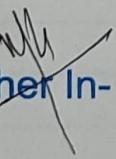


Thadomal Shahani Engineering College
Bandra (W.), Mumbai - 400 050.

CERTIFICATE

Certify that Mr./Miss Saish Bavalekar
of IT Department, Semester IV with
Roll No. 10 has completed a course of the necessary
experiments in the subject Adv Devops Lab under my
supervision in the **Thadomal Shahani Engineering College**
Laboratory in the year 2023 - 2024


Teacher In-Charge

Head of the Department

Date 21/10/23

Principal

CONTENTS

SR. NO.	EXPERIMENTS	PAGE NO.	DATE	TEACHERS SIGN.
1.	To understand the benefit of		17/7/23	7
	cloud infrastructure and setup			
	aws cloud9.			
2.	To build your application using		24/7/23	
	Aws Codebuild deploy on S3			
3.	To creating on S3 bucket using Aws		31/7/23	
	service and uploading a file into it			
4.	To understand the installation		7/8/23	
	process of Terraform			
5.	To understand the concept of		21/8/23	
	Terraform and use it to create			
	and instance using limit, plan,			
	apply and destroy command			
6.	To understand the concept of Sonar		28/8/23	
	cube & server scanner.			
7.	To learn how to use lambda		4/9/23	
	to run a simple program from			
	S3 bucket			
8.	To learn how to use lambda to		11/9/23	
	find the Content type of object			
	uploaded in S3 bucket.			
9.	To understand and install Nagios,		18/9/23	
	moniter host using Nagios			
10.	To complete study of Kubernetes		2/10/23	
11.	Theory Assignment 1			
12.	Theory Assignment 2			

Roll number:10

Name: Saish Bavalekar

Date: 23/7/23

ASSIGNMENT 1

AIM: To make an EC2 instance in AWS

THEORY:

EC2

- 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.
- You can scale the compute capacity up and down as per the computing requirement changes.

STEPS TO CREATE AN EC2 INSTANCE ON AWS:

1. Login to your aws console and go to EC2 service and click on Launch Instance

The screenshot shows the AWS EC2 Dashboard. On the left, there's a navigation sidebar with sections like EC2 Dashboard, Instances, Images, Elastic Block Store, Network & Security, and more. The main area is titled 'Resources' and shows various Amazon EC2 resources in the US East (N. Virginia) Region. It includes tables for Instances (running), Auto Scaling Groups, Dedicated Hosts, Elastic IPs, Instances, Key pairs, Load balancers, Placement groups, Security groups, Snapshots, and Volumes. Below this, a callout box suggests using the AWS Launch Wizard for Microsoft SQL Server Always On availability groups. The 'Launch instance' section contains a large orange 'Launch instance' button and a 'Migrate a server' button. To the right, there's a 'Service health' section showing the status as 'This service is operating normally'. The top right corner shows account attributes like 'Supported platforms', 'Default VPC', and 'Settings', and an 'Explore AWS' sidebar with tips for cost reduction and performance.

2. Name your EC2 instance

The screenshot shows the AWS EC2 'Launch an instance' wizard. The top navigation bar includes the AWS logo, 'Services' dropdown, search bar, and user 'Saish'. The main page title is 'EC2 > Instances > Launch an instance'. The current step is 'Launch an instance'.

Name and tags (Info)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name

Linux Instance Add additional tags

Application and OS Images (Amazon Machine Image) (Info)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux  macOS  Ubuntu  Windows  Red Hat 

Browse more AMIs 
Including AMIs from AWS, Marketplace and the Community

Summary

Number of instances [Info](#)

Software Image (AMI)
Amazon Linux 2023 AMI 2023.1.2... [read more](#)
ami-05548f9cecf47b442

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel **Launch instance** Review commands

3. Select AMI(Amazon Machine Image) you wish to proceed with. Here I've selected Amazon Linux image. Make sure the AMI is free tier.

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Quick Start



[Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI Free tier eligible

ami-05548f9cecf47b442 (64-bit (x86)) / ami-0fefbfbdaf373123d (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description
Amazon Linux 2023 AMI 2023.1.20230719.0 x86_64 HVM kernel-6.1

Architecture	AMI ID	Verified provider
64-bit (x86)	ami-05548f9cecf47b442	

4. Select the instance type. Here we have selected t2.micro which is free tier eligible. We are proceeding without using a key pair name. Create one for more security.

The screenshot shows the AWS Lambda 'Create Function' wizard. The 'Instance type' section is open, displaying the 't2.micro' option. It provides detailed pricing information for various operating systems. The 'Key pair (login)' section is also open, showing a note about using a key pair for secure connection and a dropdown menu for selecting a key pair.

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows pricing: 0.0162 USD per Hour
On-Demand SUSE pricing: 0.0116 USD per Hour
On-Demand RHEL pricing: 0.0716 USD per Hour
On-Demand Linux pricing: 0.0116 USD per Hour

Free tier eligible

All generations

Compare instance types

Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

Proceed without a key pair (Not recommended) Default value ▾

Create new key pair

- Add a security group or use a previously created security group and create the EC2 instance.

▼ Network settings [Info](#) [Edit](#)

Network [Info](#)
vpc-0d24a25c3852d6f8f

Subnet [Info](#)
No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)
Enable

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group Select existing security group

We'll create a new security group called '**launch-wizard-3**' with the following rules:

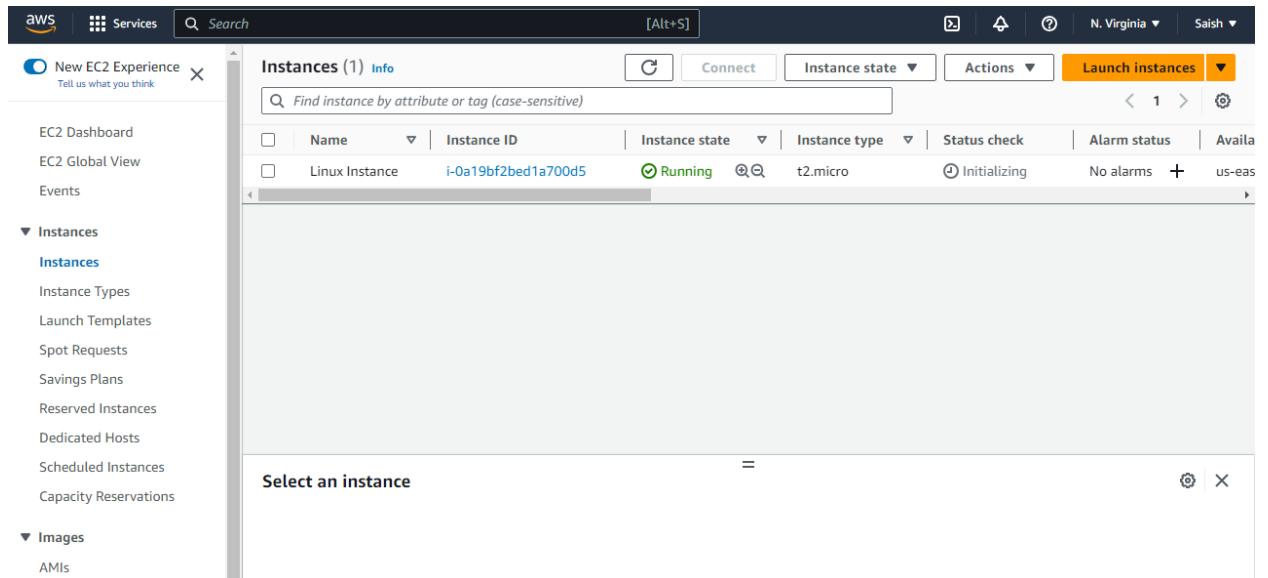
Allow SSH traffic from
Helps you connect to your instance Anywhere
0.0.0.0/0

Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. X

6. Launch the EC2 instance that has been created.



7. With the virtual linux machine at our service, we can run linux commands on the terminal.

```
(ec2-user@ip-172-31-20-119 ~)$ ifconfig
enx0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
inet 172.31.20.119 brd 172.31.20.255 netmask 255.255.240.0 broadcast 172.31.31.255
        inet6 fe80::fe80:1ff%enx0 brd fe80::f71f:21ff:fe80:1ff%enx0 scopeid 0x20<link>
          ether 0a:38:ec:71:f2:1b txqueuelen 1000 (Ethernet)
            RX packets 4431 bytes 17817271 (16.7 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 4431 bytes 316817 (309.3 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 brd 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 brd ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 12 bytes 1020 (1020.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 12 bytes 1020 (1020.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(ec2-user@ip-172-31-20-119 ~)$ echo hello
hello
(ec2-user@ip-172-31-20-119 ~)$
```

The terminal session shows the user connecting to an EC2 instance with IP 172.31.20.119. The user runs 'ifconfig' to view network interfaces, showing one interface (enx0) with IP 172.31.20.119 and another loopback interface (lo). The user then runs the 'echo' command to test the connection, outputting 'hello'.

8. Ensure to terminate the instance after use.

The screenshot shows the AWS EC2 Instances page. The main table lists one instance:

Name	Instance ID	Instance state
Linux Instance	i-0a19bf2bed1a700d5	Running

Below the table, there is a "Actions" dropdown menu with options: Stop instance, Start instance, Reboot instance, Hibernate instance, and Terminate instance. The "Terminate instance" option is highlighted.

9. After the instance has been terminated also delete the unnecessary security groups.

The screenshot shows the AWS Security Groups page. The main table lists five security groups:

Name	Security group ID	Security group name	VPC ID
-	sg-0ffb7f8c3a88d27e4	launch-wizard-1	vpc-0d24a2
-	sg-09cc4a861445af4e2	launch-wizard-2	vpc-0d24a2
-	sg-0a5c54783dd298890	default	vpc-0d24a2
-	sg-0d439fb3f39ca5		
-	sg-02176012b4f1ff		

Below the table, there is a "Actions" dropdown menu with options: Create security group, Export security groups to CSV, Export security groups inbound/outbound rules to CSV, View details, Edit inbound rules, Edit outbound rules, Manage tags, Manage stale rules, Copy to new security group, and Delete security groups. The "Delete security groups" option is highlighted.

LAB OUTCOME:

LO1 - To understand the benefits of Cloud9 infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration.

Saish Bavalekar

Roll Number:10

30/7/23

ASSIGNMENT 2

AIM: To create an AWS Cloud9 Environment.

THEORY:

AWS Cloud9

Cloud9 is virtual compiler provided by AWS. We can, code, build, test, debug and release software. AWS Cloud9 environment is a place where we can store our project files and where we run the tools to develop our application.

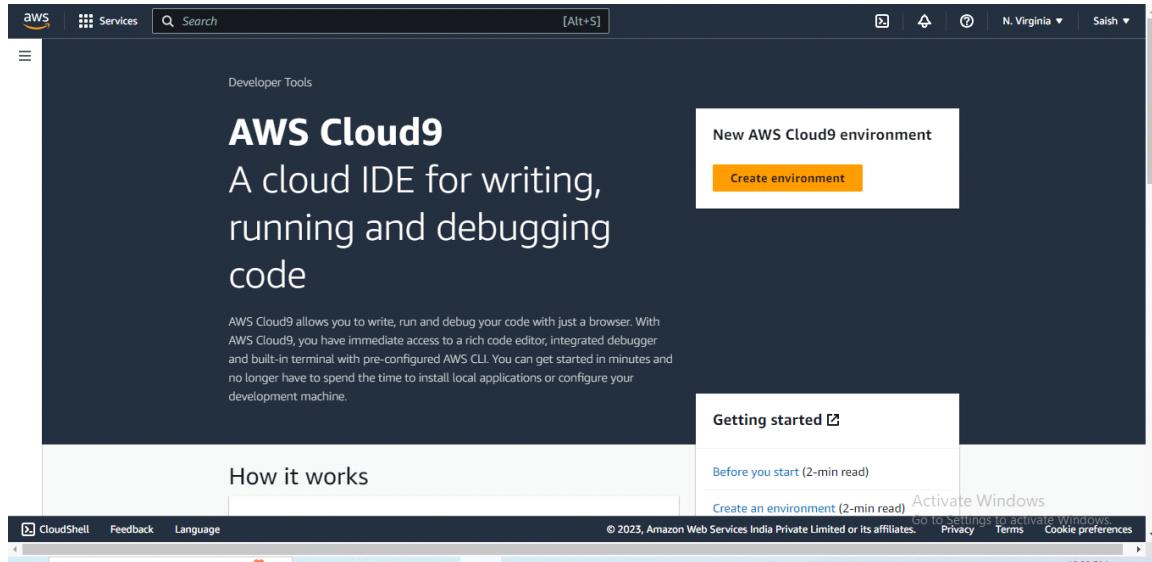
With Cloud9 Environment we can:

- Store your project files locally on the instance or server.
- Clone remote code repository into our environment.
- Work with a combination of local and cloned files in the environment.

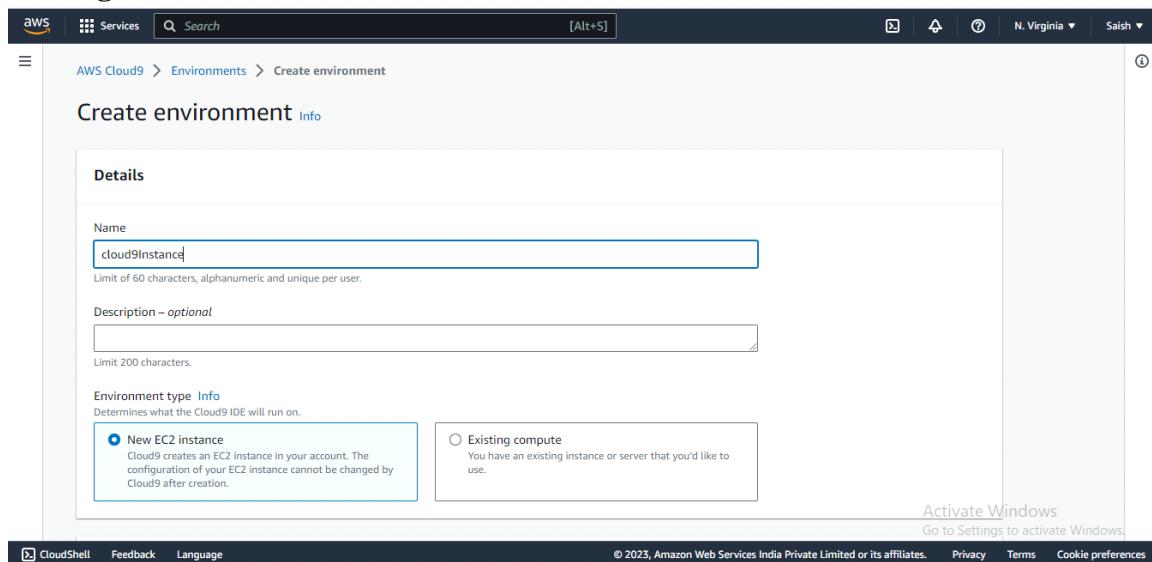
We can create and switch between multiple environments, with each environment set up for a specific development project. By storing the projects on cloud, our projects no longer need to be tied to a single computer or server setup. This enables you to do things such as easily switch between computers and more quickly onboard developers to our team.

Steps to Setup AWS Cloud9 Environment:

1. Login to your AWS console and search for Cloud9 and click on create environment.



2. Name your environment and select the environment type. We can opt to use an existing EC2 instance or create a new one.



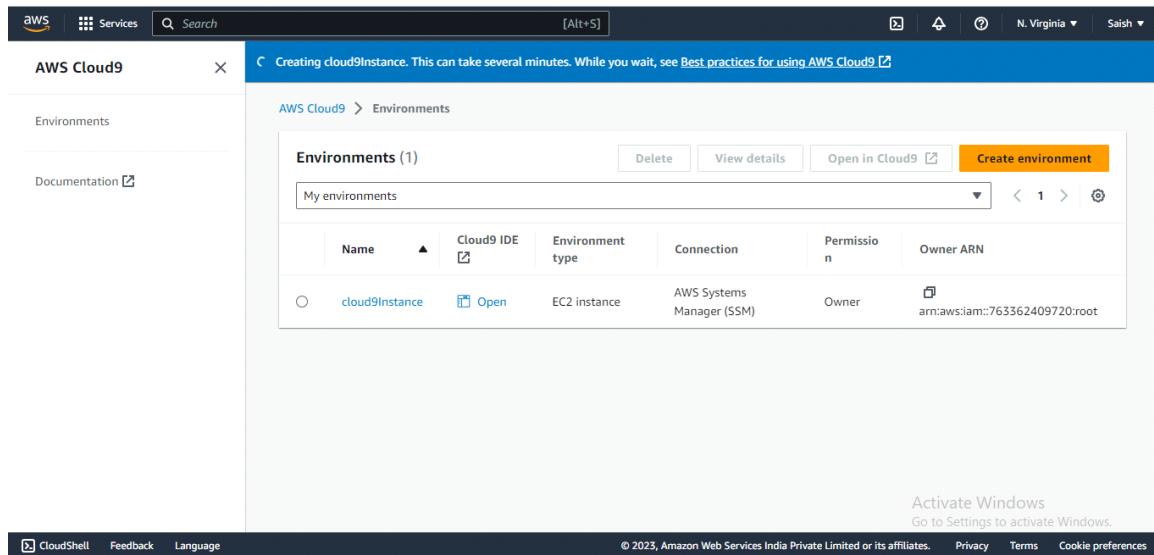
3. Select instance type as t2.micro which comes under free tier.

The screenshot shows the 'New EC2 instance' configuration page. At the top, there's a navigation bar with the AWS logo, 'Services', a search bar, and account information for 'N. Virginia' and 'Saish'. Below the navigation, the title 'New EC2 instance' is displayed. Under 'Instance type Info', it says 'The memory and CPU of the EC2 instance that will be created for Cloud9 to run on.' There are four options: 't2.micro (1 GiB RAM + 1 vCPU)' (selected), 't3.small (2 GiB RAM + 2 vCPU)', 'm5.large (8 GiB RAM + 2 vCPU)', and 'Additional instance types'. The 't2.micro' option is highlighted with a blue border. Under 'Platform Info', it says 'This will be installed on your EC2 instance. We recommend Amazon Linux 2.' A dropdown menu shows 'Amazon Linux 2'. Under 'Timeout', it says 'How long Cloud9 can be inactive (no user input) before auto-hibernating. This helps prevent unnecessary charges.' A dropdown menu shows '30 minutes'. In the bottom right corner, there's a link to 'Activate Windows' with the subtext 'Go to Settings to activate Windows'.

4. Configure the network settings as per need. We'll proceed with the default settings.

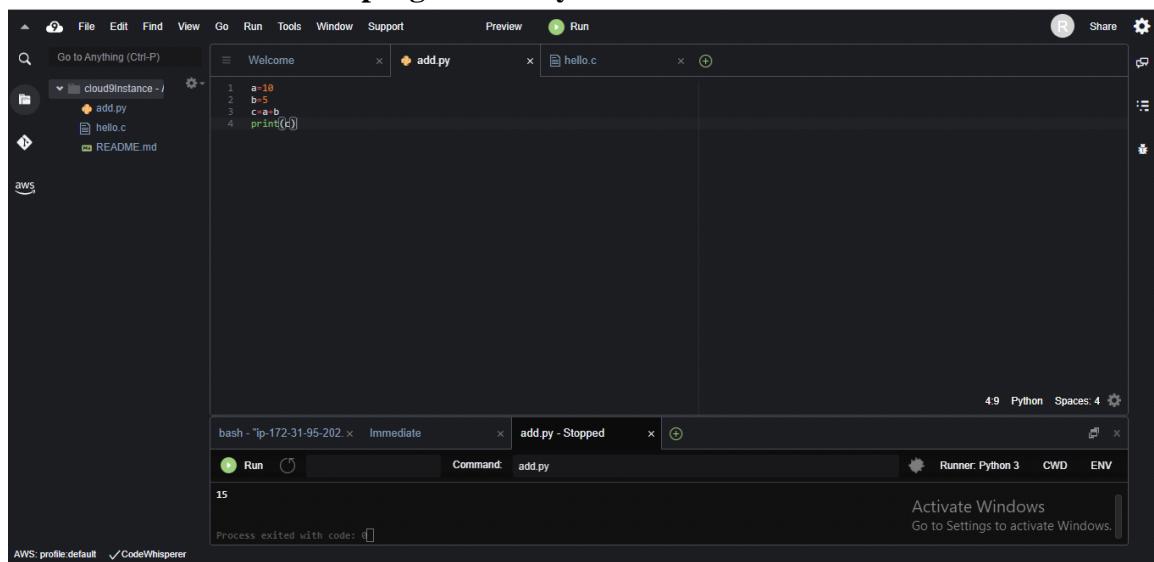
The screenshot shows the 'Network settings' configuration page. At the top, there's a navigation bar with the AWS logo, 'Services', a search bar, and account information for 'N. Virginia' and 'Saish'. Below the navigation, the title 'Network settings' is displayed. Under 'Connection', it says 'How your environment is accessed.' There are two options: 'AWS Systems Manager (SSM)' (selected) and 'Secure Shell (SSH)'. The 'AWS Systems Manager (SSM)' option is highlighted with a blue border. Below this, there's a section for 'VPC settings' with a 'Info' link. Underneath, there's a section for 'Tags – optional' with a 'Info' link. A note at the bottom states: 'The following IAM resources will be created in your account' with a list of items: 'AWSServiceRoleForAWSCloud9', 'AWSCloud9SSMAccessRole', and 'AWSCloud9SSMInstanceProfile'. The 'AWSCloud9SSMInstanceProfile' item has a 'Find out more' link. In the bottom right corner, there's a link to 'Activate Windows' with the subtext 'Go to Settings to activate Windows'.

5. After Cloud9 environment has been created Open it from the Cloud9 Dashboard.



The screenshot shows the AWS Cloud9 dashboard. On the left, there's a sidebar with 'AWS Cloud9' and 'Environments'. The main area is titled 'Creating cloud9Instance. This can take several minutes. While you wait, see Best practices for using AWS Cloud9'. Below this, it says 'AWS Cloud9 > Environments'. A table titled 'Environments (1)' lists one environment: 'cloud9Instance' (Cloud9 IDE, EC2 instance, AWS Systems Manager (SSM), Owner, arn:aws:iam::763362409720:root). There are buttons for 'Delete', 'View details', 'Open in Cloud9', and 'Create environment'. At the bottom, there are links for 'CloudShell', 'Feedback', 'Language', and copyright information: '© 2023, Amazon Web Services India Private Limited or its affiliates.' followed by 'Privacy', 'Terms', and 'Cookie preferences'.

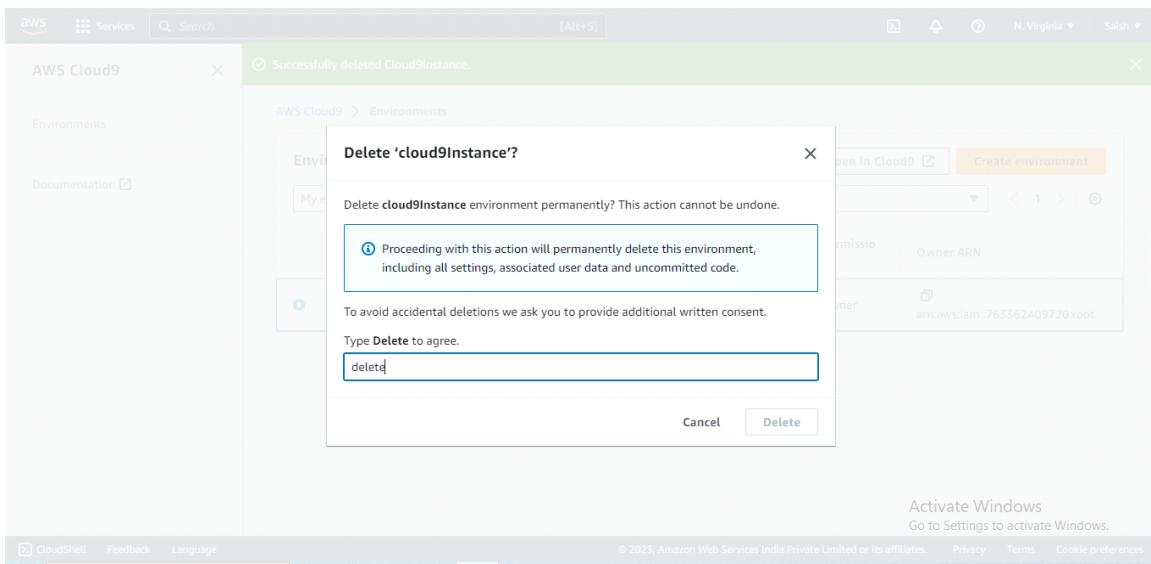
6. Cloud9 Environment acts as a virtual compiler. Using the Cloud9 Environment we have executed some basic programs in Python and C.

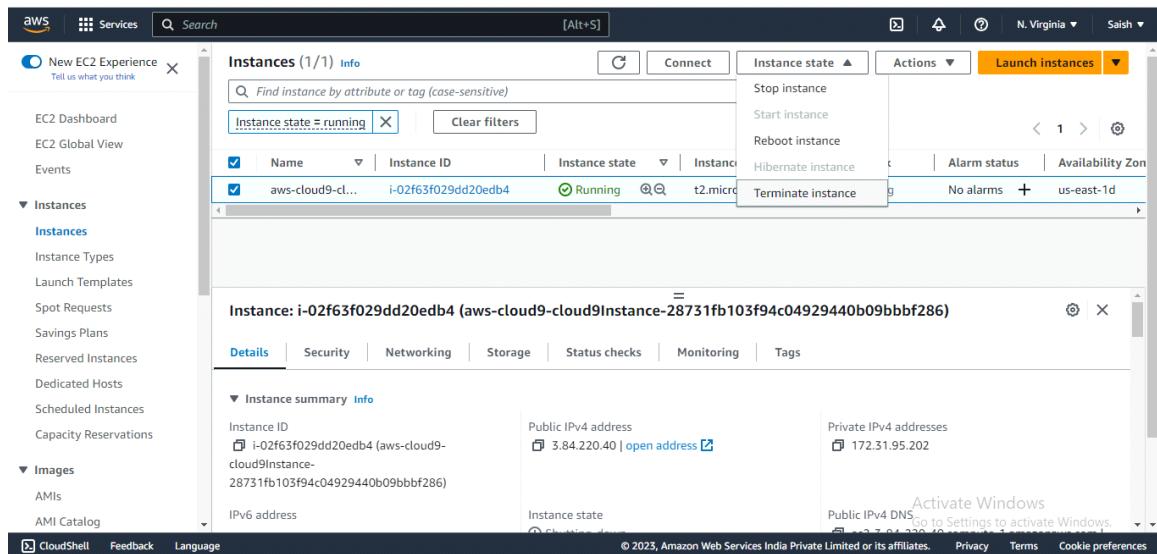


The screenshot shows the AWS Cloud9 IDE interface. On the left, there's a file explorer with files: 'add.py', 'hello.c', and 'README.md'. In the center, there are two code editors: one for 'add.py' containing the code 'a=10', 'b=5', 'c=a+b', and 'print(c)'; and another for 'hello.c'. At the bottom, there's a terminal window showing the output of a Python command: 'Process exited with code: 0'. The terminal also shows the command 'Command: add.py'. The status bar at the bottom indicates '4.9 Python Spaces: 4' and 'Activate Windows Go to Settings to activate Windows.'

