

# AWS MINI PROJECT 1

## LAB1: IAM HANDS-ON

Log in to aws management console and sign in to the aws management console as the root user.

On the right hand side corner we have our account details in that we have security and credentials click on that the below tab will be open.

The screenshot shows the AWS IAM Security Credentials page. On the left, there's a sidebar with navigation links like Dashboard, Access management, and Access reports. The main content area has a heading 'My security credentials' with a note: 'You don't have MFA assigned'. It shows account details: Account name (Spandana), Email address (bandla.spandana0115@gmail.com), AWS account ID (637423550155), and Canonical user ID (7fcfb96bfdf603fa9b367e4bd1ba1ea47d19121ad48de4b482154e6bee0a7a4d). Below this is a section for Multi-factor authentication (MFA) with a table and a 'Assign MFA device' button. At the bottom, there's an 'Access keys (0)' section with a 'Create access key' button.

After that we have to click on multi-factor authentication (MFA) Click on Assign MFA.

There are three types of Authentication methods for AWS console management account

1. Authentication app
2. Security Key
3. Hardware TOTP token

I have selected Authentication app for MFA , First we need to install Authy app on our mobiles so that we can able to do the further process. We need to give a name for MFA, and in the AWS console, a QR code will be displayed. Use the authentication app for the scan this QR code.

If our app does not support for the scanning we need to enter the provided secret key manually. The app will start generating the 6 digit codes. Enter the first code in MFA Code 1 and wait for the second code and enter in the place of MFA Code 2

The screenshot shows the 'Set up device' step of the 'Assign MFA device' wizard. It includes three numbered steps: 1. Install a compatible application such as Google Authenticator, Duo Mobile, or Authy app on your mobile device or computer. 2. Scan the QR code displayed on the screen using the app. 3. Type two consecutive MFA codes below. There are two input fields: 'MFA Code 1' and 'MFA Code 2'. A note says to wait 30 seconds between entries.

After Entering all the Information we need to confirmation so we will see the conformation message indicating that the MFA device assigned.

We will see the one change in the MFA and if we want to check that we have to logout our AWS management console and we need to login again it will ask for the MFA Code for those who choose Authentication app.

The screenshot shows the 'My security credentials' page under the IAM service. It displays account details like Account name (Spandana), Email address (baindlspandana0115@gmail.com), and AWS account ID (637423550105). In the 'Multi-factor authentication (MFA)' section, there is one entry for a Virtual device with Identifier arn:aws:iam:637423550105:mfa/personal\_phone\_MFA. The 'Access keys (0)' section is shown below.

## 2.Creating a New user with access and check its default permissions.

In the IAM dashboard we will see users in the left panel.

The screenshot shows the AWS IAM Dashboard. On the left sidebar, under 'Access management', the 'Users' option is selected. The main content area displays 'Security recommendations' with two items: 'Root user has MFA' and 'Root user has no active access keys'. Below this is an 'IAM resources' summary with counts for User groups (1), Users (3), Roles (12), Policies (1), and Identity providers (0). A 'What's new' section lists recent changes, including the addition of policy checks for public and critical resource access and recommendations for unused access. To the right, there are sections for the 'AWS Account' (Account ID: 637423550105, Account Alias: spandana-cloud) and 'Quick Links' (My security credentials, Policy simulator). The bottom right corner includes copyright information for 2024 and links for Privacy, Terms, and Cookie preferences.

Click on the Users which is there below the user groups.

The screenshot shows the 'Users' page in the AWS IAM service. The left sidebar shows 'Access management' with 'Users' selected. The main content area displays a table titled 'Users (3) Info' with three entries: 'Bhargavi', 'KOPS', and 'Spandana'. The table columns include User name, Path, Group, Last activity, MFA, Password age, Console last sign-in, and Access key ID. Each user entry has a 'Delete' button and a 'Create user' button. The bottom right corner includes copyright information for 2024 and links for Privacy, Terms, and Cookie preferences.

User name	Path	Group	Last activity	MFA	Password age	Console last sign-in	Access key ID
Bhargavi	/	0	-	-	⚠️ 109 days	-	-
KOPS	/	0	⌚ 8 days ago	-	-	-	Active - AKIAZ12L
Spandana	/	1	⚠️ 109 days ago	-	⚠️ 110 days	July 10, 2024, 22:07 (...)	Active - AKIAZ12L

When we click on the Create user this page will be open in this page we need to Enter our Username for new user and select AWS management console access and set a custom password for security purpose. If we select the Autogenerated password we can able to see that password after creation of user. So we will choose custom password for privacy and security purpose.

User name  
Firstuser

Provide user access to the AWS Management Console - optional  
If you're providing console access to a person, it's a best practice [to manage their access in IAM Identity Center](#).

**Are you providing console access to a person?**

User type  
 Specify a user in Identity Center - Recommended  
We recommend that you use Identity Center to provide console access to a person. With Identity Center, you can centrally manage user access to their AWS accounts and cloud applications.  
 I want to create an IAM user  
We recommend that you create IAM users only if you need to enable programmatic access through access keys, service-specific credentials for AWS CodeCommit or Amazon Keypaces, or a backup credential for emergency account access.

Console password  
 Autogenerated password  
You can view the password after you create the user.  
 Custom password  
Enter a custom password for the user:  
\*\*\*\*\*  
Must be at least 8 characters long  
Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # \$ % ^ & \* { } \_ - [ ] { } [ ] !  
 Show password  
 Users must create a new password at next sign-in - Recommended  
Users automatically get the IAMUserChangePassword [policy](#) to allow them to change their own password.

If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keypaces, you can generate them after you create this IAM user. [Learn more](#)

We will click on the check box below show password that is Users must create a new password at next sign in it is recommended for the security purpose. After that review and then Create user after creation we will see user's access key and secret access key save these credentials securely.

User name  
Firstuser

Console password type  
Custom password

Require password reset  
Yes

**Permissions summary**

Name	Type	Used as
IAMUserChangePassword	AWS managed	Permissions policy

**Tags - optional**  
Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.  
No tags associated with the resource.  
[Add new tag](#)  
You can add up to 50 more tags.

[Cancel](#) [Previous](#) **Create user**

User created successfully

You can view and download the user's password and email instructions for signing in to the AWS Management Console.

IAM > Users > Create user

Step 1  
Specify user details

Step 2  
Set permissions

Step 3  
Review and create

Step 4  
Retrieve password

Retrieve password

Console sign-in details

Console sign-in URL  
https://spandana-cloud.signin.aws.amazon.com/console

User name  
Firstuser

Console password  
\*\*\*\*\* Show

Email sign-in instructions

Cancel Download .csv file Return to users list

## Check Default Permissions:

After creation of user click the user of newly created IAM user for which we want to check the default permissions. We will move to the user details page, Click on the permissions tab. The permissions tab is nothing but below image. Default permissions for the user is **IAMUserChangePassword** this is the default permission for the user.

Identity and Access Management (IAM)

Summary

ARN: am:awsiam::637423550105:user/Firstuser

Console access: Enabled without MFA

Access key 1: Create access key

Created: October 28, 2024, 10:59 (UTC+0:30)

Last console sign-in: Never

Permissions (1)

Permissions policies (1)

Policy name: IAMUserChangePassword

Type: AWS managed

Attached via: Directly

Permissions boundary (not set)

Generate policy based on CloudTrail events

Generate policy

#### 4. Assign full permissions to the user

We can able to see the Add Permissions click on Add Permissions > Attach existing policies directly.

Add permissions

Step 1  
Add permissions

Step 2  
Review

Permissions options

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, inline policies, and any existing permissions boundaries 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.

Permissions policies (1/1245)

Filter by Type: All types, 1 match

Policy name	Type	Attached entities
AmazonEC2FullAccess	AWS managed	0

Cancel Next

In the search bar, type **AmazonEC2FullAccess** and check the box next to the policy. Click Next and then Add permissions.

Identity and Access Management (IAM)

Firstuser Info

Summary

ARN: arn:aws:iam::657423550105:user/Firstuser	Console access: Enabled without MFA	Access key 1: Create access key
Created: October 28, 2024, 10:59 (UTC+05:30)	Last console sign-in: Never	

Permissions | Groups | Tags | Security credentials | Last Accessed

Permissions policies (2)

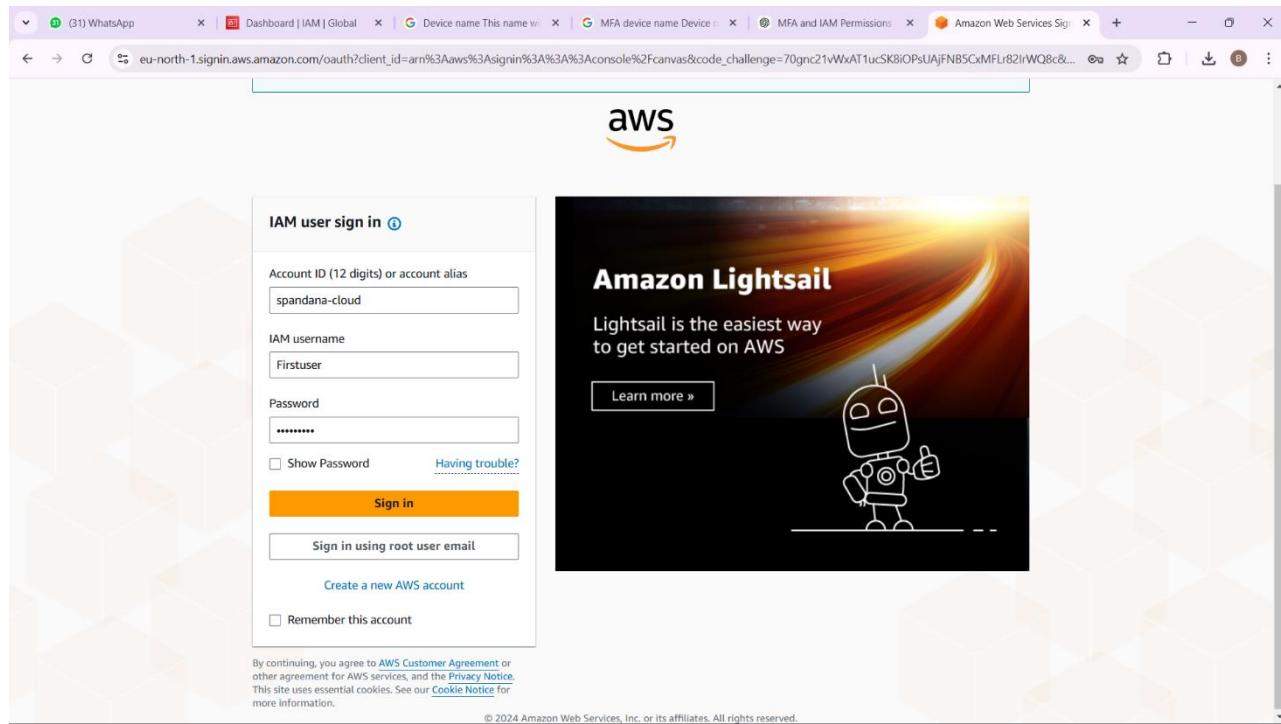
Permissions are defined by policies attached to the user directly or through groups.

Policy name	Type	Attached via
AmazonEC2FullAccess	AWS managed	Directly
IAMUserChangePassword	AWS managed	Directly

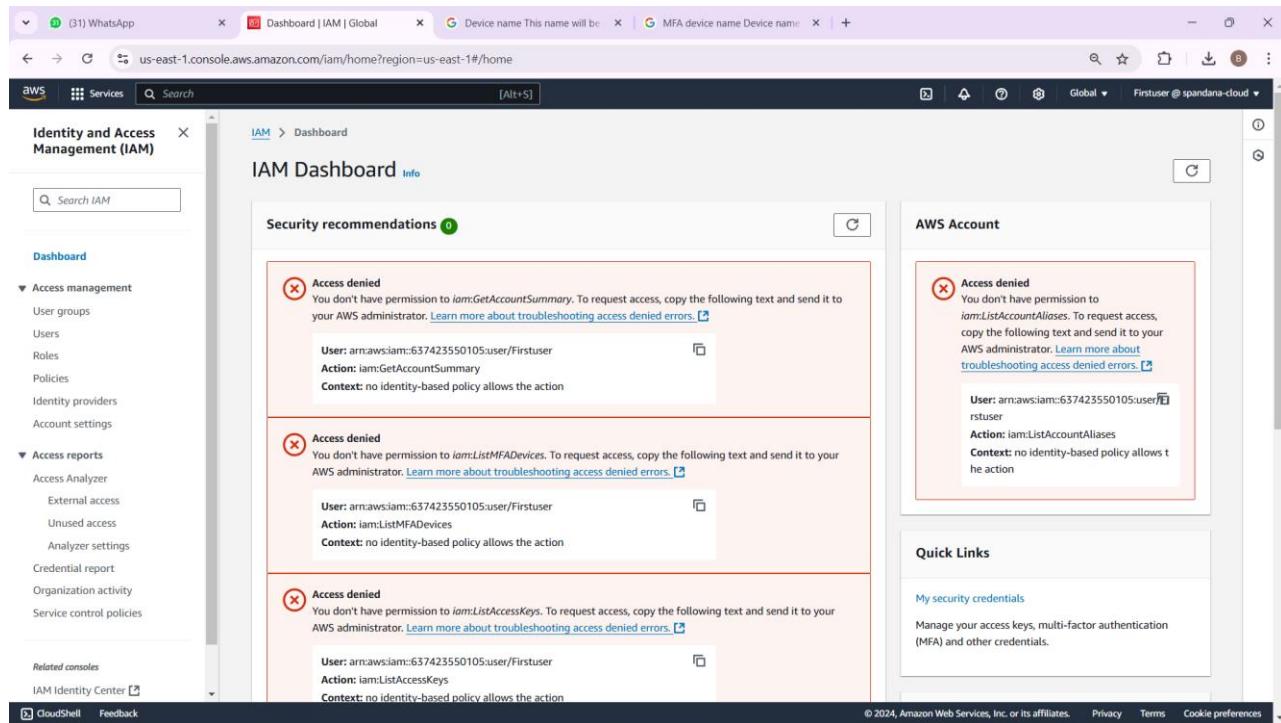
Permissions boundary (not set)

Generate policy based on CloudTrail events

Log in as the new user: Use the credentials of the new user to log in to the AWS Management console.



