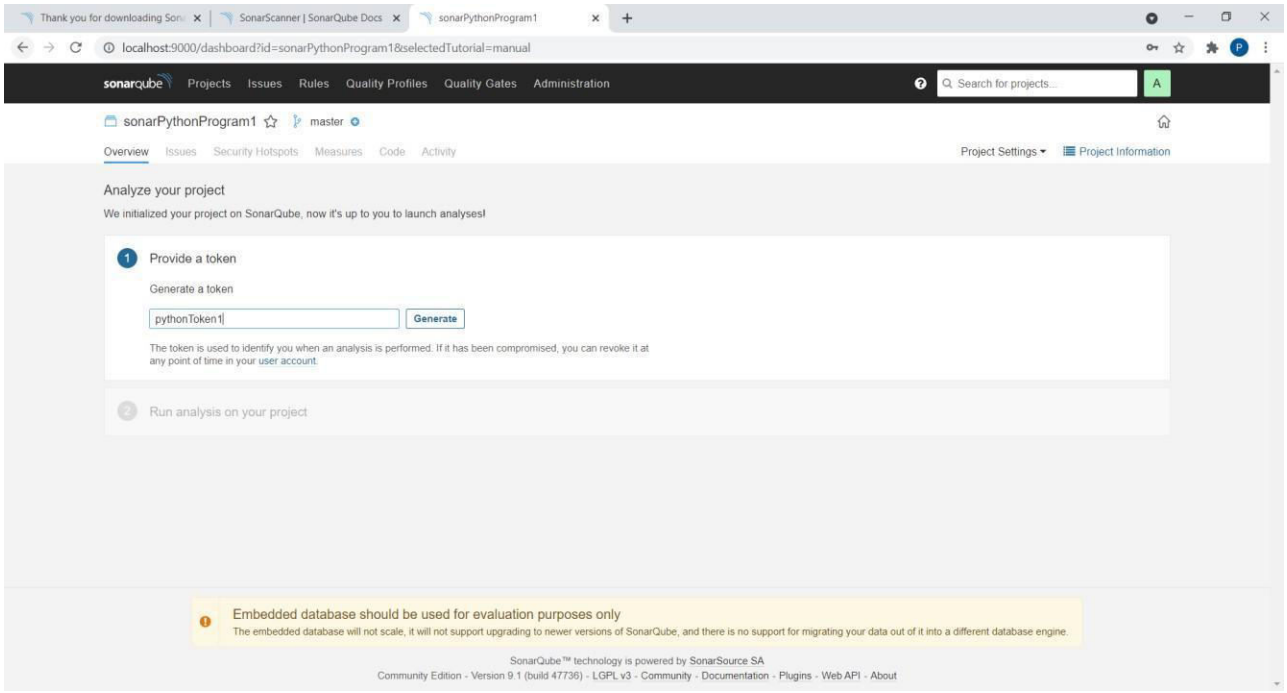
6) Click on Create a project Manually.



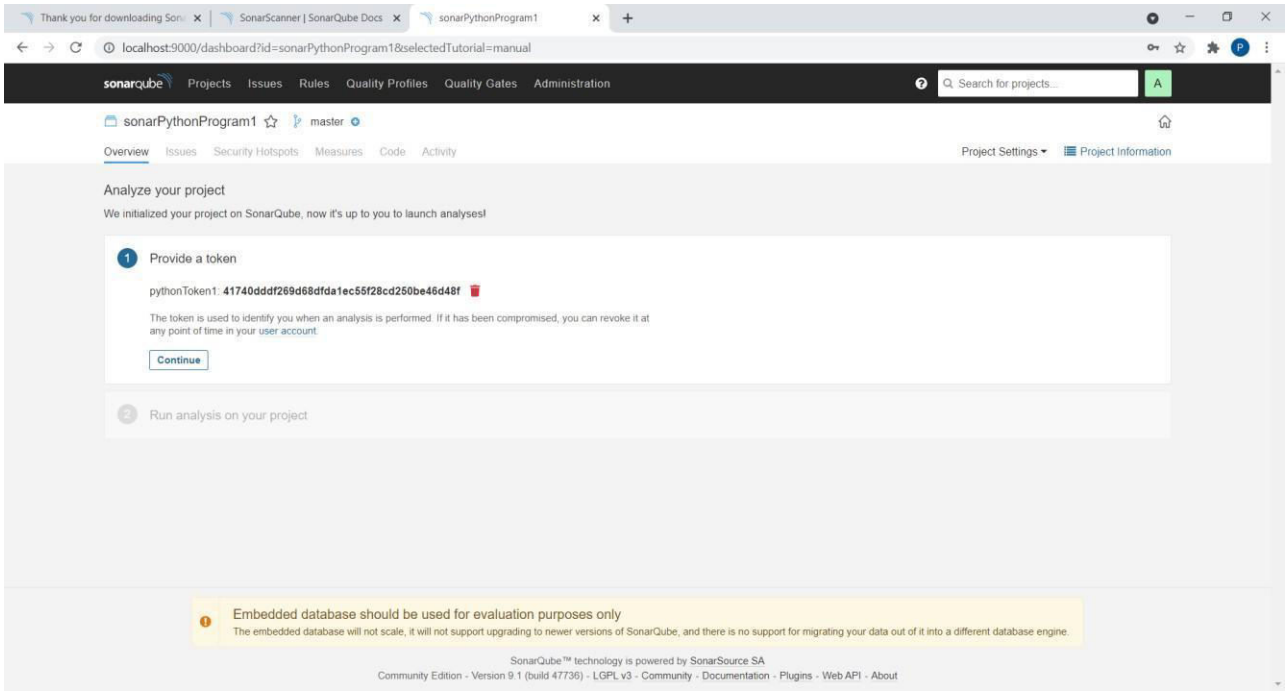
7) Give any Project display name.



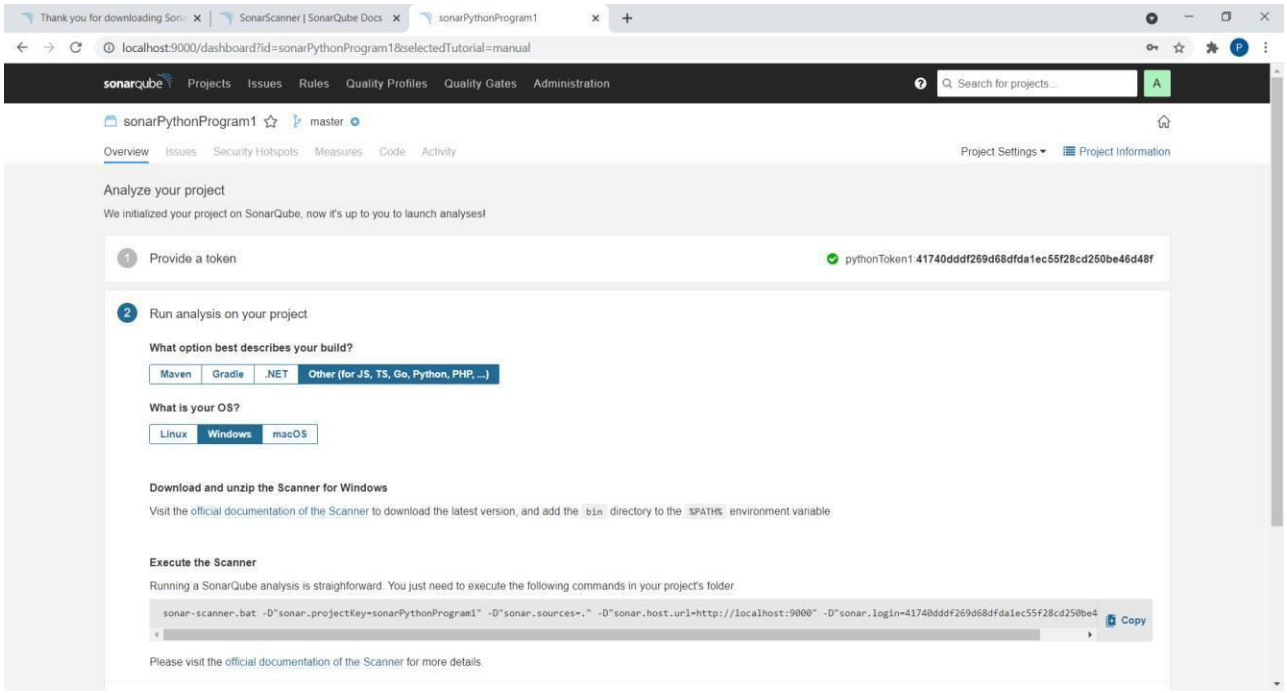
Click on Locally.



9) Give any name to token and click on Generate.



Click on Continue.

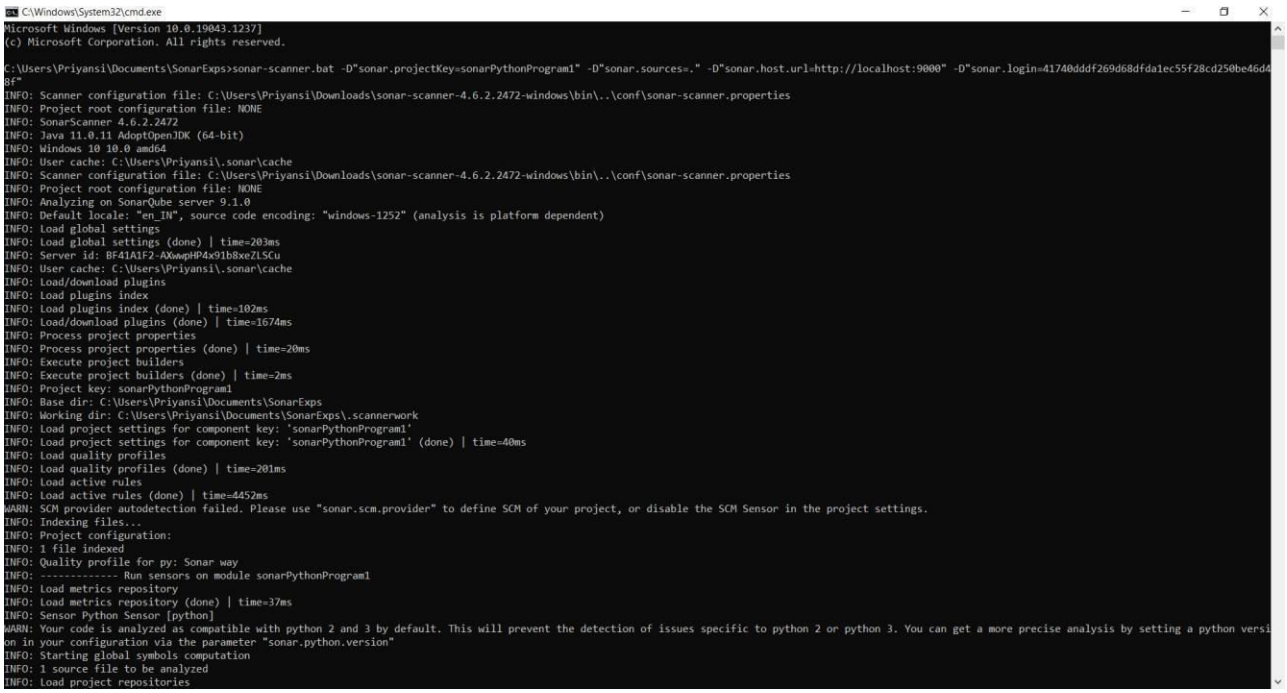


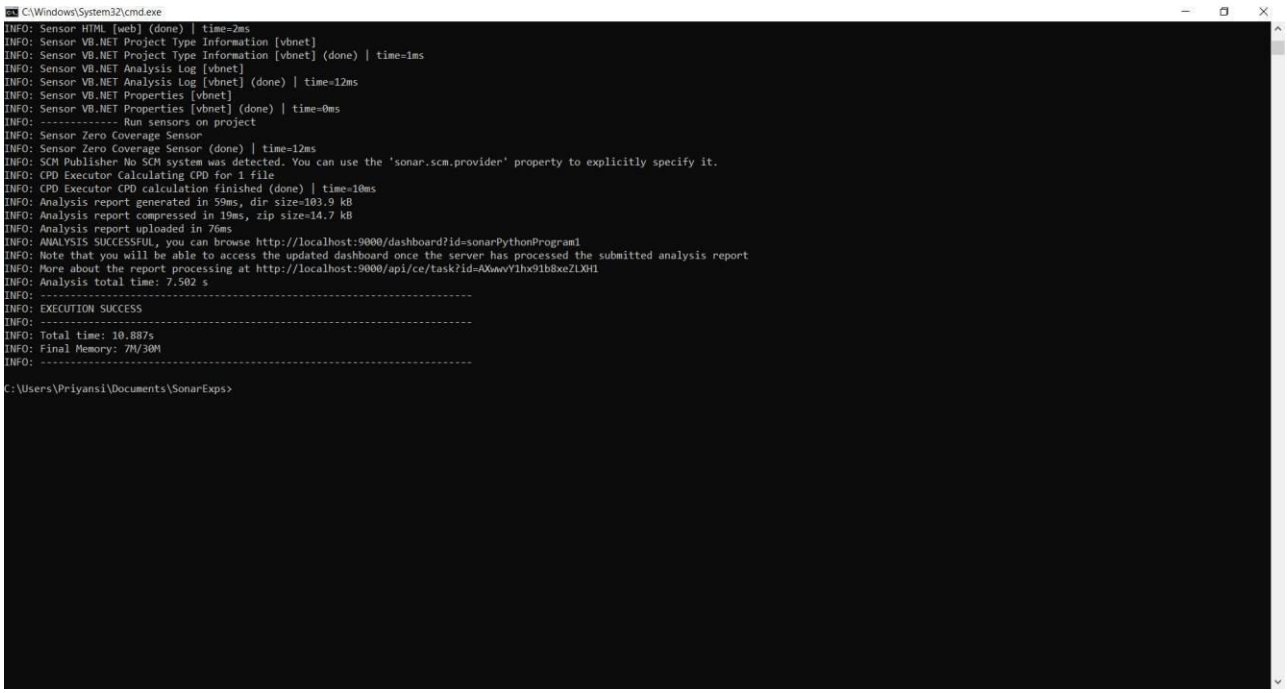
(IN SONAR SCANNER FLODER)

10) Save a Python program in a folder. class Solution(object):
def romanToInt(self, s):

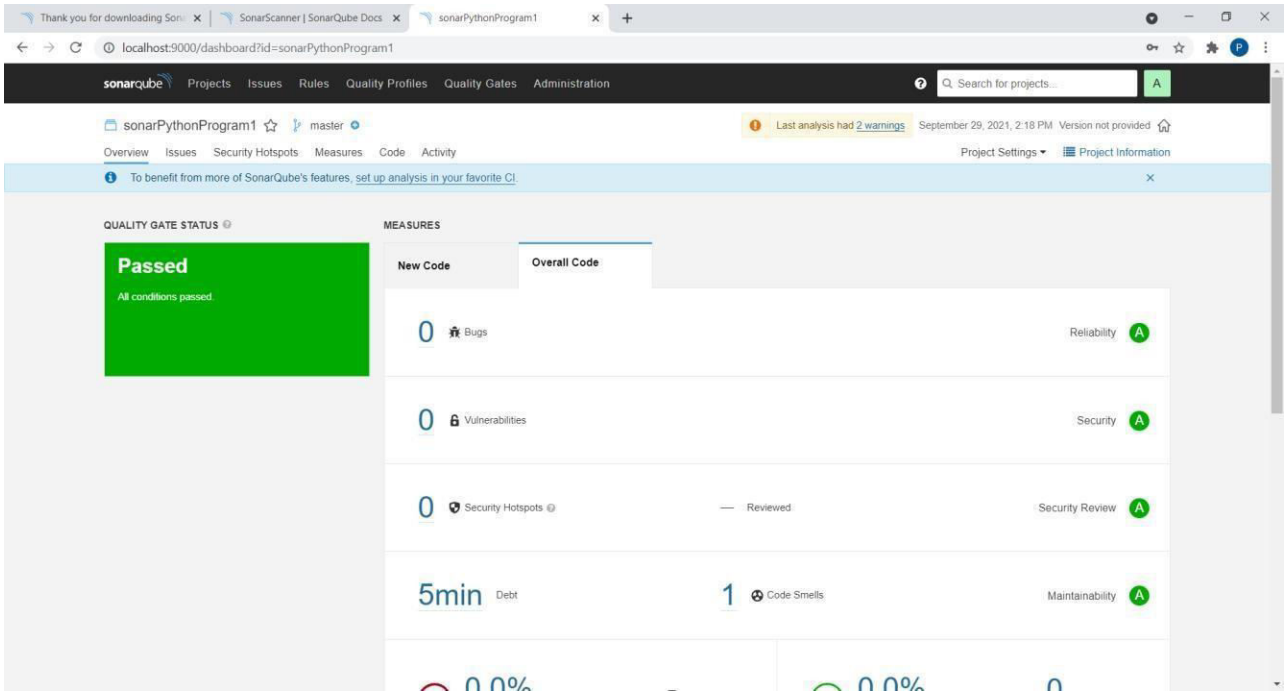
```
    roman =
{'I':1,'V':5,'X':10,'L':50,'C':100,'D':500,'M':1000,'IV':4,'IX':9,'XL':40,'XC':90,'CD':400,'CM':
900}
    i = 0
    num = ''
    ''
    while i < len(s):
      if i+1<len(s) and s[i:i+2] in roman:
        num+=roman[s[i:i+2]]
        i+=2
      else:
        #print(i)
        num+=roman[s[
        i]] i+=1
    return num
ob1 =
Solution()
print(ob1.romanToInt('III'))
print(ob1.romanToInt('CDXL
III'))
```

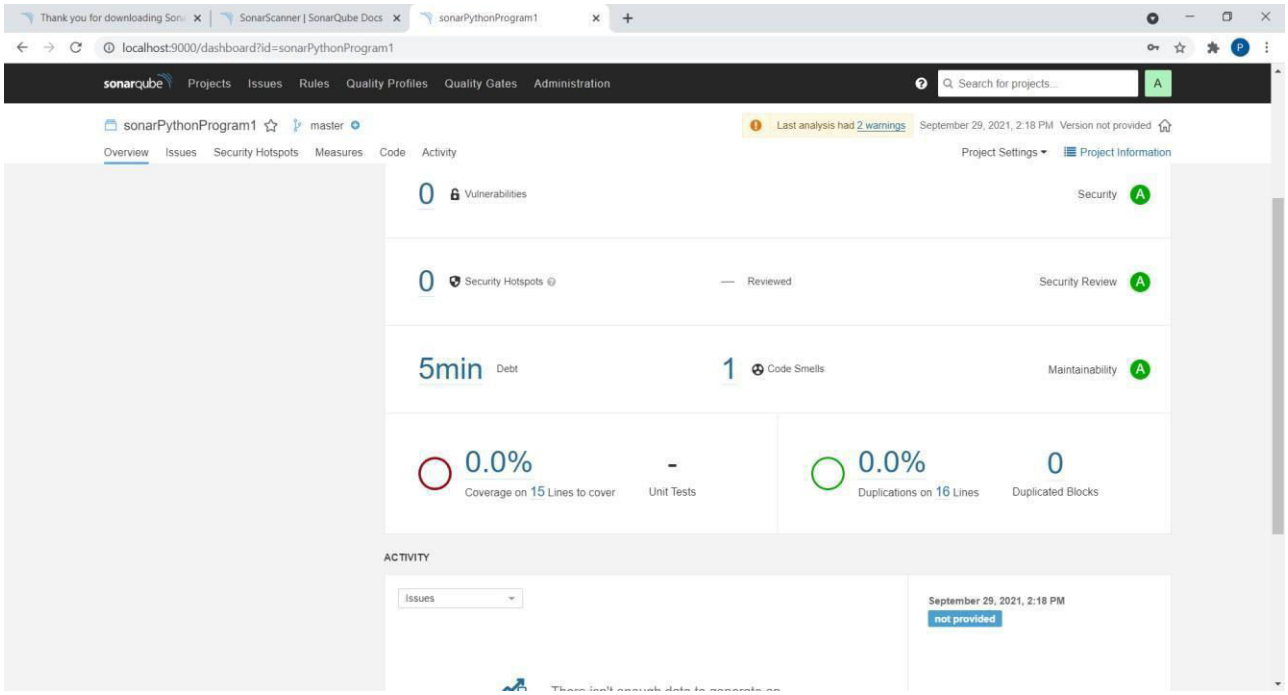
11) Open command prompt in this folder and Run program using copied command. "sonar-scanner.bat -D"sonar.projectKey=sonarPythonProgram1" -D"sonar.sources=." -D"sonar.host.url=http://localhost:9000" -D"sonar.login=41740dddf269d68dfda1ec55f28cd250be46d48f"