A screenshot of a terminal window titled "hello.c - Stopped". The window shows the command "Command: hello.c" and the output "Running /home/cc2-user/environment/hello.c" followed by "Hello World". Below the terminal, a status bar indicates "Process exited with code: 0".

7. Delete the Cloud9 Environment. Also ensure to terminate the EC2 instance after use.





LAB OUTCOME:

L01-To understand the benefits of Cloud9 infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration.

ASSIGNMENT 3

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

LO1: To understand the fundamentals of Cloud Computing and be fully proficient with Cloud based DevOps solution deployment options to meet your business requirements.

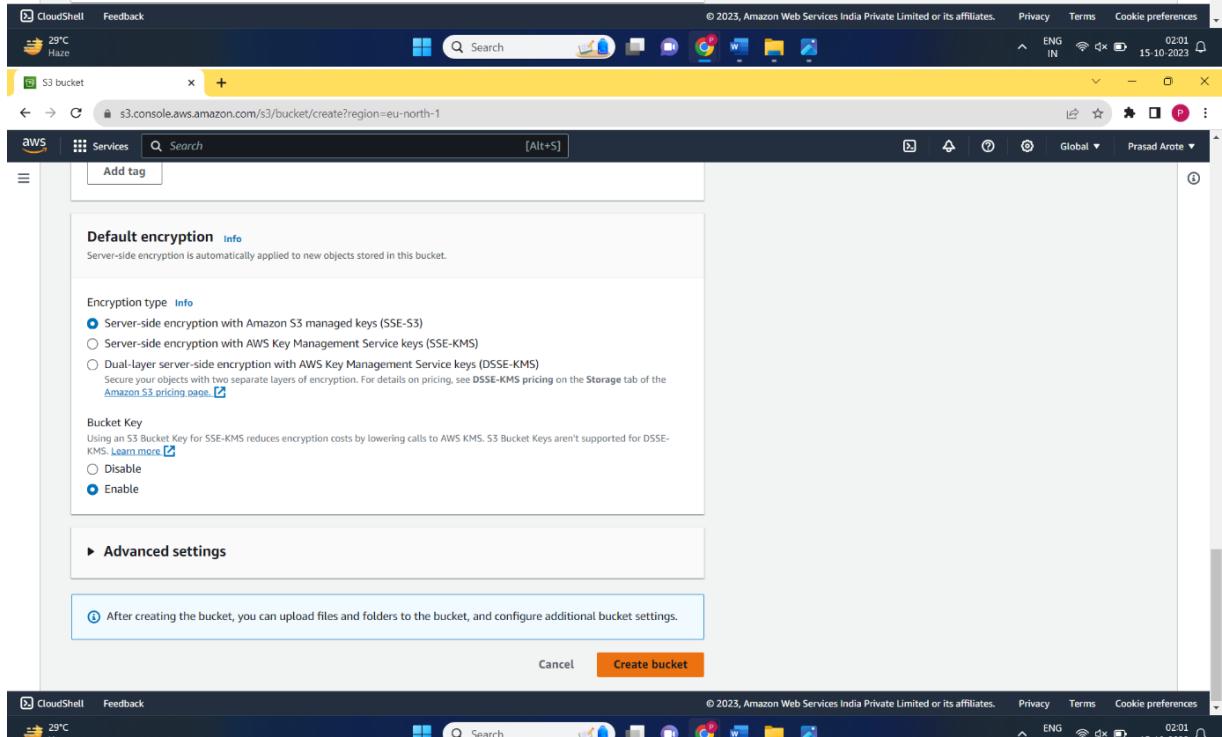
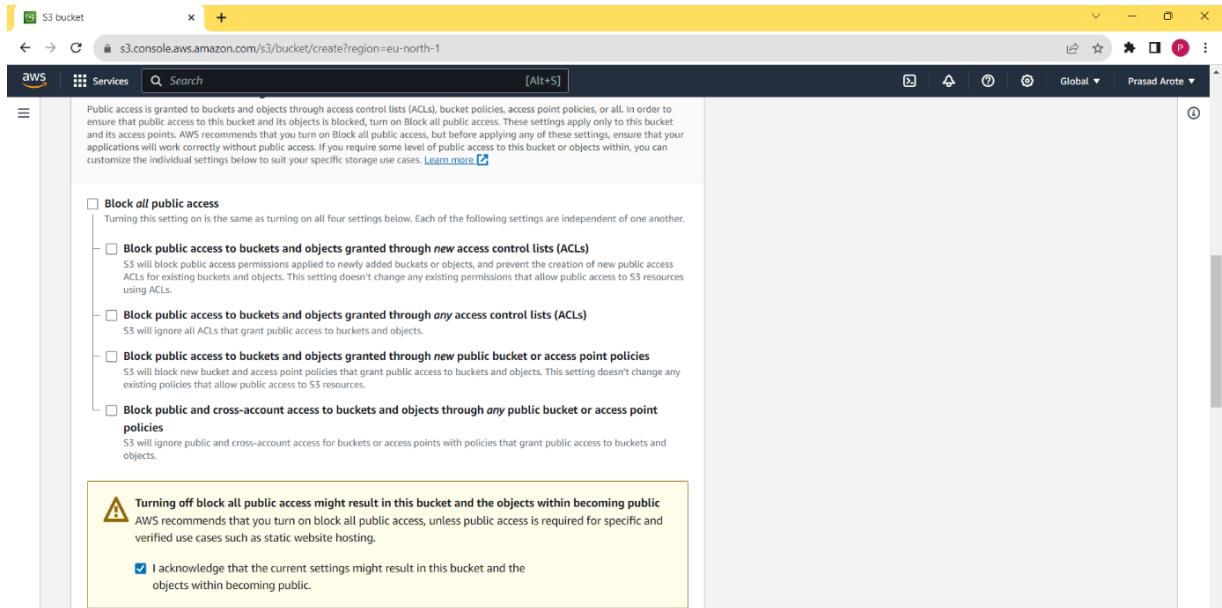
THEORY:

1. Create a S3 bucket.

The screenshot shows the AWS S3 Management Console. At the top, there's a banner for 'Amazon S3' with the tagline 'Store and retrieve any amount of data from anywhere'. Below this, a 'How it works' section features a video thumbnail titled 'Introduction to Amazon S3'. To the right, there are sections for 'Pricing' (no minimum fees) and 'Resources'.

A prominent 'Create a bucket' modal is open on the right side. It contains instructions: 'Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.' It has a large orange 'Create bucket' button.

Below the main page, a separate window shows the 'Create bucket' wizard. The first step, 'General configuration', is visible. It asks for a 'Bucket name' (input field containing 'prasad-website'), 'AWS Region' (set to 'Europe (Stockholm) eu-north-1'), and provides an optional 'Copy settings from existing bucket - optional' section with a 'Choose bucket' button. The second step, 'Object Ownership', is partially visible below it.



2. Upload the files of web application.

Screenshot of the AWS S3 Management Console showing the upload process for a website.

The browser window shows the URL: s3.console.aws.amazon.com/s3/upload/prasad-website?region=eu-north-1

Upload Info

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

Files and folders (0)

All files and folders in this table will be uploaded.

Name	Folder	Type	Size
No files or folders			

You have not chosen any files or folders to upload.

Destination

Destination

CloudShell Feedback © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences ENG IN 02:01 15-10-2023

Upload Info

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add Files** or **Add folder**.

Files and folders (24 Total, 89.5 KB)

All files and folders in this table will be uploaded.

Name	Folder	Type	Size
Bun 1.svg	-	image/svg+xml	865.0 B
Bun 1@2x.png	-	image/png	8.6 KB
Cheese.svg	-	image/svg+xml	619.0 B
Cheese@2x.png	-	image/png	1.4 KB
Lettuce.svg	-	image/svg+xml	629.0 B
Lettuce@2x.png	-	image/png	2.4 KB
Onion.svg	-	image/svg+xml	831.0 B
Onion@2x.png	-	image/png	2.8 KB
Patty.svg	-	image/svg+xml	639.0 B
Patty@2x.png	-	image/png	3.9 KB

Upload succeeded
View details below.

Upload: status

The information below will no longer be available after you navigate away from this page.

Summary

Destination	Succeeded	Failed
s3://prasad-website	24 files, 89.5 KB (100.00%)	0 files, 0 B (0%)

Files and folders (24 Total, 89.5 KB)

Name	Folder	Type	Size	Status	Error
Bun 1.svg	-	image/svg+xml	865.0 B	Success	
Bun 1@2x.png	-	image/png	8.6 KB	Success	
Cheese.svg	-	image/svg+xml	619.0 B	Success	
Cheese@2x.png	-	image/png	1.4 KB	Success	
Lettuce.svg	-	image/svg+xml	629.0 B	Success	
Lettuce@2x.png	-	image/png	2.4 KB	Success	
Onion.svg	-	image/svg+xml	831.0 B	Success	
Onion@2x.png	-	image/png	2.8 KB	Success	
Patty.svg	-	image/svg+xml	639.0 B	Success	
Patty@2x.png	-	image/png	3.9 KB	Success	

CloudShell Feedback © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences ENG IN 02:09 15-10-2023

3. Enable Static website hosting

The screenshot shows the AWS S3 console with the URL s3.console.aws.amazon.com/s3/bucket/prasad-website/property/website/edit?region=eu-north-1. The page is titled 'Edit static website hosting'. It has two main sections: 'Static website hosting' and 'Hosting type'.

Static website hosting: A note says 'Use this bucket to host a website or redirect requests.' with a 'Learn more' link. Below are radio buttons for 'Disable' (unchecked) and 'Enable' (checked). A note below says: 'For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see [Using Amazon S3 Block Public Access](#)'.

Hosting type: A radio button for 'Host a static website' (checked) is selected. It says 'Use the bucket endpoint as the web address.' with a 'Learn more' link. Another radio button for 'Redirect requests for an object' is unchecked. A note below says: 'Redirect requests to another bucket or domain.' with a 'Learn more' link.

Index document: A note says 'Specify the home or default page of the website.' with a 'Learn more' link. A text input field contains 'index.html'.

Error document - optional: A note says 'This is returned when an error occurs.' with a 'Learn more' link. A text input field contains 'error.html'.

The bottom of the page includes standard AWS navigation links like CloudShell, Feedback, and various icons. The status bar shows the date and time as 15-10-2023 02:11.

This screenshot is identical to the one above, showing the 'Edit static website hosting' configuration page for the 'prasad-website' bucket. The content, including the 'Static website hosting' section, 'Hosting type' section, 'Index document' field, 'Error document' field, and the note about Block Public Access, are all the same. The scroll position is lower than in the first screenshot.

prasad-website - S3 bucket

Successfully edited static website hosting.

Amazon S3 > Buckets > prasad-website

prasad-website [Info](#)

Objects [Properties](#) Permissions Metrics Management Access Points

Bucket overview

AWS Region	Amazon Resource Name (ARN)	Creation date
Europe (Stockholm) eu-north-1	arn:aws:s3:::prasad-website	October 15, 2023, 02:01:26 (UTC+05:30)

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

[Edit](#)

Bucket Versioning
Disabled
Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API.

CloudShell Feedback 28°C Haze Search Privacy Terms Cookie preferences © 2023, Amazon Web Services India Private Limited or its affiliates ENG IN 02:21 15-10-2023

5. Change the Bucket Policy

prasad-website - S3 bucket

s3.console.aws.amazon.com/s3/bucket/prasad-website/property/policy/edit?region=eu-north-1

Amazon S3 > Buckets > prasad-website > Edit bucket policy

Edit bucket policy [Info](#)

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

[Policy examples](#) [Policy generator](#)

Bucket ARN
arn:aws:s3:::prasad-website

Policy

1 | [Edit statement](#)

Select a statement

Select an existing statement in the policy or add a new statement.

+ Add new statement

CloudShell Feedback 28°C Haze Search Privacy Terms Cookie preferences © 2023, Amazon Web Services India Private Limited or its affiliates ENG IN 02:21 15-10-2023

AWS Policy Generator

The AWS Policy Generator is a tool that enables you to create policies that control access to Amazon Web Services (AWS) products and resources. For more information about creating policies, see key concepts in Using AWS Identity and Access Management. Here are sample policies.

Step 1: Select Policy Type

A Policy is a container for permissions. The different types of policies you can create are an IAM Policy, an S3 Bucket Policy, an SNS Topic Policy, a VPC Endpoint Policy, and an SQS Queue Policy.

Select Type of Policy S3 Bucket Policy IAM Policy SNS Topic Policy VPC Endpoint Policy SQS Queue Policy

Step 2: Add Statement(s)

A statement is the formal description of a single permission. See a description of elements that you can use in statements.

Effect Allow Deny

Principal

Use a comma to separate multiple values.

AWS Service All Services (*)

Actions Action(s) Selected All Actions (*)

Amazon Resource Name (ARN)

ARN should follow the following format: arn:aws:s3:::\${BucketName}/\${KeyName}.
Use a comma to separate multiple values.

Add Conditions (Optional)

Add Statement

You added the following statements. Click the button below to Generate a policy.

Principal(s)	Effect	Action	Resource	Conditions
* *	Allow	s3:GetObject	arn:aws:s3:::prasad-website	None

Step 3: Generate Policy

A policy is a document (written in the Access Policy Language) that acts as a container for one or more statements.

Generate Policy **Start Over**

This AWS Policy Generator is provided for informational purposes only; you are still responsible for your use of Amazon Web Services technologies and ensuring that your use is in compliance with all applicable terms and conditions. This AWS Policy Generator is provided as is without warranty of any kind, whether express, implied, or statutory. This AWS Policy Generator does not modify the applicable terms and conditions governing your use of Amazon Web Services technologies.

©2010, Amazon Web Services LLC or its affiliates. All rights reserved.
An [amazon.com](#) company

The screenshot shows the AWS Policy Generator interface. At the top, there are tabs for 'prasad-website - S3 bucket', 'AWS Policy Generator', and 'prasad-website.s3.eu-north-1.amazonaws.com'. The main area has a title 'Policy JSON Document' with a note: 'Click below to edit. To save the policy, copy the text below to a text editor. Changes made below will **not** be reflected in the policy generator tool.' Below this is a code editor containing the following JSON policy:

```
[{"Id": "Policy1697316791653", "Version": "2012-10-17", "Statement": [ {"Sid": "Stmt1697316788348", "Action": [ "s3:GetObject" ], "Effect": "Allow", "Resource": "arn:aws:s3:::prasad-website/*", "Principal": "*" } ]}
```

At the bottom of the editor, a message reads: 'This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring that your use is in compliance with all applicable terms and conditions. This AWS Policy Generator is provided as-is without warranty of any kind, whether express, implied, or statutory. This AWS Policy Generator does not modify the applicable terms and conditions governing your use of Amazon Web Services technologies.' A 'Close' button is at the bottom right.

The screenshot shows the AWS S3 console with a tab for 'prasad-website - S3 bucket'. The main area is titled 'Policy' and displays the same JSON policy as the previous screenshot. On the right side, there is a sidebar titled 'Edit statement' with the sub-section 'Select a statement'. It contains the text: 'Select an existing statement in the policy or add a new statement.' and a button '+ Add new statement'.

prasad-website - S3 bucket | AWS Policy Generator | prasad-website.s3.eu-north-1.amazonaws.com | +

Services Search [Alt+S]

Successfully edited bucket policy.

Amazon S3 > Buckets > prasad-website

prasad-website info

Objects Properties **Permissions** Metrics Management Access Points

Permissions overview

Access
Objects can be public

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Edit

Block all public access
Off

► Individual Block Public Access settings for this bucket

CloudShell Feedback © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences 28°C Haze ENG IN 02:24 15-10-2023

6. Now open the link (given in the bucket below) in browser and you can see the static website hosted.

prasad-website - S3 bucket | BRRRGRRR | BRRRGRRR | +

Not secure | prasad-website.s3-website.eu-north-1.amazonaws.com

HUNGRY? GRAB A BRRRGRRR

Ingredients

- Patty
- Cheese
- Tomatoes
- Onions
- Lettuce

PRICES

Whole wheat bun	20
Patty	80
Onions	20
Tomatoes	20
Lettuce	20
Cheese slice	10

Choose what goes into your burger

Patty Cheese Tomatoes Onions Lettuce

Current Order Total
INR 170
To Pay

CloudShell Feedback © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences 28°C Haze ENG IN 02:25 15-10-2023

CONCLUSION:

Here we studied to host a static website on S3 bucket.

ASSIGNMENT 4

AIM: To understand terraform lifecycle, core concepts/terminologies and install it.

LO3: To apply best practices for managing infrastructure as code environments and use terraform to define and deploy cloud infrastructure.

THEORY:

Terraform is one of the most popular Infrastructure-as-code (IaC) tool, used by DevOps teams to automate infrastructure tasks. It is used to automate the provisioning of your cloud resources. Terraform is an open-source, cloud-agnostic provisioning tool developed by HashiCorp and written in GO language.

Benefits of Terraform:

- Does orchestration, not just configuration management.
- Supports multiple providers such as AWS, Azure, Oracle, GCP, and many more.
- Provide immutable infrastructure where configuration changes smoothly.
- Uses easy to understand language, HCL (HashiCorp configuration language).
- Easily portable to any other provider.

TERRAFORM LIFECYCLE

Terraform lifecycle consists of – **init**, **plan**, **apply**, and **destroy**.



1. **Terraform init** initializes the (local) Terraform environment. Usually executed only once per session.
2. **Terraform plan** compares the Terraform state with the as-is state in the cloud, build and display an execution plan. This does not change the deployment (read-only).
3. **Terraform apply** executes the plan. This potentially changes the deployment.
4. **Terraform destroy** deletes all resources that are governed by this specific terraform environment.

TERRAFORM CORE CONCEPTS/TERMINOLOGIES

1. Variables: Terraform has input and output variables, it is a key-value pair. Input variables are used as parameters to input values at run time to customize our deployments. Output variables are return values of a terraform module that can be used by other configurations.
2. Provider: Terraform users provision their infrastructure on the major cloud providers such as AWS, Azure, OCI, and others. A provider is a plugin that interacts with the various APIs required to create, update, and delete various resources.
3. Module: Any set of Terraform configuration files in a folder is a module. Every Terraform configuration has at least one module, known as its root module.
4. State: Terraform records information about what infrastructure is created in a Terraform state file. With the state file, Terraform is able to find the resources it created previously, supposed to manage and update them accordingly.
5. Resources: Cloud Providers provides various services in their offerings, they are referenced as Resources in Terraform. Terraform resources can be anything from compute instances, virtual networks to higher-level components such as DNS records. Each resource has its own attributes to define that resource.
6. Data Source: Data source performs a read-only operation. It allows data to be fetched or computed from resources/entities that are not defined or managed by Terraform or the current Terraform configuration.
7. Plan: It is one of the stages in the Terraform lifecycle where it determines what needs to be created, updated, or destroyed to move from the real/current state of the infrastructure to the desired state.
8. Apply: It is one of the stages in the Terraform lifecycle where it applies the changes real/current state of the infrastructure in order to achieve the desired state.

INSTALLATION:

- 1) Download Terraform

To install Terraform, First Download the Terraform Cli Utility for windows from terraforms official website

website: <https://www.terraform.io/downloads.html>

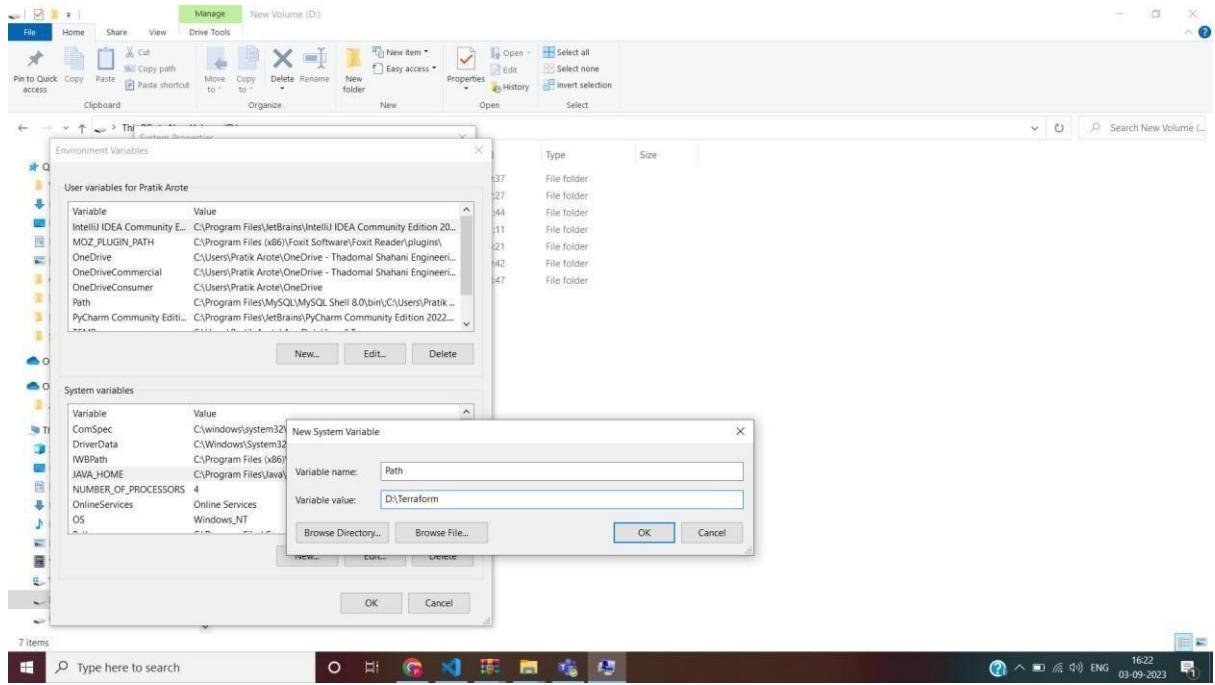
Select the Operating System Windows followed by either 32bit or 64 bit based on your OS type.

The screenshot shows the HashiCorp Terraform website's download page. At the top, it says "Install or update to v1.5.6 (latest version) of Terraform to get started." Below this, there are tabs for Operating System (macOS, Windows, Linux, FreeBSD, OpenBSD, Solaris), with Windows selected. Under "Binary download for Windows", there are two options: "386" (Version: 1.5.6) and "AMD64" (Version: 1.5.6), each with a "Download" button. Below these, there is a "Release information" section with a "Changelog" link (Version: 1.5.6) and a "GitHub" link. A "Notes" section follows. To the right, there are boxes for "About Terraform", "Featured docs" (Introduction to Terraform, Configuration Language, Terraform CLI, Terraform Cloud, Provider Use), and "Terraform Cloud" (Automate your infrastructure provisioning at any scale, Try Terraform Cloud for free). At the bottom, a cookie consent message is present.

2. Extract the downloaded setup file Terraform.exe in C:\Terraform Directory

The screenshot shows a Windows File Explorer window with a ZIP file named "terraform_1.5.6_windows_amd64.zip" selected. An "Extraction path and options" dialog box is open, showing the destination path as "C:\Terraform". Under "Update mode", "Extract and replace files" is selected. Under "Overwrite mode", "Ask before overwrit..." is selected. Other options like "Extract and update files", "Fresh existing files only", "Overwrite without prompt", "Skip existing files", and "Rename automatically" are available. There are also "Miscellaneous" options for "Keep broken files" and "Display files in Explorer". The "OK" button is highlighted in red. The background shows a desktop with various icons and a taskbar at the bottom.

3. Set the System path for Terraform in Environment Variables.



4. Open PowerShell with Admin Access. Open Terraform in PowerShell and check its functionality.

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Pratik Arote> cd D:
PS D:\> Terraform
Usage: terraform [global options] <subcommand> [args]

The available commands for execution are listed below.
The primary workflow commands are given first, followed by
less common or more advanced commands.

Main commands:
  init      Prepare your working directory for other commands
  validate  Validate the configuration is valid
  plan     Show changes required by the current configuration
  apply    Create or update infrastructure
  destroy   Destroy previously-created infrastructure

All other commands:
  console   Try Terraform expressions at an interactive command prompt
  fmt       Reformat your configuration in the standard style
  force-unlock Release a stuck lock on the current workspace
  get       Install or upgrade remote Terraform modules
  graph    Generate a dependency graph of the steps in an operation
  import   Associate existing resources with Terraform
  login    Obtain and save credentials for a remote host
  logout   Remove locally-stored credentials for a remote host
  metadata Metadata related commands
  output   Show output values from your root module
  provision Show provisions defined for this configuration
  refresh  Update the state to match remote systems
  show     Show the current state or a saved plan
  state    Advanced state management
  taint    Mark a resource instance as no longer functional
  test     Execute tests for individual integration testing
  untaint  Remove the "tainted" state from a resource instance
  version  Show the current Terraform version
  workspace Workspace management

Global options (use these before the subcommand, if any):
  -chdir=DIR  Switch to a different working directory before executing the
             given subcommand.
  -help      Show this help output, or the help for a specified subcommand.
  -version   An alias for the "version" subcommand.