If we check the access to EC2 we can able to perform EC2 related tasks and we have all permissions to access EC2 but when we navigate to other services like IAM, S3 and confirm that we do not have access because EC2 full permissions should restrict access to other services.



The screenshot shows the AWS IAM 'Users' page. On the left, there's a navigation sidebar with 'Identity and Access Management (IAM)' selected. The main area displays a table with one row, which has a red border around it. The error message in the table says: 'Access denied. You don't have permission to iam>ListUsers. To request access, copy the following text and send it to your AWS administrator. Learn more about troubleshooting access denied errors.' Below the table, there are buttons for 'Create user' and 'Delete'.

#### 4.PROVIDING ADMINISTRATIVE PERMISSIONS TO THE USER.

Go back to the IAM dashboard and select the user again click on the Permissions tab click on Add permissions > Attach existing policies directly and search for **AdministratorAccess** and select that policy click next and then Add permissions.

The screenshot shows the 'Add permissions' step for the 'Firstuser' user. It includes sections for 'Permissions options' (with 'Attach policies directly' selected), 'Permissions policies' (listing 'AdministratorAccess' and other policies), and a 'Review' section.

The screenshot shows the AWS Identity and Access Management (IAM) console. The left sidebar is titled 'Identity and Access Management (IAM)' and includes sections for Dashboard, Access management (User groups, Roles, Policies, Identity providers, Account settings), Access reports (Access Analyzer, External access, Unused access, Analyzer settings, Credential report, Organization activity, Service control policies), and Related consoles (IAM Identity Center). The main content area is titled 'Permissions' and shows 'Permissions policies (3)'. It lists three policies: 'AdministratorAccess' (AWS managed - job function, Attached via Directly), 'AmazonEC2FullAccess' (AWS managed, Attached via Directly), and 'IAMUserChangePassword' (AWS managed, Attached via Directly). Below this is a section for 'Permissions boundary (not set)'. At the bottom, there is a section for 'Generate policy based on CloudTrail events' with a 'Generate policy' button.

The screenshot shows the AWS IAM 'Users' section. The left sidebar is identical to the previous screenshot. The main content area shows the details for the user 'Firstuser'. The 'Summary' section displays the ARN (arn:aws:iam::637423550105:user/Firstuser), Console access (Enabled without MFA), Last console sign-in (Today), and an 'Access key 1' button. Below this is a 'Permissions' tab, which is currently selected, showing 'Permissions policies (3)' with the same three policies listed as in the previous screenshot. The 'Groups', 'Tags', 'Security credentials', and 'Last Accessed' tabs are also present.

Login to the AWS Management console again as the user and confirm that now we have access to all services including the ability to manage IAM users and roles.

The screenshot shows the IAM Dashboard with the following details:

- Security recommendations:**
  - Root user has MFA (Green)
  - Add MFA for yourself (Yellow)
  - Your user, Firstuser, does not have any active access keys that have been unused for more than a year. (Green)
- IAM resources:**

User groups	Users	Roles	Policies	Identity providers
1	4	12	1	0
- What's new:**
  - AWS IAM Access Analyzer now offers policy checks for public and critical resource access. 5 months ago
- AWS Account:**
  - Account ID: 637423550105
  - Account Alias: spandana-cloud [Edit](#) | [Delete](#)
  - Sign-in URL for IAM users in this account: <https://spandana-cloud.signin.aws.amazon.com/console>
- Quick Links:**
  - [My security credentials](#): Manage your access keys, multi-factor authentication (MFA) and other credentials.
- Tools:**
  - [Policy simulator](#): The simulator evaluates the policies that you choose and determines the effective permissions for each of the selected users or groups.

## LAB 2-SETUP BILLING ALARM

Go to the AWS Management Console and log in to your account search for Billing and Cost Management and select it.

The screenshot shows the Billing and Cost Management home page with the following details:

- Cost summary:**
  - Month-to-date cost: **\$0.33** (↑ 6% compared to last month for same period)
  - Last month's cost for same time period: **\$0.31** (Sep 1–28)
  - Total forecasted cost for current month: **\$0.33** (↑ 8% compared to last month's total costs)
  - Last month's total cost: **\$0.31**
- Cost breakdown:**
  - Group costs by: Service
  - Costs (\$): A bar chart showing costs from May 2024 to Oct 2024. The legend includes:
    - Amazon Elastic Compute Cloud - Compute
    - Tax
    - EC2 - Other
    - AWS Key Management Service
    - Amazon Simple Storage Service
    - Others
- Recommended actions (2):**
  - Budget threshold exceeded:** 1 of your budget alert thresholds have been exceeded. [Review budgets](#)
  - Getting started:** Add an additional billing contact. [Update billing contact](#).

The screenshot shows the 'Create topic' page in the AWS SNS console. The 'Type' section is expanded, showing two options: 'FIFO (first-in, first-out)' and 'Standard'. 'Standard' is selected, indicated by a blue outline. Below it, a list of features includes: Best-effort message ordering, At-least once message delivery, Highest throughput in publishes/second, and Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints. The 'Name' field is set to 'Standard'. The 'Display name - optional' field contains 'My Topic'. A note states: 'Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.' The bottom right corner shows the copyright notice: '© 2024, Amazon Web Services, Inc. or its affiliates.'

Create a cost budget, your desired threshold amount, and configure email alerts. This setup ensures you receive notifications whenever spending approaches or exceeds the specified budget, helping you monitor and manage costs effectively.

The screenshot shows the 'Create subscription' page in the AWS SNS console. The 'Details' section is expanded, showing the 'Topic ARN' field containing 'arn:aws:sns:us-east-1:657423550105:BillingAlertTopic'. The 'Protocol' field is set to 'Email'. The 'Endpoint' field contains 'baindaspandana0115@gmail.com'. A note below the endpoint says: 'After your subscription is created, you must confirm it.' The 'Subscription filter policy - optional' section is collapsed. The 'Redrive policy (dead-letter queue) - optional' section is also collapsed. The bottom right corner shows the copyright notice: '© 2024, Amazon Web Services, Inc. or its affiliates.'

Subscription to BillingAlertTopics created successfully.

The screenshot shows the AWS SNS console with a success message: "Subscription to BillingAlertTopic created successfully. The ARN of the subscription is arn:aws:sns:us-east-1:637423550105:BillingAlertTopic:aadec8c1-0b6c-4124-95d1-2677254fa34d." Below this, the "Details" section displays the ARN, endpoint, topic, and principal. The status is "Pending confirmation" and the protocol is "EMAIL".

We have to go to Cloud Watch click on the alarm in the left side select billing under the metrics click on create alarm .

The screenshot shows the AWS CloudWatch Metrics Alarms page. The sidebar includes options like Dashboards, Alarms, Logs, Metrics, X-Ray traces, Events, Application Signals, Network monitoring, and Insights. The main area shows a table for alarms, which is currently empty. A prominent orange "Create alarm" button is located at the top right of the table area.

Select metric choose Billing and select Total estimated charge in our preference currency click on select metric

The screenshot shows the 'Specify metric and conditions' step of the CloudWatch Create alarm wizard. In the 'Metric' section, a graph displays 'EstimatedCharges' over time from 10/22 to 10/26. A red horizontal line at the top represents the threshold, and a blue line shows the metric value. The 'Threshold type' is set to 'Static'. In the 'Conditions' section, the threshold is defined as 'No unit' with a value of '4'. The 'Namespace' is set to 'AWS/Billing', 'Metric name' to 'EstimatedCharges', 'Currency' to 'USD', 'Statistic' to 'Maximum', and 'Period' to '6 hours'. The sidebar on the left lists steps: Step 1 (Specify metric and conditions), Step 2 (Configure actions), Step 3 (Add name and description), and Step 4 (Preview and create).

In the configuration actions section under notification choose send notification to the following SNS topic

Select the BillingAlertTopic optionally we can specify additional SNS topics if needed by adding topic ARN. Under the alarm state select the alarm to notify us whenever the alarm thresholds crossed.

The screenshot shows the 'Configure actions' step of the CloudWatch Create alarm wizard. In the 'Notification' section, the 'Alarm state trigger' is set to 'In alarm'. The 'Send a notification to...' section shows a selected SNS topic named 'BillingAlertTopic'. The 'Lambda action' section is currently empty. The sidebar on the left lists steps: Step 1 (Specify metric and conditions), Step 2 (Configure actions), Step 3 (Add name and description), and Step 4 (Preview and create).

In the below image we will see our Billing alaram setup is done successfully.

The screenshot shows the AWS CloudWatch Alarms interface. The left sidebar is collapsed, and the main area displays a green success message: "Successfully created alarm MonthlyBillingAlarm." Below this, the "CloudWatch > Alarms" section is shown. A table lists the "Billing alarms (1)". The single entry is "MonthlyBillingAlarm", which is currently in an "Insufficient data" state. The last update was on "2024-10-28 06:20:16". The condition for the alarm is "EstimatedCharges <= 3 for 1 datapoints within 6 hours", and the status is "Actions enabled".

Name	State	Last state update (UTC)	Conditions	Actions
MonthlyBillingAlarm	Insufficient data	2024-10-28 06:20:16	EstimatedCharges <= 3 for 1 datapoints within 6 hours	Actions enabled

## LAB-3 S3 BUCKET

In the console search for S3 and open it we can able to see below image.

The screenshot shows the AWS S3 console interface. The left sidebar is titled "Amazon S3" and includes sections for Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, and Storage Lens settings. Below these are sections for Feature spotlight and AWS Marketplace for S3. The main content area is titled "Amazon S3" and features an "Account snapshot - updated every 24 hours" section with a link to "All AWS Regions". It also includes a "Storage lens provides visibility into storage usage and activity trends" message with a "Learn more" link and a "View Storage Lens dashboard" button. The "General purpose buckets" tab is selected, showing a list of buckets. The table has columns for Name, AWS Region, IAM Access Analyzer, and Creation date. Two buckets are listed: "cf-templates-60219jnrm7bg-us-east-1" and "elasticbeanstalk-us-east-1-637423550105". Each bucket row includes links to "View analyzer for us-east-1".

Name	AWS Region	IAM Access Analyzer	Creation date
<a href="#">cf-templates-60219jnrm7bg-us-east-1</a>	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	July 30, 2024, 21:54:57 (UTC+05:30)
<a href="#">elasticbeanstalk-us-east-1-637423550105</a>	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	July 30, 2024, 22:29:21 (UTC+05:30)

Click on create bucket and we have to give unique name for the bucket.

The screenshot shows the 'Create bucket' configuration page in the AWS S3 console. The 'General configuration' section is visible, showing the 'Bucket name' field set to 'my-bucket-ss-001'. The 'Bucket type' dropdown is open, with 'General purpose' selected (indicated by a blue border). Other options like 'Directory' are also shown. Below the bucket name, there's a note about uniqueness and naming rules. The 'Object Ownership' section is partially visible at the bottom. The status bar at the bottom right indicates 'CloudShell Feedback'.

Choose other setting default for now and click on create bucket.

The screenshot shows the 'Object Ownership' configuration page in the AWS S3 console. The 'ACLs disabled (recommended)' option is selected (indicated by a blue border). The 'Object Ownership' section below it shows 'Bucket owner enforced'. The 'Block Public Access settings for this bucket' section is expanded, showing four options under 'Block all public access': 'Block public access to buckets and objects granted through new access control lists (ACLS)', 'Block public access to buckets and objects granted through any access control lists (ACLS)', 'Block public access to buckets and objects granted through new public bucket or access point policies', and 'Block public and cross-account access to buckets and objects through any public bucket or access point policies'. The status bar at the bottom right indicates 'CloudShell Feedback'.