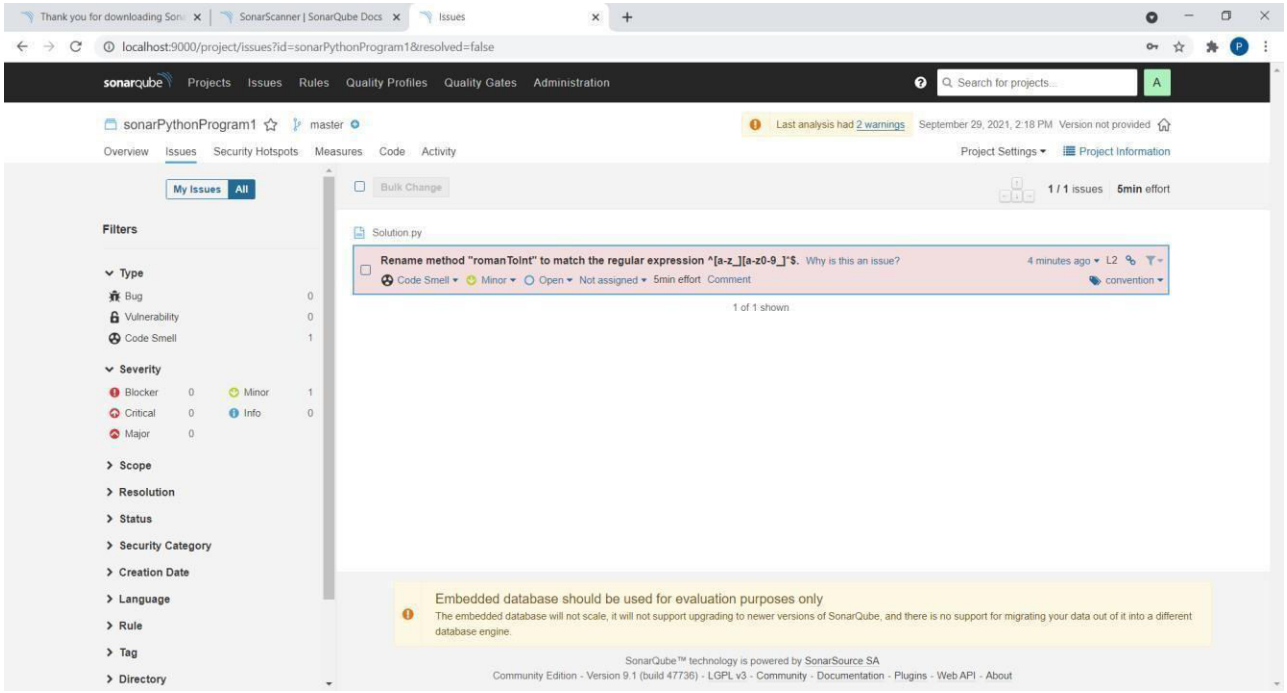




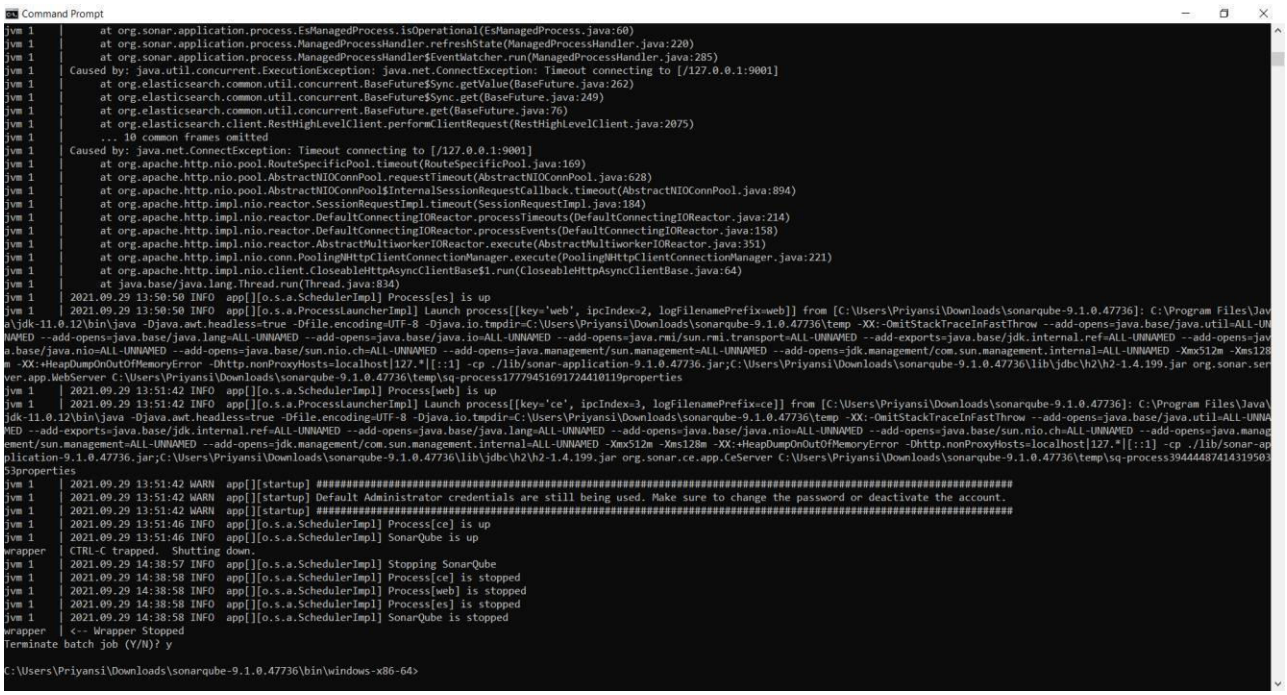
13) Given below is the inspection of code quality to perform automatic reviews with static analysis of code to detect bugs, code smells, and security vulnerabilities.







Press “Ctrl + C” to stop the server.



Conclusion :-

The SonarQube SAST process is a powerful tool for performing static analysis on Python programs. It can help you to identify and fix security vulnerabilities and code quality issues, improving the overall security and quality of your software.

Name :-Niranjan Rajesh Joshi
Roll No:- 2105051
Batch:- T13
Date Of Performance :- 12/09/2023

Experiment 8

Aim:-

To understand continuous monitoring using Nagios

Theory:-

Nagios is an open-source monitoring system that provides monitoring of services, applications, and network resources. It is designed to alert system administrators about potential issues before they become critical problems. Nagios allows you to monitor the entire IT infrastructure, including servers, switches, applications, and services. It provides a comprehensive monitoring solution for both small and large organizations. Some key features and capabilities of Nagios include:

Monitoring Capabilities: Nagios can monitor a wide variety of network services including SMTP, POP3, HTTP, NNTP, ICMP, SNMP, FTP, SSH, and many more.

Alerting and Notification: It provides alerting and notification functionalities to notify system administrators when something goes wrong. Nagios can send alerts via email, SMS, or other methods to ensure that the right people are notified in real-time.

Plugin Architecture: Nagios has a modular architecture that allows users to develop their plugins and addons to monitor specific devices and services that are not covered by default.

Customizable Dashboards and Reports: Nagios offers customizable dashboards and reporting capabilities that provide insights into the performance and health of the monitored resources.

Scalability and Flexibility: Nagios can scale to monitor complex, large-scale IT infrastructures. It is highly flexible and can be customized to meet specific monitoring and alerting requirements.

Extensibility: Nagios can be extended through various addons and plugins, allowing it to integrate with other tools and services, and enabling the monitoring of a wide range of devices and applications.

Historical Monitoring and Trend Analysis: Nagios can store historical data and provide trend analysis, allowing system administrators to identify patterns and plan for future infrastructure needs.

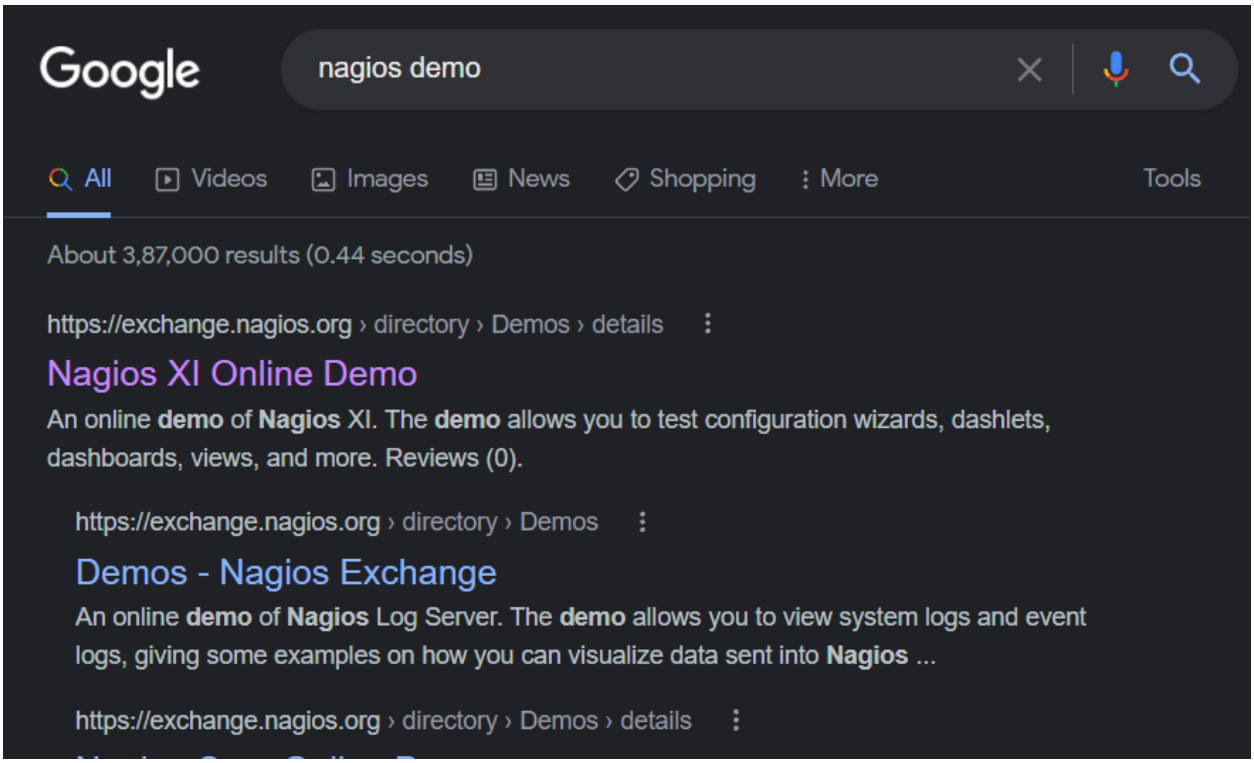
Community Support and Active Development: Being an open-source project, Nagios has a vibrant community that contributes to its development and support. This community-driven approach ensures that the software remains updated and robust.

Centralized Monitoring: Nagios provides a centralized view of the entire IT infrastructure, allowing administrators to have a comprehensive overview of the health and performance of all monitored resources from a single location.

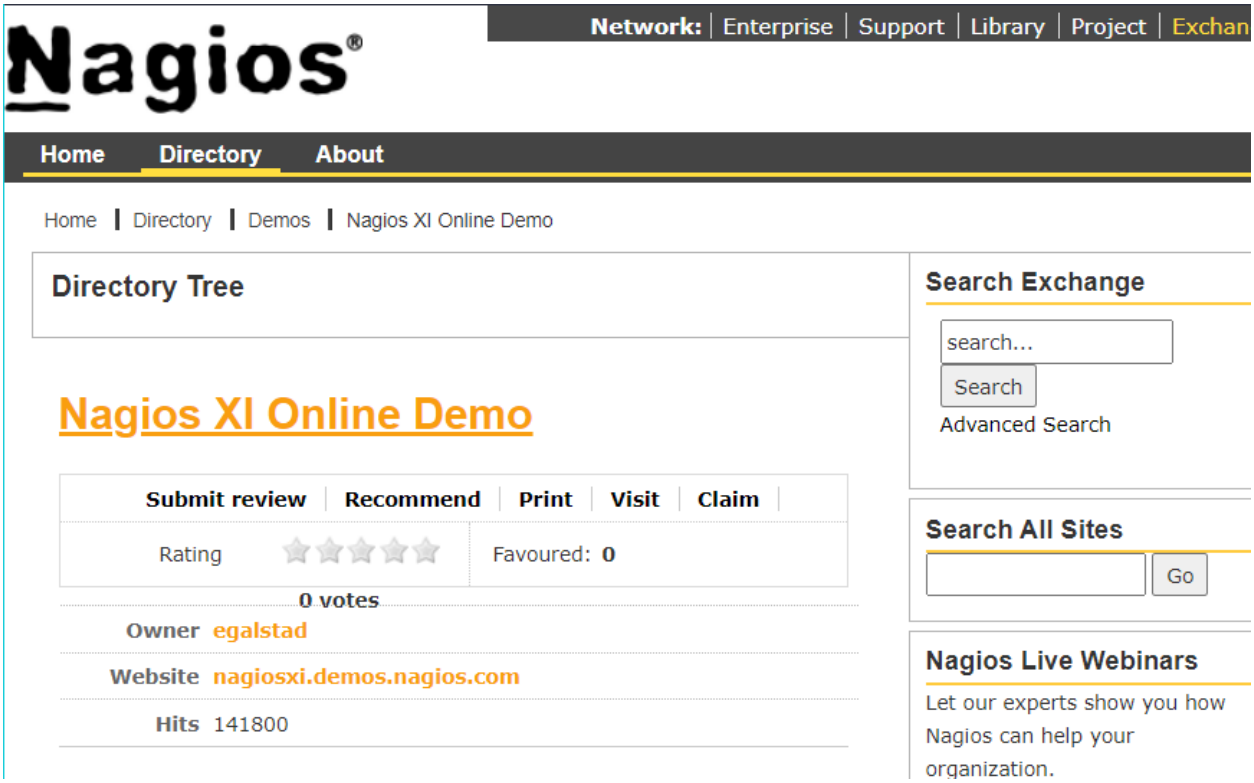
Integration with Third-Party Tools: Nagios can integrate with various third-party tools and services, making it a versatile monitoring solution that can fit into different IT ecosystems and workflows.

Steps :-

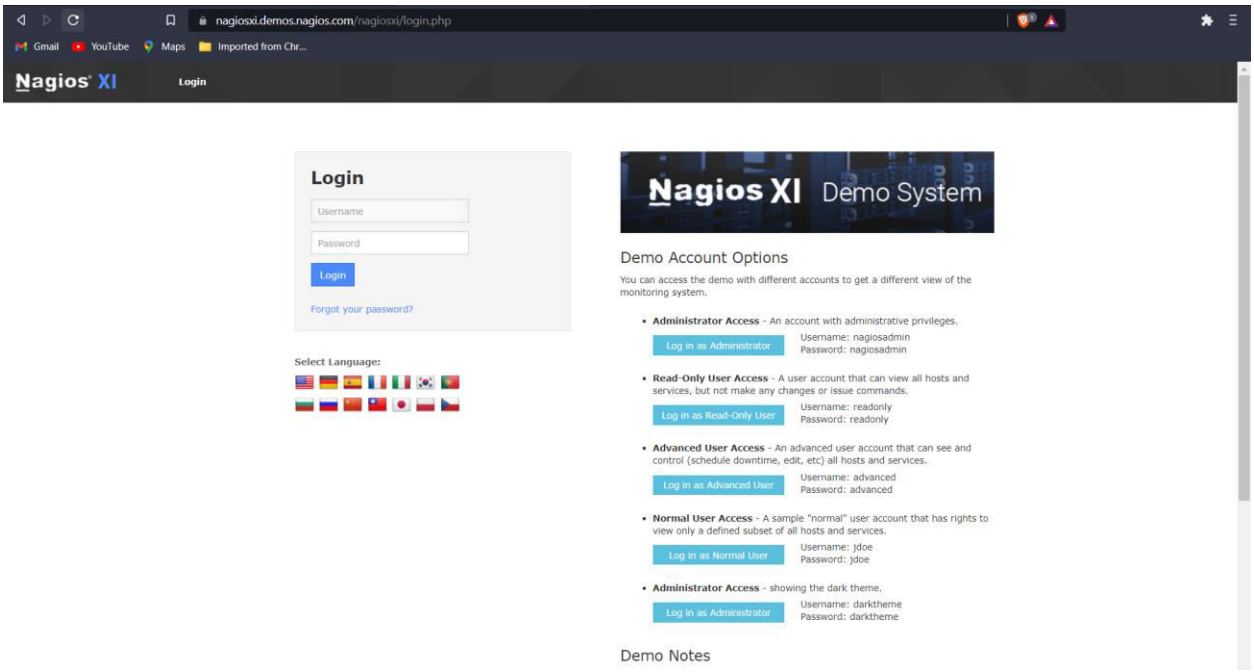
1) Go to google.com, Search Nagios Demo
Click on the first link shown below