PS D:\>

```

CONCLUSION:

Here, we studied the about the terraform lifecycle and terminologies/concepts of terraform and installed it on our system.

ASSIGNMENT 5

AIM: To Build, change, and destroy AWS infrastructure Using Terraform.

LO1: To understand the fundamentals of Cloud Computing and be fully proficient with Cloud based DevOps solution deployment options to meet your business requirements.

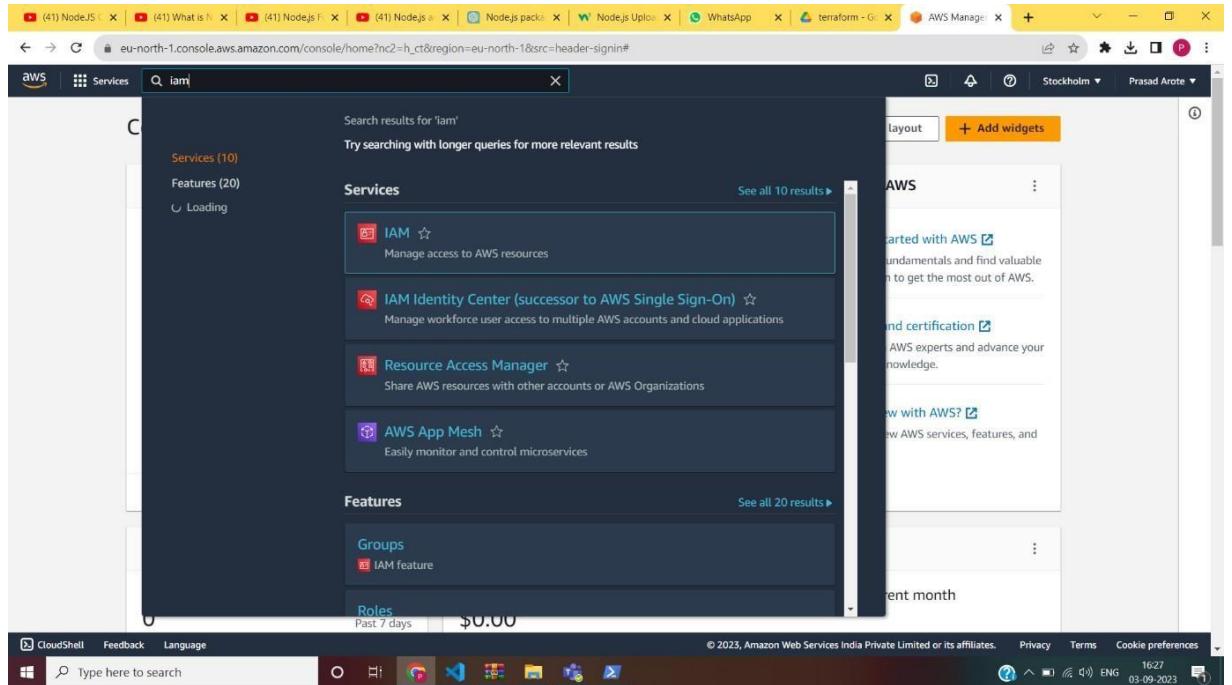
LO5: To use Continuous Monitoring Tools to resolve any system errors (low memory, unreachable server etc.) before they have any negative impact on the business productivity.

Theory:

- 1) Make dir Terraform Scripts Open aws.amazon.com

Login to your account

Search IAM



- 2) Click on Users (on the LHS)

The screenshot shows the AWS IAM service interface. The left sidebar has sections for Identity and Access Management (IAM), Dashboard, Access management (User groups, Users, Roles, Policies, Identity providers, Account settings), Access reports (Access analyzer, Archive rules, Analyzers, Settings, Credential report, Organization activity), and Service control policies (SCPs). The main content area is titled 'Users (0) info' and contains a message: 'An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.' A search bar and a table header ('User name, Path, Group, Last activity') are present. Below the table, it says 'No resources to display'. At the bottom right of the main area are 'Delete' and 'Create user' buttons. The status bar at the bottom shows the URL 'us-east-1.console.aws.amazon.com/iamv2/home?region=eu-north-1#users', the date '03-09-2023', and the time '16:27'.

3) Click Add users

The screenshot shows the 'Create user' wizard, Step 1: Specify user details. The left sidebar shows 'Step 1: Specify user details', 'Step 2: Set permissions', and 'Step 3: Review and create'. The main content area is titled 'Specify user details' and contains a 'User details' section. It includes a 'User name' input field with 'prasad' typed in, a note about character restrictions, and an optional checkbox for 'Provide user access to the AWS Management Console'. A note at the bottom explains how to generate programmatic access keys. At the bottom right are 'Cancel' and 'Next' buttons. The status bar at the bottom shows the URL 'us-east-1.console.aws.amazon.com/iamv2/home?region=eu-north-1#users/create', the date '03-09-2023', and the time '16:28'.

4) Set Permissions -> AmazonEC2FullAccess

Screenshot of the AWS IAM 'Create user' wizard, Step 2: Set permissions.

The 'Permissions options' section shows three choices:

- 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.

The 'Permissions policies' list (1127) is displayed, filtered by Type (All types). Policies listed include:

Policy name	Type	Attached entities
AccessAnalyzerServiceRolePolicy	AWS managed	0
AdministratorAccess	AWS managed - job function	0
AdministratorAccess-Amplify	AWS managed	0

In the second screenshot, the 'AmazonEC2FullAccess' policy is selected (indicated by a checked checkbox).

The URL shown in the browser is <https://us-east-1.console.aws.amazon.com/iamv2/home?region=eu-north-1#/users/create>.

5) Create User

The screenshot shows the AWS IAM console with a success message: "User created successfully". The message states: "You can view and download the user's password and email instructions for signing in to the AWS Management Console." Below this, the "Users" table is displayed, showing one user named "prasad". The table includes columns for User name, Path, Group, Last activity, MFA, Password age, and Console last sign-in.

6) Create access key

The screenshot shows the "Create access key" step in the AWS IAM console. It displays "Access key best practices & alternatives" with a note: "Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives." A list of use cases is shown, with "Command Line Interface (CLI)" selected:

- Command Line Interface (CLI)
You plan to use this access key to enable the AWS CLI to access your AWS account.
- Local code
You plan to use this access key to enable application code in a local development environment to access your AWS account.
- Application running on an AWS compute service
You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.
- Third-party service
You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.
- Application running outside AWS
You plan to use this access key to enable an application running on an on-premises host, or to use a local AWS client or third-party AWS plugin.
- Other

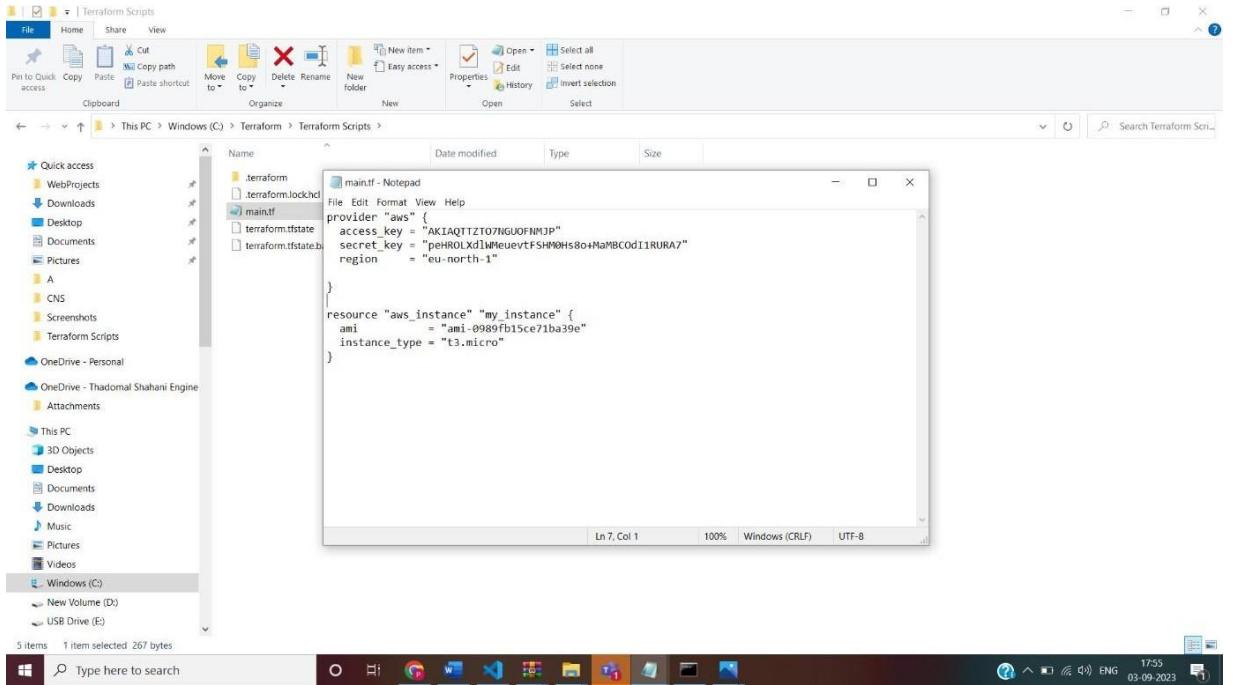
7) Download the .csv file

The screenshot shows the AWS IAM 'Create access key' page. A green header bar at the top says 'Access key created'. Below it, a message states: 'This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.' The main section is titled 'Retrieve access keys' and shows a table with one row. The table has two columns: 'Access key' and 'Secret access key'. The 'Access key' column contains the value 'AKIAQTTZTO7NGUOFNMJP'. The 'Secret access key' column is partially visible as '***** Show'. Below the table is a section titled 'Access key best practices' with a list of four items: 'Never store your access key in plain text, in a code repository, or in code.', 'Disable or delete access key when no longer needed.', 'Enable least-privilege permissions.', and 'Rotate access keys regularly.' At the bottom right of the page are 'Download .csv file' and 'Done' buttons.

8) Open EC2 Instances and Copy the AMI ID of any one of them.

The screenshot shows the AWS EC2 'Launch instances' page. The left sidebar lists 'Amazon Machine Image (AMI)' and 'Ubuntu Server 22.04 LTS (HVM), SSD Volume Type' is selected. The right side shows a 'Summary' section with a 'Number of instances' input field set to '1'. Below it are fields for 'Software Image (AMI)', 'Virtual server type (instance type)', 'Firewall (security group)', and 'Storage (volumes)'. A modal window is open in the center, titled 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per year'. It includes a 'Cancel' button and a large 'Launch instance' button. The bottom of the screen shows the Windows taskbar.

9) Configure the main.tf file



- 10) Run the commands in cmd -> `terraform init` , `terraform validate` , `terraform plan`, `terraform apply` to create EC2 instance using terraform.

```

C:\Windows\System32\cmd.exe
C:\Terraform\Terraform Scripts>terraform init
Initializing the backend...
Initializing provider plugins...
  - Finding latest version of hashicorp/aws...
  - Installing hashicorp/aws v5.15.0...
  - Installed hashicorp/aws v5.15.0 (signed by HashiCorp)

Terraform has created a lock file .terraform.lock.hcl to record the provider
selection 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,
run this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

C:\Terraform\Terraform Scripts>terraform validate
Success! The configuration is valid.

C:\Terraform\Terraform Scripts>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.my_instance will be created
  + resource "aws_instance" "my_instance" {
      + ami           = "ami-0989fb15ce71ba39e"
      + subnet_id    = "(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_stop = "(known after apply)"
      + enable_ip_termination = "(known after apply)"
      + ephemeral_optimized = "(known after apply)"
      + get_password_data = false
      + host_id       = "(known after apply)"
      + host_resource_group_arn = "(known after apply)"
    }

```

```
C:\Windows\System32\cmd.exe
Terraform will detect and report changes to your configuration
when you run "terraform apply" or similar commands. If you
forget, other commands will detect it and remind you to do so if necessary.

C:\Terraform\Terraform Scripts>terraform validate
Success! The configuration is valid.

C:\Terraform\Terraform Scripts>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.my_instance will be created
+ resource "aws_instance" "my_instance" {
  + ami                                = "ami-0989fb15ce71ba39e"
  + arn                                = (known after apply)
  + associate_public_ip_address        = (known after apply)
  + available_to_zones                = (known after apply)
  + cpu_core_count                    = (known after apply)
  + cpu_threads_per_core              = (known after apply)
  + disable_api_stop                 = (known after apply)
  + disable_api_termination           = (known after apply)
  + ebs_optimized                     = (known after apply)
  + get_password_data                = false
  + host_id                            = (known after apply)
  + host_resource_group_arn           = (known after apply)
  + iam_instance_profile              = (known after apply)
  + id                                 = (known after apply)
  + instance_initiated_shutdown_behavior = (known after apply)
  + instance_lifecycle                = (known after apply)
  + instance_state                   = (known after apply)
  + instance_type                     = "t3.micro"
  + ipv6_address_count               = (known after apply)
  + ipv6_addresses                    = (known after apply)
  + key_name                           = (known after apply)
  + monitoring                         = (known after apply)
  + outpost_arn                       = (known after apply)
  + password_data                     = (known after apply)
  + placement_group                  = (known after apply)
  + placement_partition_number        = (known after apply)
  + primary_network_interface_id     = (known after apply)
  + private_dns                        = (known after apply)
  + private_ip                         = (known after apply)
  + public_dns                          = (known after apply)
  + public_ip                           = (known after apply)
  + secondary_private_ips             = (known after apply)
  + security_groups                   = (known after apply)
}

Type here to search  O: 17:11 ENG 03-09-2023
```

```
C:\Windows\System32\cmd.exe
+ get_password_data          = false
+ host_id                     = (known after apply)
+ host_resource_group_arn    = (known after apply)
+ iam_instance_profile       = (known after apply)
+ id                          = (known after apply)
+ instance_initiated_shutdown_behavior = (known after apply)
+ instance_lifecycle          = (known after apply)
+ instance_state              = (known after apply)
+ instance_type               = "t3.micro"
+ ipv6_address_count         = (known after apply)
+ ipv6_addresses              = (known after apply)
+ key_name                    = (known after apply)
+ monitoring                 = (known after apply)
+ outpost_arn                 = (known after apply)
+ password_data               = (known after apply)
+ placement_group             = (known after apply)
+ placement_partition_number = (known after apply)
+ primary_network_interface_id = (known after apply)
+ private_dns                 = (known after apply)
+ private_ip                  = (known after apply)
+ public_dns                  = (known after apply)
+ public_ip                   = (known after apply)
+ secondary_private_ips       = (known after apply)
+ security_groups              = (known after apply)
)

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

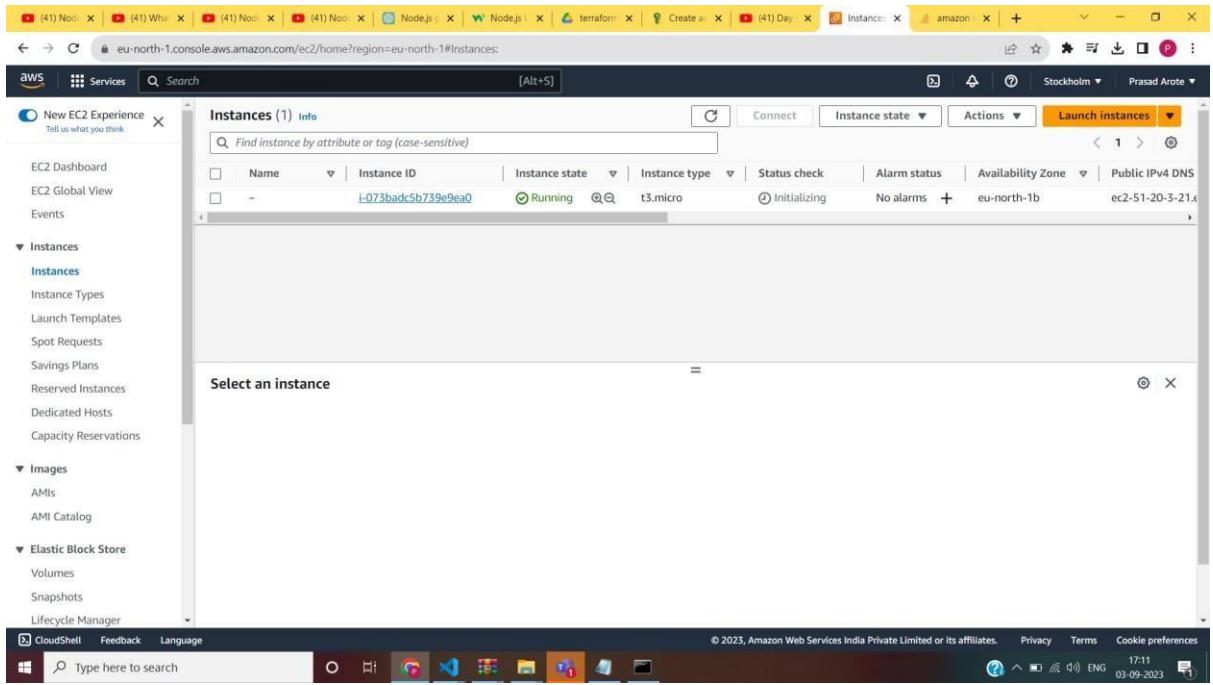
Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

Enter a value: yes

aws_instance.my_instance: Creating...
aws_instance.my_instance: Still creating... [10s elapsed]
aws_instance.my_instance: Creation complete after 16s [id=i-073badc5b739e9ea0]

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

C:\Terraform\Terraform Scripts>
Type here to search  O: 17:11 ENG 03-09-2023
```



11) Destroy the instance using terraform destroy.

```

C:\Windows\System32\cmd.exe
+ maintenance_options {
  auto_recovery = "default" -> null
}
+ metadata_options {
  http_endpoint           = "enabled" -> null
  http_protocol_ipv6      = "disabled" -> null
  http_put_response_hop_limit = 1 -> null
  http_tokens              = "optional" -> null
  instance_metadata_tags   = "disabled" -> null
}
+ private_dns_name_options {
  enable_resource_name_dns_a_record = false -> null
  enable_resource_name_dns_aaaa_record = false -> null
  hostname_type                  = "ip-name" -> null
}
+ root_block_device {
  delete_on_termination = true -> null
  device_name           = "/dev/sda1" -> null
  encrypted              = false -> null
  iops                   = 100 -> null
  tags                   = {} -> null
  throughput             = 0 -> null
  volume_id              = "vol-07db9163befc9c2f1" -> null
  volume_size             = 8 -> null
  volume_type             = "gp2" -> null
}
)
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.my_instance: Destroying... [id=i-073badc5b739e9ea0]
aws_instance.my_instance: Still destroying... [id=i-073badc5b739e9ea0, 10s elapsed]
aws_instance.my_instance: Still destroying... [id=i-073badc5b739e9ea0, 20s elapsed]
aws_instance.my_instance: Still destroying... [id=i-073badc5b739e9ea0, 30s elapsed]
aws_instance.my_instance: Still destroying... [id=i-073badc5b739e9ea0, 40s elapsed]
aws_instance.my_instance: Destruction complete after 42s

destroy complete! Resources: 1 destroyed.

C:\Terraform\Terraform Scripts>

```

CONCLUSION: Here, we understood the use of terraform and we have successfully created a EC2 instances and destroyed it using terraform.

ASSIGNMENT 6

AIM: To perform static analysis on Python programs using SonarQube SAST process.

LO4: To identify and remediate application vulnerabilities earlier and help integrate security in the development process using SAST Techniques.

THEORY:

SonarQube:

Overview: SonarQube is an open-source platform for continuous inspection of code quality. It is used to analyze and measure code quality and security issues in a codebase.

Features:

Static Code Analysis: SonarQube scans source code to identify bugs, code smells, and security vulnerabilities.

Continuous Integration: It integrates seamlessly with CI/CD pipelines, providing automated code analysis during the development process.

Security Analysis: While it primarily focuses on code quality, it also has some security rules to catch common security issues.

Maintainability Metrics: SonarQube provides maintainability metrics and helps teams understand code complexity and maintainability.

Dashboard and Reporting: It offers dashboards and reports for tracking code quality and issues over time.

Use Case: SonarQube is used for improving code quality, maintainability, and to catch some common code security issues. It's more about general code quality and development best practices.

SAST (Static Application Security Testing):

Overview: SAST is a security testing method that analyzes source code, bytecode, or binary code for vulnerabilities without executing the application. It is primarily focused on identifying security issues and vulnerabilities in the code.

Features:

Code Scanning: SAST tools examine the source code or compiled code to identify potential security vulnerabilities, such as SQL injection, cross-site scripting, and more.

Early Detection: SAST is used early in the development process to find security issues before they can be exploited.

Language Support: SAST tools support various programming languages and frameworks.

Integration: They can be integrated into CI/CD pipelines to automatically scan code before deployment.

Use Case: SAST is used for finding and fixing security vulnerabilities in code. It helps secure applications by identifying potential security threats early in the development lifecycle.

1. INSTALL sonarqube (docker images) and sonarscanner zip file from <https://docs.sonarsource.com/sonarqube/latest/analyzing-sourcecode/scanners/sonarscanner/> and set up config file as given in docs.

```
Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Pratik Arote>docker pull sonarqube
Using default tag: latest
latest: Pulling from library/sonarqube
43f89b94cd7d: Pull complete
50431c77a77b: Pull complete
dfd8e860e672: Pull complete
637e2db99ae6: Pull complete
7de1c2853278: Pull complete
d2152ffce821: Pull complete
519cf218564f: Pull complete
Digest: sha256:c6c8096375002d4cb2ef64b89a2736ad572812a87a2917d92e7e59384b9f6f65
Status: Downloaded newer image for sonarqube:latest
docker.io/library/sonarqube:latest

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview sonarqube

C:\Users\Pratik Arote>docker pull sonarsource/sonar-scanner-cli
Using default tag: latest
latest: Pulling from sonarsource/sonar-scanner-cli
9398808236ff: Pull complete
4f4fb700ef54: Pull complete
3cd77fb28e46: Pull complete
f78b288abc31: Pull complete
Digest: sha256:494ecc3b5b1ee1625bd377b3905c4284e4f0cc155cff397805a244dee1c7d575
Status: Downloaded newer image for sonarsource/sonar-scanner-cli:latest
docker.io/sonarsource/sonar-scanner-cli:latest

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview sonarsource/sonar-scanner-cli
```

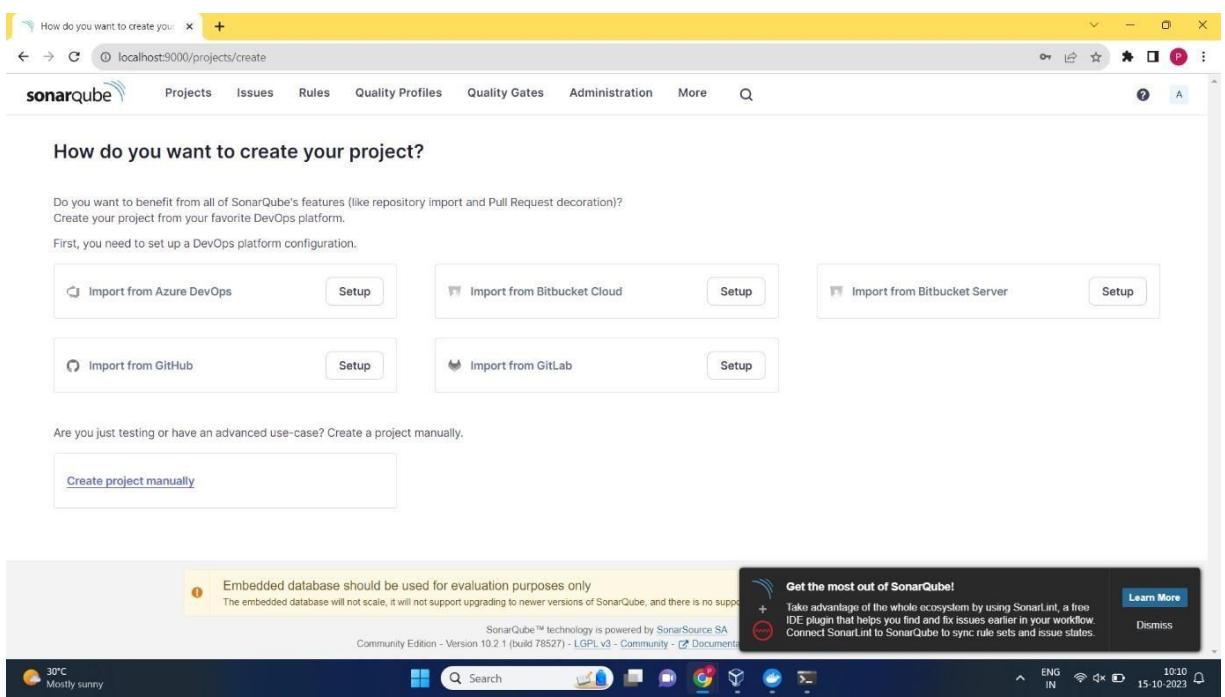
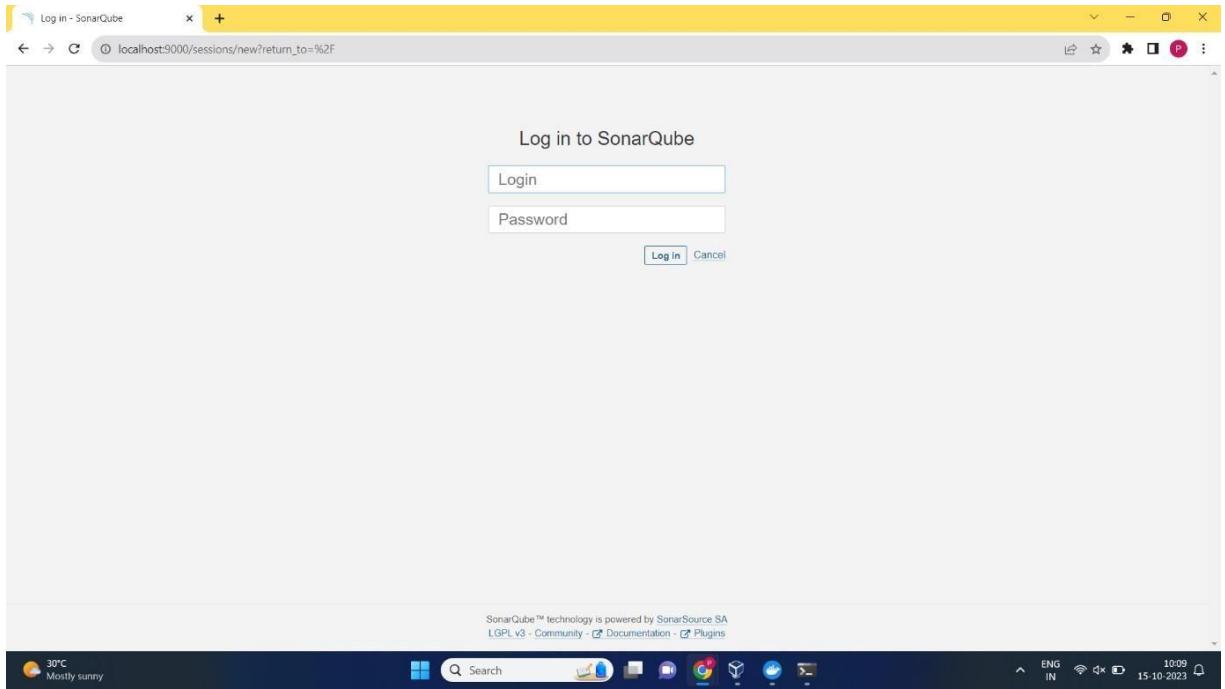
2. Spin up the container