In the below image we can able to see our new bucket has been created successfully.

The screenshot shows the AWS S3 buckets dashboard. At the top, a green success message states: "Successfully created bucket 'my-bucket-ss-001'. To upload files and folders, or to configure additional bucket settings, choose View details." Below this, the "General purpose buckets" section lists three buckets: "cf-templates-60219nrm7bg-us-east-1", "elasticbeanstalk-us-east-1-637423550105", and "my-bucket-ss-001". The "my-bucket-ss-001" row is selected, highlighted with a blue border. The table columns include Name, AWS Region, IAM Access Analyzer, and Creation date. The "Create bucket" button is visible at the top right of the table area.

In S3 dashboard click on the name of the bucket we have just created to open it. We can able to see our bucket and there is a upload button on the right corner of the object.

The screenshot shows the "my-bucket-ss-001" bucket details page. The top navigation bar includes tabs for Objects, Properties, Permissions, Metrics, Management, and Access Points. The "Objects" tab is selected. The main content area displays a table with one row: "No objects". A note below the table states: "You don't have any objects in this bucket." At the bottom right of the table area, there is a prominent orange "Upload" button.

Click on upload choose add files and add folder and select the files and folders we want to upload and click on Upload to confirm.

The screenshot shows the AWS S3 'Upload' interface. At the top, there are three tabs: 'WhatsApp' (active), 'Upload objects - S3 bucket my...', and 'AWS Notification - Subscription...'. The main area is titled 'Upload Info'. It contains a large dashed box for dragging and dropping files or folders. Below this is a table titled 'Files and folders (0)' with columns for 'Name', 'Type', 'Size', 'Status', and 'Error'. A search bar and pagination controls are also present. The 'Destination' section shows 's3://my-bucket-ss-001' selected. The status bar at the bottom includes links for CloudShell, Feedback, and cookie preferences.

We can able to see our uploaded files and folder in my bucket.

The screenshot shows the 'Upload succeeded' summary page. A green banner at the top says 'Upload succeeded' with a link to 'View details below.' Below this is a 'Summary' section with a table showing the destination 's3://my-bucket-ss-001' and results: 'Succeeded' (7 files, 4.8 MB (100.00%)) and 'Failed' (0 files, 0 B (0%)). There are tabs for 'Files and folders' (selected) and 'Configuration'. The 'Files and folders' table lists seven items with columns for Name, Folder, Type, Size, Status, and Error. A tooltip in the bottom right corner says 'Screenshot copied to clipboard and saved. Select here to mark up and share.' The status bar at the bottom includes links for CloudShell, Feedback, and cookie preferences.

The below files and folder I have uploaded in my S3 Bucket.

The screenshot shows the AWS S3 console interface. The left sidebar is titled 'Amazon S3' and includes sections for Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Block Public Access settings, Storage Lens, Dashboards, Storage Lens groups, AWS Organizations settings, Feature spotlight, and AWS Marketplace for S3. The main content area shows the 'my-bucket-ss-001' bucket details. The 'Objects' tab is selected, displaying three items:

Name	Type	Last modified	Size	Storage class
AWS Devops/	Folder	-	-	-
certificate (1).pdf	pdf	October 28, 2024, 12:03:32 (UTC+05:30)	1.2 MB	Standard
Firstuser_credentials.csv	csv	October 28, 2024, 12:03:33 (UTC+05:30)	117.0 B	Standard

Go to bucket and click on the Permission tab. We will notice that Block all public access is usually enable by default this restricts public access to the bucket.

The screenshot shows the 'Permissions' tab for the 'my-bucket-ss-001' bucket. The left sidebar is identical to the previous screenshot. The main content area shows the 'Permissions' tab selected. Under 'Permissions overview', it says 'Access finding' and provides a link to 'View analyzer for us-east-1'. Under 'Block public access (bucket settings)', it says 'Block all public access' and shows 'Off'. There is a note: '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.' A 'Edit' button is available. Under 'Bucket policy', it says '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.' A 'Edit' and 'Delete' button are available.

In the permissions tab of the file find the public access settings click edit and choose Grant Public read access to this object confirm by clicking save changes.

The screenshot shows the AWS S3 console with the 'Permissions' tab selected for the bucket 'my-bucket-ss-001'. In the 'Block all public access' section, the 'On' radio button is selected. A note at the bottom of this section states: 'Public access is blocked because Block Public Access settings are turned on for this bucket. To determine which settings are turned on, check your Block Public Access settings for this bucket. Learn more about using Amazon S3 Block Public Access.' There is an 'Edit' button next to the 'On' radio button.

Access the file with browser we can able to see it like this because images are shown like this I have taken an certificate image for the process.

The screenshot shows a browser window displaying a PDF file named 'certificate+(1).pdf' from the URL 'my-bucket-ss-001.s3.us-east-1.amazonaws.com/certificate+(1).pdf'. The PDF content is an XML error response with the following structure:

```

<Error>
<Code>AccessDenied</Code>
<Message>Access Denied</Message>
<RequestId>5P8X77CMBTPZQ26H</RequestId>
<HostId>62B6GC5MyNC2BwGh/axPp0ijPa5BLiIw846LiduPqsAxqbgelhW38AUneLmpOAMd9CuLmhfl+Ts=</HostId>
</Error>

```

## STEP 2

Go to the bucket and go to the Properties tab of the bucket

The screenshot shows the AWS S3 console with the 'Objects' tab selected. The table displays the following data:

Name	Type	Last modified	Size	Storage class
AWS Devops/	Folder	-	-	-
certificate (1).pdf	pdf	October 28, 2024, 12:03:32 (UTC+05:30)	1.2 MB	Standard
Firstuser_credentials.csv	csv	October 28, 2024, 12:03:33 (UTC+05:30)	117.0 B	Standard

Scroll to the Bucket Versioning we will see Bucket versioning is Disable in the below image.

The screenshot shows the AWS S3 console with the 'Properties' tab selected. The 'Bucket Versioning' section contains the following information:

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](#)

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. [Learn more](#)

Multi-factor authentication (MFA) delete  
Disabled

Click on Edit option and enable Bucket Versioning and then click on save changes.

The screenshot shows the AWS S3 Bucket Properties page for 'my-bucket-ss-001'. The 'Properties' tab is selected. A green success message at the top states: 'Successfully edited Bucket Versioning. To transition, archive, or delete older object versions, configure lifecycle rules for this bucket.' Under the 'Bucket overview' section, the 'Bucket Versioning' status is shown as 'Enabled'. An 'Edit' button is located in the top right corner of this section. The 'Tags (0)' section also has an 'Edit' button. The bottom navigation bar includes CloudShell, Feedback, and links to © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

Create one text file on local machine (e.g., Example.txt) with some content. This is original version.

The screenshot shows the AWS S3 Upload page for 'my-bucket-ss-001'. The 'Upload' tab is selected. A message at the top says: '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.' Below this is a large dashed blue box for dragging files. A table below shows 'Files and folders (1 Total, 104.0 B)'. It lists 'Example.txt' with options to 'Remove', 'Add files', or 'Add folder'. A search bar 'Find by name' is present. The 'Destination' section shows 's3://my-bucket-ss-001'. The 'Destination details' section contains the note: 'Bucket settings that impact new objects stored in the specified destination.' The bottom navigation bar includes CloudShell, Feedback, and links to © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

In the click on upload select add files and choose Example.txt Click upload to Confirm and we can able to see that our text file uploaded successfully.

The screenshot shows the AWS S3 console interface. At the top, there are three tabs: 'Upload objects - S3 bucket my' (active), 'Upload objects - S3 bucket my', and 'AWS Notification - Subscription'. The browser address bar shows 'us-east-1.console.aws.amazon.com/s3/upload/my-bucket-ss-001?region=us-east-1&bucketType=general'. Below the tabs, the AWS logo and 'Services' button are visible. A green success message box displays 'Upload succeeded' and 'View details below.' The main content area is titled 'Upload: status'. It includes a summary table and a detailed table of uploaded files. The summary table shows 'Destination s3://my-bucket-ss-001' with 'Succeeded' (1 file, 104.0 B (100.0%)) and 'Failed' (0 files, 0 B (0%)). The detailed table lists 'Files and folders (1 Total, 104.0 B)' with one entry: 'Example.txt' (text/plain, 104.0 B, Status: Succeeded). The bottom of the page includes standard AWS footer links: CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

We can able to see Example.txt filein the S3 Bucket.

The screenshot shows the AWS S3 console interface. The top navigation bar has tabs for 'Upload objects - S3 bucket my' (active), 'my-bucket-ss-001 - S3 bucket', and 'AWS Notification - Subscription'. The browser address bar shows 'us-east-1.console.aws.amazon.com/s3/buckets/my-bucket-ss-001?region=us-east-1&bucketType=general&tab=objects'. Below the tabs, the AWS logo and 'Services' button are visible. A green success message box displays 'Upload succeeded' and 'View details below.' The main content area is titled 'my-bucket-ss-001' with an 'Info' link. It includes a navigation menu with 'Objects' selected, followed by 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. The 'Objects' section shows a table of four items: 'AWS Devops/' (Folder), 'certificate (1).pdf' (pdf, 1.2 MB, Standard), 'Example.txt' (txt, 162.0 B, Standard), and 'Firstuser\_credentials.csv' (csv, 117.0 B, Standard). The bottom of the page includes standard AWS footer links: CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

This is the Content of the text file before updating the file.

The screenshot shows a browser window with multiple tabs open. The active tab displays the content of a file named 'Example.txt' from an S3 bucket. The content of the file is:

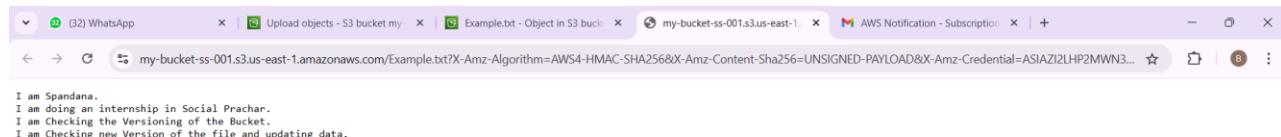
```
I am Spandana.  
I am doing an internship in Social Prachar.  
I am Checking the Versioning of the Bucket.
```

Update the Example.txt file on local machine and modify its content (This is updated version). Save the changes and upload the file and confirm AWS S3 automatically treat this as a new version of the file. We can able to see that there are 2 Example.txt files in S3 Bucket.

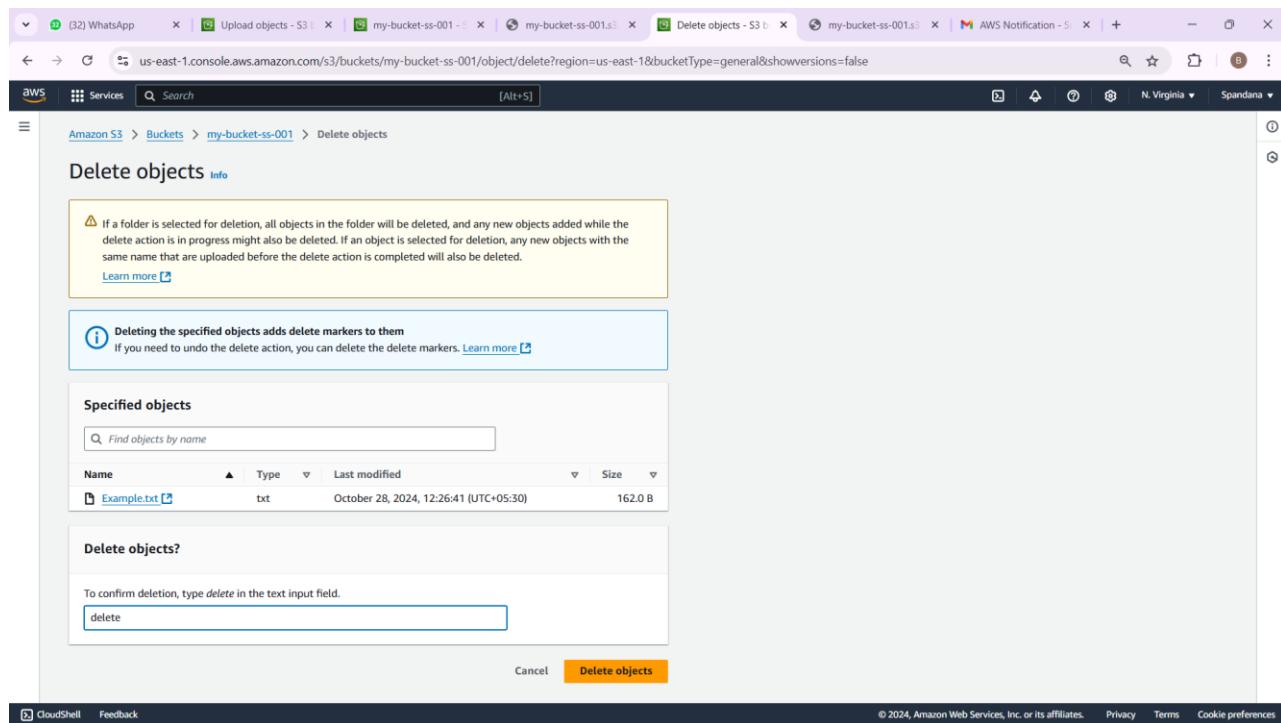
The screenshot shows the AWS S3 console in the 'Objects' view for the 'my-bucket-ss-001' bucket. There are five objects listed:

Name	Type	Version ID	Last modified	Size	Storage class
AWS Devops/	Folder	-	-	-	-
certificate (1).pdf	pdf	null	October 28, 2024, 12:05:32 (UTC+05:30)	1.2 MB	Standard
Example.txt	txt	QvM6E6QHk zuSzQZfFwE9 k0b9FNcmK MY_	October 28, 2024, 12:26:41 (UTC+05:30)	162.0 B	Standard
Example.txt	txt	QpspYFi6NC. xnR5UPqmc VycKec4Gd_v	October 28, 2024, 12:23:59 (UTC+05:30)	104.0 B	Standard
Firstruser_credentials.csv	csv	null	October 28, 2024, 12:05:33 (UTC+05:30)	117.0 B	Standard

This is the Updated Version of Example.txt file I have updated last line.