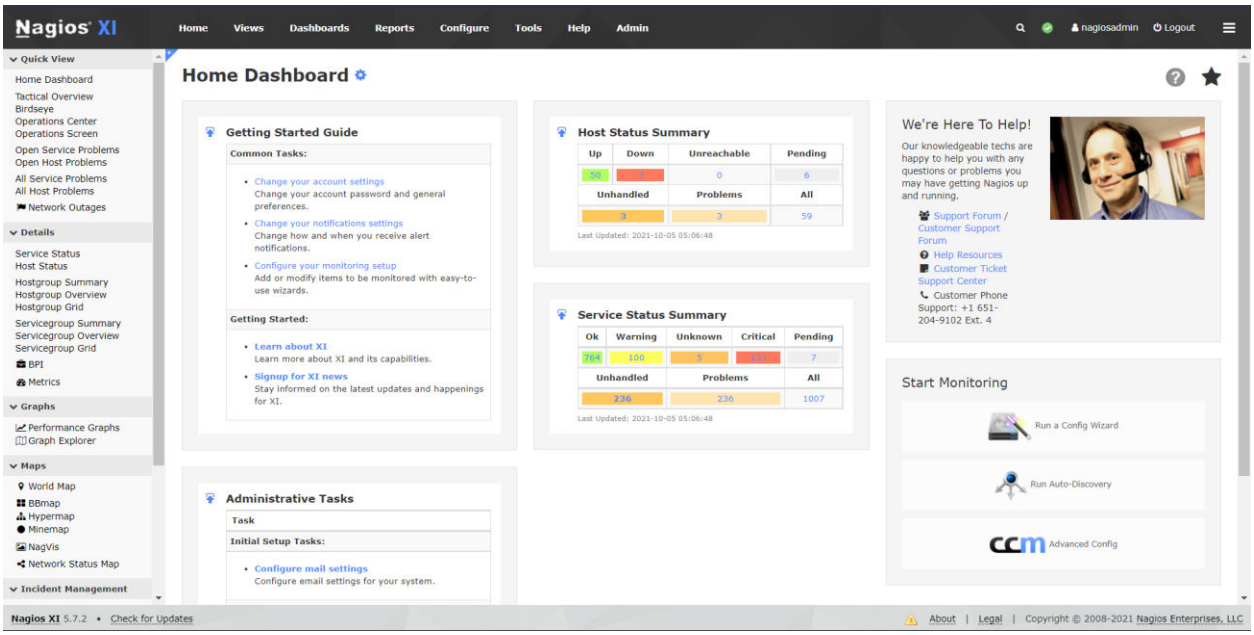


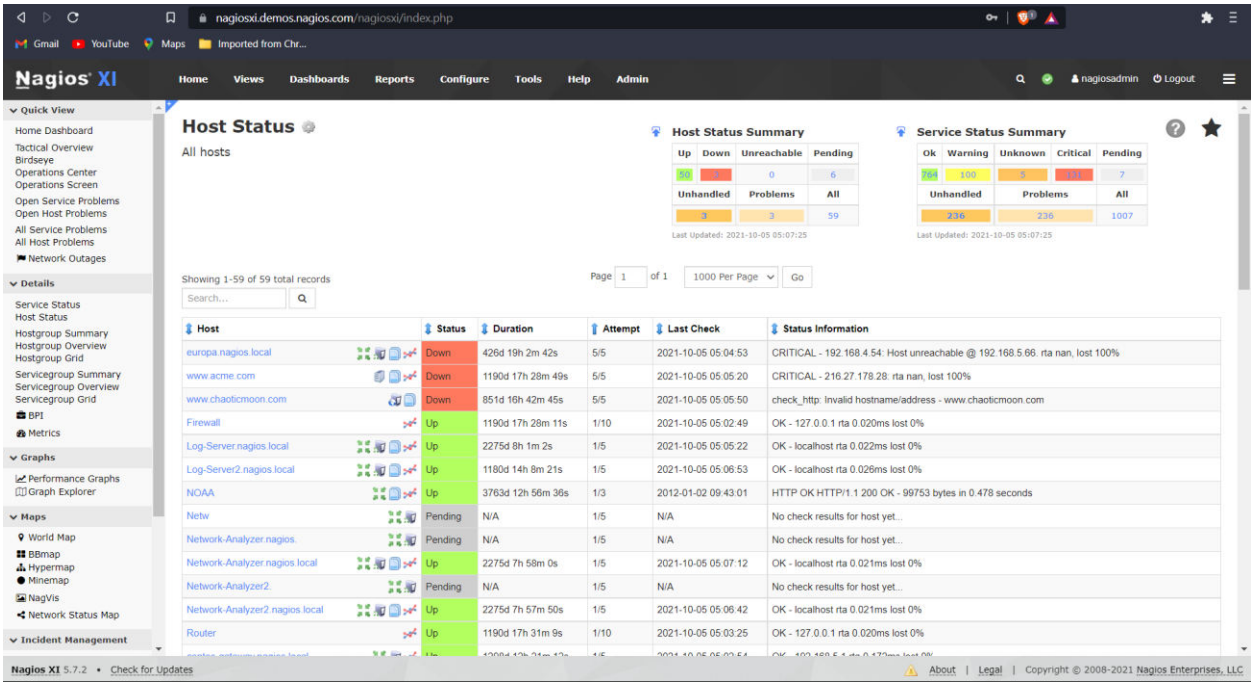
2) Now click on the website-



3) Now click on login as administrator

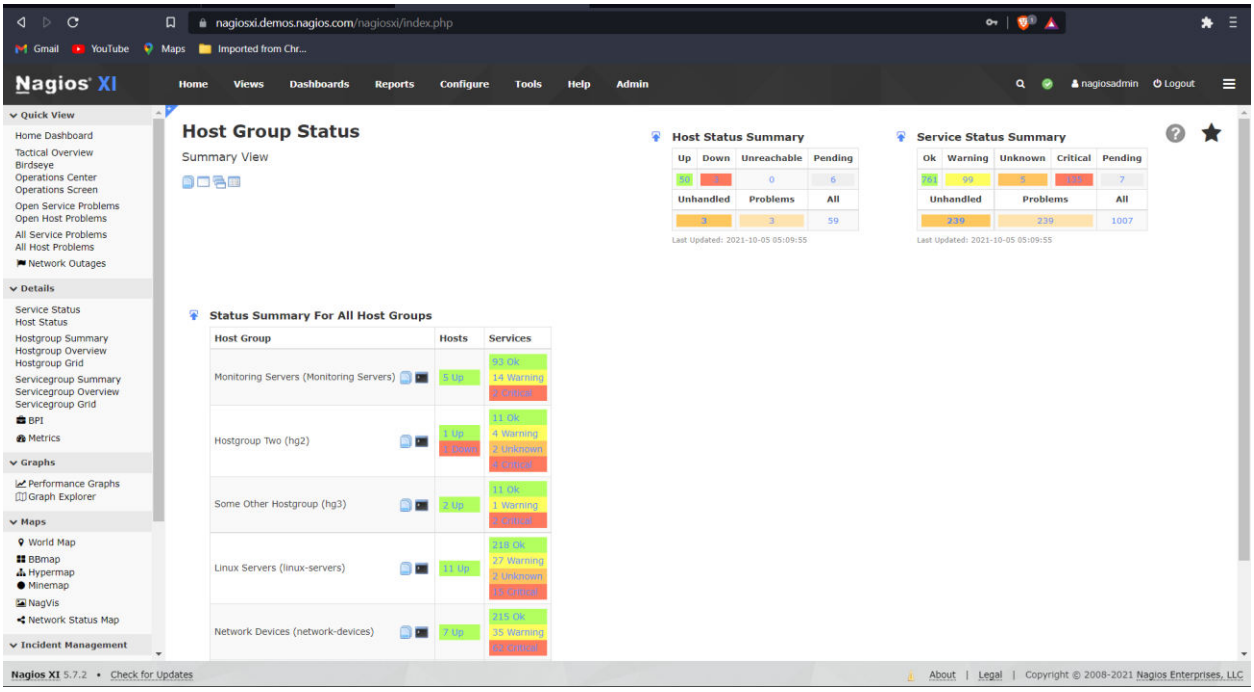






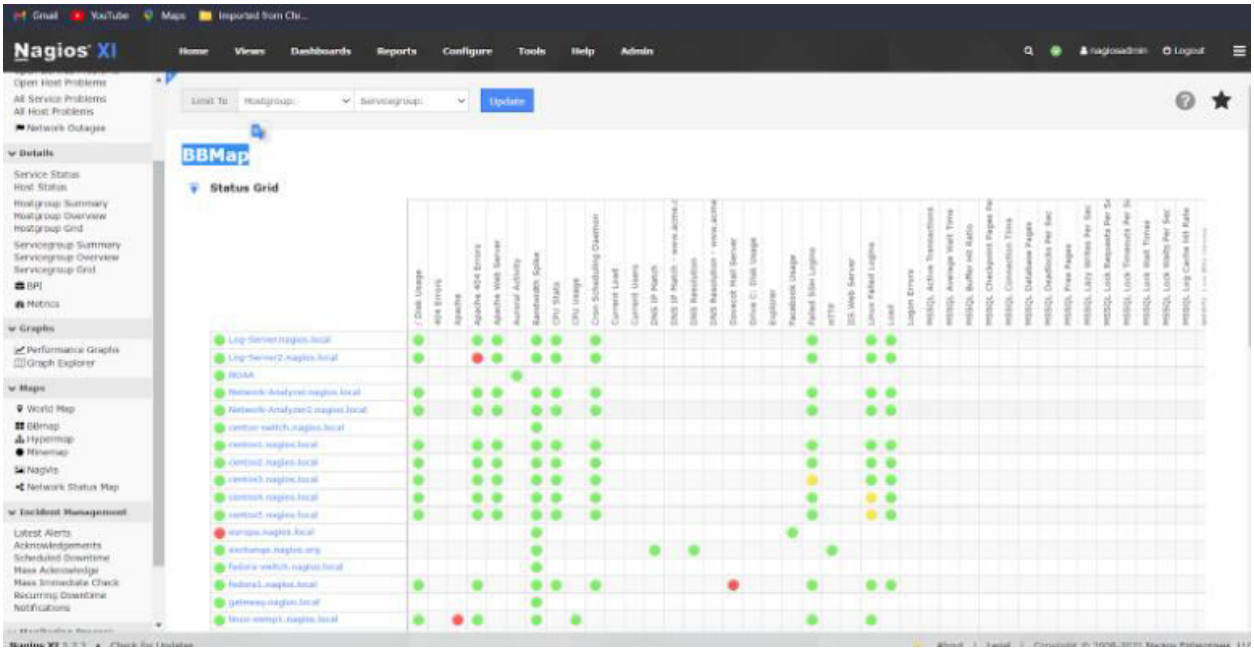
In the above image one can see Host Status Summary and Service Status Summary also how many host are up, down and also errors in detail

5) Now click on Host Group Status.



6) Now we click on BBMap

In this we can see status of following stuff in each host-



7) Now we have Network status map which is graphical representation of the network status

Conclusion:-

Continuous monitoring with Nagios enables proactive detection of system issues, ensuring minimal downtime and enhanced operational efficiency. Through customizable alerts and comprehensive reporting, Nagios empowers administrators to maintain optimal performance across diverse IT environments. Its scalable architecture and robust community support make it an invaluable tool for streamlined and centralized monitoring.

Name :-Niranjan Rajesh Joshi
Roll No:- 2105051
Batch:- T13
Date Of Performance :- 05/09/2023

Experiment 9

Aim:-

To understand AWS lambda functions and create a lambda function using python to log “an image has been added” message , once the file has been added in S3 bucket

Theory:-

AWS Lambda is a serverless computing service provided by Amazon Web Services (AWS) that allows you to run code without provisioning or managing servers. With AWS Lambda, you can upload your code and the service will automatically run and scale your code based on the incoming requests or events. Here are some key features and aspects of AWS Lambda:

Event-Driven Computing: AWS Lambda allows you to execute your code in response to events such as changes to data in an Amazon S3 bucket, updates to a DynamoDB table, HTTP requests via Amazon API Gateway, or custom events from various AWS services.

Serverless Architecture: It enables you to build applications and services without the need to manage infrastructure. You are charged only for the compute time you consume, with no charge when your code is not running.

Support for Multiple Languages: AWS Lambda supports multiple programming languages including Node.js, Python, Java, Go, Ruby, and .NET Core, allowing you to choose the language that best suits your application.

Automatic Scaling: Lambda automatically scales your application by running code in response to each trigger. It can handle a few requests per day or thousands of requests per second.

Microservices and Backend Services: Lambda is often used to build scalable and cost-effective back-end services for mobile, web, and other applications. It is also commonly used in the development of microservices-based architectures.

Integration with Other AWS Services: Lambda seamlessly integrates with other AWS services, enabling you to create powerful applications using a combination of AWS services without managing servers.

Security and Access Control: AWS Lambda provides built-in security features, allowing you to control the execution role and access permissions for your Lambda functions through AWS Identity and Access Management (IAM).

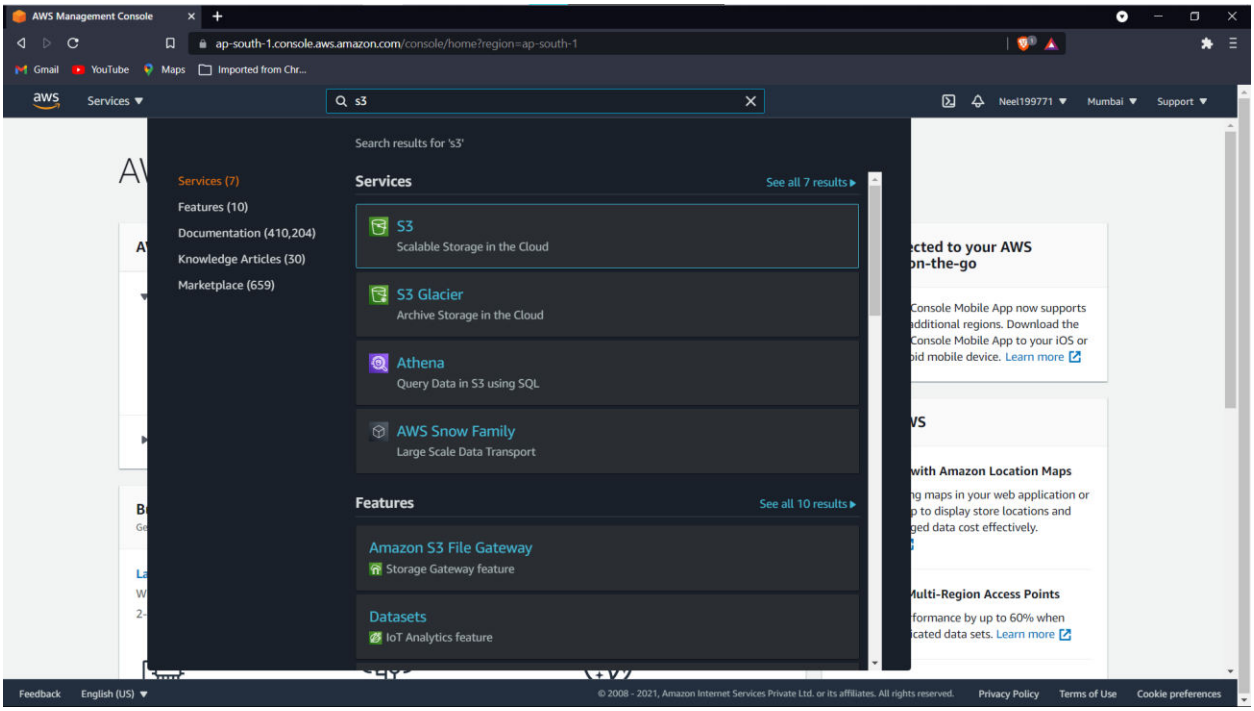
Easy Deployment and Management: Lambda makes it easy to deploy and manage your code. You can update your code without downtime, and the service handles all the operational and administrative activities, such as capacity provisioning, monitoring, and logging.

High Availability and Fault Tolerance: AWS Lambda automatically replicates your code across multiple availability zones to ensure high availability and fault tolerance.

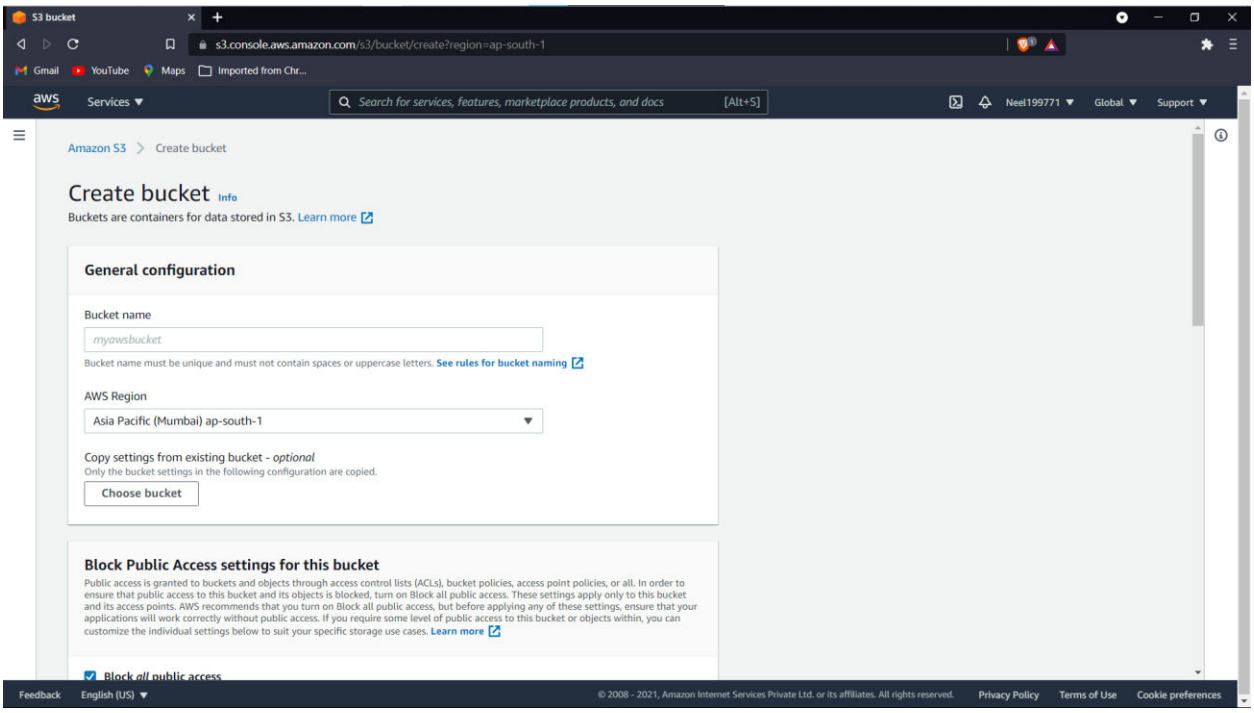
Monitoring and Logging: Lambda provides detailed monitoring and logging through Amazon CloudWatch, allowing you to monitor performance metrics, set alarms, and troubleshoot issues for your Lambda functions.