```
C:\Users\Pratik Arote>docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
f3630dbc2ffa6e5598ad922085026400a1f9f1564416b0606b5348000f6d1377

C:\Users\Pratik Arote>docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
sonarqube           latest   3183d6818c6e  42 hours ago  716MB
sample-web-app      latest   713c7cdaaf78  2 weeks ago   42.7MB
myimage              latest   438bb56a50a3  2 weeks ago   122MB
sonarsource/sonar-scanner-cli  latest   2f384fb1bbd5  5 weeks ago   358MB
ubuntu               latest   c6b84b685f35  8 weeks ago   77.8MB
hello-world          latest   9c7a54a9a43c  5 months ago  13.3kB

C:\Users\Pratik Arote>docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS                 NAMES
f3630dbc2ffa        sonarqube:latest   "/opt/sonarqube/dock..."   27 minutes ago    Up 27 minutes   0.0.0.0:9000->9000/tcp   sonarqube
```

3. Open <http://localhost:9000> on the browser. Enter login and password both as “admin” and then set up new password.



4. Create a project

Screenshot of the SonarQube 'Create a project' wizard, Step 1: Set up project for Clean as You Code.

The page shows the following fields:

- Project display name ***: sonarPythonProgram
- Project key ***: sonarPythonProgram
- Main branch name ***: main

A note states: "The name of your project's default branch [Learn More](#)". A "Next" button is at the bottom.

At the top right, there is a banner: "Get the most out of SonarQube! Take advantage of the whole ecosystem by using SonarLint, a free IDE plugin that helps you find and fix issues earlier in your workflow. Connect SonarLint to SonarQube to sync rule sets and issue states." with "Learn More" and "Dismiss" buttons.

The browser status bar shows: "localhost:5000/projects/create?mode=manual&setncd=true".

The screenshot shows the SonarQube web interface with a yellow header bar. The URL in the address bar is `localhost:9000/tutorials?id=sonarPythonProgram1`. The main content area has a heading "How do you want to analyze your repository?". It contains several options:

- With Jenkins**
- With GitHub Actions**
- With Bitbucket Pipelines**
- With GitLab CI**
- With Azure Pipelines**
- Other CI**: A note stating "SonarQube integrates with your workflow no matter which CI tool you're using."
- Locally**: A note stating "Use this for testing or advanced use-case. Other modes are recommended to help you set up your CI environment."

A yellow info box at the bottom left says "Embedded database should be used for evaluation purposes only". A black info box at the bottom right says "Get the most out of SonarQube!" with a "Learn More" button and a "Dismiss" button.

5. Provide token

The screenshot shows the SonarQube web interface with a yellow header bar. The URL in the address bar is `localhost:9000/tutorials?id=sonarPythonProgram1&selectedTutorial=local`. The main content area has a heading "1 Provide a token". It displays a token value: `pythonToken: sqp_c8922aed03df90f4cc0dfb26200772ff42a9032b`. A note below it says: "The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point in time in your [user account](#)". A blue "Continue" button is visible. A yellow info box at the bottom left says "Embedded database should be used for evaluation purposes only". A black info box at the bottom right says "Get the most out of SonarQube!" with a "Learn More" button and a "Dismiss" button.

2 Run analysis on your project

What option best describes your build?

Maven Gradle .NET Other (for JS, TS, Go, Python, PHP, ...)

What is your OS?

Linux Windows macOS

Download and unzip the Scanner for Windows

Visit the [official documentation of the Scanner](#) to download the latest version, and add the `bin` directory to the `%PATH%` environment variable.

Execute the Scanner

Running a SonarQube analysis is straightforward. You just need to execute the following commands in your project's folder.

```
sonar-scanner.bat -D"sonar.projectKey=sonarPythonProgram1" -D"sonar.sources=." -D"sonar.host.url=http://localhost:9000" -D"sonar.token=sqp_c8922aed03df90f4cc0dfb26200772ff42a9032b"
```

Please visit the [official documentation of the Scanner](#) for more details.

6. Enter the following command

```
C:\Windows\System32\cmd.exe > Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.

C:\Program Files\sonar-scanner-5.0.1.3006-windows\bin>sonar-scanner.bat -D"sonar.projectKey=sonarPythonProgram1" -D"sonar.sources=C:\Users\Pratik Arote\Desktop\sastPython" -D"sonar.host.url=http://localhost:9000" -D"sonar.token=sqp_c8922aed03df90f4cc0dfb26200772ff42a9032b" -D"sonar.projectBaseDir=C:\Users\Pratik Arote\Desktop\sastPython"
INFO: Scanner configuration file: C:\Program Files\sonar-scanner-5.0.1.3006-windows\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: SonarScanner 5.0.1.3006
INFO: Java 17.0.7 Eclipse Adoptium (64-bit)
INFO: Windows 11 10.0 amd64
INFO: User cache: C:\Users\Pratik Arote\.sonar\cache
INFO: Analyzing on SonarQube server 10.2.1.78527
INFO: Default locale: "en_IN", source code encoding: "windows-1252" (analysis is platform dependent)
INFO: Load global settings
INFO: Load global settings (done) | time=58ms
INFO: Server id: 1478411F-AYsxFDZoQL-ruFd2_S5
INFO: User cache: C:\Users\Pratik Arote\.sonar\cache
INFO: Load/download plugins
INFO: Load plugins index
INFO: Load plugins index (done) | time=338ms
INFO: Load/download plugins (done) | time=8251ms
INFO: Process project properties
INFO: Process project properties (done) | time=40ms
INFO: Execute project builders
INFO: Execute project builders (done) | time=7ms
INFO: Project key: sonarPythonProgram1
INFO: Base dir: C:\Users\Pratik Arote\Desktop\sastPython
INFO: Working dir: C:\Users\Pratik Arote\Desktop\sastPython\scannerwork
INFO: Load project settings for component key: 'sonarPythonProgram1'
INFO: Load project settings for component key: 'sonarPythonProgram1' (done) | time=122ms
WARN: SCM provider autodetection failed. Please use "sonar.scm.provider" to define SCM of your project, or disable the SCM Sensor in the project settings.
INFO: Load quality profiles
INFO: Load quality profiles (done) | time=597ms
INFO: Load active rules
INFO: Load active rules (done) | time=7984ms
INFO: Load analysis cache
INFO: Load analysis cache (404) | time=60ms
INFO: Load project repositories
INFO: Load project repositories (done) | time=295ms
```

```
C:\Windows\System32\cmd.e + v
INFO: Sensor VB.NET Properties [vbnet] (done) | time=2ms
INFO: Sensor IaC Docker Sensor [iac]
INFO: 0 source files to be analyzed
INFO: 0/0 source files have been analyzed
INFO: Sensor IaC Docker Sensor [iac] (done) | time=206ms
INFO: ----- Run sensors on project
INFO: Sensor Analysis Warnings import [csharp]
INFO: Sensor Analysis Warnings import [csharp] (done) | time=7ms
INFO: Sensor Zero Coverage Sensor
INFO: Sensor Zero Coverage Sensor (done) | time=47ms
INFO: SCM Publisher No SCM system was detected. You can use the 'sonar.scm.provider' property to explicitly specify it.
INFO: CPD Executor 1 file had no CPD blocks
INFO: CPD Executor Calculating CPD for 0 files
INFO: CPD Executor CPD calculation finished (done) | time=0ms
INFO: Analysis report generated in 253ms, dir size=136.5 kB
INFO: Analysis report compressed in 48ms, zip size=17.5 kB
INFO: Analysis report uploaded in 201ms
INFO: ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=sonarPythonProgram1
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
INFO: More about the report processing at http://localhost:9000/api/ce/task?id=AYsx47EpoQL-ruFd3M3Y
INFO: Analysis total time: 22.756 s
INFO: More about the report processing at http://localhost:9000/api/ce/task?id=AYsx47EpoQL-ruFd3M3Y
INFO: EXECUTION SUCCESS
INFO:
INFO: Total time: 35.565s
INFO: Final Memory: 23M/77M
INFO:

C:\Program Files\sonar-scanner-5.0.1.3006-windows\bin>
```

7. See the result of the test

The screenshot shows a web browser window with the SonarQube interface. The URL in the address bar is `localhost:9000/dashboard?id=sonarPythonProgram1`. The dashboard has a green 'Passed' status for the Quality Gate. It includes sections for Measures (Reliability, Security, Coverage, Duplications), Issues, Security Hotspots, and Activity. The browser's address bar shows 'localhost:9000/dashboard?id=sonarPythonProgram1'.

CONCLUSION:

Here we have successfully performed static analysis of python programs.

ASSIGNMENT 7

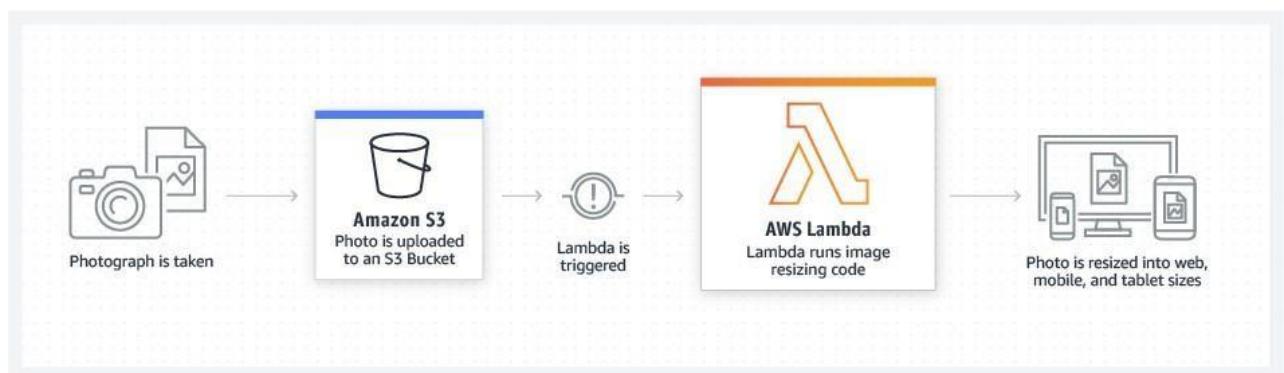
AIM: To understand AWS Lambda functions and create a Lambda function using Python to log “An Image has been added” message, once a file is added to a S3 bucket.

LO6: To engineer a composition of nano services using AWS Lambda and Step Functions with the Serverless Framework.

THEORY:

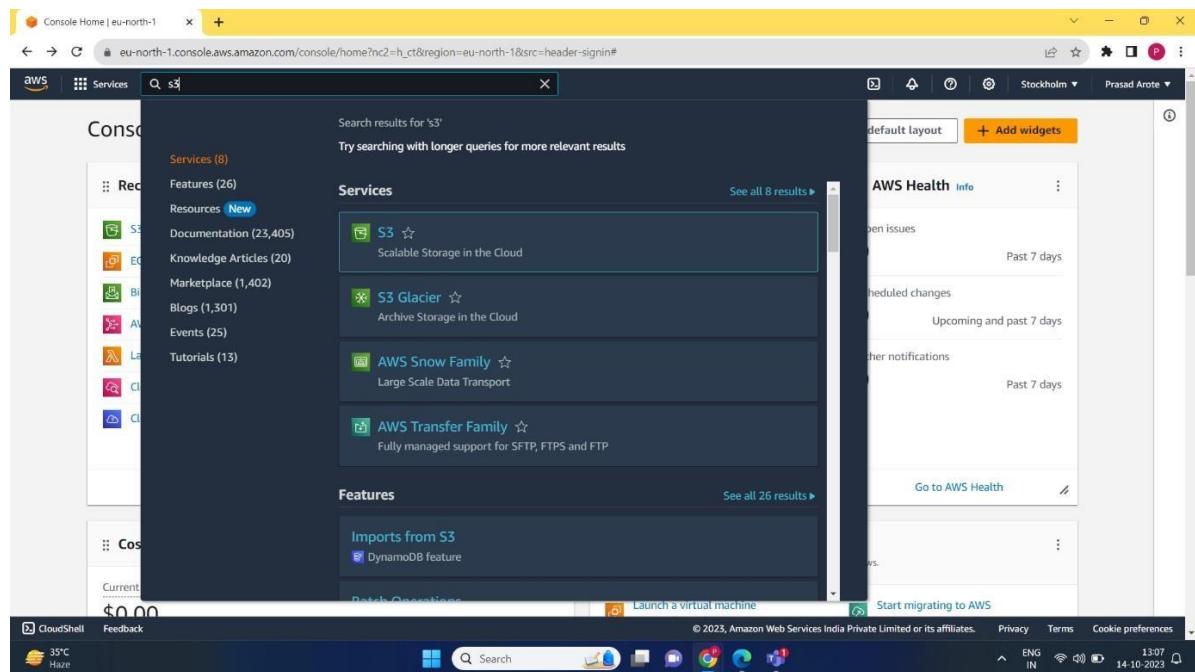
LAMBDA FUNCTION

AWS Lambda is a serverless, event-driven compute service that lets you run code for virtually any type of application or backend service without provisioning or managing servers. You can trigger Lambda from over 200 AWS services and software as a service (SaaS) applications, and only pay for what you use.



Installation:

1. Create a S3 bucket



The screenshot shows the 'Create bucket' page in the AWS S3 console. The 'General configuration' section is filled with the following details:

- Bucket name:** prasadDev
- AWS Region:** Asia Pacific (Mumbai) ap-south-1
- Copy settings from existing bucket - optional:** Choose bucket (button)

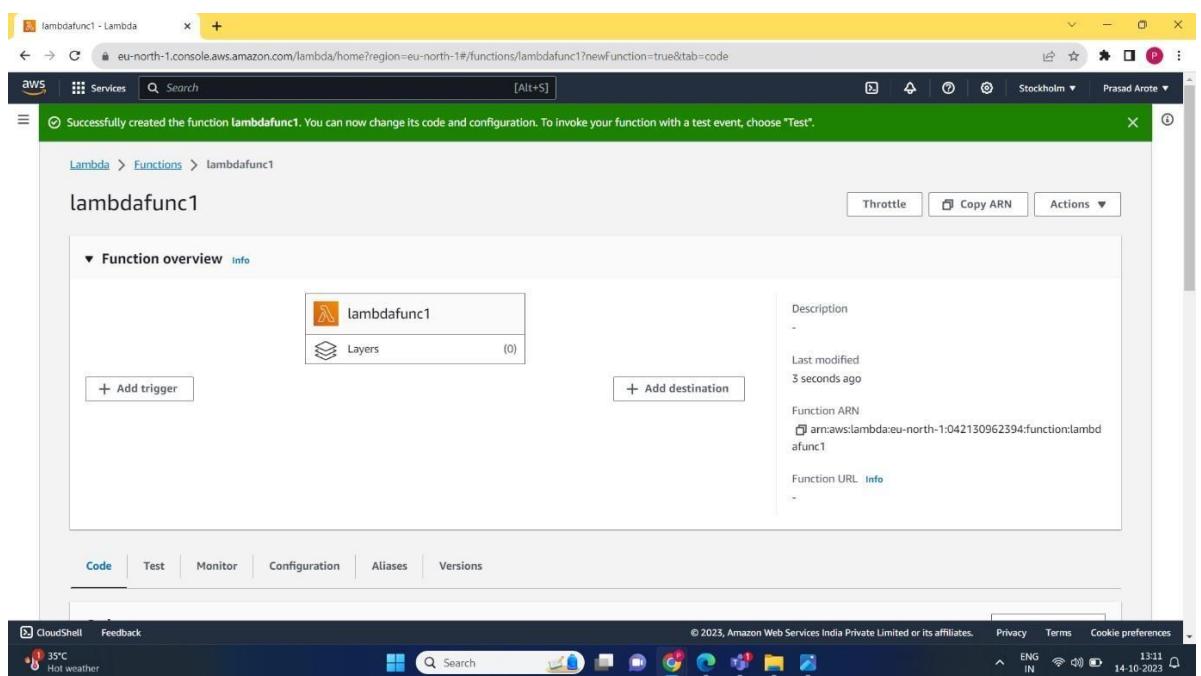
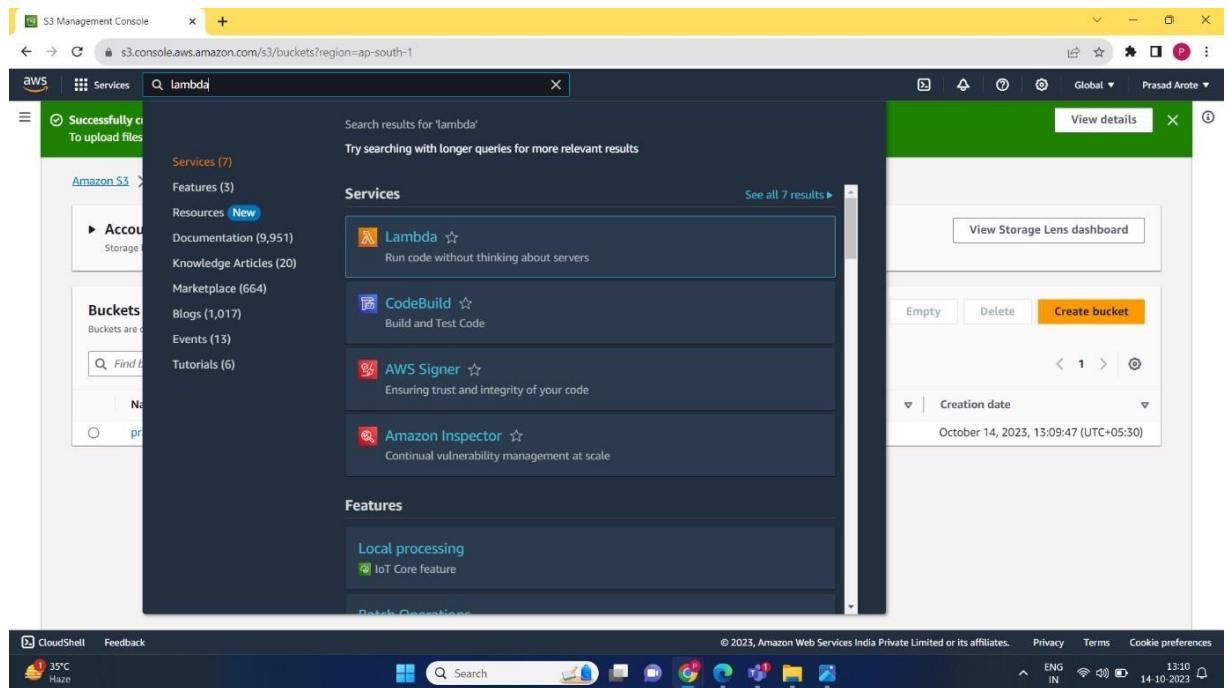
The 'Object Ownership' section shows that ACLs are disabled (recommended). The status bar at the bottom indicates it's 35°C Haze.

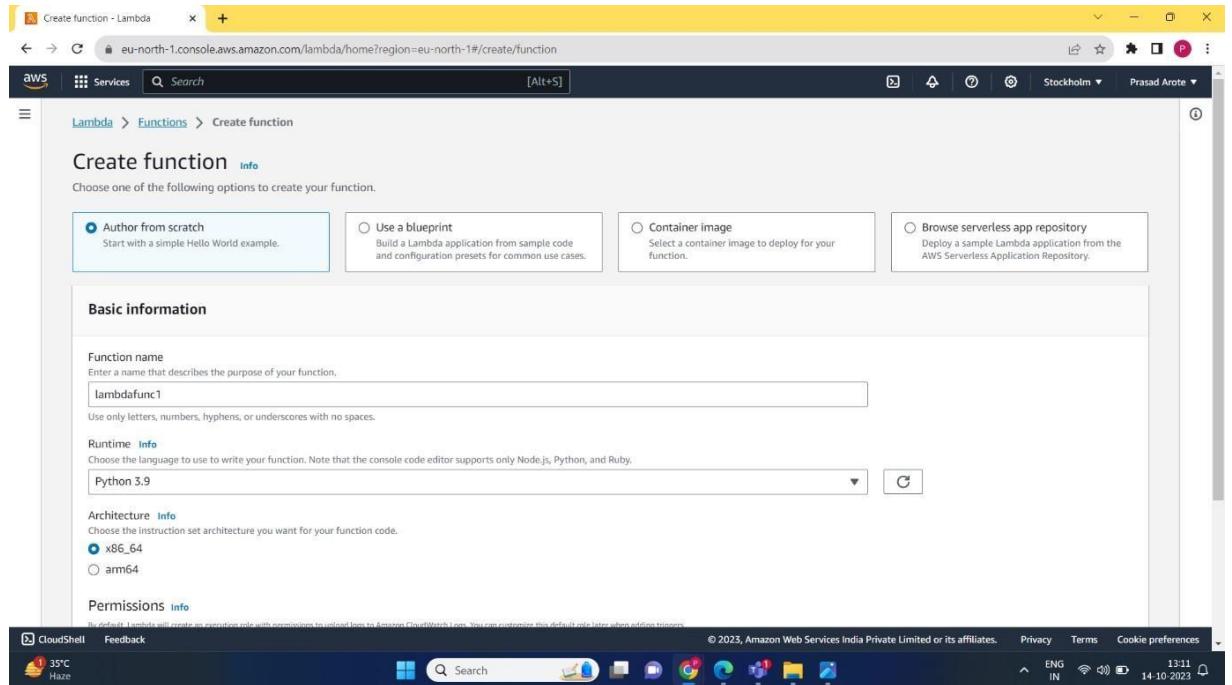
The screenshot shows the 'Buckets' page in the AWS S3 Management Console. A green success message states: "Successfully created bucket 'prasad.dev'. To upload files and folders, or to configure additional bucket settings, choose View details." Below this, the 'Account snapshot' section is visible. The main table lists the single bucket 'prasad.dev' with the following details:

Name	AWS Region	Access	Creation date
prasad.dev	Asia Pacific (Mumbai) ap-south-1	Bucket and objects not public	October 14, 2023, 13:09:47 (UTC+05:30)

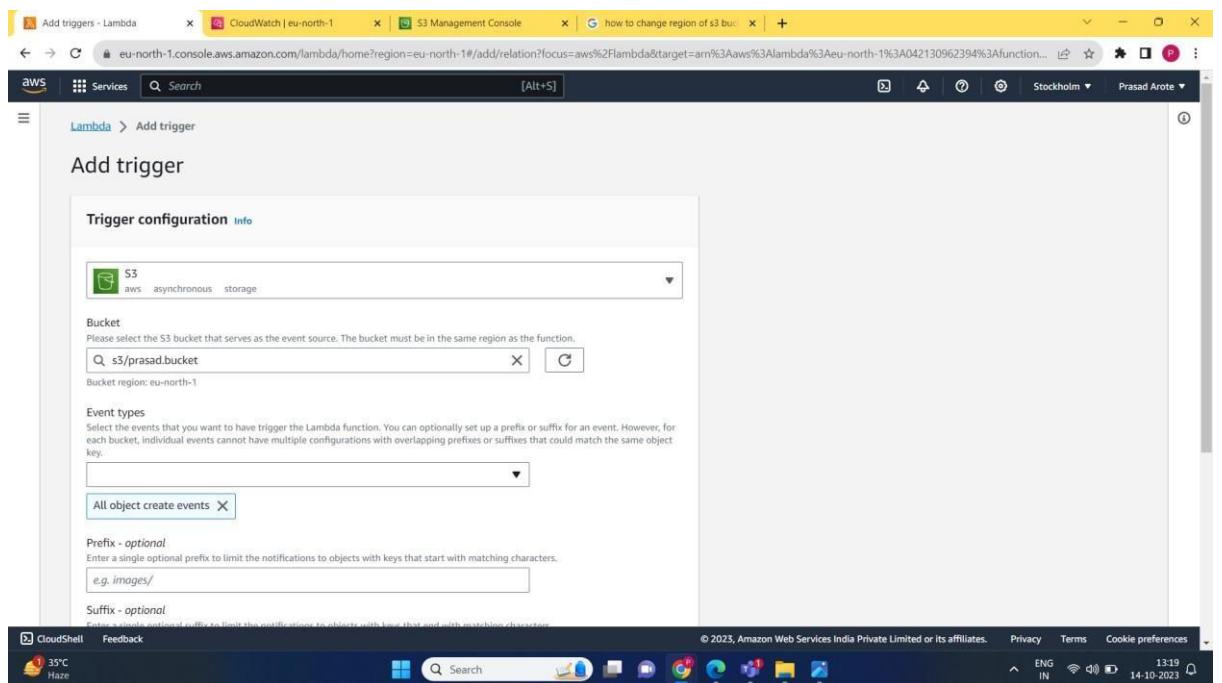
The status bar at the bottom indicates it's 35°C Haze.

2. Create a Lambda function.





3. Create a trigger

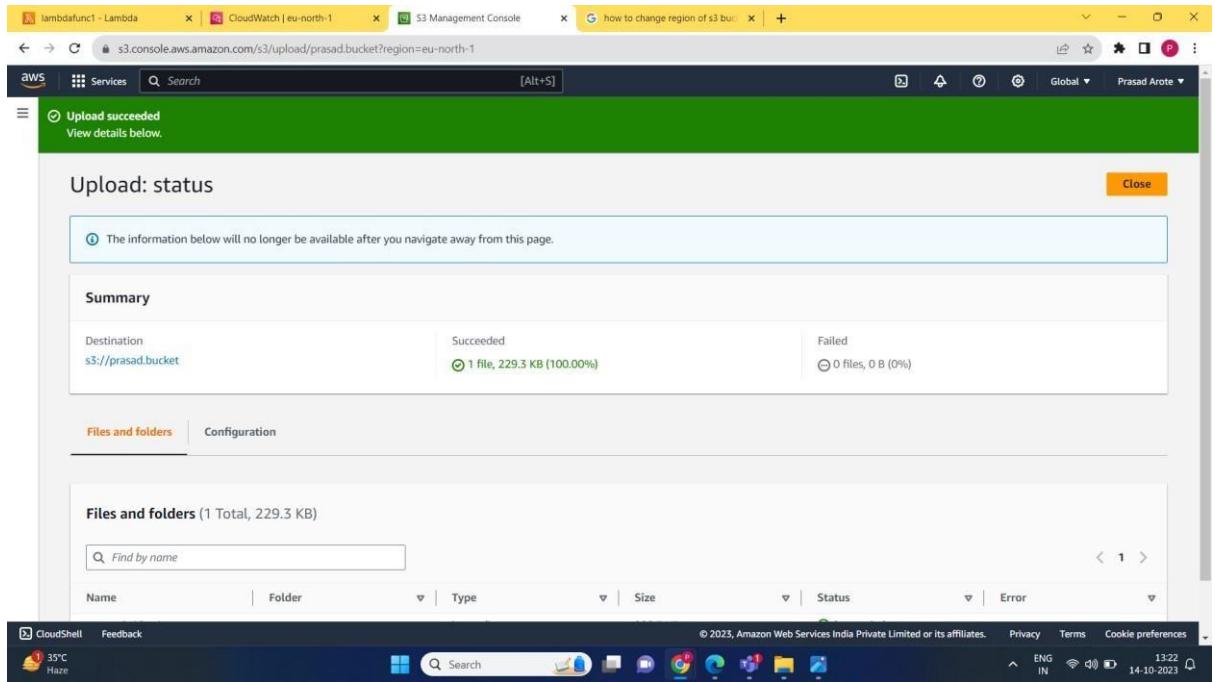


The screenshot shows the AWS Lambda console interface. The top navigation bar includes tabs for 'Add triggers - Lambda', 'CloudWatch | eu-north-1', 'S3 Management Console', and a search bar. The main content area is titled 'Event types' and shows configuration for a new trigger. It includes fields for 'Event type' (set to 'All object create events'), 'Prefix - optional' (containing 'e.g. images/'), and 'Suffix - optional' (containing 'e.g. jpg'). A note about recursive invocation is present, along with a checked checkbox acknowledging the use of the same S3 bucket for both input and output. A note also states that Lambda will add necessary permissions for AWS S3. At the bottom are 'Cancel' and 'Add' buttons.

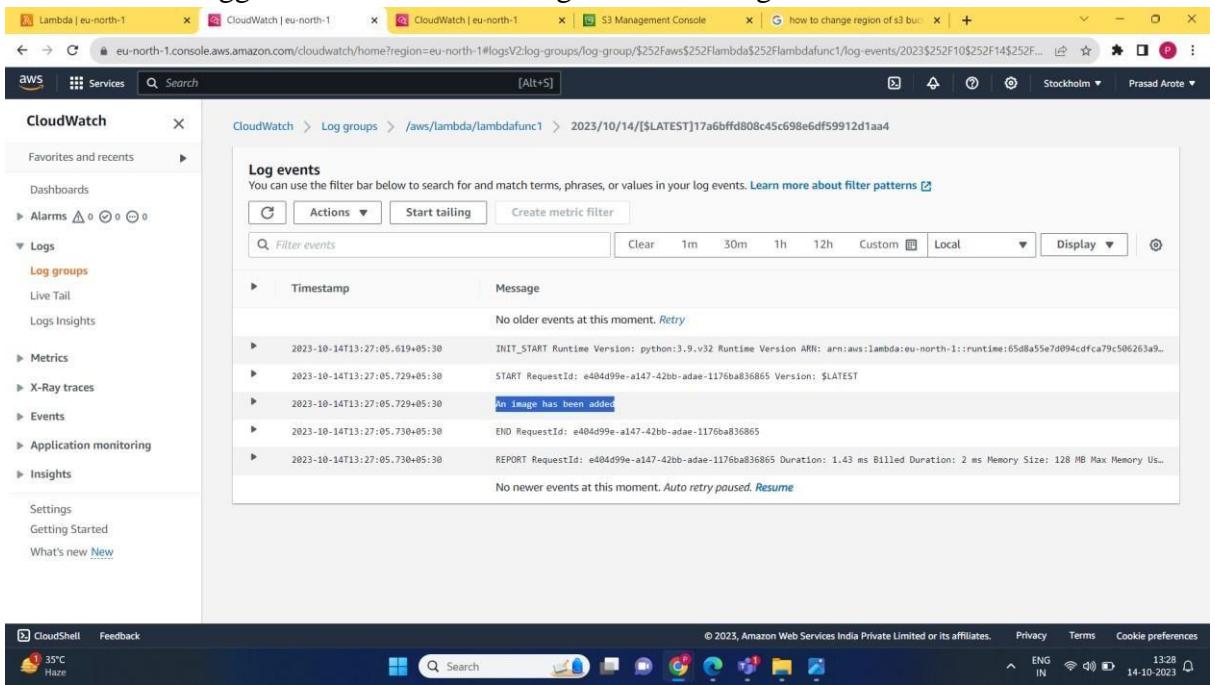
The second part of the screenshot shows the Lambda function configuration page for 'lambdafunc1'. The top navigation bar is identical. The main content area shows the function overview. It lists the trigger 'prasad.bucket' successfully added to the function. The 'Configuration' tab is selected, showing details like the ARN (arn:aws:lambda:eu-north-1:042130962394:function:lambdafunc1), last modified time (8 minutes ago), and function URL. Below the configuration are tabs for 'Code', 'Test', 'Monitor', 'Configuration' (selected), 'Aliases', and 'Versions'.

The screenshot shows the AWS S3 Management Console interface for uploading files to a bucket named 'prasad.bucket'. The process is divided into several steps:

- Upload Info:** A large text area at the top allows users to drag and drop files or choose 'Add files' or 'Add folder'. It includes a note about uploading files larger than 160GB.
- Files and folders (0):** A table with columns for Name, Folder, Type, and Size. A search bar and sorting options are available. The message 'No files or folders' is displayed.
- Destination:** A section where the user can select the destination bucket. In this step, the bucket 'prasad.bucket' is selected.
- Destination details:** A sub-section showing bucket settings for new objects stored in the specified destination.
- Permissions:** An optional section for granting public access or access to other AWS accounts.
- Properties:** An optional section for specifying storage class, encryption settings, and tags.
- Buttons:** At the bottom right are 'Cancel' and 'Upload' buttons.



4. Thus we have triggered the function that logs when an image is added to S3 Bucket.



Conclusion: We have successfully created an lambda functions that logs when an image is added in S3 bucket.

Roll No:10

Saish Bavalekar

ASSIGNMENT 8

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

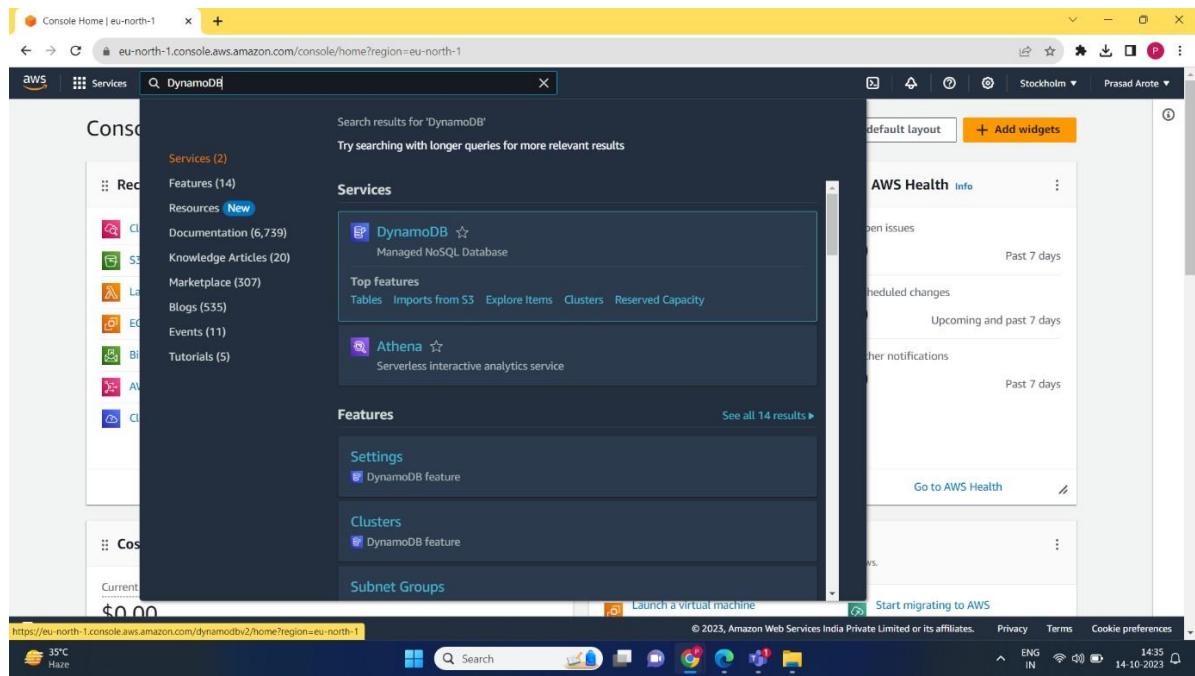
LO6: To engineer a composition of nano services using AWS Lambda and Step Functions with the serverless framework.

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. Create a table

The screenshot shows the 'Create table' wizard in the Amazon DynamoDB console. In the 'Table details' section, the table name is set to 'Student' and the partition key is 'id'. In the 'Table settings' section, the sort key is optional and left empty. The 'Table settings' section also includes a 'CloudWatch Metrics' checkbox which is unchecked.

2. Create a role using IAM

The screenshot shows the AWS IAM console with a search query 'IAM' entered in the search bar. The search results list several services under the 'Services' category, including IAM, IAM Identity Center, Resource Access Manager, and AWS App Mesh. The IAM service card is highlighted, showing its purpose: 'Manage access to AWS resources'. The search interface also includes a sidebar with navigation links like 'Documentation', 'Features', 'Groups', and 'Policies'.

3. Add permissions – AmazonDynamoFullAccess

Select trusted entity

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

Add permissions

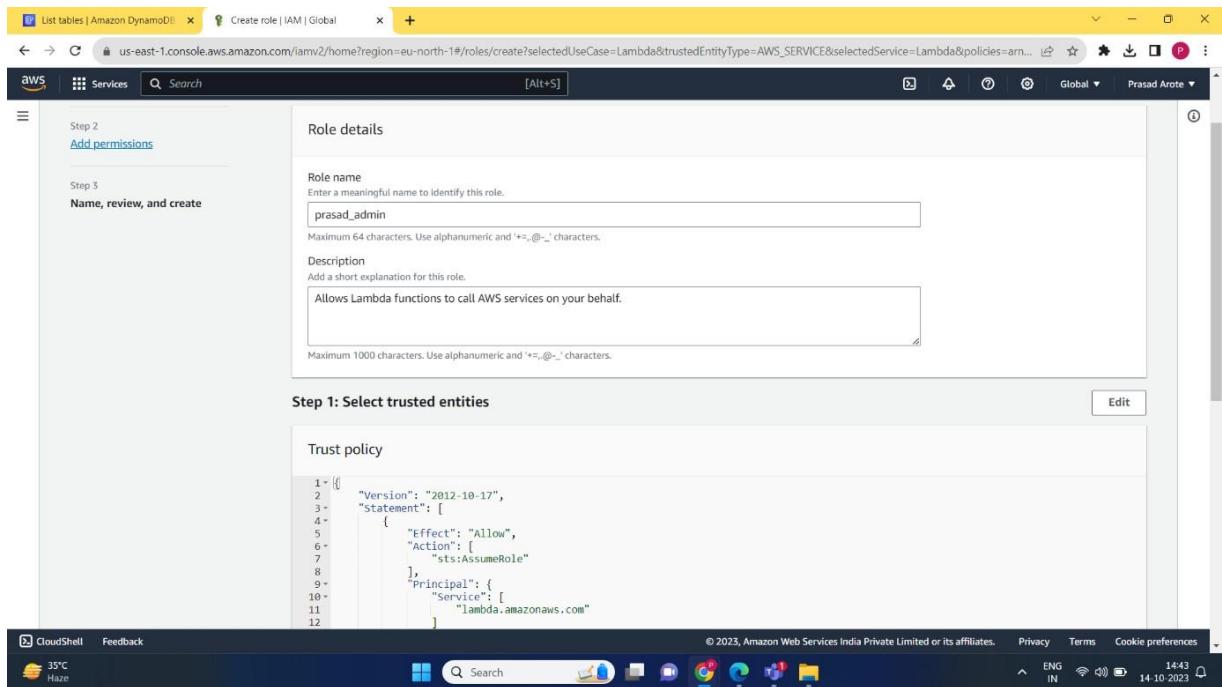
Permissions policies (1/887)

Choose one or more policies to attach to your new role.

Policy name	Type	Description
<input checked="" type="checkbox"/> AmazonDynamoDBFullAccess	AWS managed	Provides full access to Amazon DynamoDB...
<input type="checkbox"/> AmazonDynamoDBReadOnlyAccess	AWS managed	Provides read only access to Amazon Dyn...
<input type="checkbox"/> AWSLambdaDynamoDBExecutionRole	AWS managed	Provides list and read access to DynamoD...
<input type="checkbox"/> AWSLambdaInvocation-DynamoDB	AWS managed	Provides read access to DynamoDB Strea...

▶ Set permissions boundary - *optional*

Cancel Previous Next

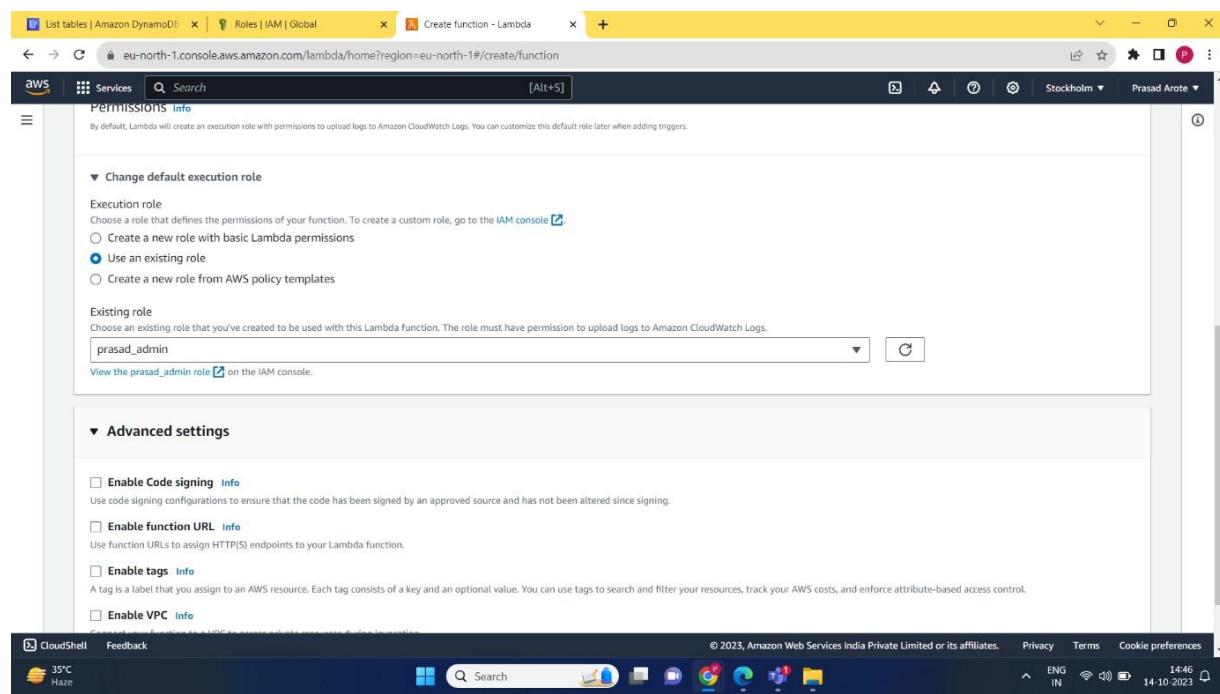
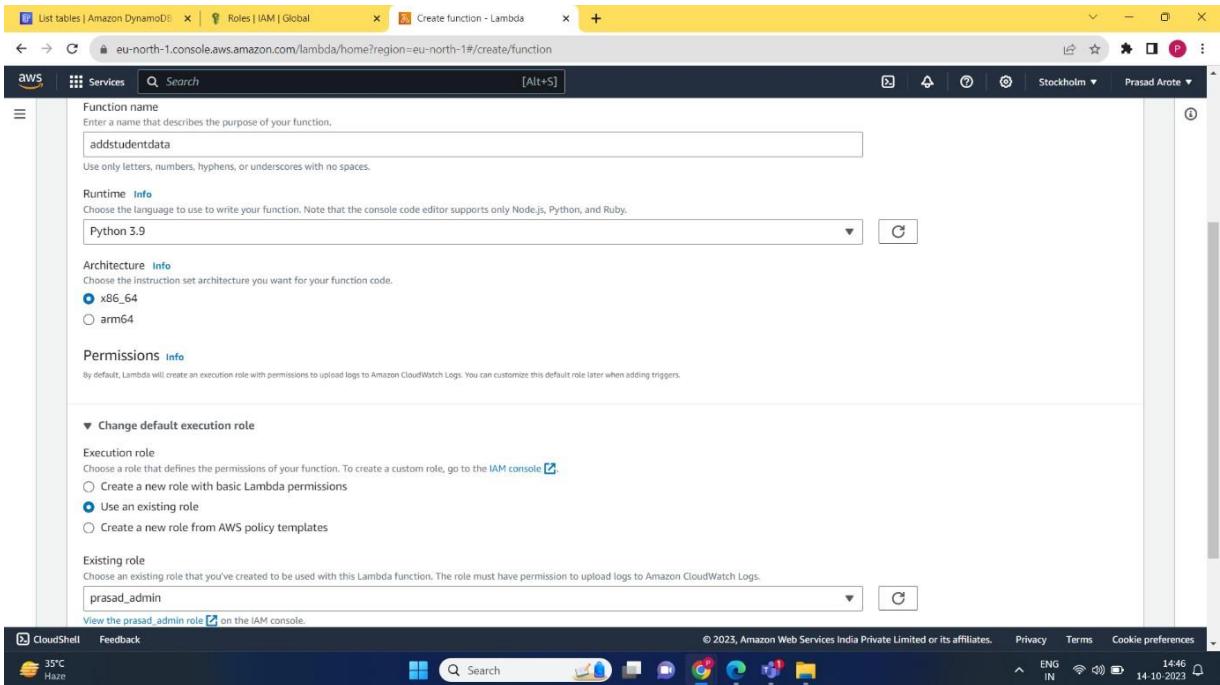


Role name	Trusted entities	Last activity
AWSCloud9SSMAccessRole	AWS Service: ec2, and 1 more	75 days ago
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable	AWS Service: dynamodb.application-	-
AWSServiceRoleForAWSCloud9	AWS Service: cloud9 (Service-Linked)	75 days ago
AWSServiceRoleForSupport	AWS Service: support (Service-Linked)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service)	-
lambdafunc1-role-11c5lj6u	AWS Service: lambda	1 hour ago
prasad_admin	AWS Service: lambda	-
PyRole	AWS Service: lambda	68 days ago
Runpython	AWS Service: lambda	68 days ago

4. Create a Lambda Function

The screenshot shows the AWS Lambda search results page. The search bar at the top contains the query 'lambda'. Below the search bar, there is a message: 'Search results for "lambda" Try searching with longer queries for more relevant results'. The results are categorized under 'Services' and 'Features'. Under 'Services', the 'Lambda' service is highlighted with a yellow box, followed by 'CodeBuild', 'AWS Signer', and 'Amazon Inspector'. Under 'Features', there is a section for 'Local processing' which includes 'IoT Core feature'. On the right side of the page, there is a sidebar titled 'AWS Health' with sections for 'open issues', 'scheduled changes', 'Upcoming and past 7 days', 'other notifications', and a link to 'Go to AWS Health'. The bottom of the page shows the URL 'https://eu-north-1.console.aws.amazon.com/lambda/home?region=eu-north-1', the AWS logo, and various browser tabs.

The screenshot shows the 'Create function' wizard. The title bar says 'Create function - Lambda'. The main heading is 'Create function' with an 'Info' link. Below it, a sub-instruction says 'Choose one of the following options to create your function.' There are four options: 'Author from scratch' (selected), 'Use a blueprint', 'Container image', and 'Browse serverless app repository'. The 'Author from scratch' option has a sub-instruction: 'Start with a simple Hello World example.' The 'Basic information' section contains fields for 'Function name' (set to 'addstudentdata'), 'Runtime' (set to 'Python 3.9'), 'Architecture' (set to 'x86_64'), and 'Permissions' (with a note about creating a CloudWatch Logs permission). At the bottom, there are links for 'CloudShell' and 'Feedback', and the footer includes the URL 'https://eu-north-1.console.aws.amazon.com/lambda/home?region=eu-north-1#/create/function', the AWS logo, and various browser tabs.



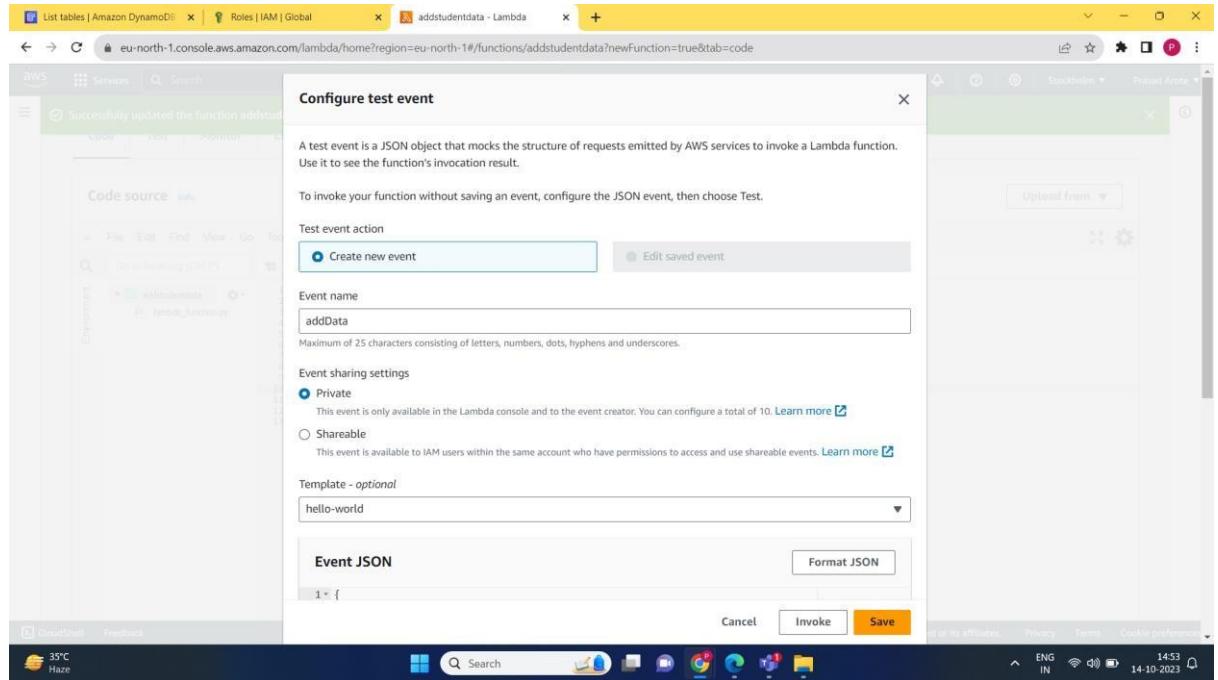
5. Write the following code

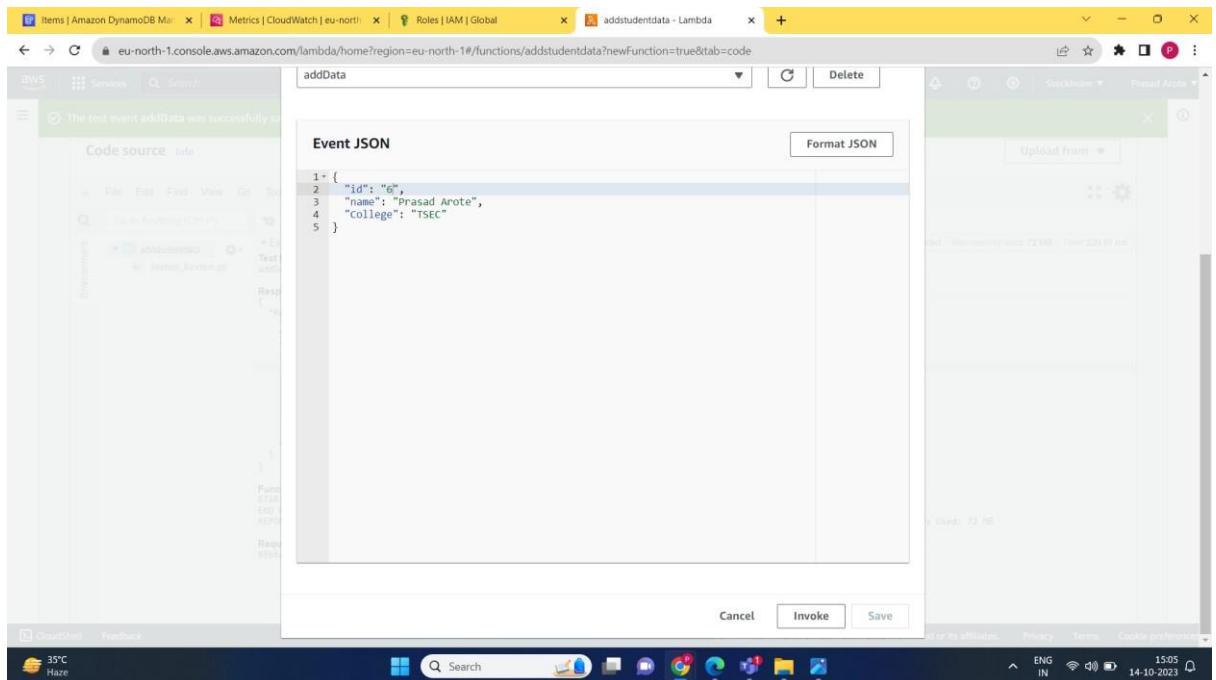
```