In the Bucket locate Example.txt select it and click on Delete. Confirm the deletion.



We can able to see that the object is deleted successfully.

The screenshot shows the AWS S3 console with multiple tabs open. The current tab is 'Delete objects - S3 b' with the URL [us-east-1.console.aws.amazon.com/s3/buckets/my-bucket-ss-001/object/delete?region=us-east-1&bucketType=general&showversions=false](https://us-east-1.console.aws.amazon.com/s3/buckets/my-bucket-ss-001/object/delete?region=us-east-1&bucketType=general&showversions=false). A green banner at the top says 'Successfully deleted objects' with a link to 'View details below.' Below it, a summary table shows:

Source	Successfully deleted	Failed to delete
s3://my-bucket-ss-001	1 object, 162.0 B	0 objects

Below the summary, there are tabs for 'Failed to delete' (selected) and 'Configuration'. Under 'Failed to delete', it says '(0)' and 'No objects failed to delete.'

Delete the Example.txt file permanently so that we can able to see the versioning properly.

The screenshot shows the AWS S3 console with the path 'Amazon S3 > Buckets > my-bucket-ss-001 > Delete objects'. The main area displays a warning message about deleting objects from a folder and a 'Learn more' link. Below it is a table of 'Specified objects' containing 'Example.txt'. At the bottom, a 'Permanently delete objects?' dialog box asks to type 'permanently delete' into a text input field. Buttons for 'Cancel' and 'Delete objects' are at the bottom right.

Go back to the versions of the objects tab. We will find the previous versions, including the one marked as “deleted marked”. Restore the deleted version to recover the file delete the delete marker version of Example.txt once we remove the delete marker the file will be restored to its previous version.

Name	Type	Last modified	Size	Storage class
AWS Devops/	Folder	-	-	-
Example.txt	txt	October 28, 2024, 12:26:41 (UTC+05:30)	162.0 B	Standard
Firstuser_credentials.csv	csv	October 28, 2024, 12:03:33 (UTC+05:30)	117.0 B	Standard

We can able to see the restore of the file ad content of the file.

```
I am Spandana.  
I am doing an internship in Social Prachar.  
I am Checking the Versioning of the Bucket.  
I am Checking new Version of the file and updating data.
```

## LAB 4 - EC2-INSTANCE

In AWS console search for EC2 and open it.

The screenshot shows the AWS EC2 Dashboard. On the left, a sidebar navigation includes EC2 Dashboard, EC2 Global View, Events, Instances (with sub-options like Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations), Images (AMIs, AMI Catalog), Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), Network & Security (Security Groups, Elastic IPs, Placement Groups), and CloudShell/Feedback. The main content area has tabs for Resources, Launch instance, Service health, and Zones. The Resources tab displays metrics for Instances (running: 0), Auto Scaling Groups (0), Capacity Reservations (0), Dedicated Hosts (0), Elastic IPs (0), Instances (0), Key pairs (1), Load balancers (0), Placement groups (0), Security groups (15), Snapshots (0), and Volumes (0). The Service health tab shows AWS Health Dashboard, Region (US East (N. Virginia)), Status (This service is operating normally), and Account attributes for Default VPC (vpc-014e267882d4c57b4) and Settings (Data protection and security). The Zones tab lists Zone names (us-east-1a, us-east-1b, us-east-1c) and Zone IDs (use1-az2, use1-az4, use1-az5).

Click on Launch instance that we are seeing in the below image at the right top corner.

The screenshot shows the AWS EC2 Instances page. The sidebar is identical to the previous dashboard. The main content area has tabs for Instances info, Actions, and Launch instances. The Instances info tab shows a search bar, filters (Last updated: less than a minute ago, All states), and a table header for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, and Public IPv6 DNS. Below the table, a message says "No instances" and "You do not have any instances in this region". A large "Launch instances" button is prominently displayed. A modal window titled "Select an instance" is partially visible at the bottom.

When we click on launch instance new page will be appear that we are seeing in the below image. It is nothing but the configuration of instances firstly we have to give name for the instance and select AMI based on our choice. I have selected ubuntu server AMI.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. In the 'Name and tags' step, the instance name is set to 'Putty'. The 'Software Image (AMI)' dropdown is set to 'Canonical, Ubuntu, 24.04, amd64...'. The 'Virtual server type (instance type)' dropdown is set to 't2.micro'. A tooltip for 't2.micro' states: '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, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of free data transfer between Amazon S3 buckets and your instance.' The 'Launch instance' button is highlighted in orange.

Under key pair select an existing key pair or create new key pair we have to select instance type. I have selected t2.micro which is under free tier eligible.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. In the 'Amazon Machine Image (AMI)' step, the AMI selected is 'Amazon Linux 2023 AMI'. The 'Virtual server type (instance type)' dropdown is set to 't2.micro'. A tooltip for 't2.micro' states: '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, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of free data transfer between Amazon S3 buckets and your instance.' The 'Launch instance' button is highlighted in orange.

After configuration check click on create instance. We will see a notification that Success so that our instance has been created successfully.

The screenshot shows the AWS EC2 'Launch an instance' page. A green success banner at the top reads: "Successfully initiated launch of instance (i-056ad8ed7425e92d7)". Below it, a 'Next Steps' section contains several cards:

- Create billing and free tier usage alerts**: To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds. Includes a "Create billing alerts" button.
- Connect to your instance**: Once your instance is running, log into it from your local computer. Includes a "Connect to instance" button and a "Learn more" link.
- Connect an RDS database**: Configure the connection between an EC2 instance and a database to allow traffic flow between them. Includes a "Connect an RDS database" button and a "Create a new RDS database" link.
- Create EBS snapshot policy**: Create a policy that automates the creation, retention, and deletion of EBS snapshots. Includes a "Create EBS snapshot policy" button.
- Manage detailed monitoring**: Enable or disable detailed monitoring for the instance. If you enable detailed monitoring, the Amazon EC2 console displays monitoring graphs with a 1-minute period. Includes a "Create Load Balancer" button.
- Create Load Balancer**: Create a application, network gateway or classic Elastic Load Balancer. Includes a "Create Load Balancer" button.
- Create AWS budget**: AWS Budgets allows you to create budgets, forecast spend, and take action on your costs and usage from a single location. Includes a "Create AWS budget" button.
- Manage CloudWatch alarms**: Create or update Amazon CloudWatch alarms for the instance. Includes a "Manage CloudWatch alarms" button.

At the bottom, there are links for CloudShell, Feedback, and a copyright notice: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.

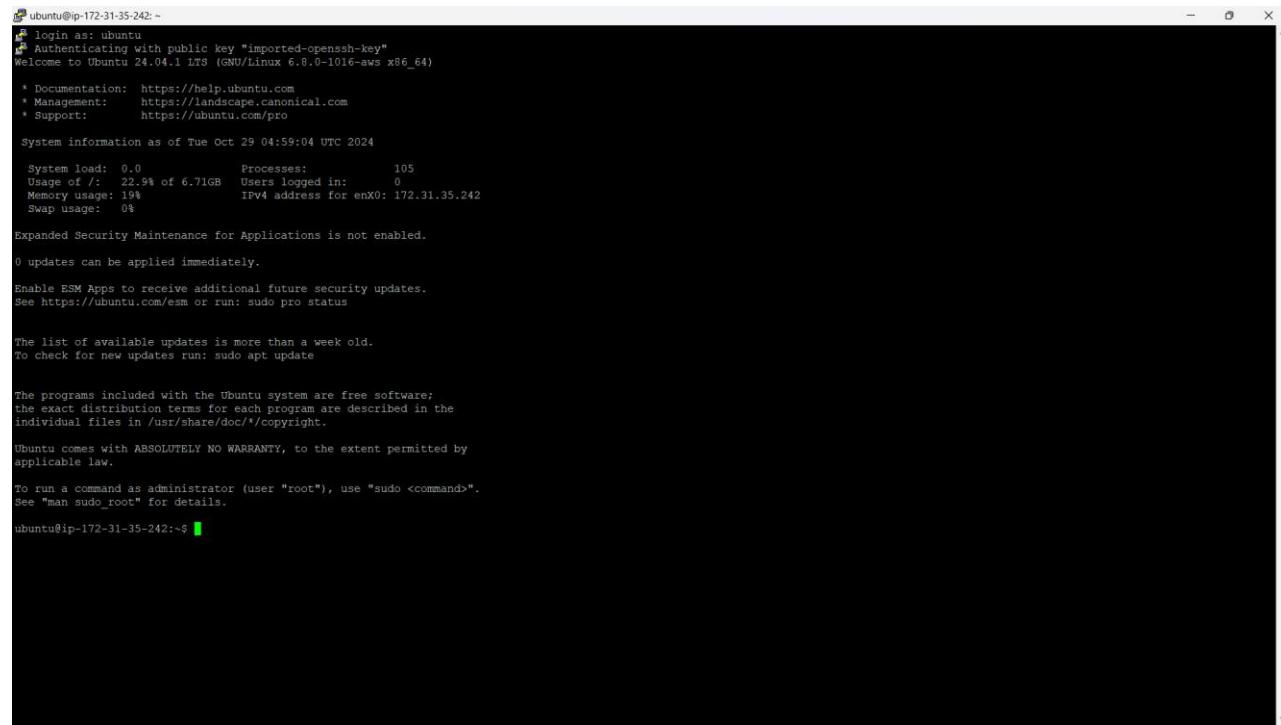
Download PuTTYgen open the PuTTYgen click load and select .pem file once loaded click save private key to save it in .ppk format This file will be used for authentication in PuTTY. Copy the public IPv4 address open PuTTY in the host name field enter public ip and go to SSH >Auth >connection and browse to select our .ppk file. Click on browse to connection and accept the security alert to connect.

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

Instance ID	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 address
i-056ad8ed7425e92d7	Initializing	View alarms +	us-east-1c	ec2-54-196-100-26.compute-1.amazonaws.com	54.196.100.26

To the left, a sidebar shows navigation options like Instances, Images, and Network & Security. A PuTTY terminal window is open on the left, showing the message: "54.196.100.26 - PuTTY" and "Authenticating with public key "imported-openssh-key"".