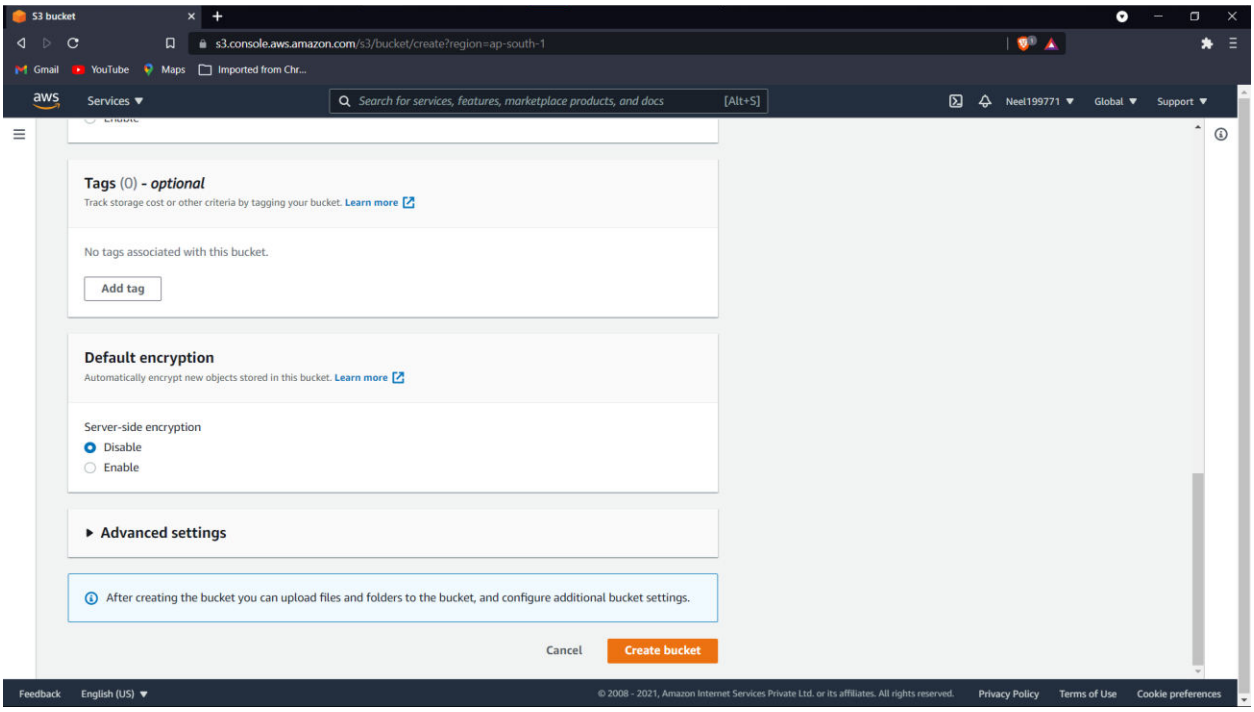
Steps :-

1) Login to AWS account then Search S3 ,click on the option below shown-

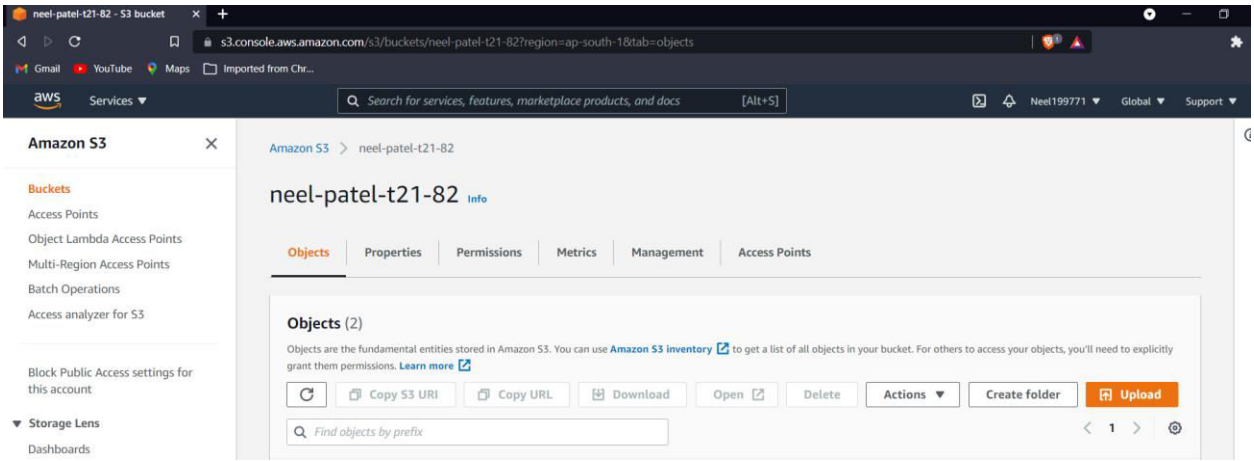


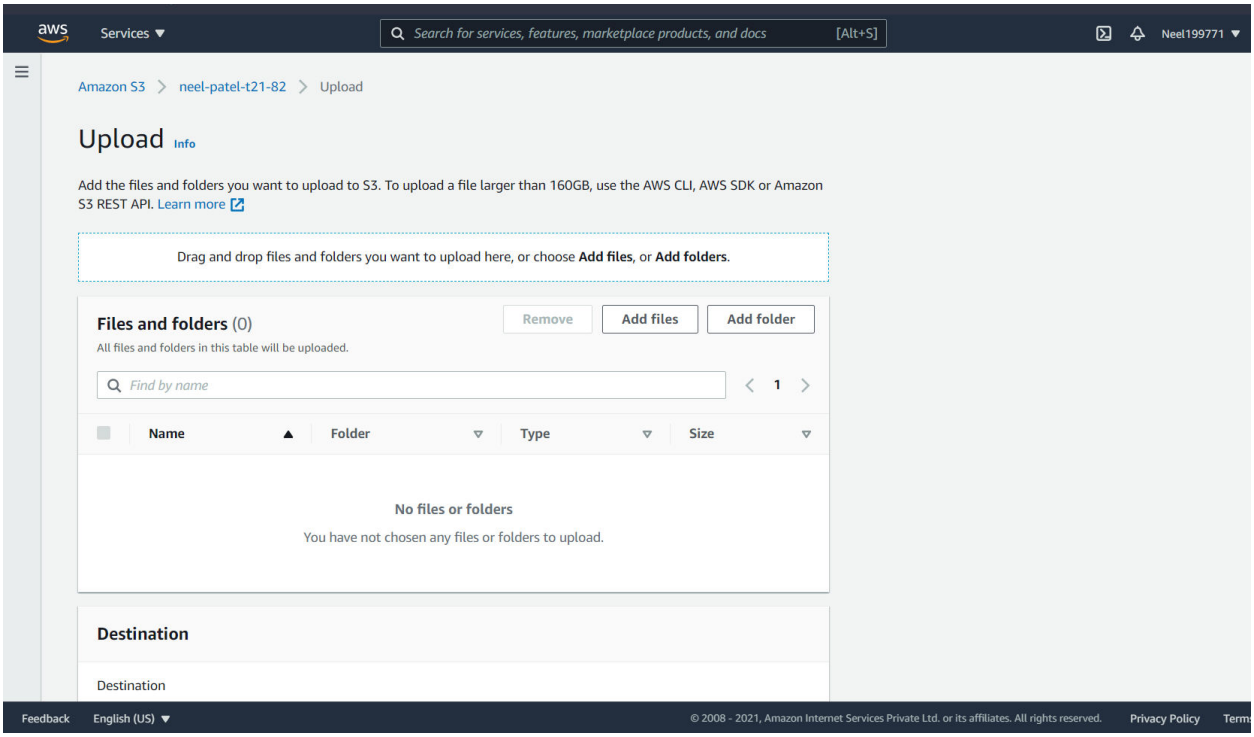
2) Create an S3 bucket by giving it a name



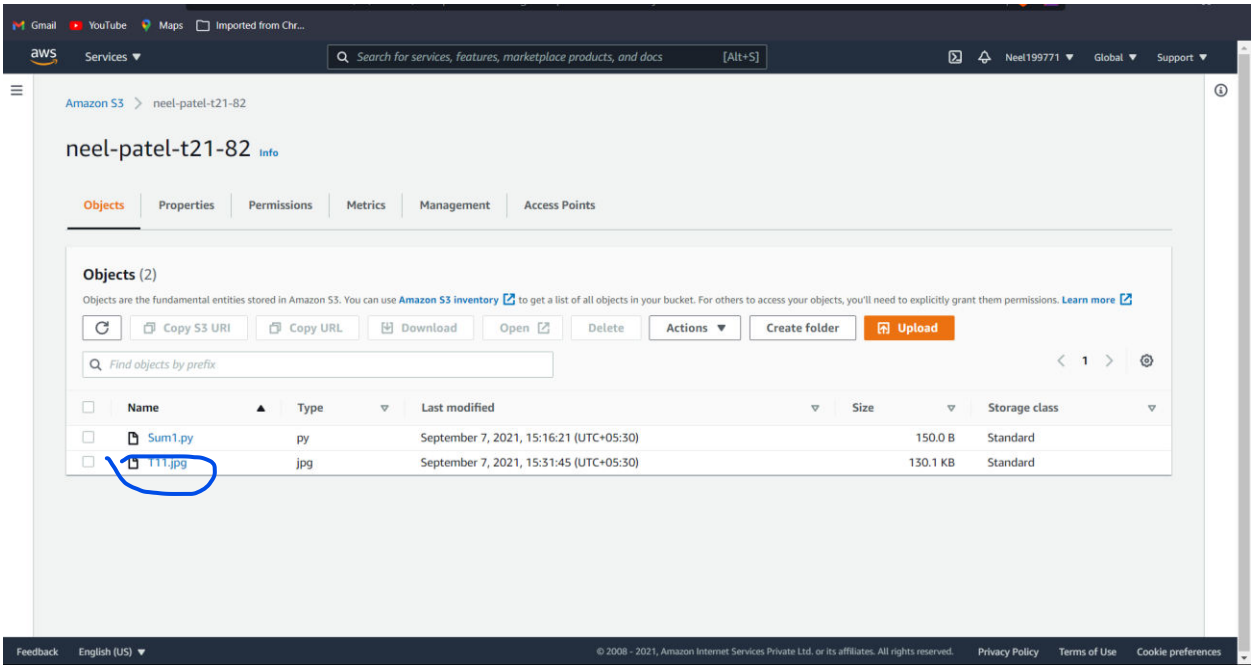


3) Click on upload button after the s3 bucket is created in the object section

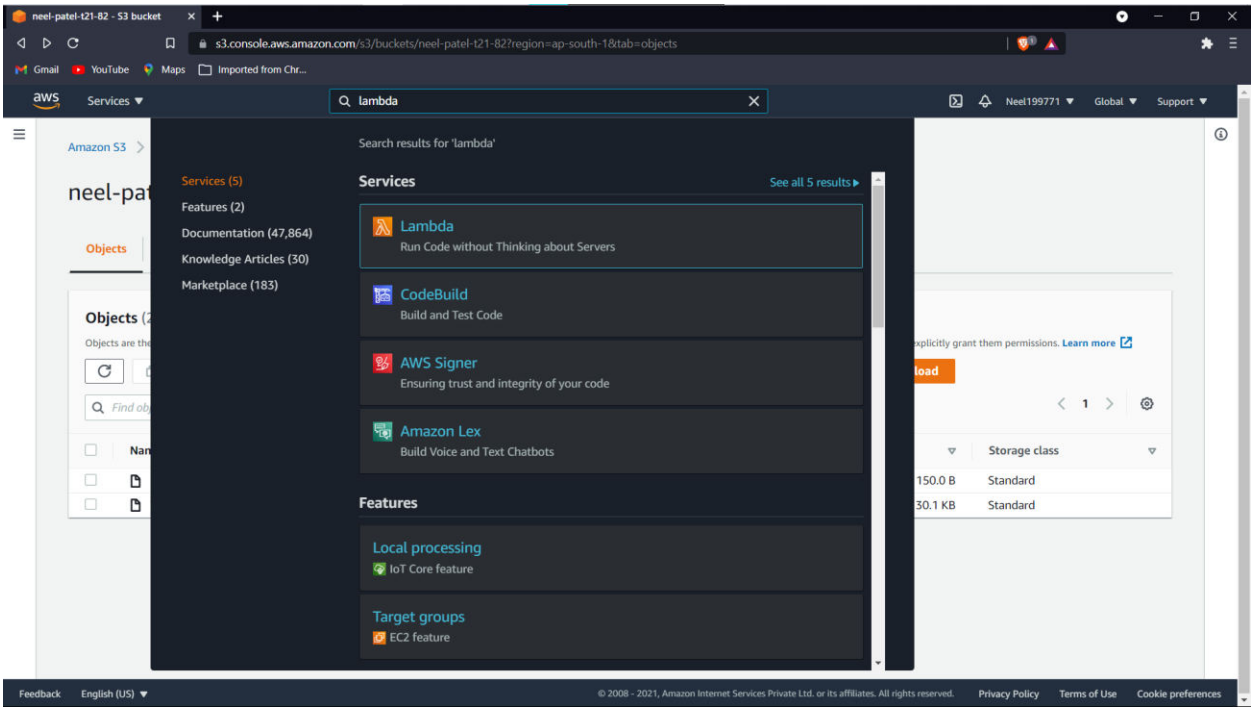




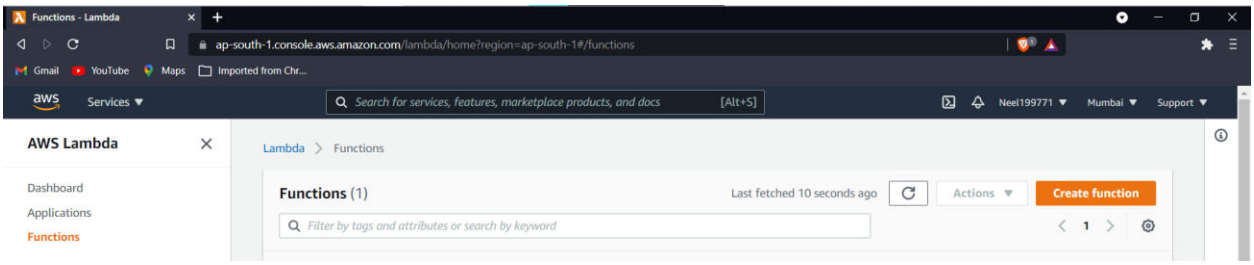
4) Add any .py or .java extenstion file and click on upload

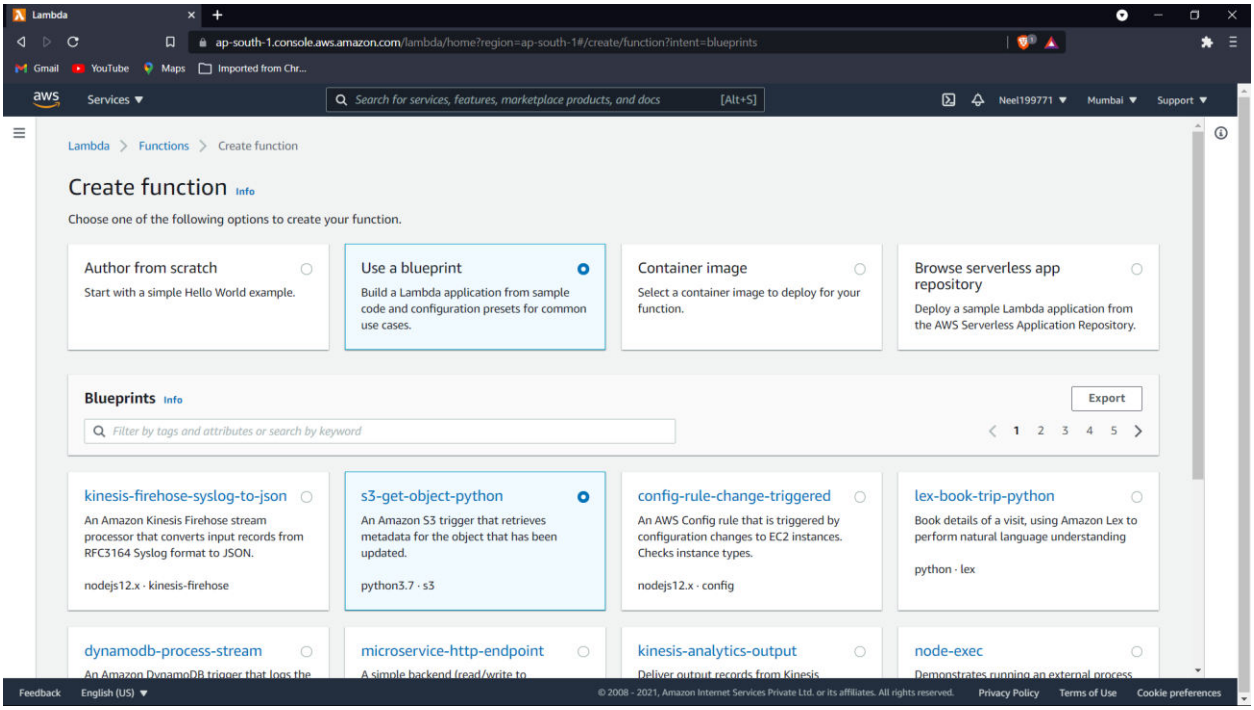


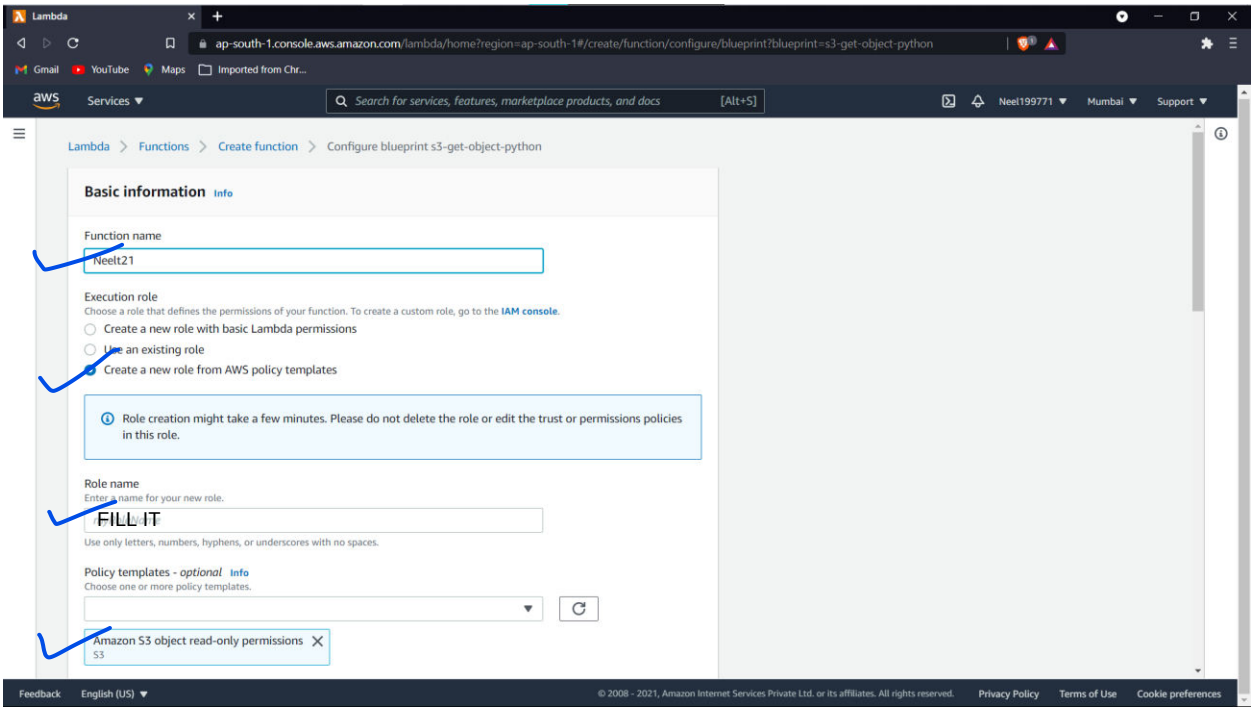
5) Now search lambda



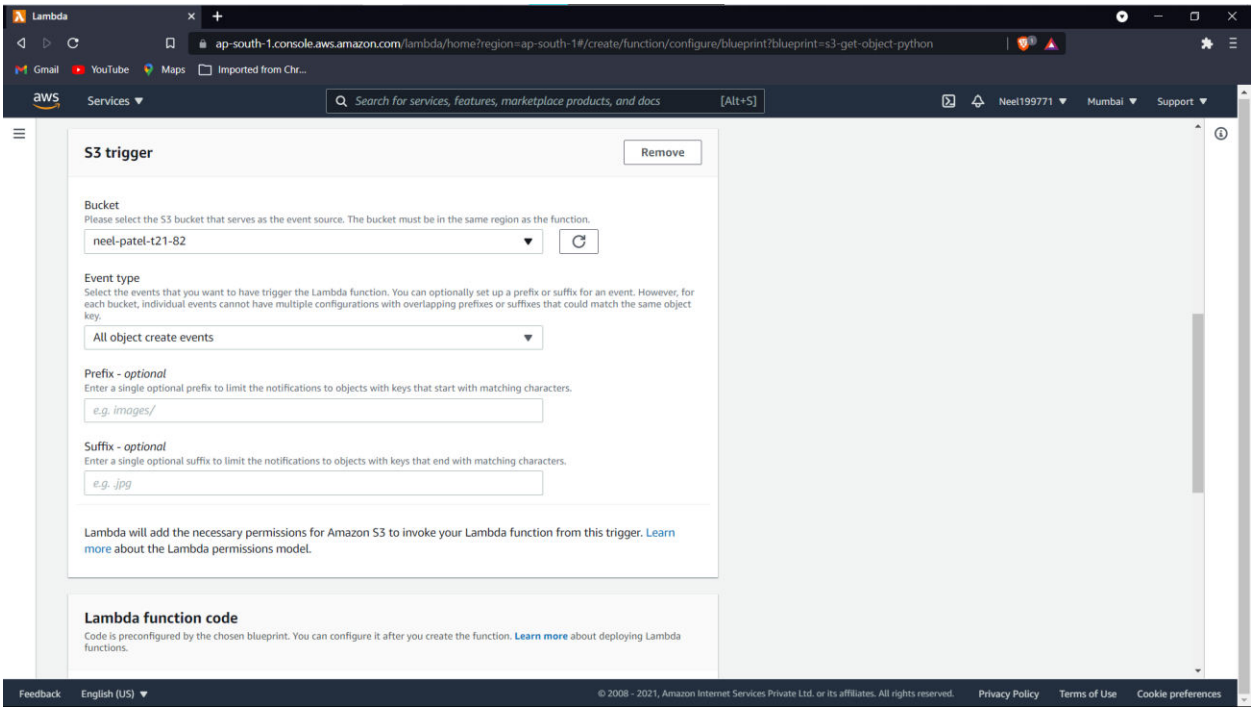
6)Click create function , click on below options and click on configure


Functions - Lambda
ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/functions
Gmail
YouTube
Maps
Imported from Chr...
aws
Services
Search for services, features, marketplace products, and docs
[Alt+S]
Neel199771
Mumbai
Support
AWS Lambda
Dashboard
Applications
Functions
Lambda > Functions
Functions (1)
Last fetched 10 seconds ago
Actions
Create function
Filter by tags and attributes or search by keyword
1