1 import json
2 import boto3
3
4 def lambda_handler(event, context):
5     # TODO Implement
6     client_dynamo = boto3.resource('dynamodb')
7     table = client_dynamo.Table('Student')
8
9     response = table.put_item(Item=event)
10
11     return response
12
13

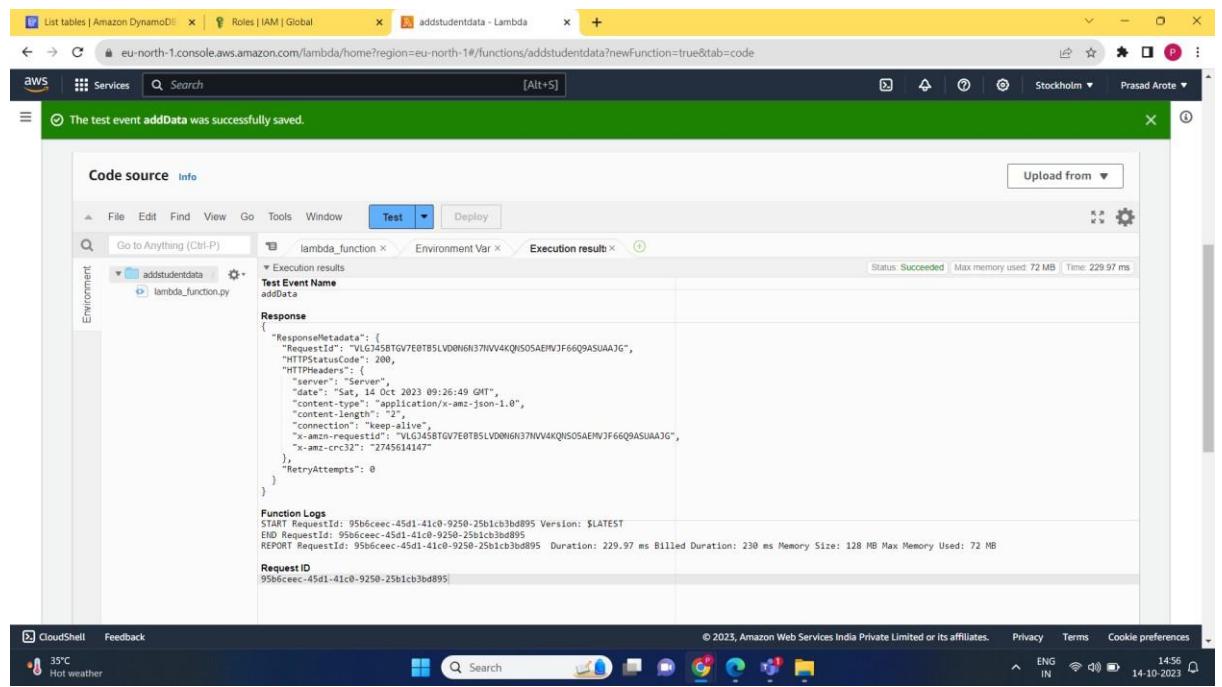
```

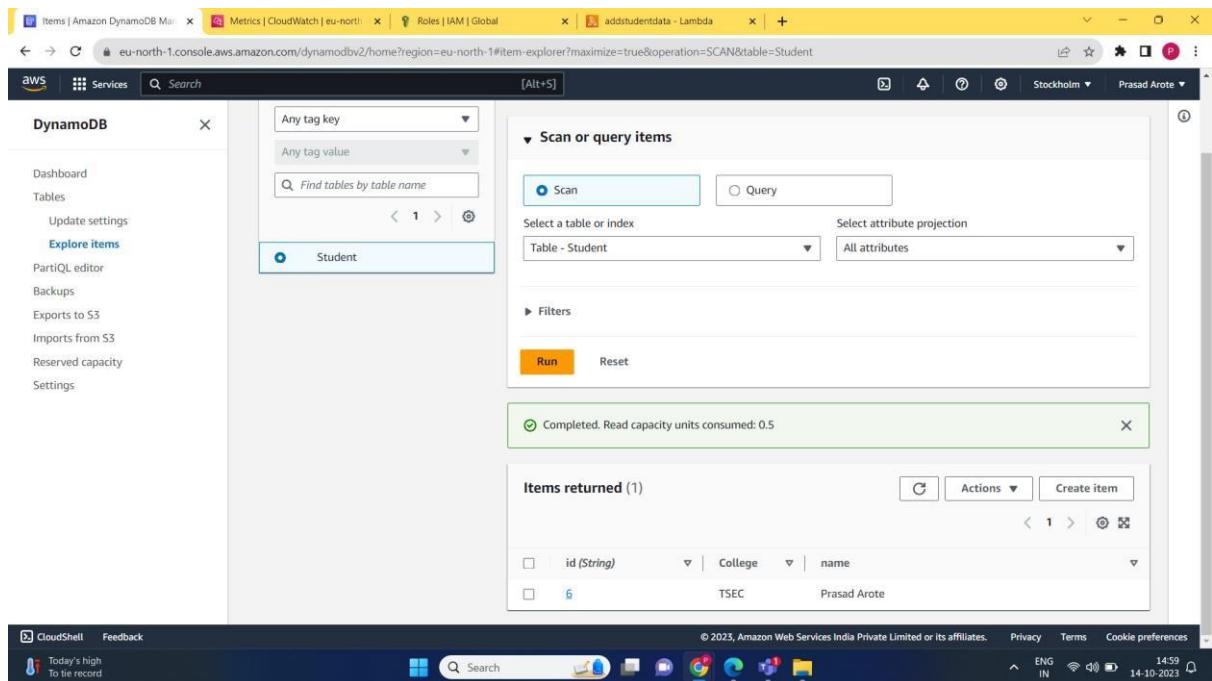
6. Configure test event and Save





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





CONCLUSION:

Thus, we have successfully inserted data in DynamoDB by using a Lambda function.

Roll No:10

Saish Bavalekar

ASSIGNMENT 9

Aim: To understand demo of Nagios and open-source tools.

LO1: To understand the fundamentals of Cloud Computing and be fully proficient with Cloud based DevOps solution deployment options to meet your business requirements.

LO5: To use Continuous Monitoring Tools to resolve any system errors (low memory, unreachable server etc.) before they have any negative impact on the business productivity.

Theory:

What is Nagios and how it works?

Nagios is an open source monitoring system for computer systems Nagios software runs periodic checks on critical parameters of application, network and server resources. For example, Nagios can monitor memory usage, disk usage, microprocessor load, the number of currently running processes and log files.

Steps-

Go to google.com, Search Nagios Demo

Google

nagios demo

All Videos Images News Shopping More Tools

About 3,87,000 results (0.44 seconds)

<https://exchange.nagios.org/directory/Demos/details> ::

Nagios XI Online Demo

An online **demo** of Nagios XI. The **demo** allows you to test configuration wizards, dashlets, dashboards, views, and more. Reviews (0).

<https://exchange.nagios.org/directory/Demos/details> ::

Demos - Nagios Exchange

An online **demo** of Nagios Log Server. The **demo** allows you to view system logs and event logs, giving some examples on how you can visualize data sent into Nagios ...

<https://exchange.nagios.org/directory/Demos/details> ::

Now click on the website-

The screenshot shows the Nagios XI Online Demo page. At the top, there's a navigation bar with links for Home, Directory (which is underlined), and About. Below the navigation is a breadcrumb trail: Home | Directory | Demos | Nagios XI Online Demo. The main content area has a title 'Directory Tree' and a section for the 'Nagios XI Online Demo'. This section includes a 'Submit review' button, a rating of 5 stars, 0 votes, and a Favoured count of 0. It also shows the owner as 'egalstad', the website as 'nagiosxi.demos.nagios.com', and hits as 141800. To the right, there are two search boxes: 'Search Exchange' and 'Search All Sites', both with 'Go' buttons. A sidebar on the right is titled 'Nagios Live Webinars' with a sub-section about experts helping organizations.

Now click on login as administrator

The screenshot shows a web browser window with two tabs: 'Nagios XI Online Demo - Nagios XI' and 'Login - Nagios XI'. The main content is a 'Login' form with fields for Username and Password, and a 'Login' button. Below the form is a 'Select Language:' dropdown with various flags. To the right, there's a banner for 'Nagios XI Demo System' and a section titled 'Demo Account Options'. This section describes five types of accounts: 'Administrator Access', 'Read-Only User Access', 'Advanced User Access', 'Normal User Access', and 'Administrator Access' (dark theme). For each account type, it lists a 'Log in as [Account Type]' button and the corresponding username and password. At the bottom, there's a 'Demo Notes' section and a taskbar with various icons.

It will have interface like this

The screenshot shows the Nagios XI Home Dashboard. On the left, there's a sidebar with sections like Quick View, Details, Graphs, and Maps. The main area has several cards: 'Getting Started Guide' with common tasks, 'Host Status Summary' (Up: 132, Down: 1, Unreachable: 1, Pending: 0), 'Service Status Summary' (Ok: 1267, Warning: 29, Unknown: 97, Critical: 1, Pending: 0), and 'Administrative Tasks'. To the right, there's a 'We're Here To Help!' section with a photo of a support team member and links to Support Forum, Help Resources, Customer Ticket Support Center, and Customer Phone Support. At the bottom, there are buttons for 'Run a Config Wizard', 'Run Auto-Discovery', and 'Advanced Config'.

Now click on Host status-

The screenshot shows the Nagios XI Host Status page. It lists 242 hosts, each with columns for Host, Status, Duration, Attempt, Last Check, and Status Information. The hosts are categorized by status: Up (green), Down (red), and Unreachable (yellow). The 'Status Information' column provides specific details for each host, such as response times or error messages. There are also summary tables for Host Status Summary and Service Status Summary at the top of the page.

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 Now click on Host Group Status.

Host Status Summary

Up	Down	Unreachable	Pending
132	2	1	0
Unhandled	Problems	All	
28	110	242	

Last Updated: 2023-08-28 00:40:48

Service Status Summary

Ok	Warning	Unknown	Critical	Pending
1267	29	97	2	0
Unhandled	Problems	All		
200	583	1850		

Last Updated: 2023-08-28 00:40:48

Status Summary For All Host Groups

Host Group	Hosts	Services
Monitoring Servers (Monitoring Servers)	7 Up	138 Ok 5 Warning 1 Unknown 7 Unreachable
Hostgroup Two (hg2)	2 Up	41 Ok 3 Warning 10 Unknown 28 Unreachable
Some Other Hostgroup (hg3)	2 Up	19 Ok 1 Unknown 1 Unreachable
Linux Servers (linux-servers)	11 Up	292 Ok 3 Warning 1 Unknown 26 Unreachable
Network Devices (network-devices)	8 Up	320 Ok

Here we can see Status Summary for All Host Groups

Now we click on BBMap

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

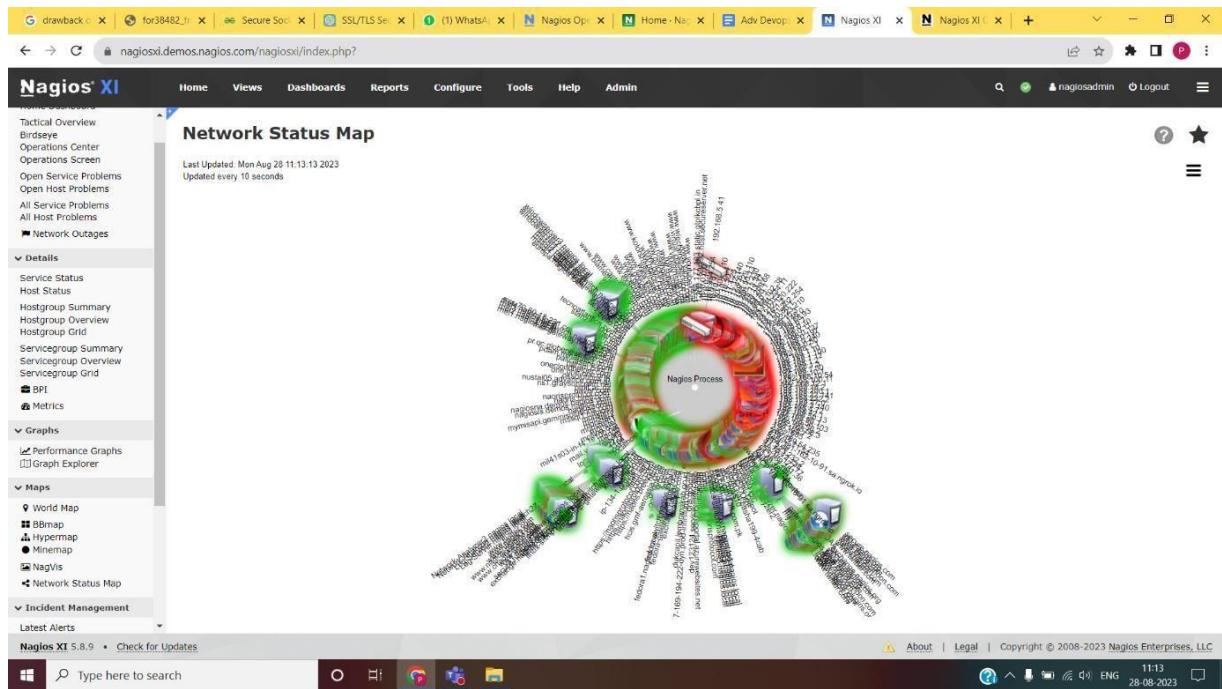
BBMap

The BBmap shows the current status of hosts/services in an icon grid layout. When a host has been acknowledged or is in downtime, the services will display semi-transparent. If a service is acknowledged or in downtime it will display the wrench icon. Hover over the round status icons to see current status information. Hover over host/service names to see full name if the name is too long to display on the map.

Status Grid

/ Disk Usage	/ Host Disk Usage	404 Errors	Active Connections	Active Directory Server	ActiveHealthChecks 1min	ActiveHealthChecks 3min	ActiveServiceCheckDelta 1min	Apache	Apache 404 Errors	Apache Web Server	Autoscale Activity	Autoscale Availability	Autoscale Latency	BGP Process - HQ Monitoring	BGP Process - HQ Network-De	BGP Process - Local Services	Bandwidth Spike	Blocklist Status	CPU Credit Balance	CPU Credit Usage	CPU Datas	CPU Usage	CPU Usage for Vmhost	CPU Utilization	Capacity Planning - HOST	Check API URL Status	Cloud VM Status	Cron Scheduling Daemon	Current Load	Current Users	DNS	DNS IP Hatch - 2024.de	DNS IP Hatch - fethi.com	DNS IP Hatch - molleweg.org	DNS IP Hatch - pdam.mwah	DNS IP Hatch - www.mwah.de
1-156.177.103.static.gtpkcbpl.in	1.200.169.192.host.secureserver.net	10.0.0.0.0	10.0.0.1	10.0.254.254	10.0.70.98	10.10.1.7	10.10.10.10	10.10.2.34	10.10.30.1	10.11.1.25	10.11.32.140	10.12.4.61	10.159.129.110																							

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



CONCLUSION:

Hence, we understood Nagios. It is a powerful monitoring tool, provided valuable insights into its capabilities and benefits for effective system monitoring and management.

ASSIGNMENT 10

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

LO2. To deploy single and multiple container applications and manage application deployments with rollouts in Kubernetes.

THEORY:

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

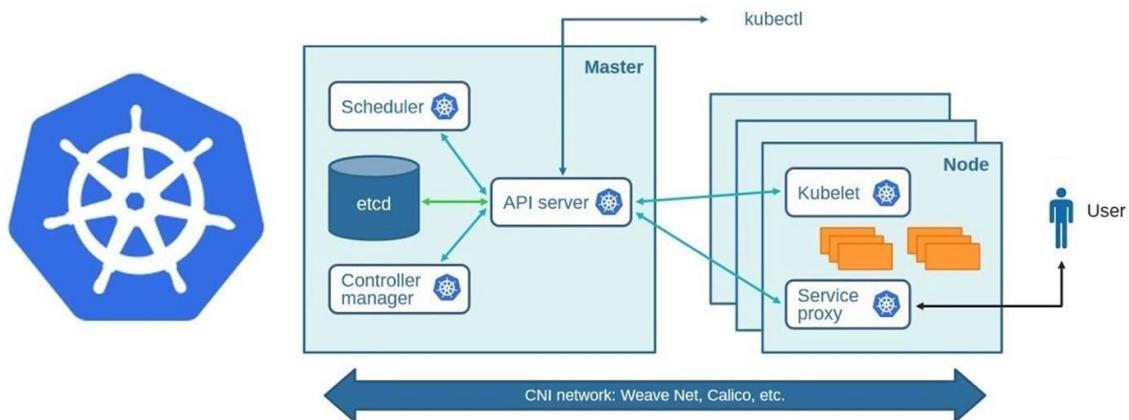
It schedules, runs, and manages isolated containers that are running on virtual/physical/cloud machines.

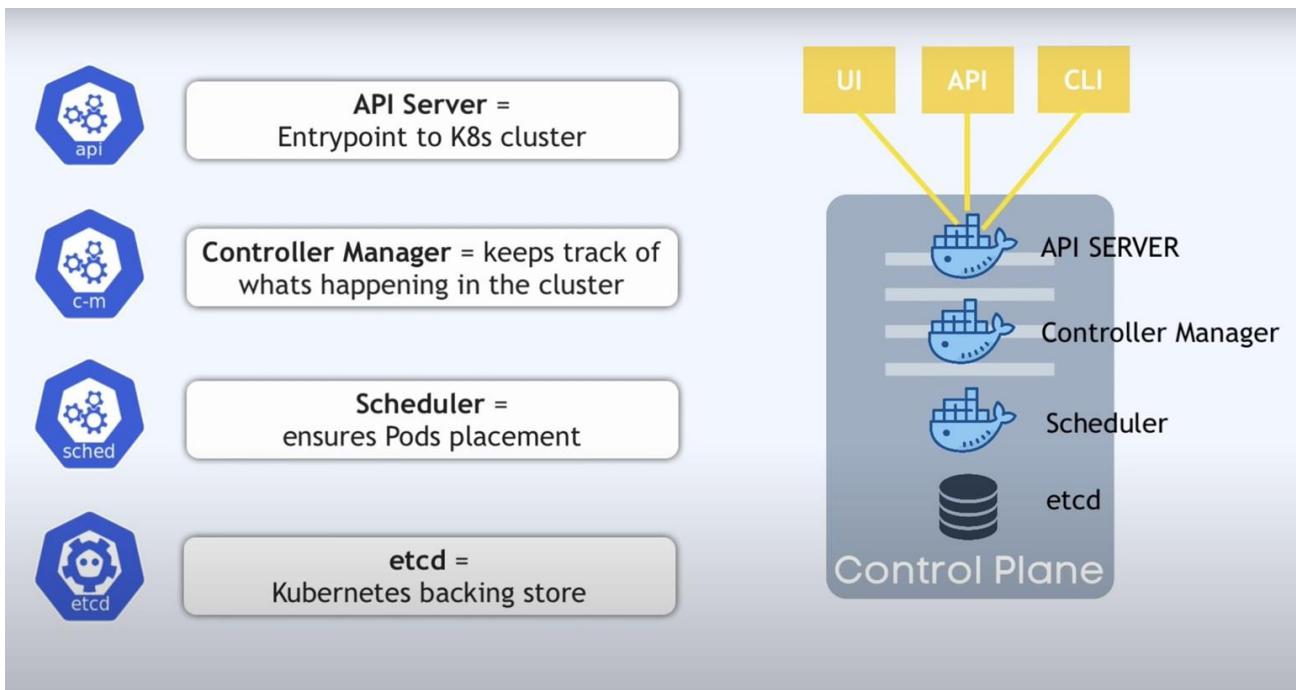
All top cloud providers support Kubernetes.

One popular name for Kubernetes is K8s.

ARCHITECTURE

Kubernetes





Working with Kubernetes

- We create a Manifest (.yml) file
- Apply those to cluster (to master) to bring it into the desired state.
- POD runs on a node, which is controlled by the master.

● Role of Master Node

- Kubernetes cluster contains containers running or Bare Metal / VM instances/cloud instances/ all mix.
- Kubernetes designates one or more of these as masters and all others as workers.
- The master is now going to run a set of K8s processes. These processes will ensure the smooth functioning of the cluster. These processes are called the ‘Control Plane’.
- Can be Multi-Master for high availability.
- Master runs control plane to run cluster smoothly.

● Components of Control Plane

■ Kube-api-server → (For all communications)

- This api-server interacts directly with the user (i.e we apply .yml or .json manifest to kube-api-server)
- This kube-api-server is meant to scale automatically as per load.
- Kube-api-server is the front end of the control plane.

■ etcd

- Stores metadata and status of the cluster.
- etcd is a consistent and high-available store (key-value-store)

- Source of truth for cluster state (info about the state of the cluster)

→ etcd has the following features

1. Fully Replicated → The entire state is available on every node in the cluster.
2. Secure → Implements automatic TLS with optional client-certificate authentication.
3. Fast → Benchmarked at 10,000 writes per second.

■ Kube-scheduler (action)

- When users request the creation & management of Pods, Kube-scheduler is going to take action on these requests.
- Handles POD creation and Management.
- Kube-scheduler match/assign any node to create and run pods.
- A scheduler watches for newly created pods that have no node assigned. For every pod that the scheduler discovers, the scheduler becomes responsible for finding the best node for that pod to run.
- The scheduler gets the information for hardware configuration from configuration files and schedules the Pods on nodes accordingly.

■ Controller-Manager

- Make sure the actual state of the cluster matches the desired state.

→ Two possible choices for controller manager—

1. If K8s is on the cloud, then it will be a cloud controller manager.
2. If K8s is on non-cloud, then it will be kube-controller-manager.

Components on the master that runs the controller

Node Controller → For checking the cloud provider to determine if a node has been detected in the cloud after it stops responding.

Route-Controller → Responsible for setting up a network, and routes on your cloud.

Service-Controller → Responsible for load Balancers on your cloud against services of type Load Balancer.

Volume-Controller → For creating, attaching, and mounting volumes and interacting with the cloud provider to orchestrate volume.

■ Nodes (Kubelet and Container Engine)

- Node is going to run 3 important pieces of software/process.

Kubelet

- The agent running on the node.
- Listens to Kubernetes master (eg- Pod creation request).
- Use port 10255.
- Send success/Fail reports to master.

Container Engine

- Works with kubelet
 - Pulling images
 - Start/Stop Containers
 - Exposing containers on ports specified in the manifest.

Kube-Proxy

- Assign IP to each pod.
 - It is required to assign IP addresses to Pods (dynamic)
 - Kube-proxy runs on each node & this makes sure that each pod will get its unique IP Address.
 - These 3 components collectively consist of ‘node’.

INSTALLATION:

1. Install Docker

```

Activities Terminal Oct 14 22:11 • prasad@prasad-VirtualBox:-
update      Update configuration of one or more containers
walt       Block until one or more containers stop, then print their exit codes

Global Options:
  -c, --config string   Location of client config files (default "/home/prasad/.docker")
  -H, --host list        Daemon socket to connect to
  -l, --log-level string Set the logging level ("debug", "info", "warn", "error", "fatal") (default "info")
  --tls                Use TLS; implied by --tlsv1.2
  --tlscacert string   Trust certs signed only by this CA (default "/home/prasad/.docker/ca.pem")
  --tlscert string     Path to TLS cert file (default "/home/prasad/.docker/cert.pem")
  --tlskey string       Path to TLS key file (default "/home/prasad/.docker/key.pem")
  --tlsv1.2              Use TLS and verify the remote
  -v, --version          Print version information and quit

Run 'docker COMMAND --help' for more information on a command.