Once the connection is done we have to go to terminal prompt and we have to login as ubuntu and now run commands on your ubuntu instance.



```
ubuntu@ip-172-31-35-242: ~
login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Tue Oct 29 04:59:04 UTC 2024

System load: 0.0          Processes:           105
Usage of /: 22.9% of 6.71GB Users logged in:      0
Memory usage: 19%          IPv4 address for enX0: 172.31.35.242
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
to check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

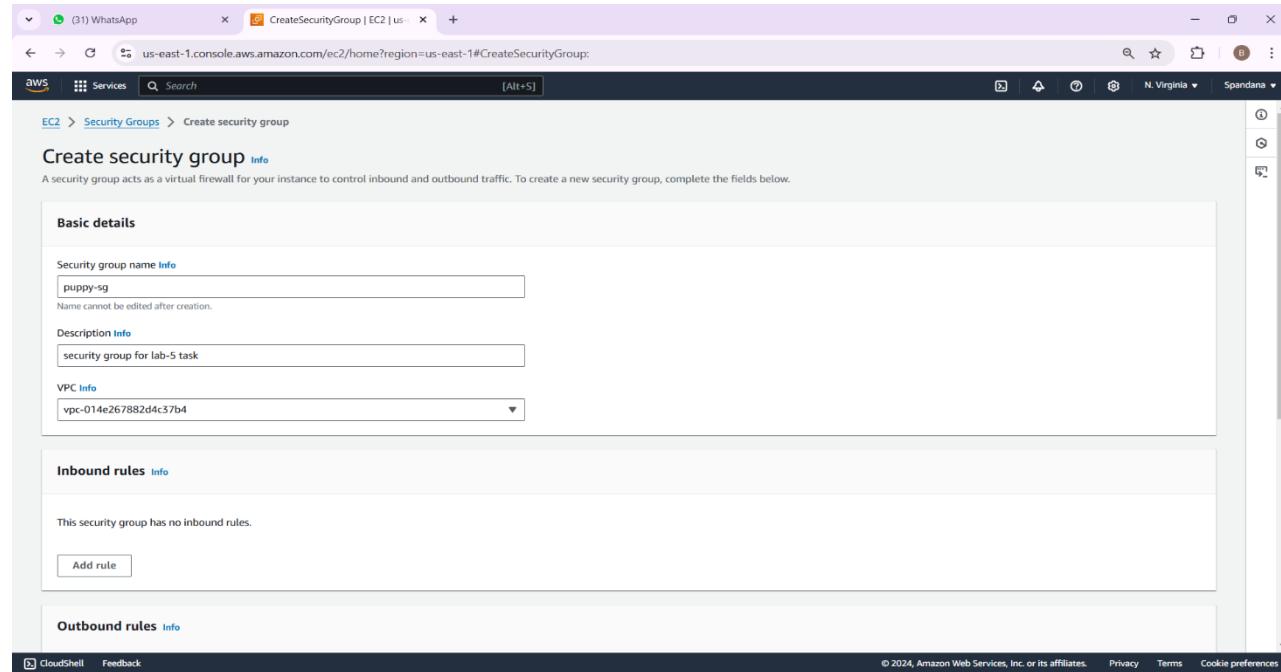
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-35-242: ~
```

## LAB 6 – VOLUMES AND SNAPSHOTS

In the left hand menu, under Network & Security , select Security groups click on create security group name security group. Provide a description select your VPC where our EC2 instance.



The screenshot shows the AWS CloudShell interface with the following details:

- Security Group Name:** puppy-sg
- Description:** security group for lab-5 task
- VPC:** vpc-014e267882d4c57b4
- Inbound Rules:** This security group has no inbound rules.
- Outbound Rules:** This section is currently empty.

In the below image we can able to see our security group has been created successfully.

The screenshot shows the AWS EC2 Security Groups console. A modal window at the top right indicates that a security group named 'sg-053a347df2c773255 | puppy-sg' was created successfully. The main view displays the details of this security group. The security group name is 'puppy-sg', the ID is 'sg-053a347df2c773255', and it is owned by user '637423550105'. It has no inbound rules and one outbound rule. The VPC ID is 'vpc-014e267882d4c37b4'. The 'Inbound rules' tab is selected, showing a table with columns for Name, Security group rule..., IP version, Type, Protocol, Port range, and Source. The table is empty, displaying the message 'No security group rules found'.

To check default rules in a new security group there are generally no inbound rules in the above image meaning all inbound traffic is denied by default. Outbound traffic typically allow traffic by default.

This screenshot shows the same AWS EC2 Security Groups console as the previous one, but with a different browser tab or session. The security group 'sg-053a347df2c773255 - puppy-sg' is displayed. The details are identical: security group name 'puppy-sg', ID 'sg-053a347df2c773255', owner '637423550105', and VPC ID 'vpc-014e267882d4c37b4'. The 'Inbound rules' tab is selected, showing the same empty table with the message 'No security group rules found'.

Check on command prompt we can able to run commands successfully in the below image.

```

ubuntu@ip-172-31-35-242: ~
login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support:   https://ubuntu.com/pro

System information as of Tue Oct 29 04:59:04 UTC 2024

System load: 0.0      Processes:          105
Usage of /: 22.9% of 6.71GB  Users logged in: 0
Memory usage: 19%      IPv4 address for enX0: 172.31.35.242
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-35-242:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-35-242:~$ 

```

Allowing inbound rules for Ports 80 and 22 in the inbound rules section click on Add rule in that we have to select Type: SSH, Protocol: TCP, Port range: 80, Type: SSH, Protocol: TCP, Port range: 22 Source type my IP/28 click on save changes and click on create security group to finalize.

We can able to see in the below image that Inbound security group rules successfully modified on security group.

Name	Security group rule...	IP version	Type	Protocol	Port range	Source
-	sgr-0a9ab91c0727c3036	IPv4	SSH	TCP	22	0.0.0.0/28
-	sgr-0314352960ce5c190	IPv4	HTTP	TCP	80	0.0.0.0/28

Navigate to instance click on the EC2 instance to which you want to attach this security group with the instance selected click Action > Security > this security group and select the new security group and remove any other security group if not needed after that update security group.

**Change security groups** Info

Amazon EC2 evaluates all the rules of the selected security groups to control inbound and outbound traffic to and from your instance. You can use this window to add and remove security groups.

**Instance details**

Instance ID i-056ad8ed7425e92d7	Network interface ID eni-03a87852ebff68001
------------------------------------	---

**Associated security groups**  
Add one or more security groups to the network interface. You can also remove security groups.

Security groups associated with the network interface (eni-03a87852ebff68001)

Security group name	Security group ID
puppy-sg	sg-053a347df2c773255

**Buttons:** Cancel, Save

**Instances (1/1) Info**

Last updated less than a minute ago

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 address
Putty	i-056ad8ed7425e92d7	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c	ec2-54-196-100-26.co...	54.196.100.26

**i-056ad8ed7425e92d7 (Putty)**

**Details** | Status and alarms | Monitoring | Security | Networking | Storage | Tags

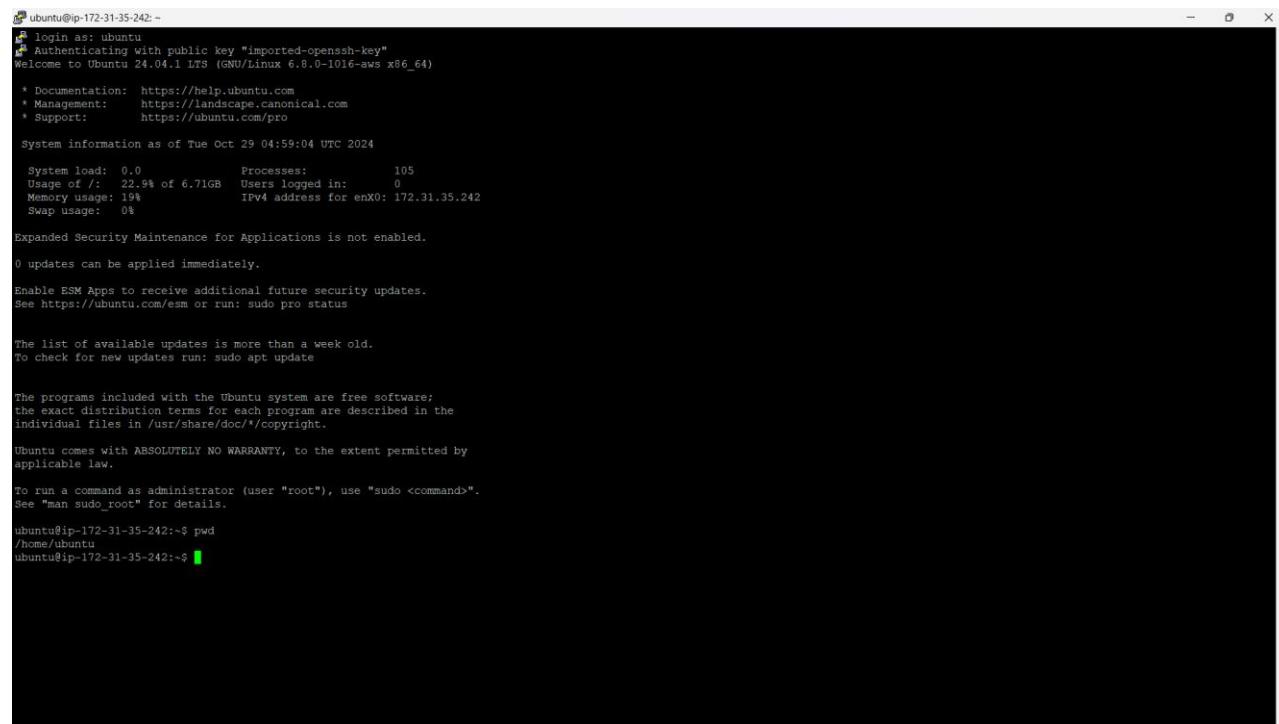
**Instance summary**

- Instance ID: i-056ad8ed7425e92d7
- IPV6 address: -
- Hostname type: IP name: ip-172-31-35-242.ec2.internal
- Answer private resource DNS name: IPv4 (A)
- Auto-assigned IP address: 54.196.100.26 [Public IP]

**Networking**

- Public IPv4 address: 54.196.100.26 [open address]
- Private IPv4 addresses: 172.31.35.242
- Public IPv4 DNS: ec2-54-196-100-26.compute-1.amazonaws.com [open address]
- Elastic IP addresses: -
- AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. [Opt-in]

Open the PuTTY and attempt to connect to the instance and we cannot able to access.



```
ubuntu@ip-172-31-35-242: ~
[1] 1: Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Tue Oct 29 04:59:04 UTC 2024

System load: 0.0 Processes: 105
Usage of /: 22.5% of 6.71GB Users logged in: 0
Memory usage: 19% IPv4 address for enx0: 172.31.35.242
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

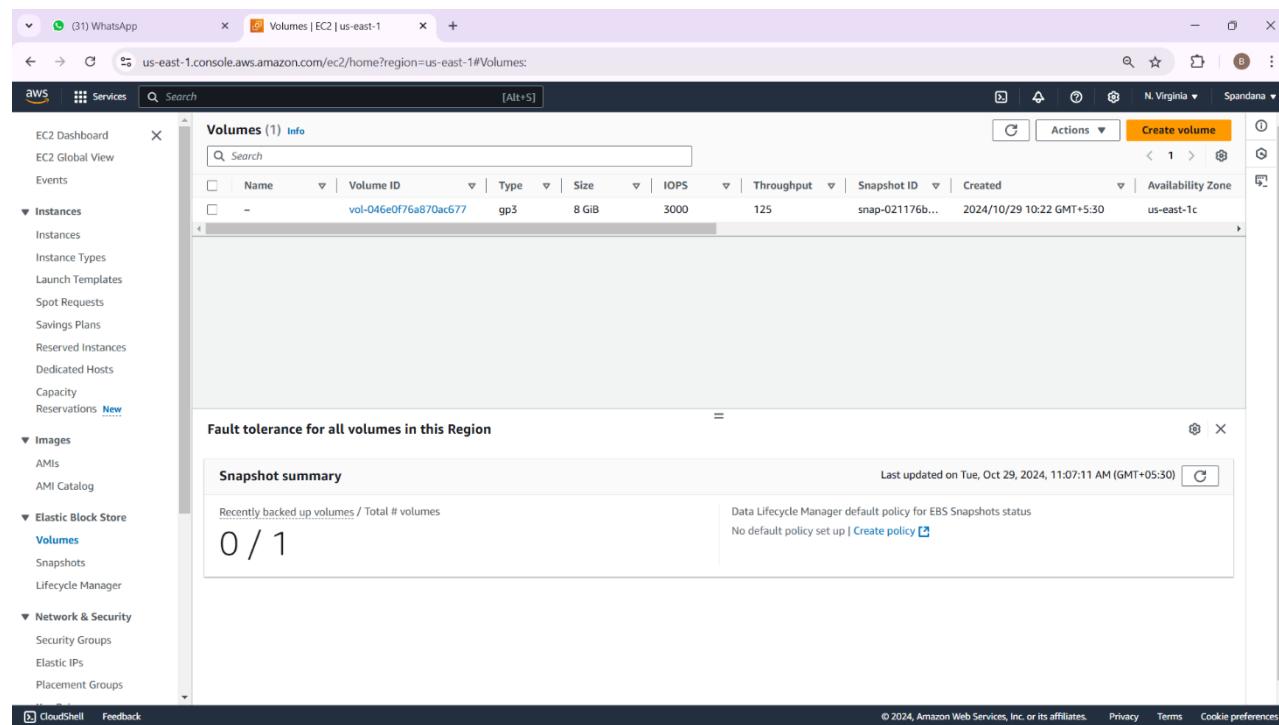
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-35-242:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-35-242:~$
```

## LAB 6 – VOLUMES AND SNAPSHOTS

Create a new volume in the left-hand menu under Elastic Block store select volumes and click on create volume.