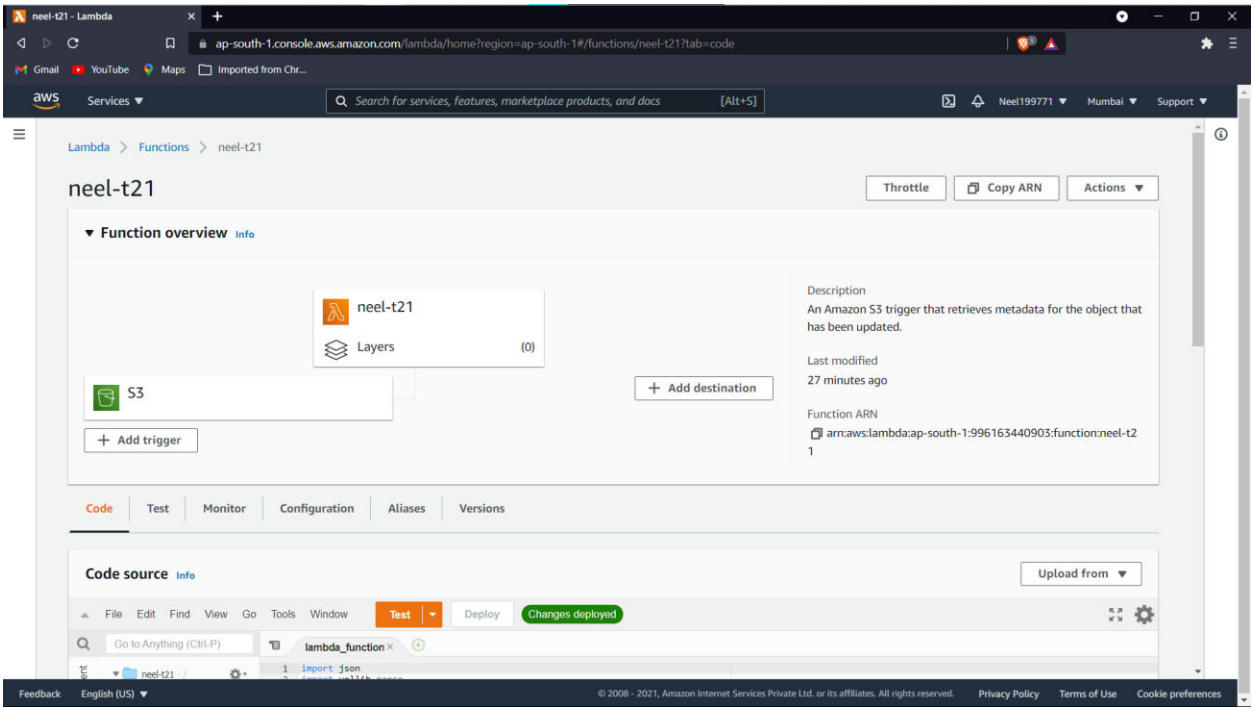

Lambda
ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/create/function?intent=blueprints
Lambda > Functions > Create function
Create function Info
Choose one of the following options to create your function.
Author from scratch
Start with a simple Hello World example.
Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.
Container image
Select a container image to deploy for your function.
Browse serverless app repository
Deploy a sample Lambda application from the AWS Serverless Application Repository.
Blueprints Info
Export
Filter by tags and attributes or search by keyword
1 2 3 4 5
kinesis-firehose-syslog-to-json
An Amazon Kinesis Firehose stream processor that converts input records from RFC3164 Syslog format to JSON.
nodejs12.x · kinesis-firehose
s3-get-object-python
An Amazon S3 trigger that retrieves metadata for the object that has been updated.
python3.7 · s3
config-rule-change-triggered
An AWS Config rule that is triggered by configuration changes to EC2 instances. Checks instance types.
nodejs12.x · config
lex-book-trip-python
Book details of a visit, using Amazon Lex to perform natural language understanding
python · lex
dynamodb-process-stream
microservice-http-endpoint
kinesis-analytics-output
node-exec
Feedback
English (US)
© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences


Lambda
ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/create/function/configure/blueprint?blueprint=s3-get-object-python
Lambda > Functions > Create function > Configure blueprint s3-get-object-python
Basic information Info
Function name
Neelt21
Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.
Create a new role with basic Lambda permissions
Use an existing role
Create a new role from AWS policy templates
Role creation might take a few minutes. Please do not delete the role or edit the trust or permissions policies in this role.
Role name
Enter a name for your new role.
FILL IT
Use only letters, numbers, hyphens, or underscores with no spaces.
Policy templates - optional Info
Choose one or more policy templates.
Amazon S3 object read-only permissions
S3
Feedback
English (US)
© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences

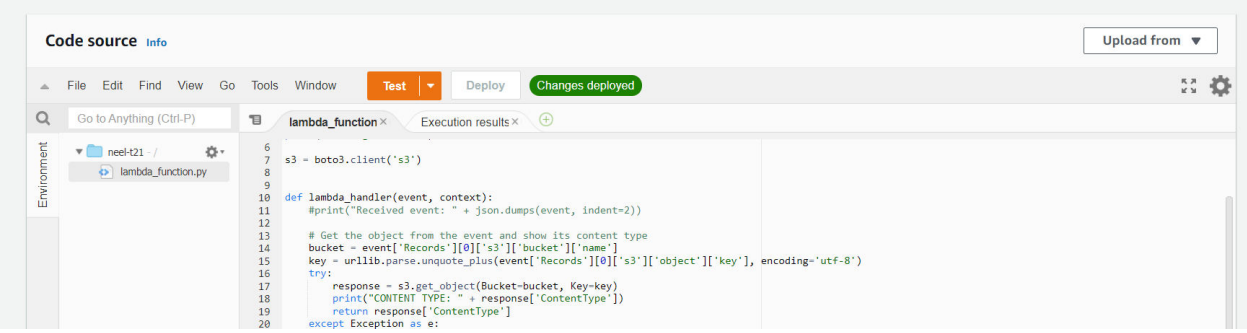
7)Select the bucket created and create trigger ,click on create function

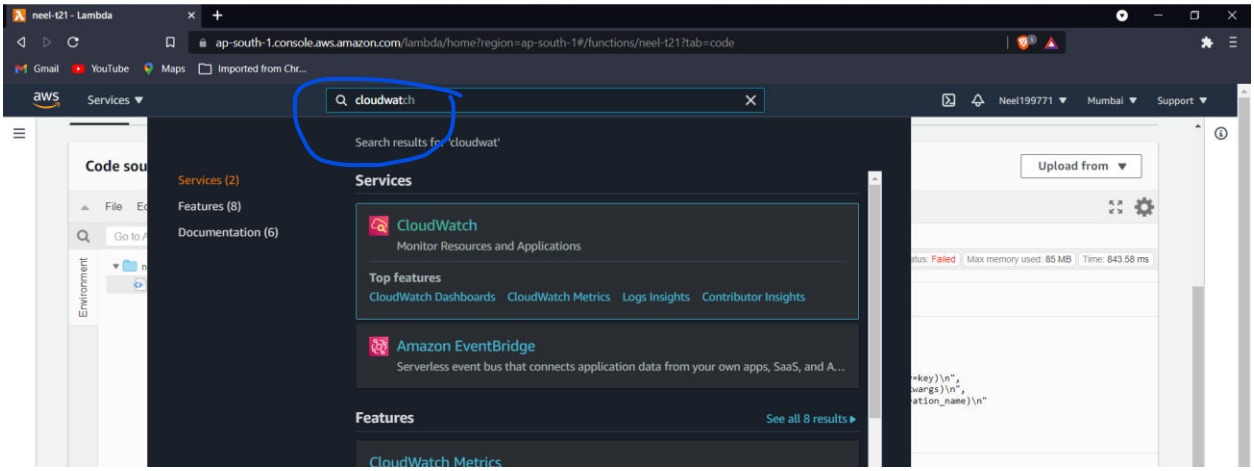


8) Check the given trigger is created

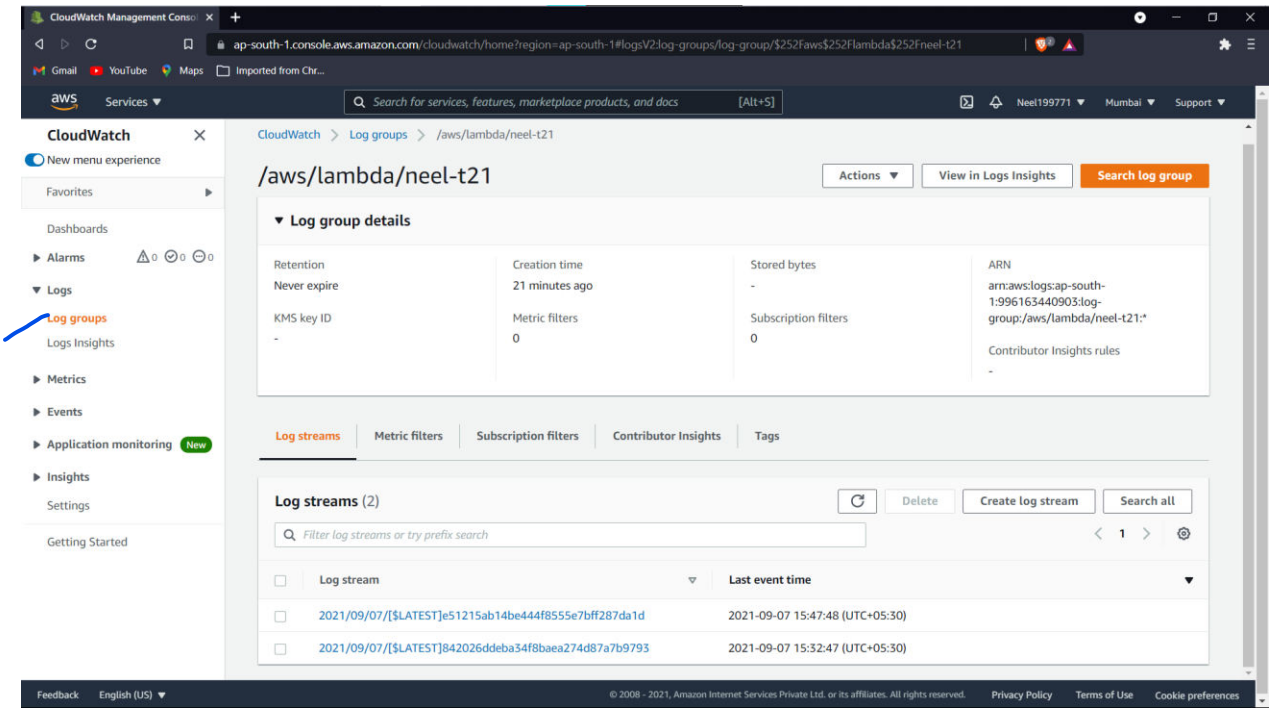


9) Click on the orange test button

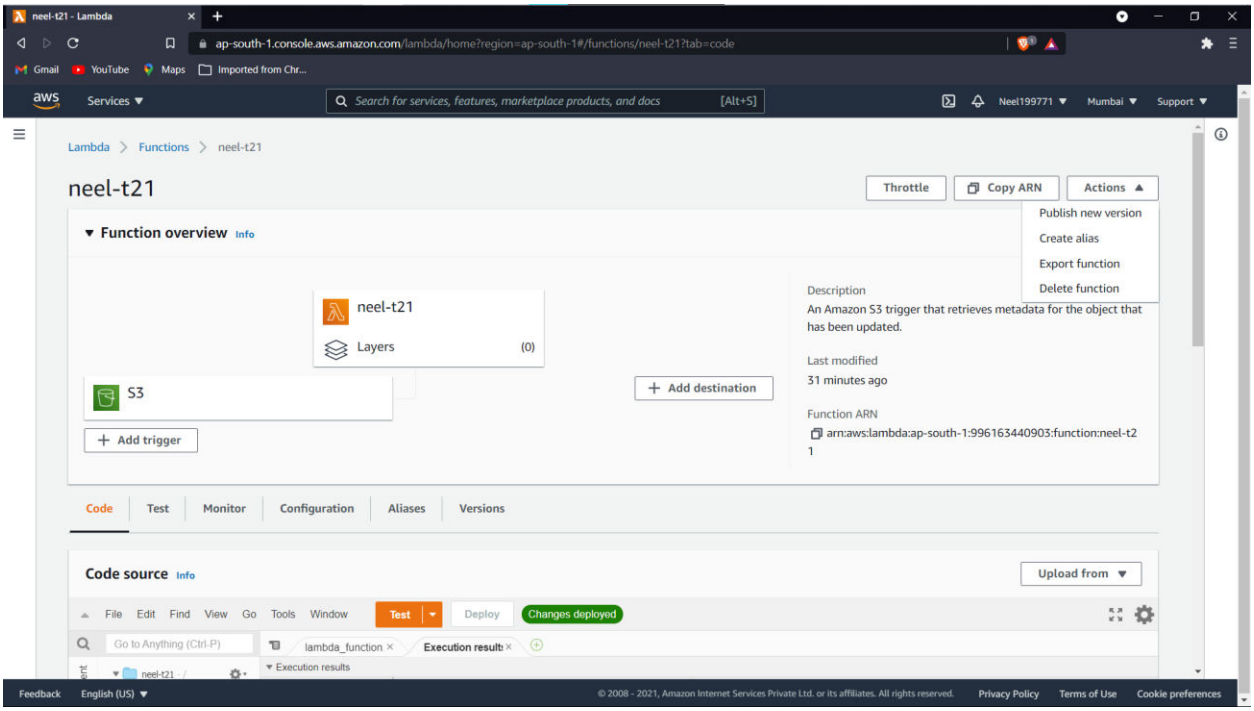


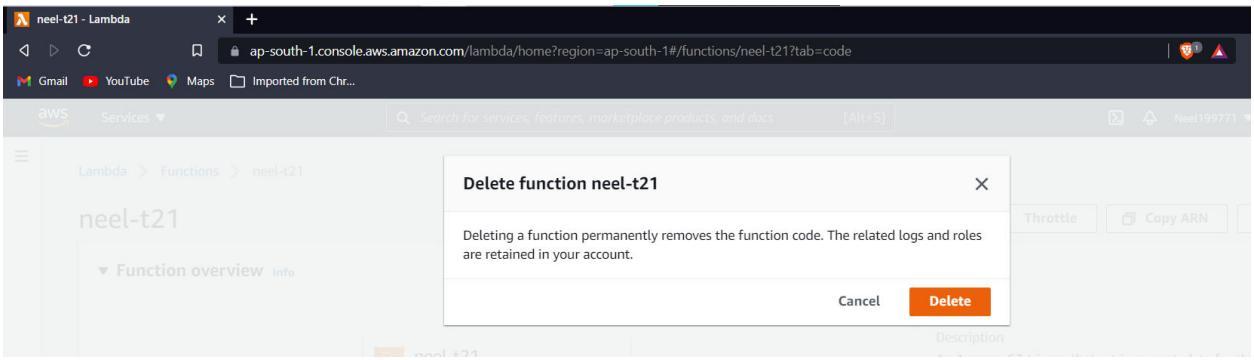


10) Check the logs , “an image has been added” message will be printed in logs

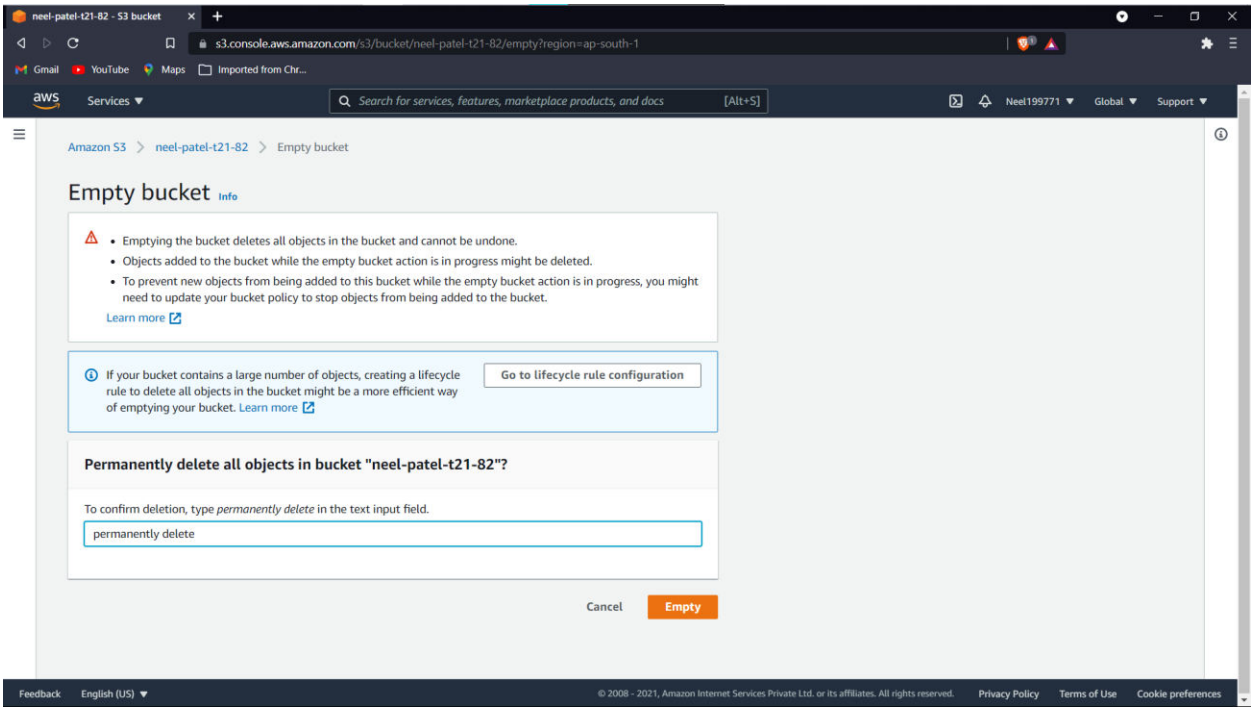


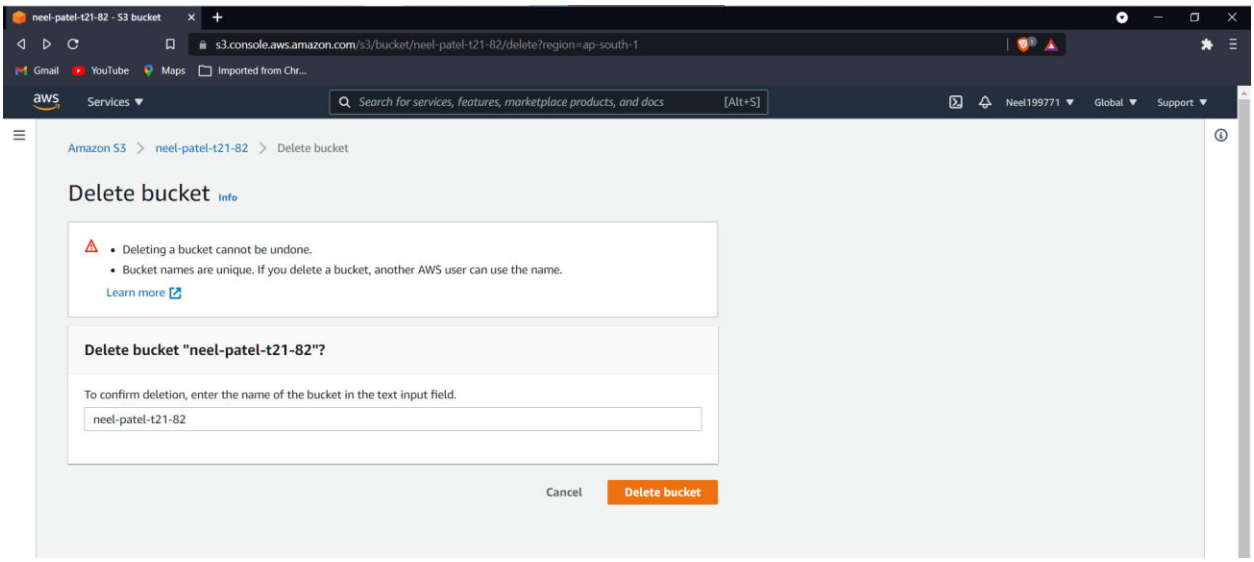
11) terminate by clicking on delete function





12) Empty and delete the bucket





Conclusion:-

Learnt about AWS Lambda function their applications in software industry and created a Lambda function using python to add image in a S3 bucket and verified if image is added or not by printing message after message is added

Name :-Niranjan Rajesh Joshi
Roll No:- 2105051
Batch:- T13
Date Of Performance :- 05/09/2023

Experiment 10

Aim:-

To create a Lambda function using Python for adding data to Dynamo DB database.