For more help on how to use Docker, head to https://docs.docker.com/go/guides/
prasad@prasad-VirtualBox:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
719385e32844: Pull complete
Digest: sha256:88ec0aca3ec199d37ea7f3588f4518c25f9d34f58ce9a0df68429c5af48e8d
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (and)
 3. The Docker daemon created a new container from that image which runs the
 executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
 to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
prasad@prasad-VirtualBox:~$ 

```

```

Activities Terminal Oct 14 22:11 • prasad@prasad-VirtualBox:-
Unpacking curl (7.68.0-ubuntu2.20) ...
Setting up curl (7.68.0-ubuntu2.20) ...
Processing triggers for man-db (2.9.1-1) ...
prasad@prasad-VirtualBox:~$ sudo install -n 0755 -d /etc/apt/keyrings
prasad@prasad-VirtualBox:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
prasad@prasad-VirtualBox:~$ echo "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" > /etc/apt/sources.list.d/docker.list
> sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
prasad@prasad-VirtualBox:~$ sudo apt-get update
Get:1 https://download.docker.com/linux/ubuntu focal InRelease [57.7 kB]
Get:2 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [33.3 kB]
Hit:3 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:4 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:5 http://archive.ubuntu.com/ubuntu focal-security InRelease
Fetched 91.0 kB in 2s (59.6 kB/s)
Reading package lists...
Done
prasad@prasad-VirtualBox:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Reading package lists...
Reading package lists...
Building dependency tree
Reading state information...
The following packages were automatically installed and are no longer required:
  adium-theme-ubuntu-command-not-found-data cpp-7 diffstat enhancet example-content fwupdatd gcc-7-base gcc-8-base gettext gir1.2-goa-1.0 gir1.2-gtksource-3.0 gir1.2-mutter-2 gnome-software-common
  gnome-user-guide guile-2.0-libs ifupdown intltool-debian iptutils-arping libappstream-glib8 libapt-pkg-perl libarchive-zip-perl libargon2-0 libart-2.0-2 libasync-mergepoint-perl
  libbb-hooks-endoscope perl libbb-hooks-op-check perl libboost-filesystem-1.65.1 libboost-thread-1.65.1
  libbbapi0 libcamel-1.1 libcapnp0 libccache0 libcdio0 libcurl4 libdash0 libdwarf0 libelf0 libevent0 libfdisk0 libfontconfig0 libgcc1 libgdbm0 libgccasan0 libgccas0 libgccjit0 libgccjit0-0.1 libgccjit0-0.2 libgccjit0-0.3 libgccjit0-0.4 libgccjit0-0.5 libgccjit0-0.6 libgccjit0-0.7 libgccjit0-0.8 libgccjit0-0.9 libgccjit0-0.10 libgccjit0-0.11 libgccjit0-0.12 libgccjit0-0.13 libgccjit0-0.14 libgccjit0-0.15 libgccjit0-0.16 libgccjit0-0.17 libgccjit0-0.18 libgccjit0-0.19 libgccjit0-0.20 libgccjit0-0.21 libgccjit0-0.22 libgccjit0-0.23 libgccjit0-0.24 libgccjit0-0.25 libgccjit0-0.26 libgccjit0-0.27 libgccjit0-0.28 libgccjit0-0.29 libgccjit0-0.30 libgccjit0-0.31 libgccjit0-0.32 libgccjit0-0.33 libgccjit0-0.34 libgccjit0-0.35 libgccjit0-0.36 libgccjit0-0.37 libgccjit0-0.38 libgccjit0-0.39 libgccjit0-0.40 libgccjit0-0.41 libgccjit0-0.42 libgccjit0-0.43 libgccjit0-0.44 libgccjit0-0.45 libgccjit0-0.46 libgccjit0-0.47 libgccjit0-0.48 libgccjit0-0.49 libgccjit0-0.50 libgccjit0-0.51 libgccjit0-0.52 libgccjit0-0.53 libgccjit0-0.54 libgccjit0-0.55 libgccjit0-0.56 libgccjit0-0.57 libgccjit0-0.58 libgccjit0-0.59 libgccjit0-0.60 libgccjit0-0.61 libgccjit0-0.62 libgccjit0-0.63 libgccjit0-0.64 libgccjit0-0.65 libgccjit0-0.66 libgccjit0-0.67 libgccjit0-0.68 libgccjit0-0.69 libgccjit0-0.70 libgccjit0-0.71 libgccjit0-0.72 libgccjit0-0.73 libgccjit0-0.74 libgccjit0-0.75 libgccjit0-0.76 libgccjit0-0.77 libgccjit0-0.78 libgccjit0-0.79 libgccjit0-0.80 libgccjit0-0.81 libgccjit0-0.82 libgccjit0-0.83 libgccjit0-0.84 libgccjit0-0.85 libgccjit0-0.86 libgccjit0-0.87 libgccjit0-0.88 libgccjit0-0.89 libgccjit0-0.90 libgccjit0-0.91 libgccjit0-0.92 libgccjit0-0.93 libgccjit0-0.94 libgccjit0-0.95 libgccjit0-0.96 libgccjit0-0.97 libgccjit0-0.98 libgccjit0-0.99 libgccjit0-0.100 libgccjit0-0.101 libgccjit0-0.102 libgccjit0-0.103 libgccjit0-0.104 libgccjit0-0.105 libgccjit0-0.106 libgccjit0-0.107 libgccjit0-0.108 libgccjit0-0.109 libgccjit0-0.110 libgccjit0-0.111 libgccjit0-0.112 libgccjit0-0.113 libgccjit0-0.114 libgccjit0-0.115 libgccjit0-0.116 libgccjit0-0.117 libgccjit0-0.118 libgccjit0-0.119 libgccjit0-0.120 libgccjit0-0.121 libgccjit0-0.122 libgccjit0-0.123 libgccjit0-0.124 libgccjit0-0.125 libgccjit0-0.126 libgccjit0-0.127 libgccjit0-0.128 libgccjit0-0.129 libgccjit0-0.130 libgccjit0-0.131 libgccjit0-0.132 libgccjit0-0.133 libgccjit0-0.134 libgccjit0-0.135 libgccjit0-0.136 libgccjit0-0.137 libgccjit0-0.138 libgccjit0-0.139 libgccjit0-0.140 libgccjit0-0.141 libgccjit0-0.142 libgccjit0-0.143 libgccjit0-0.144 libgccjit0-0.145 libgccjit0-0.146 libgccjit0-0.147 libgccjit0-0.148 libgccjit0-0.149 libgccjit0-0.150 libgccjit0-0.151 libgccjit0-0.152 libgccjit0-0.153 libgccjit0-0.154 libgccjit0-0.155 libgccjit0-0.156 libgccjit0-0.157 libgccjit0-0.158 libgccjit0-0.159 libgccjit0-0.160 libgccjit0-0.161 libgccjit0-0.162 libgccjit0-0.163 libgccjit0-0.164 libgccjit0-0.165 libgccjit0-0.166 libgccjit0-0.167 libgccjit0-0.168 libgccjit0-0.169 libgccjit0-0.170 libgccjit0-0.171 libgccjit0-0.172 libgccjit0-0.173 libgccjit0-0.174 libgccjit0-0.175 libgccjit0-0.176 libgccjit0-0.177 libgccjit0-0.178 libgccjit0-0.179 libgccjit0-0.180 libgccjit0-0.181 libgccjit0-0.182 libgccjit0-0.183 libgccjit0-0.184 libgccjit0-0.185 libgccjit0-0.186 libgccjit0-0.187 libgccjit0-0.188 libgccjit0-0.189 libgccjit0-0.190 libgccjit0-0.191 libgccjit0-0.192 libgccjit0-0.193 libgccjit0-0.194 libgccjit0-0.195 libgccjit0-0.196 libgccjit0-0.197 libgccjit0-0.198 libgccjit0-0.199 libgccjit0-0.200 libgccjit0-0.201 libgccjit0-0.202 libgccjit0-0.203 libgccjit0-0.204 libgccjit0-0.205 libgccjit0-0.206 libgccjit0-0.207 libgccjit0-0.208 libgccjit0-0.209 libgccjit0-0.210 libgccjit0-0.211 libgccjit0-0.212 libgccjit0-0.213 libgccjit0-0.214 libgccjit0-0.215 libgccjit0-0.216 libgccjit0-0.217 libgccjit0-0.218 libgccjit0-0.219 libgccjit0-0.220 libgccjit0-0.221 libgccjit0-0.222 libgccjit0-0.223 libgccjit0-0.224 libgccjit0-0.225 libgccjit0-0.226 libgccjit0-0.227 libgccjit0-0.228 libgccjit0-0.229 libgccjit0-0.230 libgccjit0-0.231 libgccjit0-0.232 libgccjit0-0.233 libgccjit0-0.234 libgccjit0-0.235 libgccjit0-0.236 libgccjit0-0.237 libgccjit0-0.238 libgccjit0-0.239 libgccjit0-0.240 libgccjit0-0.241 libgccjit0-0.242 libgccjit0-0.243 libgccjit0-0.244 libgccjit0-0.245 libgccjit0-0.246 libgccjit0-0.247 libgccjit0-0.248 libgccjit0-0.249 libgccjit0-0.250 libgccjit0-0.251 libgccjit0-0.252 libgccjit0-0.253 libgccjit0-0.254 libgccjit0-0.255 libgccjit0-0.256 libgccjit0-0.257 libgccjit0-0.258 libgccjit0-0.259 libgccjit0-0.260 libgccjit0-0.261 libgccjit0-0.262 libgccjit0-0.263 libgccjit0-0.264 libgccjit0-0.265 libgccjit0-0.266 libgccjit0-0.267 libgccjit0-0.268 libgccjit0-0.269 libgccjit0-0.270 libgccjit0-0.271 libgccjit0-0.272 libgccjit0-0.273 libgccjit0-0.274 libgccjit0-0.275 libgccjit0-0.276 libgccjit0-0.277 libgccjit0-0.278 libgccjit0-0.279 libgccjit0-0.280 libgccjit0-0.281 libgccjit0-0.282 libgccjit0-0.283 libgccjit0-0.284 libgccjit0-0.285 libgccjit0-0.286 libgccjit0-0.287 libgccjit0-0.288 libgccjit0-0.289 libgccjit0-0.290 libgccjit0-0.291 libgccjit0-0.292 libgccjit0-0.293 libgccjit0-0.294 libgccjit0-0.295 libgccjit0-0.296 libgccjit0-0.297 libgccjit0-0.298 libgccjit0-0.299 libgccjit0-0.300 libgccjit0-0.301 libgccjit0-0.302 libgccjit0-0.303 libgccjit0-0.304 libgccjit0-0.305 libgccjit0-0.306 libgccjit0-0.307 libgccjit0-0.308 libgccjit0-0.309 libgccjit0-0.310 libgccjit0-0.311 libgccjit0-0.312 libgccjit0-0.313 libgccjit0-0.314 libgccjit0-0.315 libgccjit0-0.316 libgccjit0-0.317 libgccjit0-0.318 libgccjit0-0.319 libgccjit0-0.320 libgccjit0-0.321 libgccjit0-0.322 libgccjit0-0.323 libgccjit0-0.324 libgccjit0-0.325 libgccjit0-0.326 libgccjit0-0.327 libgccjit0-0.328 libgccjit0-0.329 libgccjit0-0.330 libgccjit0-0.331 libgccjit0-0.332 libgccjit0-0.333 libgccjit0-0.334 libgccjit0-0.335 libgccjit0-0.336 libgccjit0-0.337 libgccjit0-0.338 libgccjit0-0.339 libgccjit0-0.340 libgccjit0-0.341 libgccjit0-0.342 libgccjit0-0.343 libgccjit0-0.344 libgccjit0-0.345 libgccjit0-0.346 libgccjit0-0.347 libgccjit0-0.348 libgccjit0-0.349 libgccjit0-0.350 libgccjit0-0.351 libgccjit0-0.352 libgccjit0-0.353 libgccjit0-0.354 libgccjit0-0.355 libgccjit0-0.356 libgccjit0-0.357 libgccjit0-0.358 libgccjit0-0.359 libgccjit0-0.360 libgccjit0-0.361 libgccjit0-0.362 libgccjit0-0.363 libgccjit0-0.364 libgccjit0-0.365 libgccjit0-0.366 libgccjit0-0.367 libgccjit0-0.368 libgccjit0-0.369 libgccjit0-0.370 libgccjit0-0.371 libgccjit0-0.372 libgccjit0-0.373 libgccjit0-0.374 libgccjit0-0.375 libgccjit0-0.376 libgccjit0-0.377 libgccjit0-0.378 libgccjit0-0.379 libgccjit0-0.380 libgccjit0-0.381 libgccjit0-0.382 libgccjit0-0.383 libgccjit0-0.384 libgccjit0-0.385 libgccjit0-0.386 libgccjit0-0.387 libgccjit0-0.388 libgccjit0-0.389 libgccjit0-0.390 libgccjit0-0.391 libgccjit0-0.392 libgccjit0-0.393 libgccjit0-0.394 libgccjit0-0.395 libgccjit0-0.396 libgccjit0-0.397 libgccjit0-0.398 libgccjit0-0.399 libgccjit0-0.400 libgccjit0-0.401 libgccjit0-0.402 libgccjit0-0.403 libgccjit0-0.404 libgccjit0-0.405 libgccjit0-0.406 libgccjit0-0.407 libgccjit0-0.408 libgccjit0-0.409 libgccjit0-0.410 libgccjit0-0.411 libgccjit0-0.412 libgccjit0-0.413 libgccjit0-0.414 libgccjit0-0.415 libgccjit0-0.416 libgccjit0-0.417 libgccjit0-0.418 libgccjit0-0.419 libgccjit0-0.420 libgccjit0-0.421 libgccjit0-0.422 libgccjit0-0.423 libgccjit0-0.424 libgccjit0-0.425 libgccjit0-0.426 libgccjit0-0.427 libgccjit0-0.428 libgccjit0-0.429 libgccjit0-0.430 libgccjit0-0.431 libgccjit0-0.432 libgccjit0-0.433 libgccjit0-0.434 libgccjit0-0.435 libgccjit0-0.436 libgccjit0-0.437 libgccjit0-0.438 libgccjit0-0.439 libgccjit0-0.440 libgccjit0-0.441 libgccjit0-0.442 libgccjit0-0.443 libgccjit0-0.444 libgccjit0-0.445 libgccjit0-0.446 libgccjit0-0.447 libgccjit0-0.448 libgccjit0-0.449 libgccjit0-0.450 libgccjit0-0.451 libgccjit0-0.452 libgccjit0-0.453 libgccjit0-0.454 libgccjit0-0.455 libgccjit0-0.456 libgccjit0-0.457 libgccjit0-0.458 libgccjit0-0.459 libgccjit0-0.460 libgccjit0-0.461 libgccjit0-0.462 libgccjit0-0.463 libgccjit0-0.464 libgccjit0-0.465 libgccjit0-0.466 libgccjit0-0.467 libgccjit0-0.468 libgccjit0-0.469 libgccjit0-0.470 libgccjit0-0.471 libgccjit0-0.472 libgccjit0-0.473 libgccjit0-0.474 libgccjit0-0.475 libgccjit0-0.476 libgccjit0-0.477 libgccjit0-0.478 libgccjit0-0.479 libgccjit0-0.480 libgccjit0-0.481 libgccjit0-0.482 libgccjit0-0.483 libgccjit0-0.484 libgccjit0-0.485 libgccjit0-0.486 libgccjit0-0.487 libgccjit0-0.488 libgccjit0-0.489 libgccjit0-0.490 libgccjit0-0.491 libgccjit0-0.492 libgccjit0-0.493 libgccjit0-0.494 libgccjit0-0.495 libgccjit0-0.496 libgccjit0-0.497 libgccjit0-0.498 libgccjit0-0.499 libgccjit0-0.500 libgccjit0-0.501 libgccjit0-0.502 libgccjit0-0.503 libgccjit0-0.504 libgccjit0-0.505 libgccjit0-0.506 libgccjit0-0.507 libgccjit0-0.508 libgccjit0-0.509 libgccjit0-0.510 libgccjit0-0.511 libgccjit0-0.512 libgccjit0-0.513 libgccjit0-0.514 libgccjit0-0.515 libgccjit0-0.516 libgccjit0-0.517 libgccjit0-0.518 libgccjit0-0.519 libgccjit0-0.520 libgccjit0-0.521 libgccjit0-0.522 libgccjit0-0.523 libgccjit0-0.524 libgccjit0-0.525 libgccjit0-0.526 libgccjit0-0.527 libgccjit0-0.528 libgccjit0-0.529 libgccjit0-0.530 libgccjit0-0.531 libgccjit0-0.532 libgccjit0-0.533 libgccjit0-0.534 libgccjit0-0.535 libgccjit0-0.536 libgccjit0-0.537 libgccjit0-0.538 libgccjit0-0.539 libgccjit0-0.540 libgccjit0-0.541 libgccjit0-0.542 libgccjit0-0.543 libgccjit0-0.544 libgccjit0-0.545 libgccjit0-0.546 libgccjit0-0.547 libgccjit0-0.548 libgccjit0-0.549 libgccjit0-0.550 libgccjit0-0.551 libgccjit0-0.552 libgccjit0-0.553 libgccjit0-0.554 libgccjit0-0.555 libgccjit0-0.556 libgccjit0-0.557 libgccjit0-0.558 libgccjit0-0.559 libgccjit0-0.560 libgccjit0-0.561 libgccjit0-0.562 libgccjit0-0.563 libgccjit0-0.564 libgccjit0-0.565 libgccjit0-0.566 libgccjit0-0.567 libgccjit0-0.568 libgccjit0-0.569 libgccjit0-0.570 libgccjit0-0.571 libgccjit0-0.572 libgccjit0-0.573 libgccjit0-0.574 libgccjit0-0.575 libgccjit0-0.576 libgccjit0-0.577 libgccjit0-0.578 libgccjit0-0.579 libgccjit0-0.580 libgccjit0-0.581 libgccjit0-0.582 libgccjit0-0.583 libgccjit0-0.584 libgccjit0-0.585 libgccjit0-0.586 libgccjit0-0.587 libgccjit0-0.588 libgccjit0-0.589 libgccjit0-0.590 libgccjit0-0.591 libgccjit0-0.592 libgccjit0-0.593 libgccjit0-0.594 libgccjit0-0.595 libgccjit0-0.596 libgccjit0-0.597 libgccjit0-0.598 libgccjit0-0.599 libgccjit0-0.510 libgccjit0-0.511 libgccjit0-0.512 libgccjit0-0.513 libgccjit0-0.514 libgccjit0-0.515 libgccjit0-0.516 libgccjit0-0.517 libgccjit0-0.518 libgccjit0-0.519 libgccjit0-0.520 libgccjit0-0.521 libgccjit0-0.522 libgccjit0-0.523 libgccjit0-0.524 libgccjit0-0.525 libgccjit0-0.526 libgccjit0-0.527 libgccjit0-0.528 libgccjit0-0.529 libgccjit0-0.5210 libgccjit0-0.5211 libgccjit0-0.5212 libgccjit0-0.5213 libgccjit0-0.5214 libgccjit0-0.5215 libgccjit0-0.5216 libgccjit0-0.5217 libgccjit0-0.5218 libgccjit0-0.5219 libgccjit0-0.5220 libgccjit0-0.5221 libgccjit0-0.5222 libgccjit0-0.5223 libgccjit0-0.5224 libgccjit0-0.5225 libgccjit0-0.5226 libgccjit0-0.5227 libgccjit0-0.5228 libgccjit0-0.5229 libgccjit0-0.52210 libgccjit0-0.52211 libgccjit0-0.52212 libgccjit0-0.52213 libgccjit0-0.52214 libgccjit0-0.52215 libgccjit0-0.52216 libgccjit0-0.52217 libgccjit0-0.52218 libgccjit0-0.52219 libgccjit0-0.52220 libgccjit0-0.52221 libgccjit0-0.52222 libgccjit0-0.52223 libgccjit0-0.52224 libgccjit0-0.52225 libgccjit0-0.52226 libgccjit0-0.52227 libgccjit0-0.52228 libgccjit0-0.52229 libgccjit0-0.522100 libgccjit0-0.522110 libgccjit0-0.522120 libgccjit0-0.522130 libgccjit0-0.522140 libgccjit0-0.522150 libgccjit0-0.522160 libgccjit0-0.522170 libgccjit0-0.522180 libgccjit0-0.522190 libgccjit0-0.522200 libgccjit0-0.522210 libgccjit0-0.522220 libgccjit0-0.522230 libgccjit0-0.522240 libgccjit0-0.522250 libgccjit0-0.522260 libgccjit0-0.522270 libgccjit0-0.522280 libgccjit0-0.522290 libgccjit0-0.522101 libgccjit0-0.522111 libgccjit0-0.522121 libgccjit0-0.522131 libgccjit0-0.522141 libgccjit0-0.522151 libgccjit0-0.522161 libgccjit0-0.522171 libgccjit0-0.522181 libgccjit0-0.522191 libgccjit0-0.522201 libgccjit0-0.522211 libgccjit0-0.522221 libgccjit0-0.522231 libgccjit0-0.522241 libgccjit0-0.522251 libgccjit0-0.522261 libgccjit0-0.522271 libgccjit0-0.522281 libgccjit0-0.522291 libgccjit0-0.522102 libgccjit0-0.522112 libgccjit0-0.522122 libgccjit0-0.522132 libgccjit0-0.522142 libgccjit0-0.522152 libgccjit0-0.522162 libgccjit0-0.522172 libgccjit0-0.522182 libgccjit0-0.522192 libgccjit0-0.522202 libgccjit0-0.522212 libgccjit0-0.522222 libgccjit0-0.522232 libgccjit0-0.522242 libgccjit0-0.522252 libgccjit0-0.522262 libgccjit0-0.522272 libgccjit0-0.522282 libgccjit0-0.522292 libgccjit0-0.522103 libgccjit0-0.522113 libgccjit0-0.522123 libgccjit0-0.522133 libgccjit0-0.522143 libgccjit0-0.522153 libgccjit0-0.522163 libgccjit0-0.522173 libgccjit0-0.522183 libgccjit0-0.522193 libgccjit0-0.522203 libgccjit0-0.522213 libgccjit0-0.522223 libgccjit0-0.522233 libgccjit0-0.522243 libgccjit0-0.522253 libgccjit0-0.522263 libgccjit0-0.522273 libgccjit0-0.522283 libgccjit0-0.522293 libgccjit0-0.522104 libgccjit0-0.522114 libgccjit0-0.522124 libgccjit0-0.522134 libgccjit0-0.522144 libgccjit0-0.522154 libgccjit0-0.522164 libgccjit0-0.522174 libgccjit0-0.522184 libgccjit0-0.522194 libgccjit0-0.522204 libgccjit0-0.522214 libgccjit0-0.522224 libgccjit0-0.522234 libgccjit0-0.522244 libgccjit0-0.522254 libgccjit0-0.522264 libgccjit0-0.522274 libgccjit0-0.522284 libgccjit0-0.522294 libgccjit0-0.522105 libgccjit0-0.522115 libgccjit0-0.522125 libgccjit0-0.522135 libgccjit0-0.522145 libgccjit0-0.522155 libgccjit0-0.522165 libgccjit0-0.522175 libgccjit0-0.522185 libgccjit0-0.522195 libgccjit0-0.522205 libgccjit0-0.522215 libgccjit0-0.522225 libgccjit0-0.522235 libgccjit0-0.522245 libgccjit0-0.522255 libgccjit0-0.522265 libgccjit0-0.522275 libgccjit0-0.522285 libgccjit0-0.522295 libgccjit0-0.522106 libgccjit0-0.522116 libgccjit0-0.522126 libgccjit0-0.522136 libgccjit0-0.522146 libgccjit0-0.522156 libgccjit0-0.522166 libgccjit0-0.522176 libgccjit0-0.522186 libgccjit0-0.522196 libgccjit0-0.522206 libgccjit0-0.522216 libgccjit0-0.522226 libgccjit0-0.522236 libgccjit0-0.522246 libgccjit0-0.522256 libgccjit0-0.522266 libgccjit0-0.522276 libgccjit0-0.522286 libgccjit0-0.522296 libgccjit0-0.522107 libgccjit0-0.522117 libgccjit0-0.522127 libgccjit0-0.522137 libgccjit0-0.522147 libgccjit0-0.522157 libgccjit0-0.522167 libgccjit0-0.522177 libgccjit0-0.522187 libgccjit0-0.522197 libgccjit0-0.522207 libgccjit0-0.522217 libgccjit0-0.522227 libgccjit0-0.522237 libgccjit0-0.522247 libgccjit0-0.522257 libgccjit0-0.522267 libgccjit0-0.522277 libgccjit0-0.522287 libgccjit0-0.522297 libgccjit0-0.522108 libgccjit0-0.522118 libgccjit0-0.522128 libgccjit0-0.522138 libgccjit0-0.522148 libgccjit0-0.522158 libgccjit0-0.522168 libgccjit0-0.522178 libgccjit0-0.522188 libgccjit0-0.522198 libgccjit0-0.522208 libgccjit0-0.522218 libgccjit0-0.522228 libgccjit0-0.522238 libgccjit0-0.522248 libgccjit0-0.522258 libgccjit0-0.522268 libgccjit0-0.522278 libgccjit0-0.522288 libgccjit0-0.522298 libgccjit0-0.522109 libgccjit0-0.522119 libgccjit0-0.522129 libgccjit0-0.522139 libgccjit0-0.522149 libgccjit0-0.522159 libgccjit0-0.522169 libgccjit0-0.522179 libgccjit0-0.522189 libgccjit0-0.522199 libgccjit0-0.522209 libgccjit0-0.522219 libgccjit0-0.522229 libgccjit0-0.522239 libgccjit0-0.522249 libgccjit0-0.522259 libgccjit0-0.522269 libgccjit0-0.522279 libgccjit0-0.522289 libgccjit0-0.522299 libgccjit0-0.522110 libgccjit0-0.522120 libgccjit0-0.522130 libgccjit0-0.522140 libgccjit0-0.522150 libgccjit0-0.522160 libgccjit0-0.522170 libgccjit0-0.522180 libgccjit0-0.522190 libgccjit0-0.522200 libgccjit0-0.522210 libgccjit0-0.522220 libgccjit0-0.522230 libgccjit0-0.522240 libgccjit0-0.522250 libgccjit0-0.522260 libgccjit0-0.522270 libgccjit0-0.522280 libgccjit0-0.522290 libgccjit0-0.522111 libgccjit0-0.522121 libgccjit0-0.522131 libgccjit0-0.522141 libgccjit0-0.522151 libgccjit0-0.522161 libgccjit0-0.522171 libgccjit0-0.522181 libgccjit0-0.522191 libgccjit0-0.522201 libgccjit0-0.522211 libgccjit0-0.522221 libgccjit0-0.522231 libgccjit0-0.522241 libgccjit0-0.522251 libgccjit0-0.522261 libgccjit0-0.522271 libgccjit0-0.522281 libgccjit0-0.522291 libgccjit0-0.522112 libgccjit0-0.522122 libgccjit0-0.522132 libgccjit0-0.522142 libgccjit0-0.522152 libgccjit0-0.522162 libgccjit0-0.522172 libgccjit0-0.522182 libgccjit0-0.522192 libgccjit0-0.522202 libgccjit0-0.522212 libgccjit0-0.522222 libgccjit0-0.522232 libgccjit0-0.522242 libgccjit0-0.522252 libgccjit0-0.522262 libgccjit0-0.522272 libgccjit0-0.
```

```

Oct 14 22:27 ●
prasad@prasad-VirtualBox: ~
prasad@prasad-VirtualBox: $ curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
prasad@prasad-VirtualBox: ~
prasad@prasad-VirtualBox: $ ls
prasad@prasad-VirtualBox: ~
prasad@prasad-VirtualBox: $ sudo install minikube-linux-amd64 /usr/local/bin/minikube
[sudo] password for prasad:
prasad@prasad-VirtualBox: $ minikube start --driver=docker
minikube v1.31.2 on Ubuntu 20.04 (vbox/amd64)
Using the docker driver based on user configuration

  Exiting due to PROVIDER_DOCKER_NEWRGP: "docker version --format <no value>:<no value>:<no value>" exit status 1: permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock. Get "http://$Fvar%2Frun%2Fdocker.sock/v1.24/version": dial unix /var/run/docker.sock: connect: permission denied
  Suggestion: Add your user to the 'docker' group: 'sudo usermod -aG docker $USER && newgrp docker'
  Documentation: https://docs.docker.com/engine/install/linux-postinstall/
prasad@prasad-VirtualBox: $ sudo usermod -aG docker $USER && newgrp docker
prasad@prasad-VirtualBox: ~
prasad@prasad-VirtualBox: $ minikube start --driver=docker
minikube v1.31.2 on Ubuntu 20.04 (vbox/amd64)
Using the docker driver based on user configuration

  The requested memory allocation of 1971MB does not leave room for system overhead (total system memory: 1971MB). You may face stability issues.
  Suggestion: Start minikube with less memory allocated: 'minikube start --memory=1971mb'

  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 MB / 393.21 MB 100.00% 2.85 MiB
  > gcr.io/k8s-minikube/kicbase...: 447.62 MB / 447.62 MB 100.00% 2.99 MiB
  Creating docker container (CPUs=2, Memory=1971MB) ...