The screenshot shows the AWS CloudFront console with the 'Volumes' page open. The left sidebar shows navigation links for EC2 Dashboard, EC2 Global View, Events, Instances, Images, and Elastic Block Store. Under 'Elastic Block Store', 'Volumes' is selected. The main content area displays a table titled 'Volumes (1) Info' with one row of data:

Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot ID	Created	Availability Zone
-	vol-046e0f76a870ac677	gp3	8 GiB	3000	125	snap-021176b...	2024/10/29 10:22 GMT+5:30	us-east-1c

Below the table, a section titled 'Fault tolerance for all volumes in this Region' shows a 'Snapshot summary' with the text '0 / 1' indicating no snapshots have been created. A note states 'Data Lifecycle Manager default policy for EBS Snapshots status'.

Configure volume settings size 5GiB Availability Zone select the same zone as your Ec2 instance and volume type General Purpose SSD. Click on the create volume to create EBS Volume.

The screenshot shows the 'Create volume' page in the AWS Management Console. The 'Volume settings' section is open, displaying the following configuration:

- Volume type:** General Purpose SSD (gp3)
- Size (GiB):** 5
- IOPS:** 3000
- Throughput (MiB/s):** 125
- Availability Zone:** us-east-1a
- Snapshot ID - optional:** Don't create volume from a snapshot
- Encryption:** Use Amazon EBS encryption as an encryption solution for your EBS resources associated with your EC2 instances.

At the bottom right of the form, there is a large orange button labeled 'Create volume'.

The screenshot shows the 'Create volume' page in the AWS Management Console, with the 'Summary' tab selected. The summary information includes:

- Volume type: General Purpose SSD (gp3)
- Size: 5 GiB
- Availability Zone: us-east-1c
- Encryption: Use Amazon EBS encryption as an encryption solution for your EBS resources associated with your EC2 instances.

Below the summary, there are sections for 'Tags - optional' (with a note about tags being used for search and cost tracking), 'Snapshot summary' (with a note to click refresh to view backup information), and a 'Create volume' button at the bottom right.

Volume has been created successfully we can able to see in the below image. Once the volume is created select it in the volumes list and click on Action > Attach volume in the instance file choose our running instance and select /dev/sdf or similar device name. Click on Attach volume.

Connect our PuTTY and list all the attached disks to verify our new volume using lsblk we can see the attached volumes in the below image.

```
ubuntu@ip-172-31-35-242: ~
login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Tue Oct 29 05:53:17 UTC 2024

System load: 0.0      Processes:          108
Usage of /: 23.2B of 6.71GB  Users logged in:    1
Memory usage: 21%      IPv4 address for enX0: 172.31.35.242
Swap usage:  0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

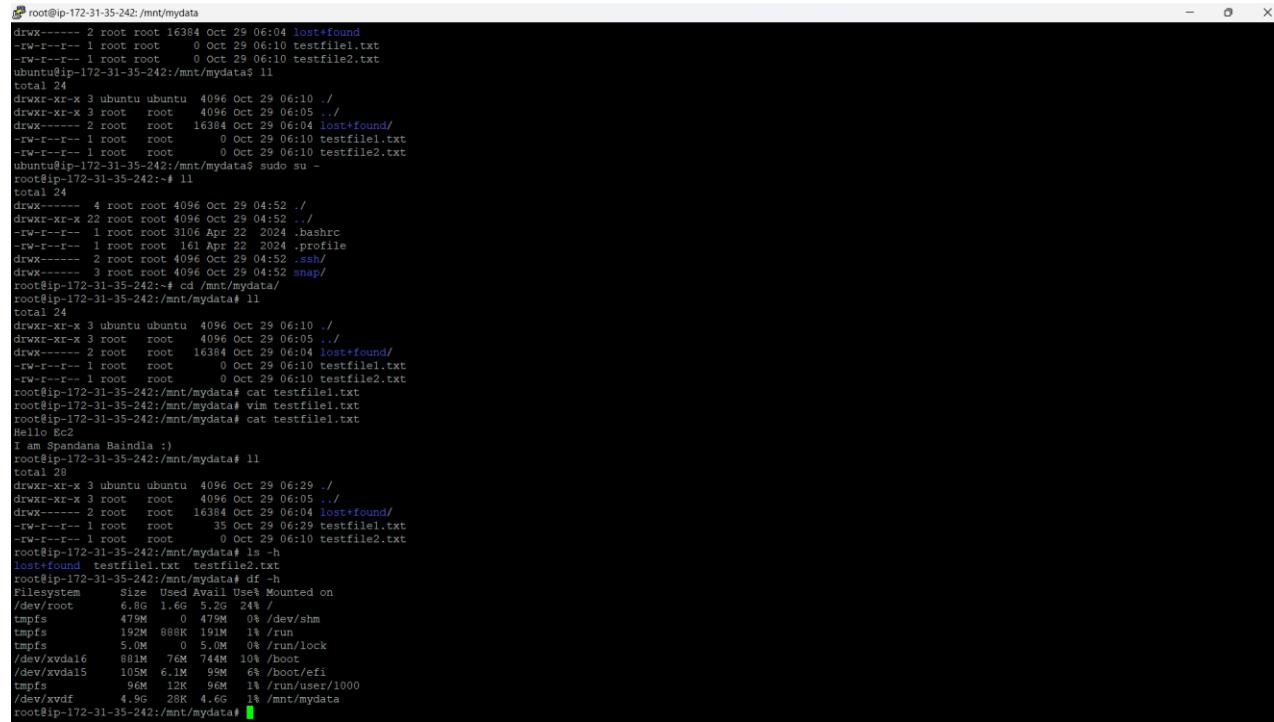
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

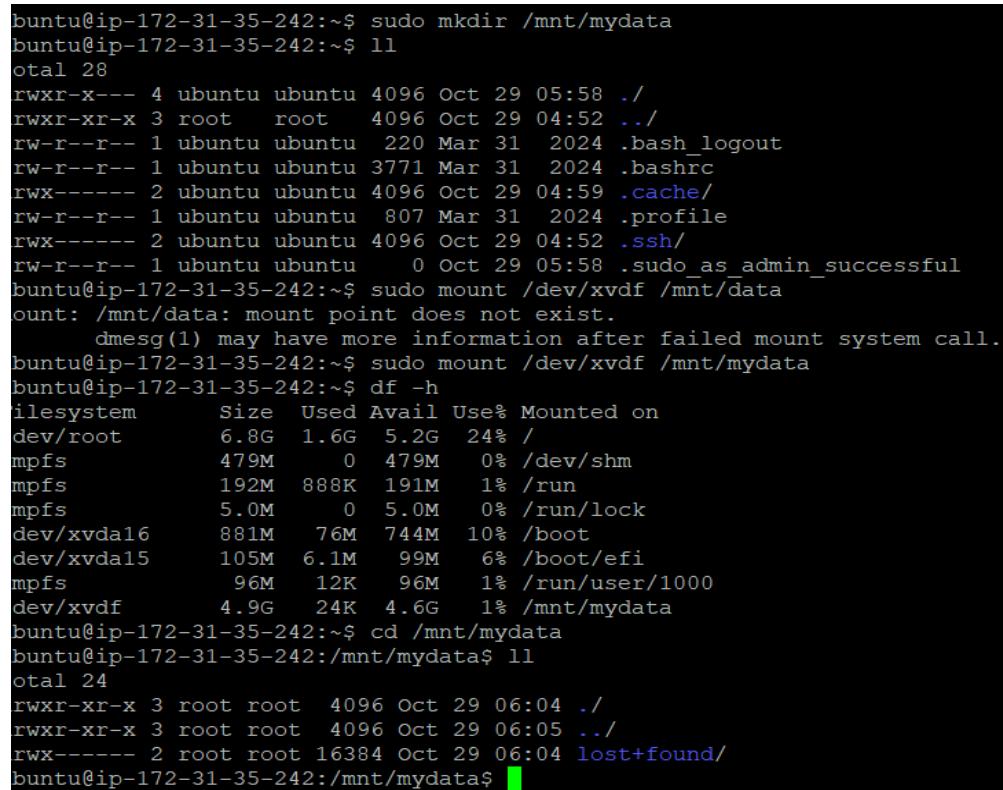
Last login: Tue Oct 29 04:59:46 2024 from 49.206.59.74
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-35-242:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-35-242:~$ lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0   7:0    0 25.2M  1 loop /snap/amazon-ssm-agent/7993
loop1   7:1    0 38.8M  1 loop /snap/snapd/21759
loop2   7:2    0 55.7M  1 loop /snap/core18/2029
xvda  202:0    0   8G  0 disk
└─xvda1 202:1    0   8G  0 part /
└─xvda4 202:14   0   4M  0 part
└─xvda5 202:15   0 106M 0 part /boot/efi
└─xvda6 259:0    0  913M 0 part /boot
xvdo  202:224   0   5G  0 disk
ubuntu@ip-172-31-35-242:~$
```

Create Test files using touch command. We can able to see 2 testfiles in the below image.

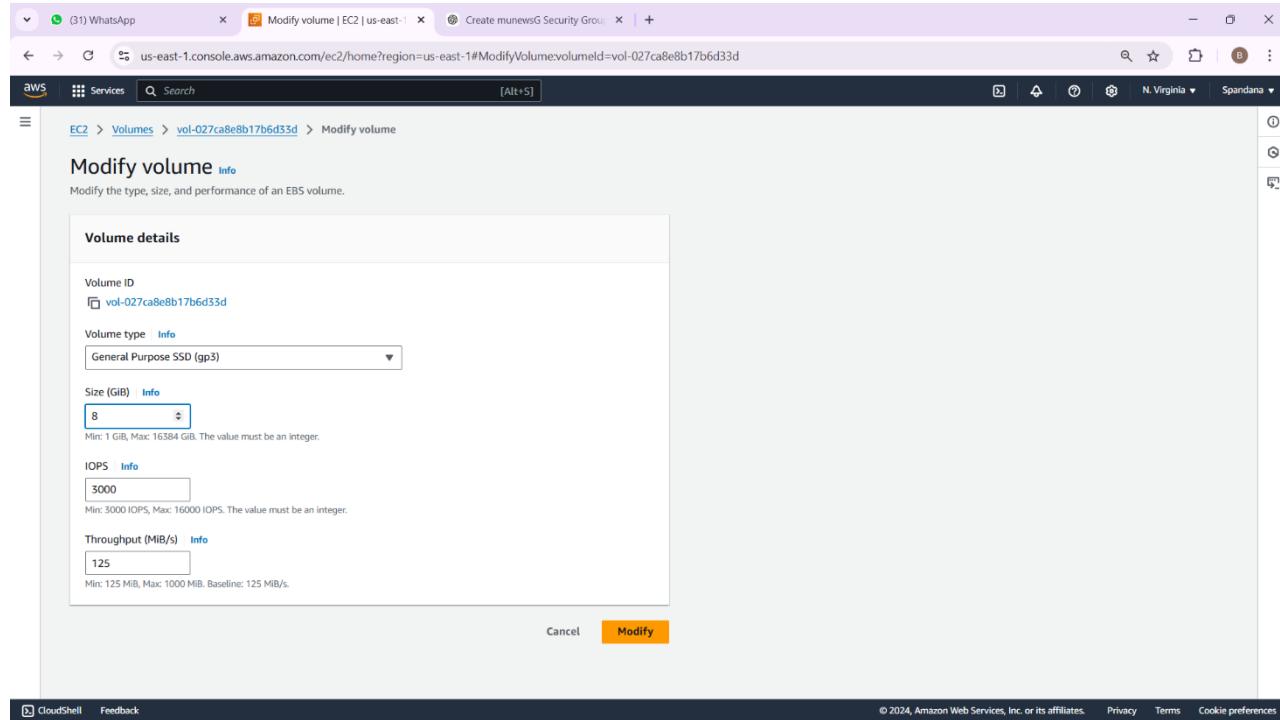


```
root@ip-172-31-35-242:/mnt/mydata
drwxr--r-- 2 root root 16384 Oct 29 06:04 lost+found
-rw-r--r-- 1 root root 0 Oct 29 06:10 testfile1.txt
-rw-r--r-- 1 root root 0 Oct 29 06:10 testfile2.txt
ubuntu@ip-172-31-35-242:/mnt/mydata$ ll
total 24
drwxr-xr-x 3 ubuntu ubuntu 4096 Oct 29 06:10 ../
drwxr-xr-x 3 root root 4096 Oct 29 06:05 ../.
drwxr--r-- 2 root root 16384 Oct 29 06:04 lost+found/
-rw-r--r-- 1 root root 0 Oct 29 06:10 testfile1.txt
-rw-r--r-- 1 root root 0 Oct 29 06:10 testfile2.txt
ubuntu@ip-172-31-35-242:/mnt/mydata$ sudo su -
root@ip-172-31-35-242:~# ll
total 24
drwxr--r-- 4 root root 4096 Oct 29 04:52 /
drwxr-xr-x 22 root root 4096 Oct 29 04:52 ../
-rw-r--r-- 1 root root 31064 Oct 22 2024 .bashrc
-rw-r--r-- 1 root root 161 Apr 22 2024 .profile
drwxr--r-- 2 root root 4096 Oct 29 04:52 .ssh/
drwxr--r-- 3 root root 4096 Oct 29 04:52 snap/
root@ip-172-31-35-242:/mnt/mydata$ cd /mnt/mydata/
root@ip-172-31-35-242:/mnt/mydata# ll
total 24
drwxr-xr-x 3 ubuntu ubuntu 4096 Oct 29 06:10 ../
drwxr-xr-x 3 root root 4096 Oct 29 06:05 ../.
drwxr--r-- 2 root root 16384 Oct 29 06:04 lost+found/
-rw-r--r-- 1 root root 0 Oct 29 06:10 testfile1.txt
-rw-r--r-- 1 root root 0 Oct 29 06:10 testfile2.txt
root@ip-172-31-35-242:/mnt/mydata$ cat testfile1.txt
root@ip-172-31-35-242:/mnt/mydata$ vim testfile1.txt
root@ip-172-31-35-242:/mnt/mydata$ cat testfile1.txt
Hello Ec2
I am Spandana Baindia :)
root@ip-172-31-35-242:/mnt/mydata# ll
total 28
drwxr-xr-x 3 ubuntu ubuntu 4096 Oct 29 06:29 ../
drwxr-xr-x 3 root root 4096 Oct 29 06:05 ../.
drwxr--r-- 2 root root 16384 Oct 29 06:04 lost+found/
-rw-r--r-- 1 root root 35 Oct 29 06:29 testfile1.txt
-rw-r--r-- 1 root root 0 Oct 29 06:10 testfile2.txt
root@ip-172-31-35-242:/mnt/mydata$ ls -h
lost+found testfile1.txt testfile2.txt
root@ip-172-31-35-242:/mnt/mydata$ df -h
Filesystem      Size   Used  Avail Use% Mounted on
/dev/root       6.0G  1.6G  5.2G  24% /
tmpfs          479M    0  479M   0% /dev/shm
tmpfs          192M  888K  191M   1% /run
tmpfs          5.0M    0  5.0M   0% /run/lock
/dev/xvda16     881M   76M  744M  10% /boot
/dev/xvda15     105M   6.1M  99M   6% /boot/efi
tmpfs          96M   12K  96M   1% /run/user/1000
/dev/xvdf       4.9G   24K  4.6G  1% /mnt/mydata
root@ip-172-31-35-242:/mnt/mydata#
```