Theory:-

DYNAMO DB

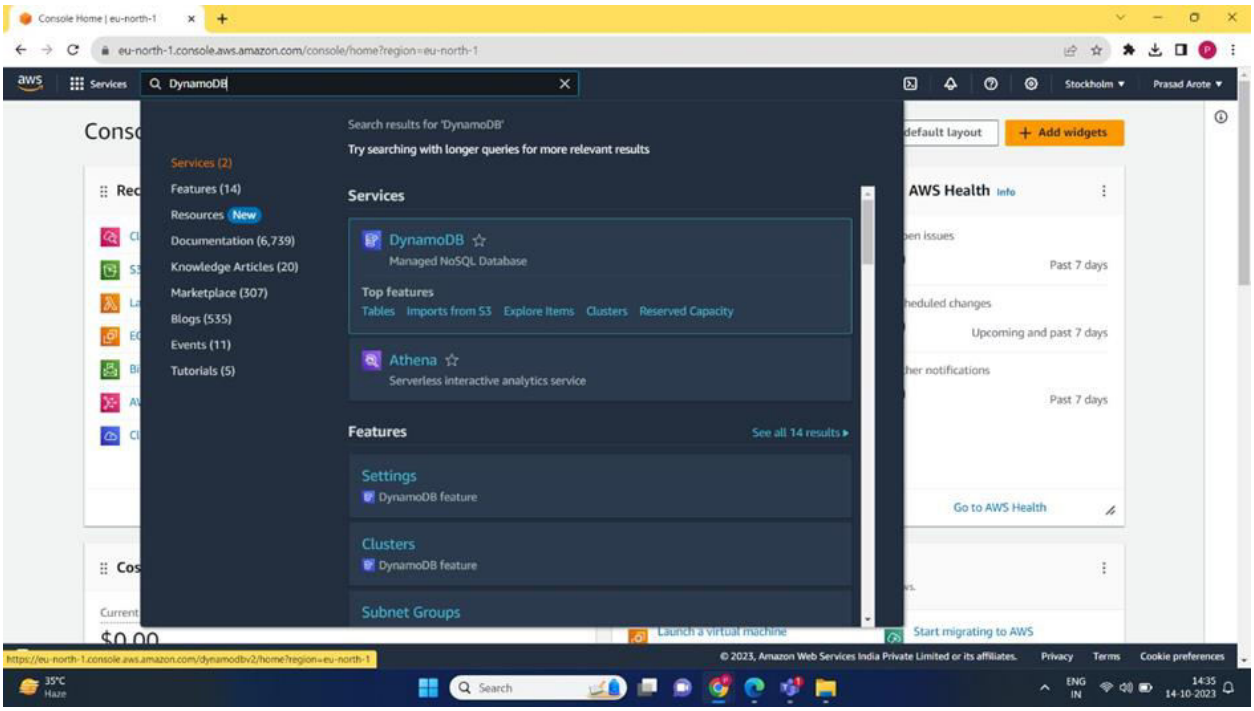
Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. DynamoDB lets you offload the administrative burdens of operating and scaling a distributed database so that you don't have to worry about hardware provisioning, setup and configuration, replication, software patching, or cluster scaling. DynamoDB also offers encryption at rest, which eliminates the operational burden and complexity involved in protecting sensitive data.

With DynamoDB, you can create database tables that can store and retrieve any amount of data and serve any level of request traffic. You can scale up or scale down your tables' throughput capacity without downtime or performance degradation. You can use the AWS Management Console to monitor resource utilization and performance metrics

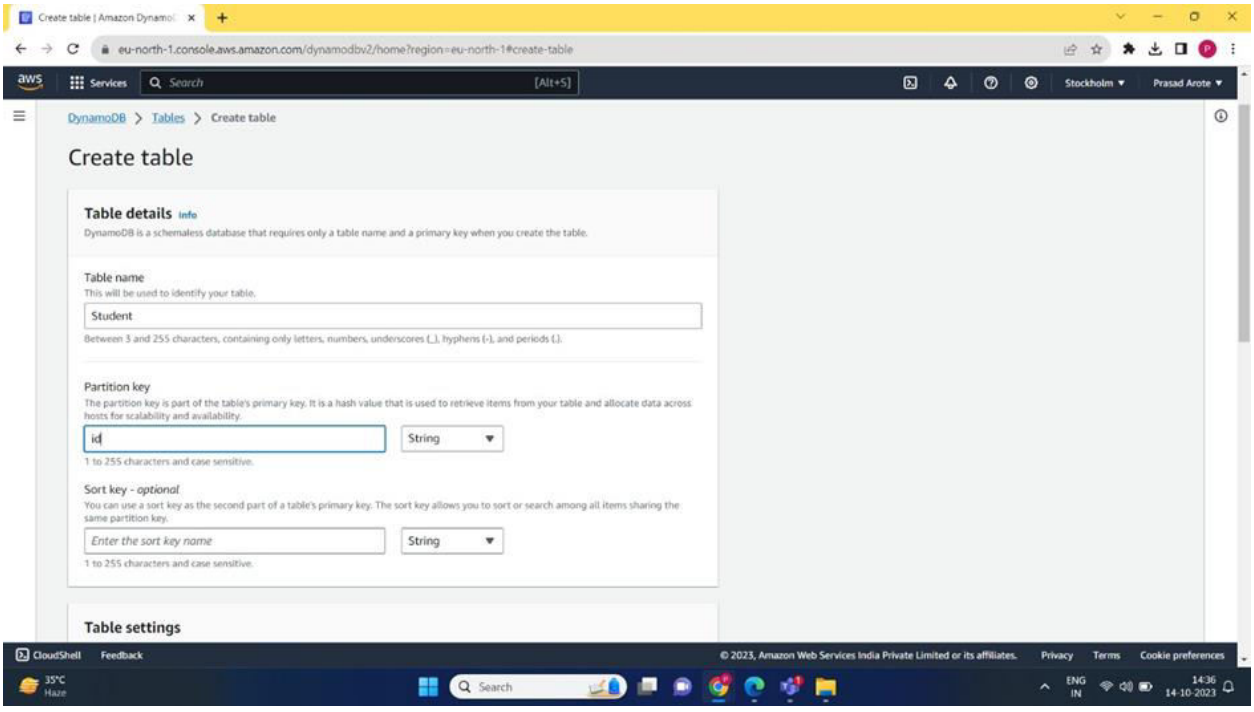
DynamoDB provides on-demand backup capability. It allows you to create full backups of your tables for long-term retention and archival for regulatory compliance needs.

Steps :-

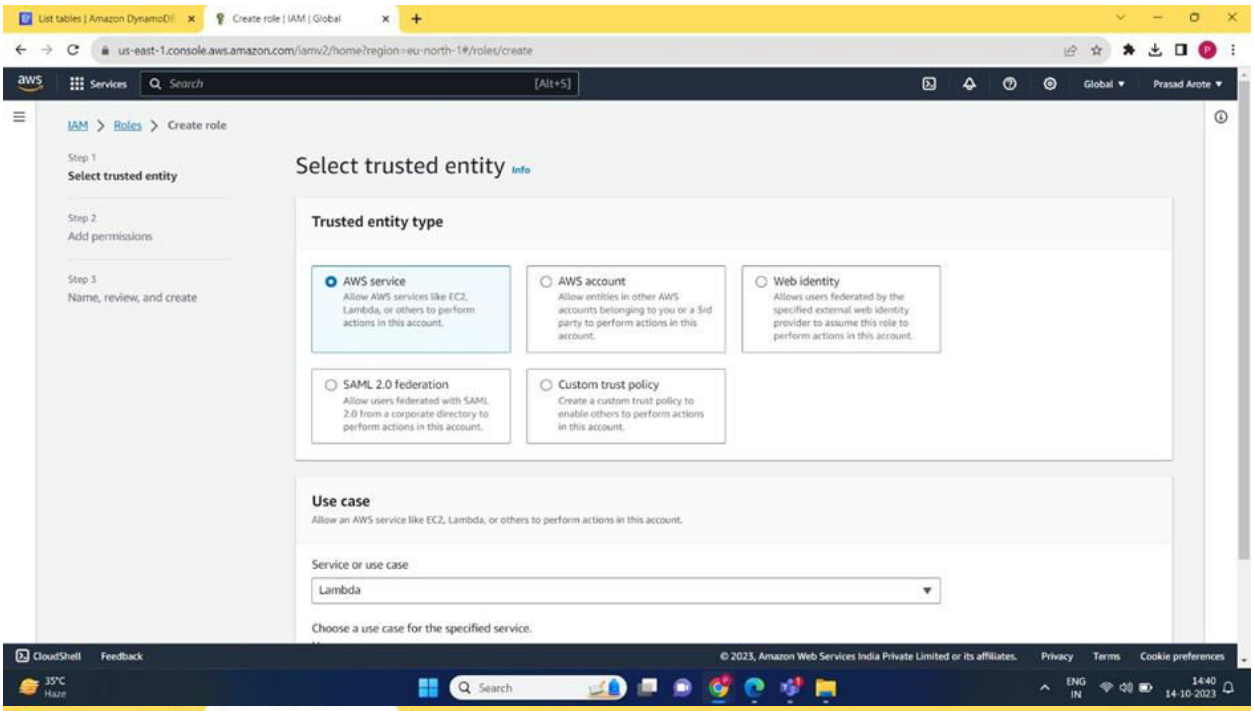
1) Login to AWS account and search for DynamoDB in search bar

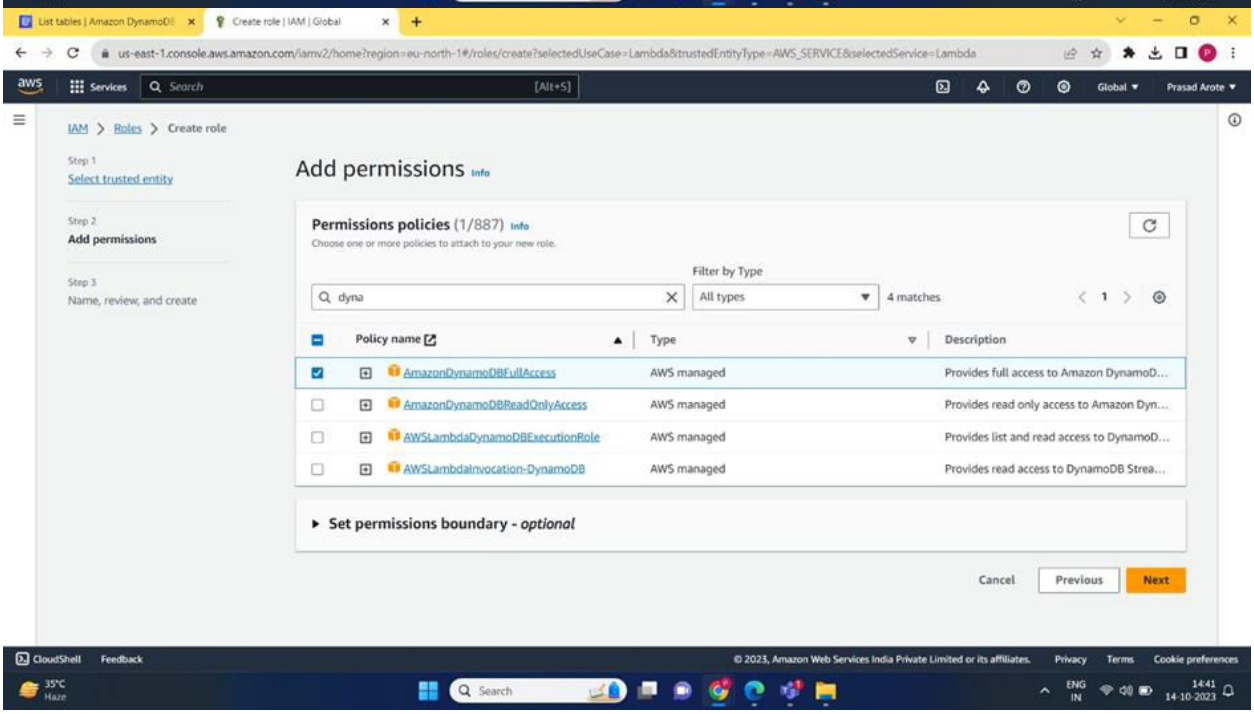


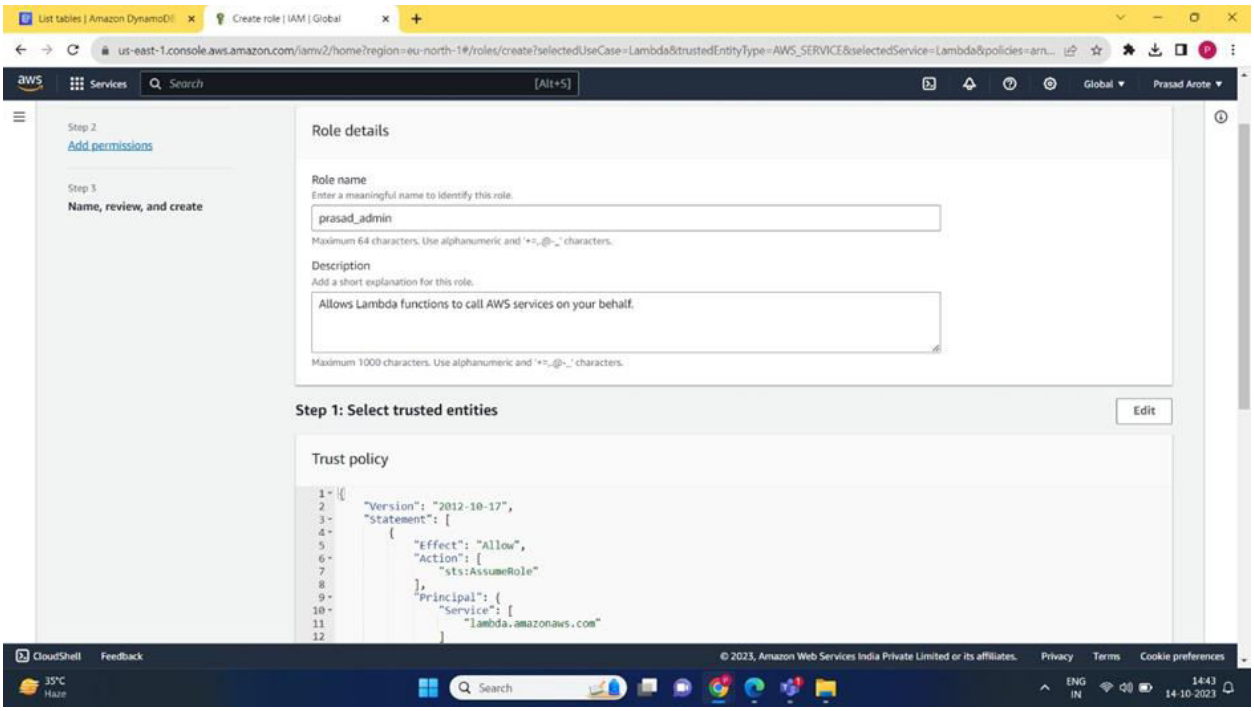
2) Click on DynamoDB option shown above and then click on create table

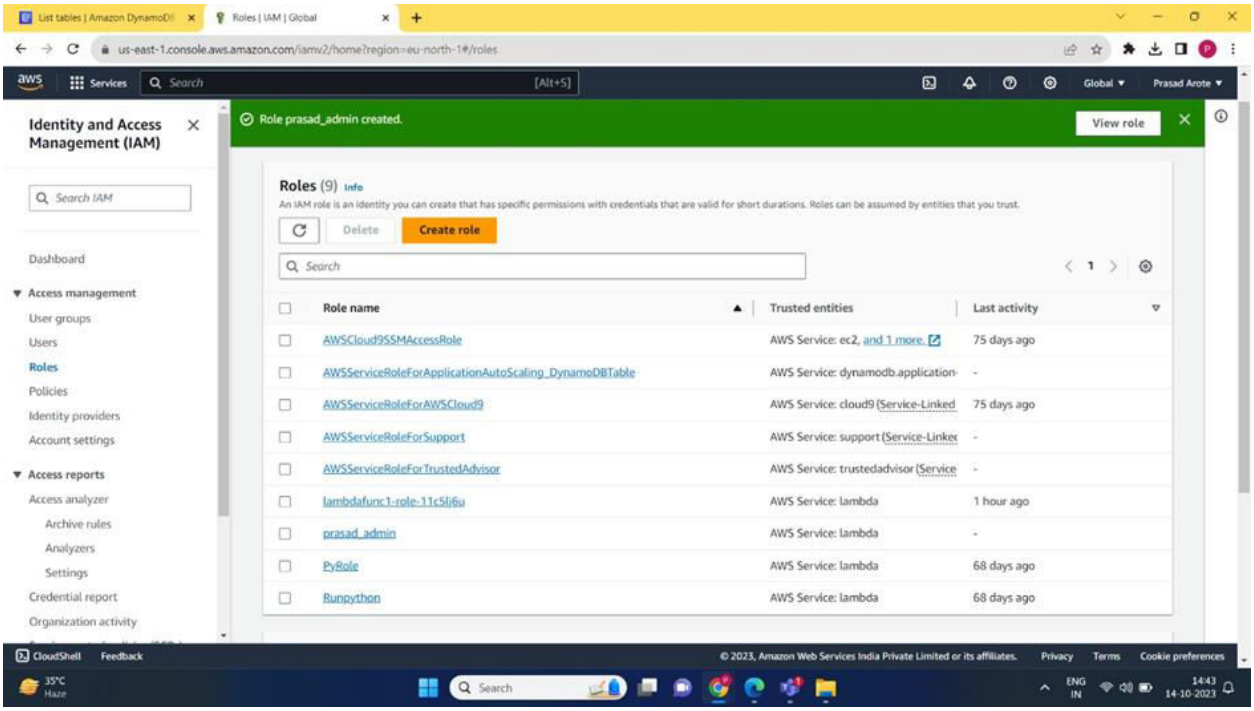


3)Then search IAM in the search box above and create a new role , give AmazonDynamoFullAccess permission to created user