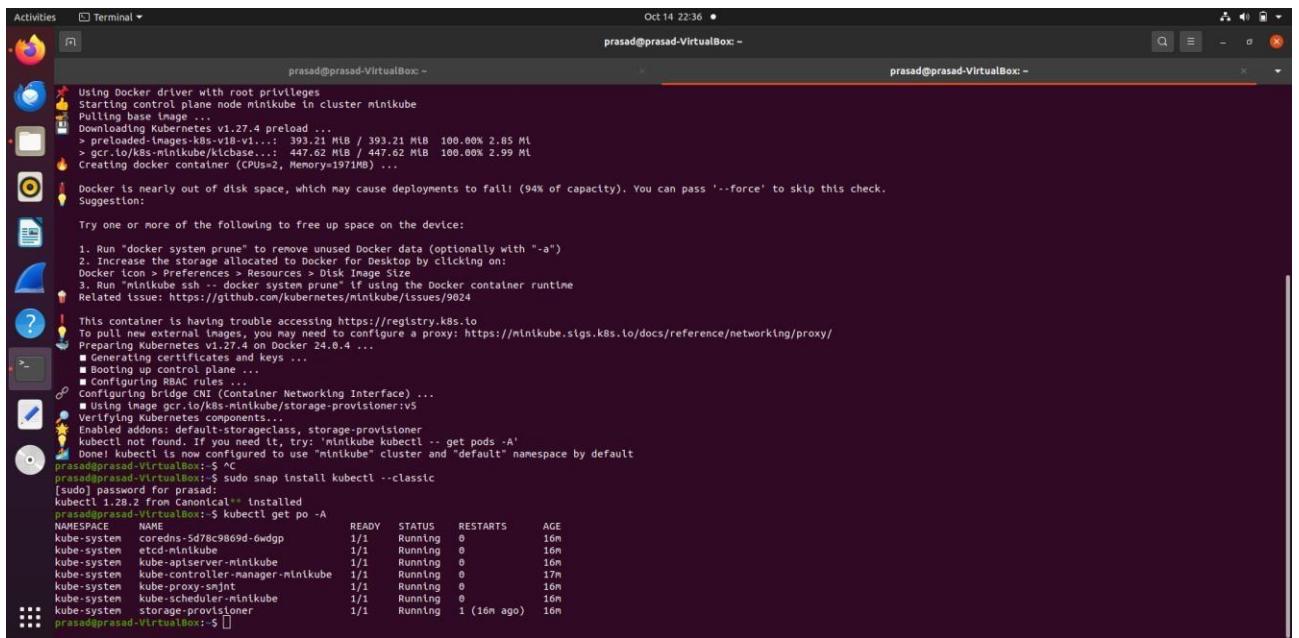
Docker is nearly out of disk space, which may cause deployments to fail! (94% of capacity). You can pass '--force' to skip this check.
Suggestion:

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 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: ~
prasad@prasad-VirtualBox: ~

```

3. Install kubectl



```

Activities Terminal Oct 14 22:36 •
prasad@prasad-VirtualBox: ~

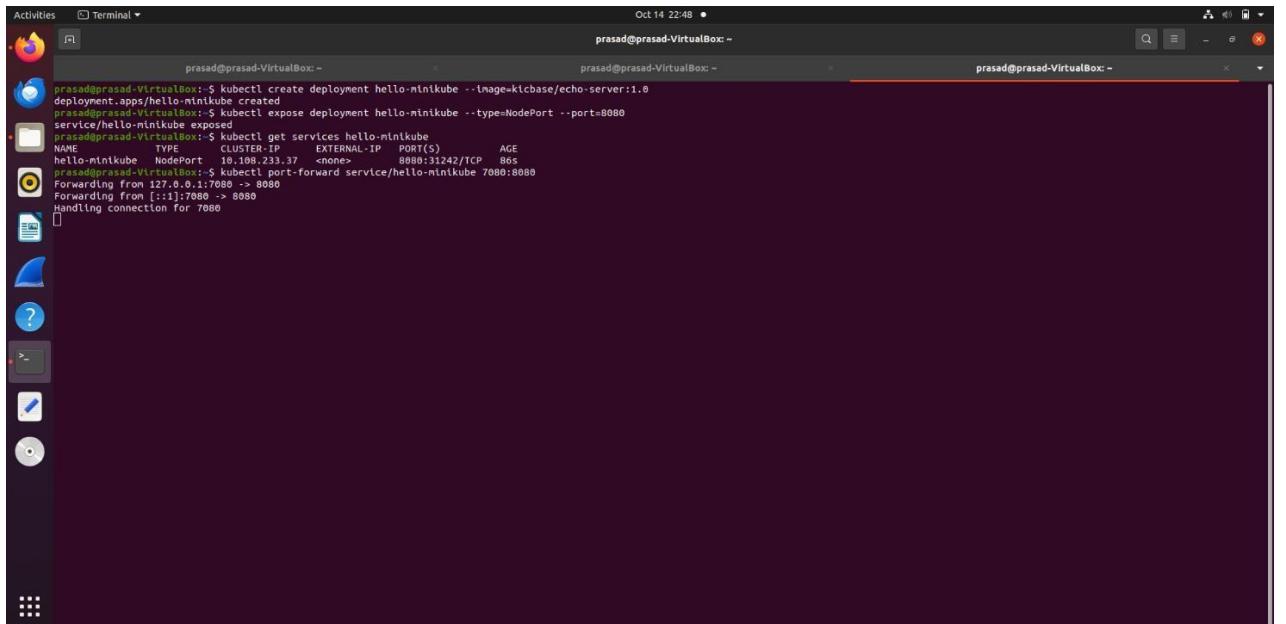
Using Docker driver with root privileges
Starting control plane node minikube in cluster minikube
Pulling base image ...
  Downloading Kubernetes v1.27.4 preload ...
    > preloaded-images-k8s-v18-v1...: 393.21 MB / 393.21 MB 100.00% 2.85 MiB
    > gcr.io/k8s-minikube/kicbase...: 447.62 MB / 447.62 MB 100.00% 2.99 MiB
Creating docker container (CPUs=2, Memory=1971MiB) ...
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/9624

This container is having trouble accessing https://registry.k8s.io
To resolve this 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...
  ■ Publishing default port forwarder class, storage-provisioner
  kubelet not found. If you need it, try: 'minikube kubectl -- get pods -A'
  Done! kubelet is now configured to use "minikube" cluster and "default" namespace by default
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
NAME          READY   STATUS    RESTARTS   AGE
coredns       1/1     Running   0          16m
etcd-minikube 1/1     Running   0          16m
kube-apiserver-minikube 1/1     Running   0          16m
kube-controller-manager-minikube 1/1     Running   0          17m
kube-proxy-smjnt 1/1     Running   0          16m
kube-scheduler-minikube 1/1     Running   0          16m
kube-storage-provisioner 1/1     Running   1 (16m ago)
prasad@prasad-VirtualBox: ~ $ 

```

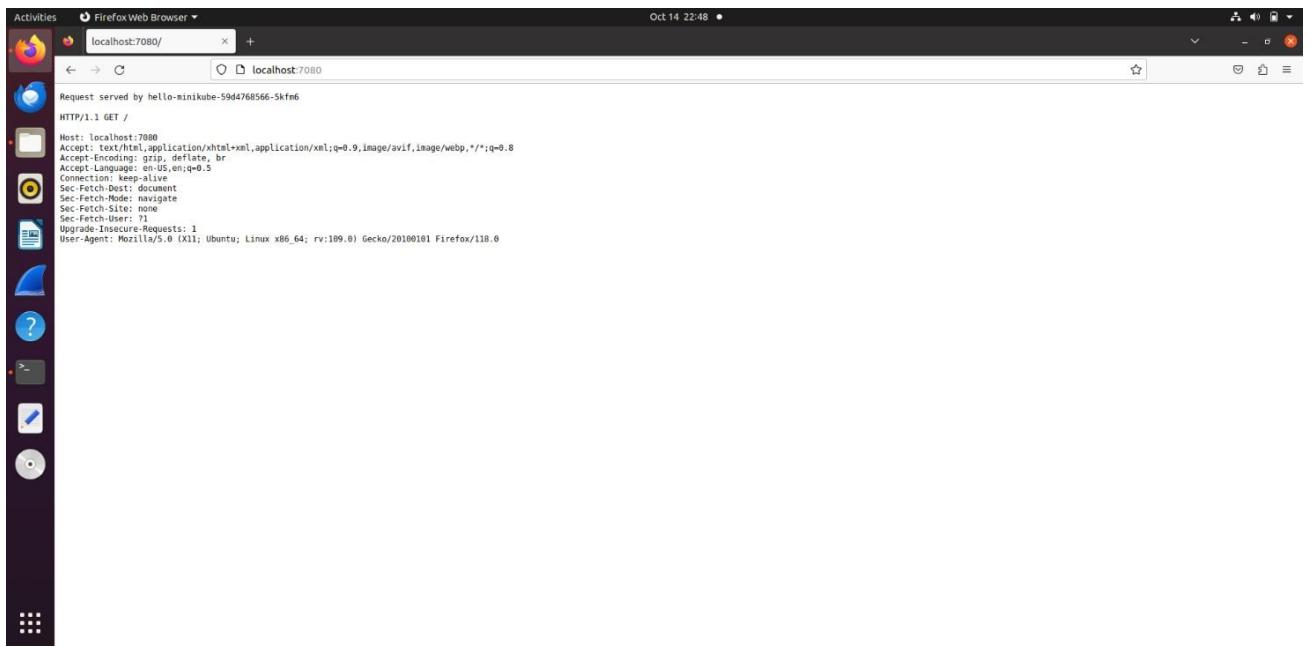
4. Create a sample deployment.



```

Activities Terminal Oct 14 22:48 •
prasad@prasad-VirtualBox: ~
prasad@prasad-VirtualBox: ~
prasad@prasad-VirtualBox: ~ $ kubectl create deployment hello-minikube --image=kicbase/echo-server:1.0
deployment.apps/hello-minikube created
prasad@prasad-VirtualBox: ~ $ kubectl expose deployment hello-minikube --type=NodePort --port:8080
service/hello-minikube exposed
prasad@prasad-VirtualBox: ~ $ kubectl get services hello-minikube
NAME           TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)        AGE
hello-minikube  NodePort   10.110.233.37   <none>        8080:31242/TCP   86s
prasad@prasad-VirtualBox: ~ $ kubectl port-forward service/hello-minikube 7080:8080
Forwarding from 127.0.0.1:7080 -> 8080
Forwarding from [::]:7080 -> 8080
Handling connection for 7080

```



CONCLUSION:

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

Written Assignment 1

1. What are the best security measures that you can take while using Kubernetes?

Ans:-

Securing a Kubernetes cluster is crucial to protect your applications, data, and infrastructure. Here are some best security measures to take while using Kubernetes:

Update Kubernetes Regularly:

Ensure you're using the latest stable version of Kubernetes with security patches and updates. Regular updates help mitigate known vulnerabilities.

Secure Kubernetes API Server:

Use strong authentication mechanisms like certificates or tokens.

Restrict access to the API server to authorized users and IP ranges.

Enable audit logs for monitoring API server access.

Implement RBAC (Role-Based Access Control):

Define specific roles and permissions for users and services to access resources. Follow the principle of least privilege, granting minimal permissions necessary for tasks.

Secure Network Communication:

Encrypt communication using TLS for all network traffic within the cluster.

Use network policies to control traffic between pods and namespaces.

Protect etcd (Cluster Data Store):

Enable encryption for etcd to secure sensitive data stored in the cluster.

Implement access controls to limit access to etcd.

Secure Container Images:

Use trusted sources for container images.

Scan images for vulnerabilities before deploying them in the cluster.

Isolate Sensitive Workloads:

Isolate sensitive workloads in separate namespaces.

Use network policies to control access to sensitive workloads.

Implement Pod Security Policies (PSP):

Define and enforce policies that dictate the security settings for pods.

Control pod capabilities, privileged access, and host namespaces.

Secure Service Accounts:

Use service accounts judiciously and limit their permissions.

Regularly review and audit service account usage.

Monitor and Audit Cluster Activity:

Use Kubernetes audit logs to monitor and track activities within the cluster.

Utilize external monitoring tools for advanced threat detection.

Implement Network Policies:

Define and enforce network policies to control pod-to-pod communication.

Allow only necessary traffic and deny unnecessary communication.

Regular Security Audits and Penetration Testing:

Conduct security audits and penetration testing to identify vulnerabilities.

Address the identified vulnerabilities promptly.

Backup and Disaster Recovery:

Implement regular backups of critical data and configurations.

Have a well-defined disaster recovery plan and test it periodically.

Educate Users and Administrators:

Educate users and administrators about secure practices and potential security threats.

Promote a security-conscious culture within the organization.

Engage Security Experts:

Consider engaging security experts or consultants to perform security assessments and provide recommendations.

By following these security measures, you can enhance the overall security posture of your Kubernetes environment and protect your applications and data from potential security threats and vulnerabilities.

Q.2 What are three 3 security techniques that can be used to protect data?

Ans:

Protecting data is a critical aspect of ensuring information security. Here are three security techniques that can be used to protect data:

Encryption:

Encryption is a fundamental technique used to protect data by encoding it in a way that only authorized individuals or systems can access and interpret it. The data is converted into ciphertext using cryptographic algorithms and can only be decrypted using the appropriate encryption key. There are two main types of encryption:

Symmetric Encryption: Uses a single secret key for both encryption and decryption.

Asymmetric Encryption (Public-key Encryption): Uses a pair of keys, a public key for encryption and a private key for decryption.

Access Control and Authorization:

Access control and authorization mechanisms ensure that only authorized users have access to specific data and resources. This involves defining user roles, permissions, and privileges based on the principle of least privilege. Key techniques include:

Role-Based Access Control (RBAC): Assigns permissions based on user roles within an organization.

Attribute-Based Access Control (ABAC): Access is granted or denied based on attributes associated with users, the resource, and the environment.

Multi-Factor Authentication (MFA): Requires multiple forms of authentication before granting access, adding an extra layer of security.

Data Masking and Anonymization:

Data masking (also known as data obfuscation) involves hiding original data with modified content to protect sensitive information while maintaining its format.

Anonymization goes a step further by irreversibly removing personally identifiable information (PII) from data sets. These techniques are used to share data for testing, analysis, or development purposes without exposing sensitive details. Common methods include:

Tokenization: Replaces sensitive data with unique tokens, retaining the format but making it meaningless.

Pseudonymization: Replaces identifiable information with pseudonyms, allowing for reversibility if needed for specific use cases.

These security techniques play a crucial role in safeguarding data from unauthorized access, breaches, and misuse, ensuring data confidentiality, integrity, and availability. Combining these techniques with a comprehensive data security strategy helps organizations maintain trust and compliance with regulatory requirements.

Q.3 How do you expose a service using ingress in Kubernetes?

Ans:

In the context of advanced DevOps practices, exposing a service using Ingress in Kubernetes is a critical aspect of managing and deploying applications in a highly efficient, scalable, and secure manner. Here's how using Ingress fits into advanced DevOps practices:

1. **Advanced Traffic Routing and Load Balancing**:**

Ingress in Kubernetes allows for advanced traffic routing and load balancing. Advanced DevOps teams leverage Ingress rules to route traffic based on various parameters like host, URL paths, HTTP methods, headers, etc. This enables sophisticated traffic distribution strategies, essential for high-availability and performance-sensitive applications.

2. **Microservices and Service Mesh Integration**:**

In advanced DevOps setups, applications are often built using microservices architecture. Ingress controllers can be configured to work seamlessly with service meshes like Istio, enabling advanced traffic management, security policies, and canary deployments within the service mesh environment.

3. **Advanced Security and SSL Termination**:**

Ingress in advanced DevOps encompasses robust security practices. SSL termination and encryption at the Ingress level are common patterns to offload SSL processing from backend services, improving performance and simplifying certificate management.

4. **Integration with WAF (Web Application Firewall)**:**

Advanced DevOps teams often integrate Ingress with a Web Application Firewall (WAF) for enhanced security. The WAF can be configured to inspect and filter traffic at the Ingress point, providing an additional layer of protection against malicious attacks.

5. **Global Load Balancing and Multi-Cloud Deployments**:**

In multi-cloud or globally distributed architectures, advanced DevOps utilizes Ingress for global load balancing. Ingress can route traffic across different cloud providers or regions based on latency, proximity, or other defined rules, optimizing user experience.

6. **Automated Ingress Configuration with GitOps**:**

Advanced DevOps embraces GitOps principles to manage infrastructure as code. Ingress configurations can be version-controlled, reviewed, and deployed

automatically through GitOps pipelines, ensuring consistency, traceability, and auditability.

7. **Dynamic Configuration Updates and Autoscaling**:**

Advanced setups often leverage Ingress controllers that support dynamic updates to configuration. Ingress rules can be updated on the fly, allowing for autoscaling and rapid adjustments to handle traffic spikes and changes in application demands.

8. **Advanced Monitoring and Analytics Integration**:**

Ingress data can be integrated into advanced monitoring and analytics platforms. Metrics and logs from the Ingress controllers provide insights into traffic patterns, performance, errors, and user behavior, enabling data-driven optimizations and troubleshooting.

In conclusion, using Ingress in Kubernetes within an advanced DevOps context goes beyond simple traffic routing. It involves advanced traffic management, security enhancements, integration with various tools and services, dynamic configurations, and the seamless orchestration of microservices in a modern, cloud-native, and highly automated DevOps environment.

Q.4 Which service protocols does Kubernetes ingress expose

Ans:

Kubernetes Ingress, by default, is primarily designed to expose HTTP and HTTPS services. It provides a way to route and manage HTTP/HTTPS traffic to services within the Kubernetes cluster based on defined rules. However, Ingress can be extended to support additional protocols beyond HTTP/HTTPS using specialized Ingress controllers and annotations.

The primary protocols exposed by Kubernetes Ingress are:

HTTP (HyperText Transfer Protocol):

Ingress is most commonly used to expose HTTP services. It allows for routing and load balancing of HTTP traffic based on defined rules and paths.

HTTPS (HTTP Secure):

Ingress can also expose HTTPS services, providing a secure way to route and manage HTTPS traffic. SSL/TLS termination can be performed at the Ingress level for HTTPS traffic.

In addition to HTTP and HTTPS, Kubernetes Ingress can be extended to support protocols such as:

TCP (Transmission Control Protocol):

Ingress can be configured to expose TCP-based services. This is useful for non-HTTP applications or services that use TCP for communication.

UDP (User Datagram Protocol):

Similarly, Ingress can be configured to expose UDP-based services. UDP is often used for applications that need a lightweight and faster communication protocol.

gRPC (gRPC Remote Procedure Calls):

Ingress can be configured to handle gRPC traffic, a high-performance, open-source and universal remote procedure call (RPC) framework.

WebSocket:

Ingress extensions can manage WebSocket protocols, which are essential for real-time applications and two-way communication between a client and a server. To support these additional protocols, specialized Ingress controllers need to be used along with appropriate annotations and configurations. These controllers extend the functionality of Kubernetes Ingress and enable routing and load balancing for a broader range of protocols beyond HTTP/HTTPS. Always refer to the specific Ingress controller's documentation to confirm the supported protocols and configurations.

Written Assignment 2

Q.1 How to deploy Lambda function on AWS?

Ans:

Deploying a Lambda function on AWS involves several steps, including creating the function, configuring its settings, and setting up triggers. Here's a step-by-step guide on how to deploy a Lambda function on AWS:

Sign in to AWS Console:

Log in to your AWS account and access the AWS Management Console.

Open Lambda service:

Navigate to the AWS Lambda service from the AWS Management Console.

Create a Lambda function:

Click on the "Create function" button.

Configure the function:

Choose an authoring option: You can either author the code inline or upload a .zip file containing the code.

Function name: Enter a unique name for your Lambda function.

Runtime: Choose the runtime that supports your code (e.g., Node.js, Python, Java, etc.).

Role: Choose an existing role or create a new one with appropriate permissions.

Function code:

If you chose inline code, you can edit the code directly in the AWS Management Console.

If you uploaded a .zip file, upload your deployment package containing the code.

Environment variables (optional):

You can set environment variables for your Lambda function.

Execution role:

Choose an existing role or create a new one with the necessary permissions for your function.

Advanced settings (optional):

Configure additional settings like memory, timeout, networking, and concurrency.

Triggers (optional):

Configure event sources that trigger your Lambda function (e.g., API Gateway, S3, SNS, etc.).

Click "Create function":

Once you've configured all the necessary settings, click the "Create function" button.

Lambda function is now deployed and ready to be triggered based on the configured event sources. We can test our function within the AWS Management Console or trigger it using the configured triggers.

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

Ans:

AWS Lambda offers several deployment options to suit various use cases and preferences. Here are the primary deployment options for AWS Lambda:

Manual Deployment:

Manually deploy a Lambda function by creating it through the AWS Management Console, AWS Command Line Interface (CLI), or AWS SDKs. You provide the function code and configuration settings during this process.

AWS Management Console:

Create, configure, and deploy Lambda functions using the AWS Management Console. This web-based interface allows for a visual and interactive way to manage your functions.

AWS Command Line Interface (CLI):

Use the AWS CLI to package and deploy Lambda functions. The CLI allows for scripting and automation of deployment processes.

AWS CloudFormation:

Define Lambda functions and related AWS resources in AWS CloudFormation templates. CloudFormation automates the deployment and management of these resources, enabling infrastructure as code (IaC) practices.

AWS SAM (Serverless Application Model):

AWS SAM is an open-source framework that extends AWS CloudFormation to simplify the deployment of serverless applications. It provides a streamlined way to define serverless applications, including Lambda functions, APIs, and more.

AWS Serverless Application Repository:

Deploy pre-built serverless applications and components from the AWS Serverless Application Repository. This allows you to use and share existing serverless applications easily.

AWS CodeDeploy:

Integrate AWS Lambda deployments with AWS CodeDeploy, a service that automates application deployments to various compute services, including Lambda. CodeDeploy can help manage updates and rollbacks for Lambda functions.

AWS SAM CLI:

Use the AWS SAM CLI to build, test, and deploy serverless applications defined using AWS SAM. The CLI provides local testing capabilities and simplifies the deployment process.

Third-Party Tools and Frameworks:

Utilize third-party tools and frameworks, such as Serverless Framework, Terraform, and others, to deploy and manage Lambda functions. These tools often provide additional features and flexibility for deployment and management.

Each deployment option has its advantages and is suited for different scenarios based on factors like automation needs, complexity, team collaboration, and integration with existing workflows. Choose the option that aligns best with your specific use case and deployment requirements.

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

Ans:

In the context of AWS (Amazon Web Services), there are three primary deployment modes commonly referred to when deploying applications or services:

Manual Deployment:

Manual deployment involves direct and hands-on configuration and setup of resources using AWS Management Console, AWS Command Line Interface (CLI), or AWS SDKs. It allows users to define and configure each component individually, making it suitable for small-scale or simple deployments.

Automated Deployment:

Automated deployment involves using automation tools and scripts to manage the deployment process. This can include using AWS services like AWS CloudFormation, AWS SAM (Serverless Application Model), AWS CodeDeploy, or third-party tools like Terraform. Automation streamlines and accelerates the deployment process, making it more efficient and repeatable.

Serverless Deployment:

Serverless deployment, often associated with AWS Lambda, involves deploying applications or functions without managing the underlying infrastructure. AWS Lambda automatically handles scaling, monitoring, and managing the compute resources needed to run the application. Serverless architectures abstract away server management tasks, allowing developers to focus solely on writing code.

These deployment modes offer varying levels of control, automation, and abstraction, catering to different use cases and preferences. Developers and organizations choose the appropriate deployment mode based on their specific requirements, such as application complexity, scalability needs, automation goals, and the level of control they desire over the deployment process.

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

Ans:

AWS Lambda is composed of three primary components that work together to enable serverless compute capabilities:

Function Code:

The function code is the actual program or script that you want AWS Lambda to execute in response to an event. It can be written in supported programming languages, including Node.js, Python, Java, C#, Go, Ruby, and PowerShell Core. You can provide the code directly within the AWS Management Console (inline code) or package it as a .zip file and upload it to AWS Lambda.

Event Sources:

Event sources are the triggers or events that invoke your AWS Lambda function. Lambda functions can be triggered by various AWS services and event sources, including Amazon S3 bucket events, Amazon DynamoDB table updates, Amazon SNS notifications, API Gateway requests, AWS CloudFormation stack updates, and more. Event sources determine when and how your Lambda function is executed.

Execution Role:

An execution role (IAM role) defines the permissions that AWS Lambda functions need to interact with other AWS services. This role grants necessary permissions for

the function to access resources, such as reading from an S3 bucket, writing to a DynamoDB table, or publishing to an SNS topic. AWS Lambda assumes this role when executing the function, allowing it to access the specified resources securely.

These three components form the foundation of AWS Lambda, enabling developers to create, configure, and run serverless functions that respond to events from various sources within the AWS ecosystem.