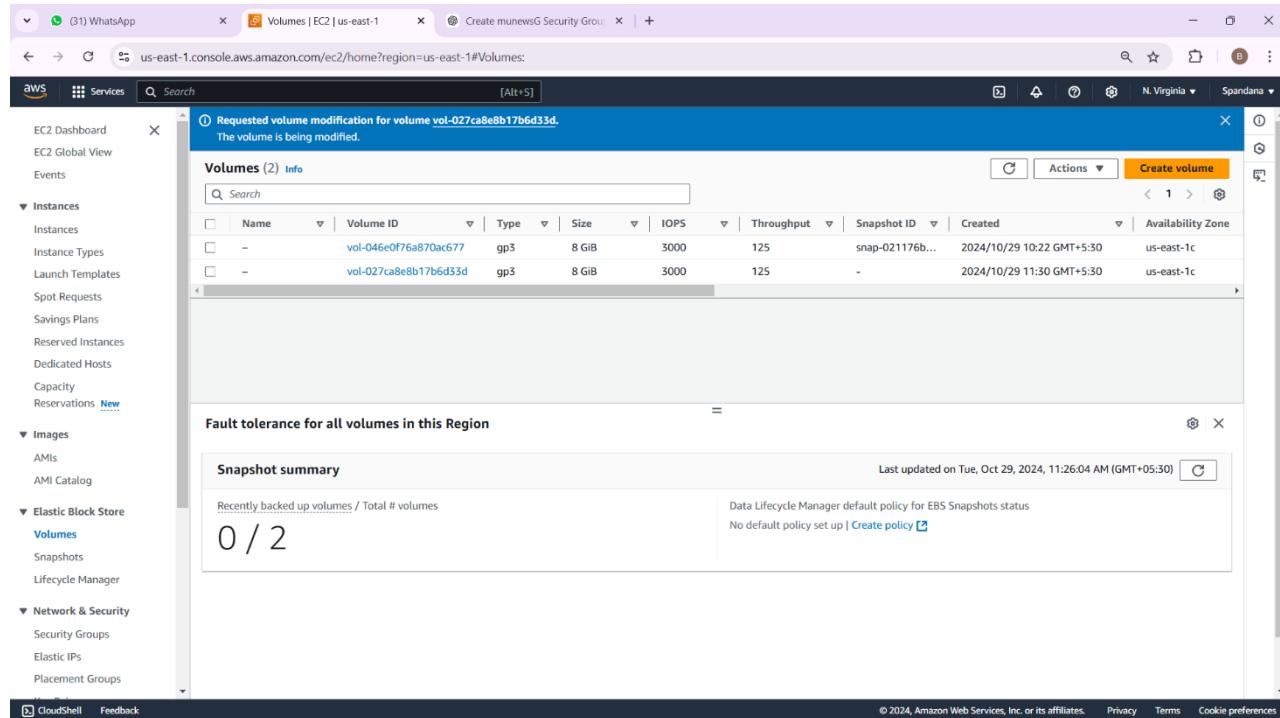


```
buntu@ip-172-31-35-242:~$ sudo mkdir /mnt/mydata
buntu@ip-172-31-35-242:~$ ll
otal 28
rwxr-x--- 4 ubuntu ubuntu 4096 Oct 29 05:58 ../
rwxr-xr-x 3 root root 4096 Oct 29 04:52 ../.
rw-r--r-- 1 ubuntu ubuntu 220 Mar 31 2024 .bash_logout
rw-r--r-- 1 ubuntu ubuntu 3771 Mar 31 2024 .bashrc
rwx----- 2 ubuntu ubuntu 4096 Oct 29 04:59 .cache/
rw-r--r-- 1 ubuntu ubuntu 807 Mar 31 2024 .profile
rwx----- 2 ubuntu ubuntu 4096 Oct 29 04:52 .ssh/
rw-r--r-- 1 ubuntu ubuntu 0 Oct 29 05:58 sudo_as_admin_successful
buntu@ip-172-31-35-242:~$ sudo mount /dev/xvdf /mnt/data
ount: /mnt/data: mount point does not exist.
        dmesg(1) may have more information after failed mount system call.
buntu@ip-172-31-35-242:~$ sudo mount /dev/xvdf /mnt/mydata
buntu@ip-172-31-35-242:~$ df -h
Filesystem      Size   Used  Avail Use% Mounted on
dev/root       6.8G  1.6G  5.2G  24% /
mpfs          479M    0  479M   0% /dev/shm
mpfs          192M  888K  191M   1% /run
mpfs          5.0M    0  5.0M   0% /run/lock
dev/xvda16     881M   76M  744M  10% /boot
dev/xvda15     105M   6.1M  99M   6% /boot/efi
mpfs          96M   12K  96M   1% /run/user/1000
dev/xvdf       4.9G   24K  4.6G  1% /mnt/mydata
buntu@ip-172-31-35-242:~$ cd /mnt/mydata
buntu@ip-172-31-35-242:/mnt/mydata$ ll
otal 24
rwxr-xr-x 3 root root 4096 Oct 29 06:04 ../
rwxr-xr-x 3 root root 4096 Oct 29 06:05 ../.
rwx----- 2 root root 16384 Oct 29 06:04 lost+found/
buntu@ip-172-31-35-242:/mnt/mydata$
```

Go to volumes and select our 5GiB volume click on Actions > Modify volume change the size to 8GiB and click on Modify. Confirm the modification. AWS will automatically resize the volume.



In the below image we can able to see that the volume size has been increased from 5GiB to 8GiB.



Return to our PuTTY and use grow part to extend the partition to occupy the increased volume space and use resize2fs to extend the filesystem to utilize the full size in the below image.

```
root@ip-172-31-35-242:/mnt/mydata# testfile1.txt testfile2.txt
root@ip-172-31-35-242:/mnt/mydata# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       6.0G  1.6G  5.2G  24% /
tmpfs          479M     0  479M   0% /tmp
tmpfs          19M    0  19M   0% /run
tmpfs          5.0M    0  5.0M   0% /run/lock
/dev/xvda16    881M  76M  744M  10% /boot
/dev/xvda15   105M  6.1M  99M   6% /boot/efi
tmpfs          96M   12K  96M   1% /run/user/1000
/dev/xvdf      4.9G  28K  4.6G  1% /mnt/mydata
root@ip-172-31-35-242:/mnt/mydata# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0    7:0    2 25.2M  1 loop /snap/amazon-ssm-agent/7993
loop1    7:1    0 38.8M  1 loop /snap/snapd/21759
loop2    7:2    0 55.7M  1 loop /snap/core18/2829
xvda   202:0    0   8G  0 disk 
└─xvda1  202:1    0   7G  0 part /
└─xvda14 202:14   0   4M  0 part
└─xvda15 202:15   0 106M  0 part /boot/efi
└─xvda16 259:0   0  913M  0 part /boot
xvdf   202:80   0   8G  0 disk /mnt/mydata
root@ip-172-31-35-242:/mnt/mydata#
```

Go to volumes in the EC2 select the volume click on Action > Create snapshot enter a description click on create snapshot. Snapshot will appear under the snapshots in the Elastic Block Store section.

The screenshot shows the AWS EC2 Volumes page. A success message at the top states: "Successfully created snapshot snap-0d724fc5872653e06 from volume vol-027ca8eb8b17b6d33d. If you need your snapshot to be immediately available consider using Fast Snapshot Restore." Below this, there is a table titled "Volumes (2) Info" showing two volumes: "vol-046e0f76a870ac677" and "vol-027ca8eb8b17b6d33d". The "Actions" button is highlighted in orange. On the right side of the page, there is a "Snapshot summary" section with the text "Fault tolerance for all volumes in this Region" and "0 / 2 Recently backed up volumes / Total # volumes".

The screenshot shows the AWS Snapshots page. A success message at the top states: "Successfully created snapshot snap-0d724fc5872653e06 from volume vol-027ca8eb8b17b6d33d. If you need your snapshot to be immediately available consider using Fast Snapshot Restore." Below this, there is a table titled "Schemas (1) Info" showing one snapshot: "snap-0d724fc5872653e06". The "Actions" button is highlighted in orange. At the bottom of the page, there is a message: "Select a snapshot above." The left sidebar shows the navigation menu for EC2 services.

Select the snapshot you created earlier with the snapshot selected click > Action > Create Volume in that Select Availability Zone as your EC2 instance select Volume size and Volume type Choose the desired type Click create Volume to create a new volume from the snapshot.

The screenshot shows the AWS Management Console interface for creating a new volume from a snapshot. The top navigation bar includes tabs for WhatsApp, Instances | EC2 | us-east-1, Snapshots | EC2 | us-east-1, Volumes | EC2 | us-east-1, and Create munewsG Security Group. The left sidebar is collapsed. The main content area displays a success message: "Successfully created volume vol-0af1b1655bb8c72265." Below this, a table lists the newly created volume:

Name	Snapshot ID	Volume size	Description	Storage tier	Snapshot status	Started	Progress
-	snap-08baafff24b8bee43f	8 GiB	snapshot create	Standard	Completed	2024/10/29 12:12 GMT+5:30	Available (100%)

A note at the bottom of the table says "Select a snapshot above." The bottom right corner of the page includes copyright information: "© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences".

In the Volumes section under Elastic Block Store find the newly created volume click Action > Attach Volume select our EC2 instance and choose a device name Click on Attach Volume to attach it to your instance.

The screenshot shows the AWS Management Console interface for attaching a new volume to an EC2 instance. The top navigation bar includes tabs for WhatsApp, Instances | EC2 | us-east-1, Snapshots | EC2 | us-east-1, Attach volume | EC2 | us-east-1, and Create munewsG Security Group. The left sidebar shows the path: EC2 > Volumes > vol-0af1b1655bb8c72265 > Attach volume. The main content area is titled "Attach volume" and contains the following fields:

- Basic details** section:
  - Volume ID: vol-0af1b1655bb8c72265
  - Availability Zone: us-east-1c
  - Instance: i-056ad8ed7425e92d7 (selected)
  - Device name: /dev/sdf
- A note in a callout box: "Newer Linux kernels may rename your devices to /dev/xvdf through /dev/xwp internally, even when the device name entered here (and shown in the details) is /dev/sdf through /dev/sdp."
- Buttons: "Cancel" and "Attach volume" (highlighted in orange).

The bottom right corner of the page includes copyright information: "© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences".

Successfully attached Volume to EC2 instance.