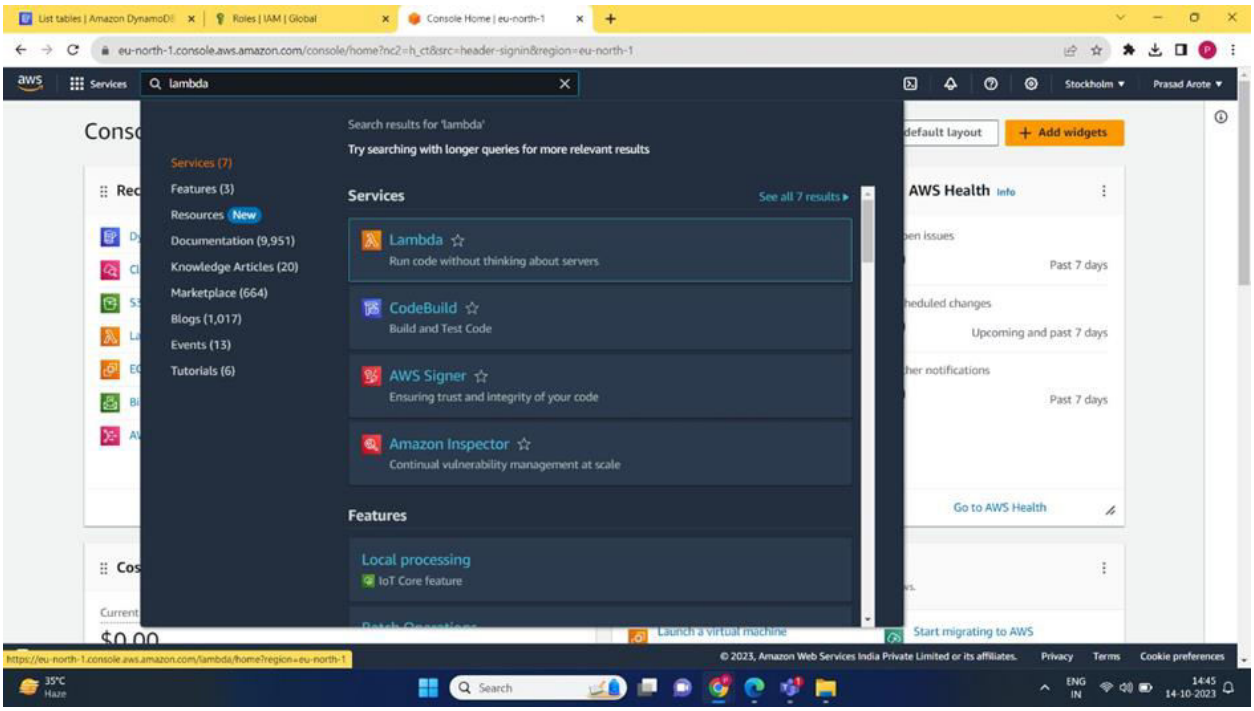

List tables | Amazon DynamoDB
Create role | IAM | Global
us-east-1.console.aws.amazon.com/iamv2/home?region=eu-north-1#/roles/create
Services
Search
[Alt+S]
Global
Prasad Arote
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Select trusted entity Info
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.
Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.
Service or use case
Lambda
Choose a use case for the specified service.
CloudShell
Feedback
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences
35°C
Haze
Search
ENG
IN
14:40
14-10-2023

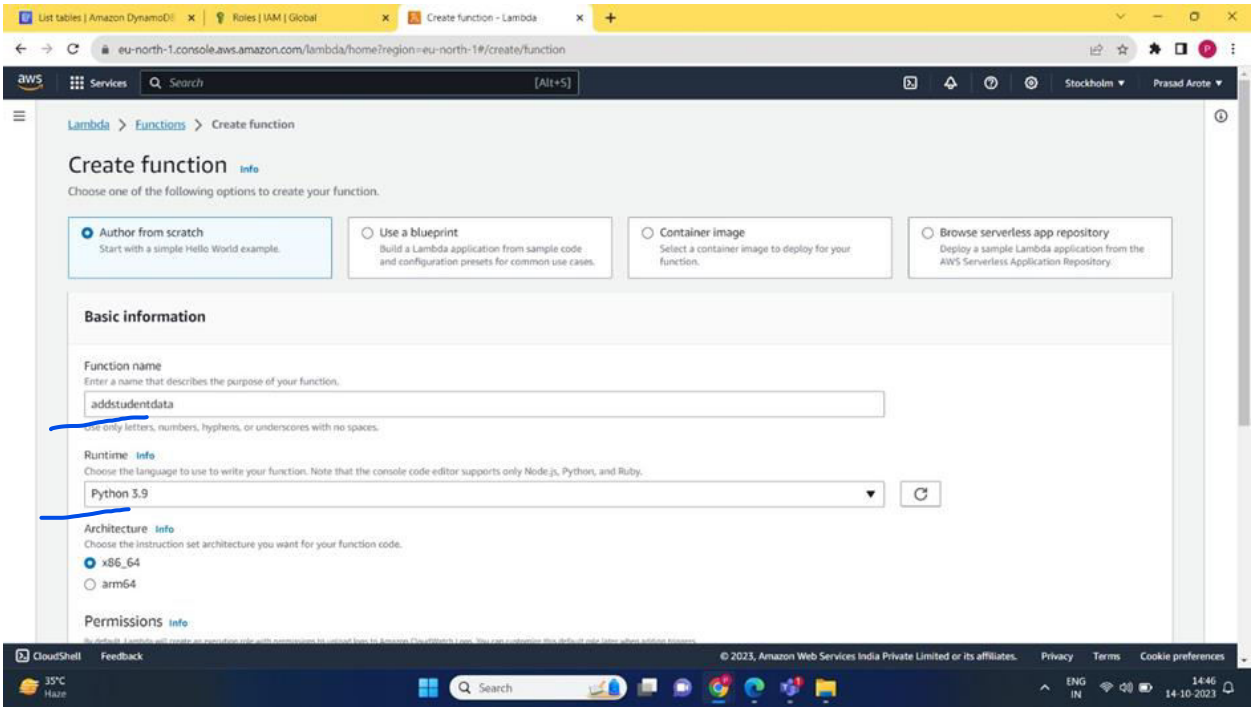

List tables | Amazon DynamoDB
Create role | IAM | Global
us-east-1.console.aws.amazon.com/iamv2/home?region=eu-north-1#/roles/create?selectedUseCase=Lambda&trustedEntityType=AWS_SERVICE&selectedService=Lambda
Services
Search
[Alt+S]
Global
Prasad Arote
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (1/887) Info
Choose one or more policies to attach to your new role.
Filter by Type
dyna
All types
4 matches
Policy name
Type
Description
AmazonDynamoDBFullAccess
AWS managed
Provides full access to Amazon DynamoD...
AmazonDynamoDBReadOnlyAccess
AWS managed
Provides read only access to Amazon Dyn...
AWSLambdaDynamoDBExecutionRole
AWS managed
Provides list and read access to DynamoD...
AWSLambdaInvocation-DynamoDB
AWS managed
Provides read access to DynamoDB Strea...
Set permissions boundary - optional
Cancel
Previous
Next
CloudShell
Feedback
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences
35°C
Haze
Search
ENG
IN
14:41
14-10-2023

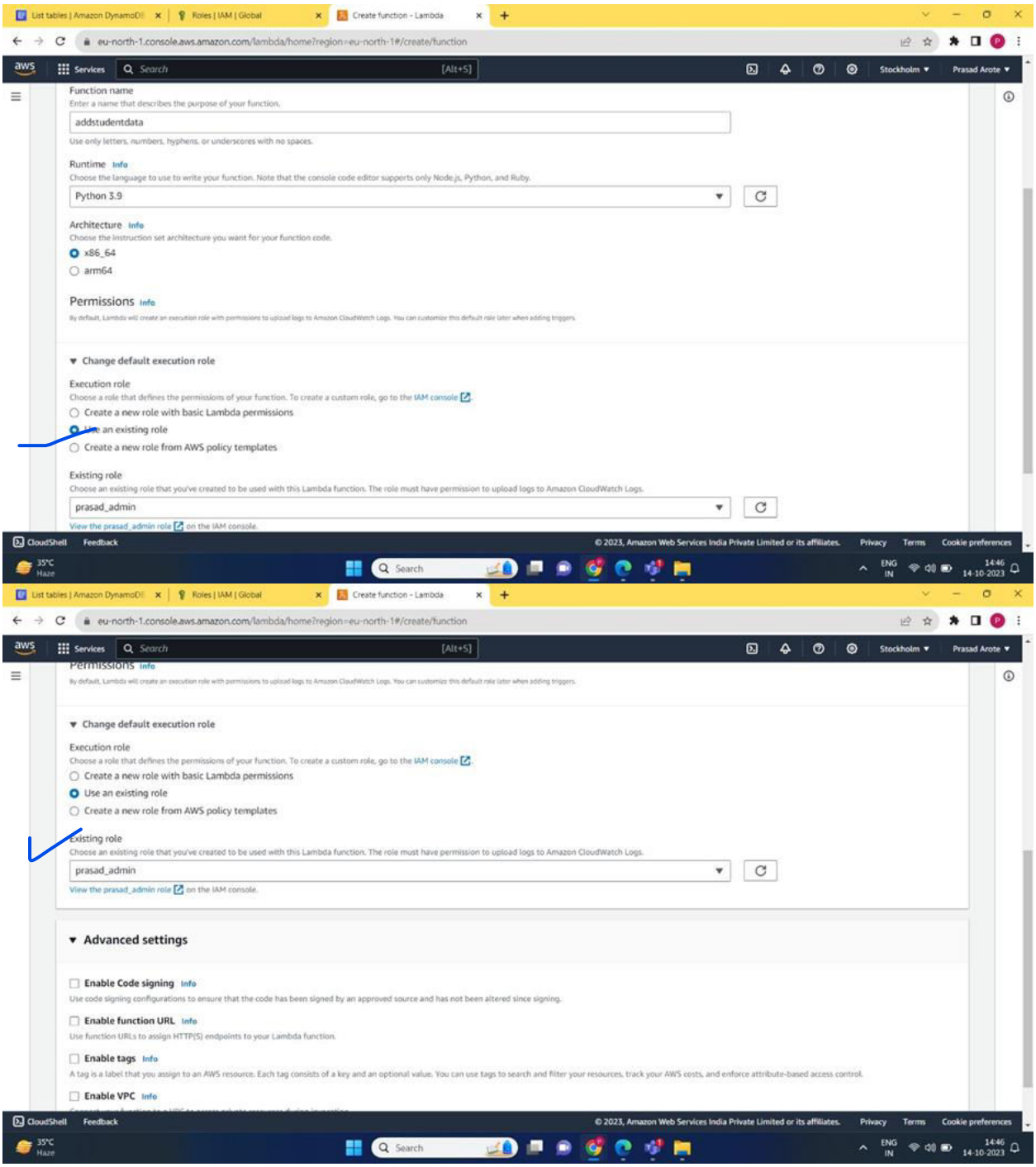




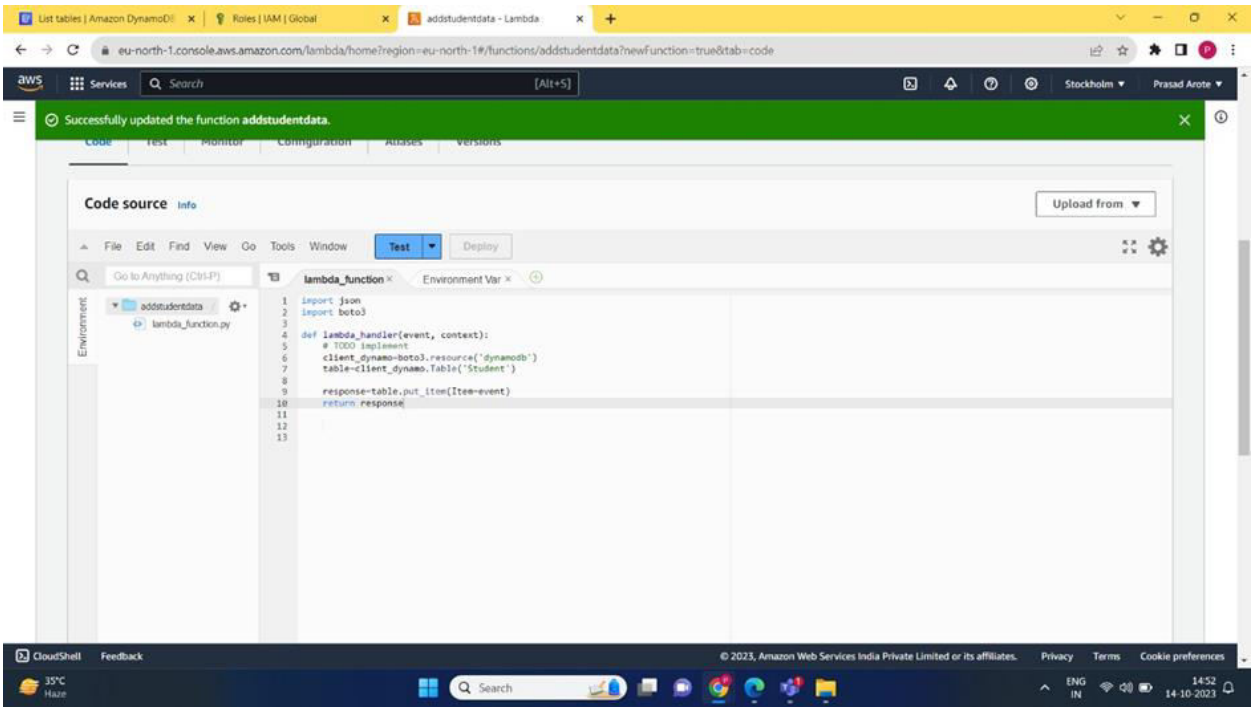
4) Search Lambda in search box and click on it , then create a new lambda function


List tables | Amazon DynamoDB
Roles | IAM | Global
Console Home | eu-north-1
eu-north-1.console.aws.amazon.com/console/home?nc2=h_ct&src=header-signin&region=eu-north-1
aws
Services
lambda
Stockholm
Prasad Arote
Search results for 'lambda'
Try searching with longer queries for more relevant results
Services (7)
Features (3)
Resources New
Documentation (9,951)
Knowledge Articles (20)
Marketplace (664)
Blogs (1,017)
Events (13)
Tutorials (6)
Services
See all 7 results
Lambda
Run code without thinking about servers
CodeBuild
Build and Test Code
AWS Signer
Ensuring trust and integrity of your code
Amazon Inspector
Continual vulnerability management at scale
Features
Local processing
IoT Core feature
default layout
Add widgets
AWS Health Info
Past 7 days
Upcoming and past 7 days
Go to AWS Health
Launch a virtual machine
Start migrating to AWS
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences
35°C
Haze
Search
ENG
IN
14:45
14-10-2023

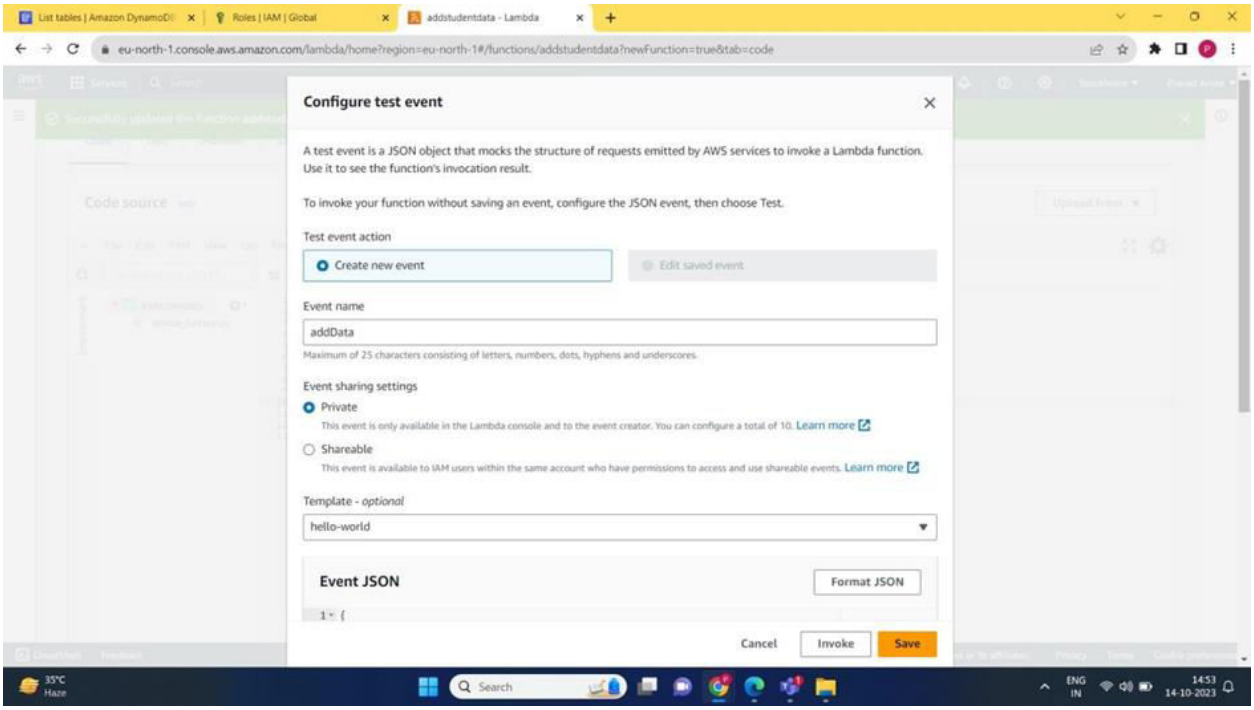

List tables | Amazon DynamoDB
Roles | IAM | Global
Create function - Lambda
eu-north-1.console.aws.amazon.com/lambda/home?region=eu-north-1#/create/function
aws
Services
Search
[Alt+S]
Stockholm
Prasad Arote
Lambda > Functions > Create function
Create function Info
Choose one of the following options to create your function.
Author from scratch
Start with a simple Hello World example.
Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.
Container image
Select a container image to deploy for your function.
Browse serverless app repository
Deploy a sample Lambda application from the AWS Serverless Application Repository.
Basic information
Function name
Enter a name that describes the purpose of your function.
addstudentdata
Use only letters, numbers, hyphens, or underscores with no spaces.
Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.9
Architecture Info
Choose the instruction set architecture you want for your function code.
x86_64
arm64
Permissions Info
CloudShell
Feedback
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences
35°C
Haze
Search
ENG
IN
14:46
14-10-2023

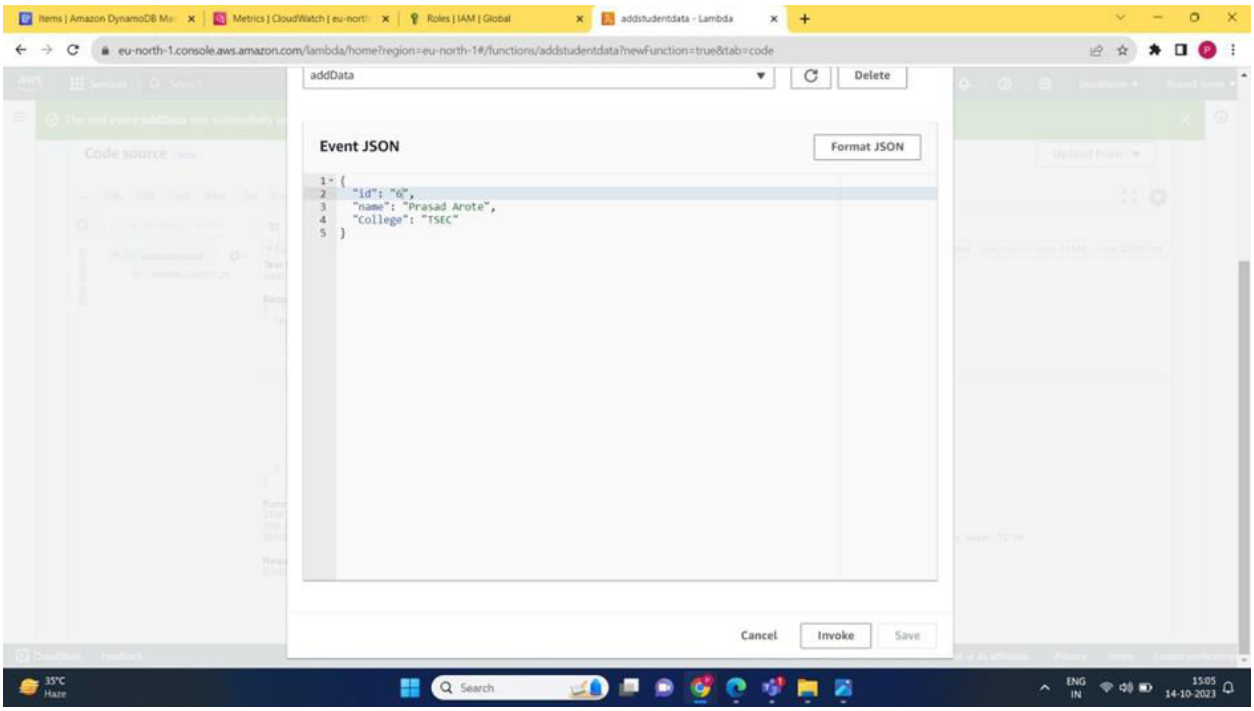


5) Write the following code in code source

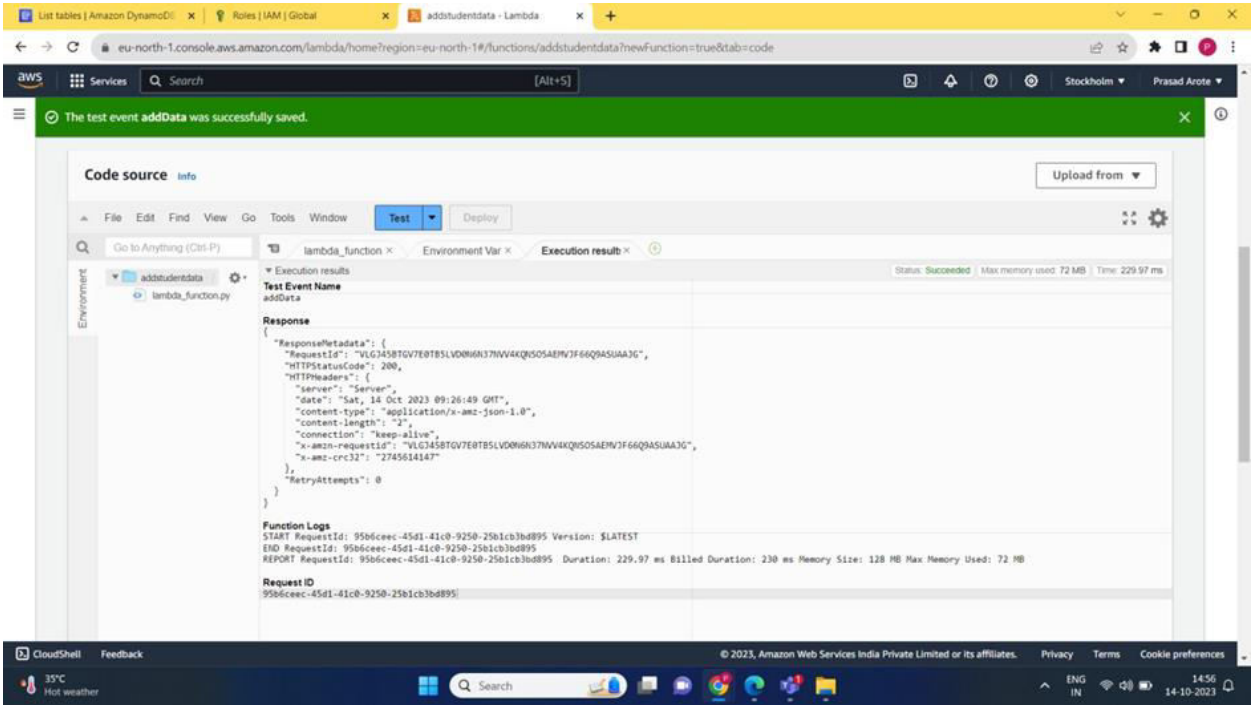


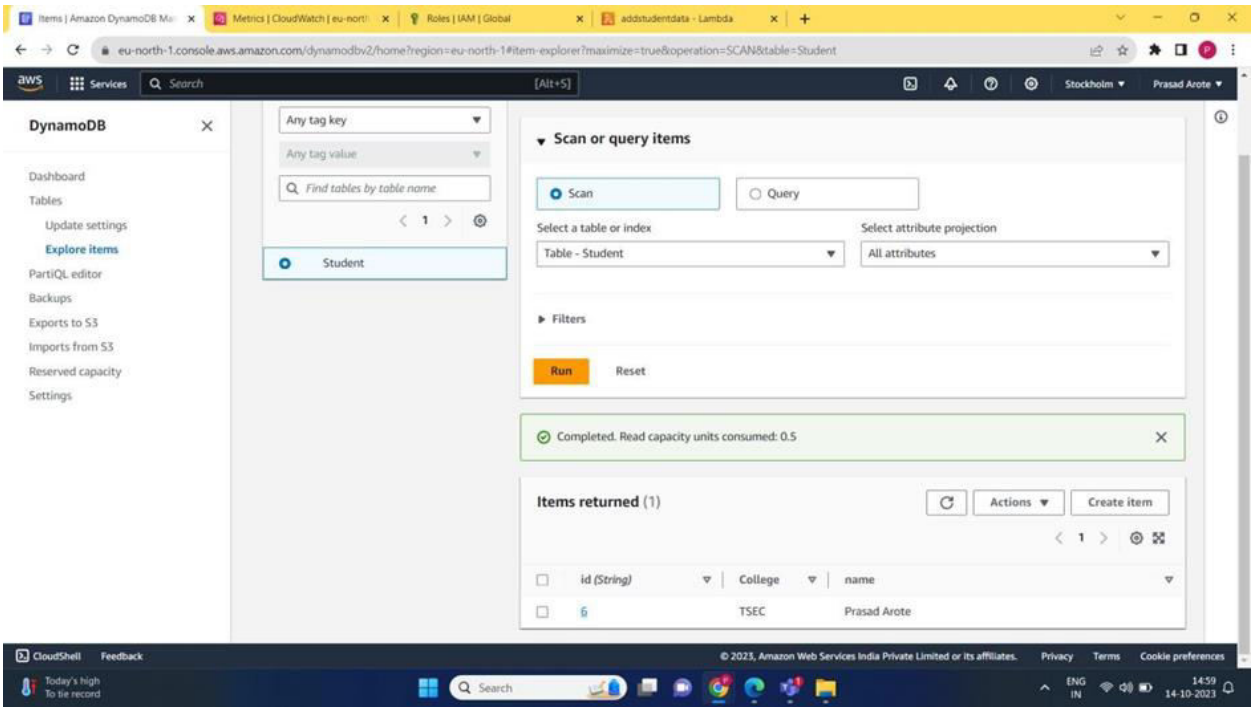
6) Configure the test event and save





7) Run the test and afterwards go to the DynamoDB>Explore items> Student where you can see the record inserted using lambda function.





Conclusion:-

Learnt about Amazon DynamoDB database service and inserted data into DynamoDB database by creating a new user , granting him permissions and then using a lambda function

Containerization is a software deployment process that bundles an application's code with all the files and libraries it needs to run on any infrastructure. Traditionally, to run any application on your computer, you had to install the version that matched your machine's operating system

Orchestration is the coordination and management of multiple computer systems, applications and/or services, stringing together multiple tasks in order to execute a larger workflow or process. These processes can consist of multiple tasks that are automated and can involve multiple systems.

The goal of orchestration is to streamline and optimize the execution of frequent, repeatable processes and thus to help data teams more easily manage complex tasks and workflows. Anytime a process is repeatable, and its tasks can be automated, orchestration can be used to save time, increase efficiency, and eliminate redundancies. For example, you can simplify data and machine learning with jobs orchestration.

Name :- Niranjan Rajesh Joshi
RollNo :- 2105051
Batch :- T13
Date Of Performance :- 01/08/2023

Kubernetes, also known as K8s, is an open-source system for automating deployment, scaling, and management of containerized applications.

Written Assignment 1:Study of Kubernetes

Q1) What security measures can be taken while using kubernetes ?

There are a number of security measures that can be taken while using Kubernetes. Some of the most important include:

1)Use Role-Based Access Control (RBAC): RBAC allows you to define who has access to the Kubernetes API and what permissions they have. This is essential for preventing unauthorized access to your cluster and its resources.
2)Use third-party authentication for the API server: This allows you to integrate Kubernetes with an existing identity provider, such as GitHub or Okta. This can make it easier to manage user accounts and permissions.
3)Protect etcd with TLS and a firewall: etcd is a distributed key-value store that stores the state of your Kubernetes cluster. It is a critical component, so it is important to protect it from unauthorized access.
4)Isolate Kubernetes nodes: Kubernetes nodes are the machines that run your containerized applications. It is important to isolate them from the rest of your network to prevent attackers from gaining access to your cluster.
5)Monitor network traffic to limit communications: Kubernetes workloads can communicate with each other over the network. It is important to monitor this traffic and limit it to only the necessary communication paths.
6)Use process whitelisting: Process whitelisting allows you to define which processes are allowed to run on your Kubernetes nodes. This can help to prevent attackers from running malicious code on your cluster.
7)Turn on audit logging: Audit logging records all activity on your Kubernetes cluster. This can help you to detect and investigate security incidents.
8)Keep Kubernetes up to date: The Kubernetes team regularly releases security updates. It is important to keep your Kubernetes cluster up to date to protect against known vulnerabilities.
9)Lock down the Kubelet: The Kubelet is a service that runs on each Kubernetes node and is responsible for managing pods. It is important to lock down the Kubelet to prevent attackers from gaining control of your nodes.

In addition to these general security measures, there are a number of other specific things you can do to secure your Kubernetes cluster, such as:

10)Use Kubernetes namespaces to isolate your workloads: Namespaces allow you to group your workloads together and isolate them from each other. This can help to prevent attackers from moving laterally between your workloads.
11)Use Pod Security Policies (PSPs) to restrict the privileges of your pods: PSPs allow you to define what resources and privileges your pods are allowed to use. This can help to prevent attackers from gaining access to sensitive data or running malicious code on your cluster.
12)Use a service mesh to manage network traffic between your workloads: A service mesh can help you to secure and manage the network traffic between your workloads. This can help to prevent attackers from communicating with your workloads or eavesdropping on their traffic.
13)Use a security scanner to scan your container images for vulnerabilities: A security scanner can help you to identify and fix vulnerabilities in your container images before they are deployed to your cluster.
14)Implement a security incident response plan: A security incident response plan will help you to respond to security incidents in a timely and effective manner.

By following these security measures, you can help to protect your Kubernetes cluster and its workloads from attack.

Q2) What are the three security techniques used to protect data ?

The three most important security techniques used to protect data are:

1) **Encryption**
2) **Access control**
3) **Backup and recovery**

Encryption is the process of converting data into a format that cannot be read without a secret key. This makes data unreadable to unauthorized individuals, even if they have access to it. Encryption can be used to protect data at rest, in transit, and in use.

Access control is the process of restricting access to data to authorized individuals. This can be done using a variety of methods, such as passwords, multi-factor authentication, and role-based access control (RBAC). Access control is important for preventing unauthorized individuals from accessing and modifying data.

Backup and recovery is the process of creating copies of data and storing them in a secure location. This is important for protecting data from loss or corruption. Backup and recovery plans should be regularly tested to ensure that they are working properly.

In addition to these three core security techniques, there are a number of other security measures that can be used to protect data, such as network security, physical security, and security awareness training.

Here are some examples of how these three security techniques can be used to protect data:

Encryption: A company can encrypt its customer database to protect it from unauthorized access, even if the database is compromised.
Access control: A hospital can use RBAC to restrict access to patient medical records to authorized personnel, such as doctors and nurses.
Backup and recovery: A government agency can regularly back up its financial data to a secure location in case of a cyberattack or natural disaster.

By using these security techniques, organizations can help to protect their data from unauthorized access, modification, and loss.

Q3) How do you expose a service using ingress in Kubernetes?

To expose a service using Ingress in Kubernetes, you need to create an **Ingress resource**. An Ingress resource specifies the rules for routing traffic to your services.

To create an Ingress resource, you can use the following command:
kubectl create ingress <ingress-name>

The Ingress resource must specify the following:

1) **The rules for routing traffic to your services.**
2) **The hostname or IP address that traffic will be routed to.**
3) **The port that traffic will be routed to.**

For example, the following Ingress resource exposes a service named my-service on port 80:

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: my-ingress
spec:
  rules:
  - http:
        paths:
        - path: /
        pathType: Prefix
```

```
      backend:
      service:
      name: my-service
      port: 80
```

Once you have created the Ingress resource, you can access the service at the hostname or IP address specified in the Ingress resource. For example, if the Ingress resource specifies the hostname my-service.example.com, you can access the service at my-service.example.com on port 80.

You can also use Ingress to expose multiple services on the same hostname or IP address. To do this, you can specify multiple rules in the Ingress resource. For example, the following Ingress resource exposes two services, my-service and my-other-service, on port 80:

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: my-ingress
spec:
  rules:
  - http:
      paths:
      - path: /
      pathType: Prefix
      backend:
      service:
      name: my-service
      port: 80
  - http:
      paths:
      - path: /other
      pathType: Prefix
      backend:
      service:
      name: my-other-service
      port: 80
```

Ingress is a powerful tool for exposing services in Kubernetes. It allows you to expose services on a specific hostname or IP address, and to expose multiple services on the same hostname or IP address

Q4) Which service protocol does Kubernetes ingress expose ?

Ingress is a Kubernetes resource that allows you to expose services running in a cluster to external traffic. **Ingress can expose services through either HTTP or HTTPS.**

To expose a service using Ingress, you need to create an Ingress resource. An Ingress resource specifies the rules for routing traffic to your services.

When you create an Ingress resource, you need to specify the following:

1) The rules for routing traffic to your services.
2) The hostname or IP address that traffic will be routed to.
3) The port that traffic will be routed to.

The Ingress resource also specifies the service protocol, which can be either HTTP or HTTPS.

To expose a service using HTTP, you need to specify the http protocol in the Ingress resource. For example, the following Ingress resource exposes a service named my-service on port 80 using HTTP:

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: my-ingress
spec:
  rules:
  - http:
        paths:
        - path: /
        pathType: Prefix
        backend:
        service:
        name: my-service
        port: 80
```

To expose a service using HTTPS, you need to specify the https protocol in the Ingress resource. For example, the following Ingress resource exposes a service named my-service on port 443 using HTTPS:

```
apiVersion: networking.k8s.io/v1
kind: Ingress
```

```
metadata:
  name: my-ingress
spec:
  tls:
  - hosts:
        - my-service.example.com
        secretName: my-tls-secret
  rules:
  - http:
        paths:
        - path: /
        pathType: Prefix
        backend:
        service:
        name: my-service
        port: 443
```

Once you have created the Ingress resource, you can access the service at the hostname or IP address specified in the Ingress resource. For example, if the Ingress resource specifies the hostname my-service.example.com, you can access the service at my-service.example.com on port 80.

You can also use Ingress to expose multiple services on the same hostname or IP address. To do this, you can specify multiple rules in the Ingress resource.

Name :- Niranjan Rajesh Joshi
RollNo :- 2105051
Batch :- T13
Date Of Performance :- 13/10/2023

Written Assignment 2

Q.1 How to deploy Lambda function on AWS?

To deploy a Lambda function on AWS, user can follow these general steps:

1) Create a Lambda Function:

a) Log in to the AWS Management Console.
b) Open the Lambda console.
c) Click on "Create function."
d) Choose the method for creating the function (Author from scratch, Use a blueprint, or Browse serverless app repository).
e) Configure the function details such as the function name, runtime, and permissions.

2) Write and Upload Code:

a) Write your function code in the selected runtime environment.
b) Upload the code either directly in the Lambda console or through an S3 bucket.

3) Configure Function:

a) Set up the function's environment variables, execution role, memory allocation, timeout, and other settings.

4) Configure Triggers:

a) You can configure triggers for your Lambda function such as API Gateway, S3, DynamoDB, etc., depending on the use case.

5) Test the Function:

a) Use the test events provided by Lambda or create your own custom test event to verify the function's behavior.

6) Deploy the Function:

a) Once the function is configured and tested, you can deploy it by clicking the "Deploy" button in the Lambda console.

7) Monitor the Function:

a) Monitor the function's performance and any potential errors using AWS CloudWatch logs and metrics.

Q.2 What are the deployment options for AWS Lambda?

AWS Lambda supports various deployment options that cater to different use cases and development preferences. Some of the key deployment options for AWS Lambda include:

AWS Management Console: Users can create, configure, and deploy Lambda functions directly through the AWS Management Console. This web-based interface allows users to set up and manage Lambda functions without the need for any additional tools or services.

AWS Command Line Interface (CLI): The AWS CLI provides a command-line interface for managing AWS services, including Lambda. Users can use the AWS CLI to create, deploy, update, and manage Lambda functions, which is particularly useful for automating deployment processes and integrating Lambda into scripts and workflows.

AWS Software Development Kits (SDKs): AWS SDKs are available for multiple programming languages, allowing users to integrate Lambda into their applications directly. Using the SDKs, users can deploy Lambda functions programmatically, providing flexibility and customization for the deployment process.

AWS CloudFormation: AWS CloudFormation enables users to define and deploy infrastructure as code. Users can use CloudFormation templates to provision and manage Lambda functions along with other AWS resources in a declarative manner, facilitating consistent and repeatable deployments.

AWS Serverless Application Model (SAM): SAM is an open-source framework that extends CloudFormation to simplify the deployment of serverless applications. It provides a simplified syntax for defining serverless applications, including Lambda functions, APIs, and other resources, making it easier to deploy complex serverless applications.

Integrated Development Environments (IDEs): Several IDEs, such as AWS Toolkit for Visual Studio, AWS Toolkit for PyCharm, and AWS Toolkit for IntelliJ, offer tools and features for developing, debugging, and deploying Lambda functions directly from the IDE environment.

These deployment options provide developers with various tools and methods to create, manage, and deploy Lambda functions on AWS, catering to diverse development workflows and preferences.

Q.3 What are the 3 full deployment modes that can be used for AWS?

AWS provides three main deployment modes that users can leverage for their applications:

1) **EC2 Deployment:**
   Description: EC2 (Elastic Compute Cloud) is a web service that provides resizable compute capacity in the cloud. It allows users to launch virtual servers (EC2 instances) and provides full control over the operating system, network, and other aspects of the infrastructure.
   Use Case: EC2 deployment is suitable for users who require complete control and customization over their computing resources. It is ideal for applications that demand a high level of configurability and flexibility.

2) **Elastic Beanstalk Deployment:**
   Description: AWS Elastic Beanstalk is a platform as a service (PaaS) that simplifies the process of deploying and scaling web applications and services. It automatically handles the details of capacity provisioning, load balancing, auto-scaling, and application health monitoring.
   Use Case: Elastic Beanstalk is well-suited for developers who want to deploy applications without managing the underlying infrastructure. It is an excellent choice for web developers who want to focus on writing code rather than managing the infrastructure.

3) **Serverless Deployment with AWS Lambda:**
   Description: AWS Lambda is a serverless computing service that allows users to run code without provisioning or managing servers. It automatically scales applications by running code in response to specific events or triggers.
   Use Case: Lambda is suitable for building serverless applications and for executing code in response to various events, such as changes to data in an Amazon S3 bucket, DynamoDB table updates, or HTTP requests via Amazon API Gateway. It is particularly beneficial for event-driven and microservices architectures where the focus is on individual functions rather than entire applications.

Each deployment mode provides distinct advantages and is designed to meet different application requirements, allowing users to choose the best fit for their specific use case and development needs.

Q.4 What are the 3 components of AWS Lambda?

Three main components of AWS Lambda include:

Lambda Function:

The Lambda function is the heart of the AWS Lambda service. It is the code that runs in response to events. Users can upload their code or write it directly in the AWS Management Console, using one of the supported programming languages such as Node.js, Python, Java, Go, and others. The function code is designed to be stateless and can be triggered by a variety of events or requests. Users can define the function's configuration, including memory size, timeouts, and environment variables.

Event Sources:

Event sources are the triggers that invoke the Lambda function. AWS Lambda can be triggered by various AWS services such as Amazon S3, Amazon DynamoDB, Amazon Kinesis, Amazon SQS, and others. Additionally, it can be integrated with custom event sources through Amazon API Gateway or other AWS services. These event sources enable users to create custom event-driven architectures that respond to specific actions or changes within their AWS environment.

Lambda Execution Environment:

The Lambda execution environment consists of the infrastructure and resources necessary to run the Lambda function. AWS manages the execution environment and automatically provisions and scales the infrastructure based on the incoming request volume. It includes the compute resources, networking, and security configurations needed to execute the function code securely and reliably. AWS Lambda handles the complexities of infrastructure management, allowing users to focus solely on writing the function code and defining the event sources.