The screenshot shows the AWS Cloud Console interface. The left sidebar navigation includes 'Instances', 'Elastic Block Store' (selected), and 'Network & Security'. The main content area is titled 'Successfully attached volume vol-0af1b1655b8c72265 to instance i-056ad8ed7425e92d7.' Below this, the 'Volumes' table lists two entries:

Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot ID	Created	Availability Zone
-	vol-046e0f76a870ac677	gp3	8 GiB	3000	125	snap-021176b...	2024/10/29 10:22 GMT+5:30	us-east-1c
-	vol-0af1b1655b8c72265	gp3	8 GiB	3000	125	snap-08baafff2...	2024/10/29 12:16 GMT+5:30	us-east-1c

Below the table, a message states 'Fault tolerance for all volumes in this Region'. The 'Snapshot summary' section indicates '0 / 1' recently backed up volumes. The bottom right corner shows the copyright notice: '© 2024, Amazon Web Services, Inc. or its affiliates.'

Open PuTTY and connect to our instance run the lsblk command to verify the attached volume. We will see the new volume listed.

```
root@ip-172-31-35-242:~#
[login as: ubuntu
authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Tue Oct 29 06:49:07 UTC 2024

System load: 0.0 Processes: 116
Usage of /: 23.2% of 6.71GB Users logged in: 1
Memory usage: 22% IPv4 address for enX0: 172.31.35.242
Swap usage: 0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Tue Oct 29 05:53:18 2024 from 49.206.59.74
ubuntu@ip-172-31-35-242:~$ sudo su -
root@ip-172-31-35-242:~# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
loop0 7:0 0 29.7M 1 loop /snap/amazon-ssm-agent/7993
loop1 7:1 0 6.8M 1 loop /snap/amazon-ssm-agent/21759
loop2 7:2 0 55.7M 1 loop /snap/core18/2829
xvda 202:0 0 8G 0 disk
└─xvda1 202:1 0 7G 0 part /
└─xvda14 202:14 0 4M 0 part
└─xvda15 202:15 0 106M 0 part /boot/efi
└─xvda16 259:0 0 913M 0 part /boot
xvdf 202:80 0 8G 0 disk /mnt/mydata
root@ip-172-31-35-242:~#
```

## LAB 7- AMI (AMAZON MACHINE IMAGE)

Select the instance and with the instance selected, click Actions > Image and templates > Create image. In the Create Image name image description optionally, add a description instance volumes verify that the root volume and any additional volumes as needed. You can adjust the size or add new volumes as needed. No reboot check this option if you want to avoid a reboot during image creation.

The screenshot shows the 'Create image' configuration page for an EC2 instance. The instance ID is i-010c0f811241e79fd (putty). The 'Image name' field is empty, and the 'Image description - optional' field contains 'Image description'. The 'Reboot instance' checkbox is checked. Under 'Instance volumes', there is one volume listed: an EBS General Purpose (SSD) volume of size 8 GiB. The 'Delete on termination' and 'Encrypted' checkboxes are checked. A note at the bottom states: 'During the image creation process, Amazon EC2 creates a snapshot of each of the above volumes.' The page includes standard AWS navigation and footer links.

The screenshot shows the 'Instances' page with one instance listed: i-010c0f811241e79fd (putty), which is currently running. The status bar at the top indicates 'Currently creating AMI ami-0f43195653edc40bb from instance i-010c0f811241e79fd. Check that the AMI status is "Available" before deleting the instance or carrying out other actions related to this AMI.' The page displays detailed information about the instance, including its public and private IP addresses, instance type (t2.micro), and various network and storage details. The left sidebar shows the navigation menu for EC2 services.

Successfully created Amazon Machine Image we can able to see in the below image.

The screenshot shows the AWS Cloud Console interface for managing Amazon Machine Images (AMIs). The left sidebar navigation includes 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Instances' (selected), 'Images', 'AMI Catalog', 'Elastic Block Store', 'Network & Security', and 'Load Balancing'. The main content area displays a table titled 'Amazon Machine Images (AMIs) (1) Info'. The table has columns for 'Name', 'AMI name', 'AMI ID', 'Source', 'Owner', 'Visibility', and 'Status'. One row is present, showing 'MyWebServerAMI' as the name, 'ami-0f43195653ed40bb' as the AMI ID, '637423550105/MyWebServerAMI' as the Source, '637423550105' as the Owner, 'Private' as the Visibility, and 'Pending' as the Status. Below the table, a modal window titled 'Select an AMI' is open, listing the same single AMI entry.

## LAB 8 – LOAD BALANCER

Launch two EC2 instances ensure both instances are in same availability zone for simplicity and assign security groups to the instances that allows only inbound traffic from load balancer on port 80.

The screenshot shows the AWS Cloud Console interface for managing EC2 instances. The left sidebar navigation includes 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Instances' (selected), 'Images', 'AMI Catalog', 'Elastic Block Store', 'Network & Security', and 'Load Balancing'. The main content area displays a table titled 'Instances (3) Info'. The table has columns for 'Name', 'Instance ID', 'Instance state', 'Instance type', 'Status check', 'Alarm status', 'Availability Zone', 'Public IPv4 DNS', 'Public IPv4...', and 'Elastic IP'. Three instances are listed: 'i-09358686387ba7b74' (Running, t2.micro, Initializing, us-east-1c, ec2-3-89-48-61.compute..., 3.89.48.61, -), 'i-0f2678099711de87' (Running, t2.micro, Initializing, us-east-1c, ec2-3-84-87-37.compute..., 3.84.87.37, -), and 'putty' (Stopped, t2.micro, -, us-east-1c, -, -, -). Below the table, a modal window titled 'Select an instance' is open, listing the three instances.

In the below image we can see that inbound traffic on port 80 (http)

The screenshot shows the 'Edit inbound rules' step of the 'ModifyInboundSecurityGroupRules' wizard. It lists two rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-096da430825d27925	SSH	TCP	22	Custom	0.0.0.0/0
-	HTTP	TCP	80	Custom	-

Buttons at the bottom include 'Cancel', 'Preview changes', and 'Save rules'.

The below image shows that the before modification of inbound rules.

The screenshot shows the 'Details' tab for a security group named 'sg-0e7553ba4ad1b9d36'. It has one inbound rule:

Name	IP version	Type	Protocol	Port range	Source	Description
sgr-0e6a96f18ef69c2f	IPv4	SSH	TCP	22	0.0.0.0/0	-

Buttons at the bottom include 'Actions', 'Manage tags', and 'Edit inbound rules'.

The below image shows that the inbound rules has been successfully updated.

The screenshot shows the AWS EC2 Security Groups page. A green banner at the top indicates that the inbound security group rules have been successfully modified on the security group (sg-0e7553ba4ad1b9d36). Below the banner, the security group details are shown, including its name (sg-0e7553ba4ad1b9d36 - launch-wizard-16), ID, owner (637423550105), and various counts of rules. The 'Inbound rules' tab is selected, displaying two entries: one for port 80 (HTTP) and another for port 22 (SSH). The table columns include Name, Security group rule..., IP version, Type, Protocol, Port range, Source, and Description.

Connect both instances using PuTTY and installing web server using ec2-user for nginx. Switch to root user through **sudo su -**, update the package index using **yum install update -y**, Install nginx using **yum install nginx -y** and start and enable nginx server using **systemctl start nginx && systemctl enable nginx**.

```

root@ip-172-31-19-221:~#
[ec2-user@ip-172-31-19-221 ~]$ sudo su -
[root@ip-172-31-19-221 ~]# yum update -y
Last metadata expiration check: 0:02:10 ago on Tue Oct 29 11:07:44 2024.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-19-221 ~]# yum install nginx -y
Last metadata expiration check: 0:02:20 ago on Tue Oct 29 11:07:44 2024.
Dependencies resolved.
=====
Package           Arch   Version      Repository Size
=====
Installing:
nginx            x86_64  1:1.24.0-1.amzn2023.0.4    amazonlinux 33 k
Installing dependencies:
generic-logos-httdp noarch  18.0.0-12.amzn2023.0.3  amazonlinux 19 k
gperftools-libs   x86_64  2.9.1-1.amzn2023.0.3    amazonlinux 300 k
libcurl          x86_64  1.4.0-5.amzn2023.0.2    amazonlinux 66 k
nginx-core       x86_64  1:1.24.0-1.amzn2023.0.4    amazonlinux 586 k
nginx-fs          noarch  1:1.24.0-1.amzn2023.0.4    amazonlinux 9.8 k
nginx-mimetypes   noarch  2.1.49-3.amzn2023.0.3    amazonlinux 21 k
=====
Transaction Summary
=====
Install 7 Packages

Total download size: 1.0 M
Installed size: 3.4 M
Downloading Packages:
(1/7): libunwind-1.4.0-5.amzn2023.0.2.x86_64.rpm 1.2 MB/s | 66 kB     0:00
(2/7): generic-logos-httdp-18.0.0-12.amzn2023.0.3 305 kB/s | 19 kB     0:00
(3/7): gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64 4.4 MB/s | 308 kB    0:00
(4/7): nginx-1.24.0-1.amzn2023.0.4.x86_64.rpm 1.6 MB/s | 33 kB     0:00
(5/7): nginx-fs-1.24.0-1.amzn2023.0.4.i686 425 kB/s | 9.8 kB    0:00
(6/7): nginx-mimetypes-2.1.49-3.amzn2023.0.3.no 975 kB/s | 21 kB     0:00
(7/7): nginx-core-1.24.0-1.amzn2023.0.4.x86_64.i686 13 MB/s | 586 kB   0:00
=====
Total                                         6.8 MB/s | 1.0 MB  0:00
Running transaction check
Transaction check succeeded.

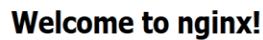
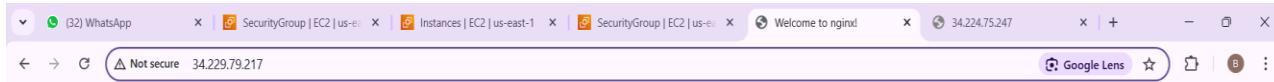
```

We can able to check the status through systemctl status and we can able to see nginx server is in active state in below image.

```
[root@ip-172-31-19-221 ~]# systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: (/usr/lib/systemd/system/nginx.service; enabled; preset: disabled)
     Active: active (running) since Tue 2024-10-29 11:10:54 UTC; 1min 6s ago
   Process: 25684 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
  Process: 25685 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
  Process: 25686 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
 Main PID: 25687 (nginx)
    Tasks: 2 (limit: 1112)
   Memory: 2.2M
      CPU: 4ms
     CGroup: /system.slice/nginx.service
           ├─25687 nginx: master process /usr/sbin/nginx"
           └─25688 nginx: worker process"

Oct 29 11:10:54 ip-172-31-19-221.ec2.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Oct 29 11:10:54 ip-172-31-19-221.ec2.internal nginx[25685]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Oct 29 11:10:54 ip-172-31-19-221.ec2.internal nginx[25685]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Oct 29 11:10:54 ip-172-31-19-221.ec2.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
[root@ip-172-31-19-221 ~]#
```

Access nginx web server through copy each instance's public IP address in browser to verify. If the server is properly connected then we can able to see like a below image and we can confirm that nginx server is connected.



If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*

Connect both instances using PuTTY and installing web server using ec2-user for Apache. Switch to root user through **sudo su -**, update the package index using **yum install update -y**, Install Apache using **yum install httpd -y**.

```
root@ip-172-31-22-170:~#
[ec2-user@ip-172-31-22-170 ~]$ sudo su -
[root@ip-172-31-22-170 ~]# yum update -y
Last metadata expiration check: 0:02:32 ago on Tue Oct 29 11:14:01 2024.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-22-170 ~]# yum install httpd -y
Last metadata expiration check: 0:02:40 ago on Tue Oct 29 11:14:01 2024.
Dependencies resolved.

Transaction Summary
=====================================================================
Installing: httpd x86_64 2.4.62-1.amzn2023
Installing dependencies:
apr x86_64 1.7.2-2.amzn2023.0.2
apr-util x86_64 1.6.3-1.amzn2023.0.1
generic-logos-httpd noarch 18.0.0-12.amzn2023.0.3
httpd-core x86_64 2.4.62-1.amzn2023
httpd-filesystem noarch 2.4.62-1.amzn2023
httpd-tools x86_64 2.4.62-1.amzn2023
libbrotli x86_64 1.0.9-4.amzn2023.0.2
mailcap noarch 2.1.49-3.amzn2023.0.3
Installing weak dependencies:
apr-util-openssl x86_64 1.6.3-1.amzn2023.0.1
mod_http2 x86_64 2.0.27-1.amzn2023.0.3
mod_lua x86_64 2.4.62-1.amzn2023

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

Total download size: 2.3 M
Installed size: 6.9 M
Downloading Packages:
(1/12): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm 266 kB/s | 17 kB 00:00
(2/12): apr-1.7.2-2.amzn2023.0.2.x86_64.rpm 1.7 MB/s | 129 kB 00:00
(3/12): generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch.rpm 1.0 MB/s | 19 kB 00:00
(4/12): httpd-2.4.62-1.amzn2023.x86_64.rpm 2.1 MB/s | 48 kB 00:00
(5/12): httpd-filesystem-2.4.62-1.amzn2023.noarch.rpm 719 kB/s | 14 kB 00:00
```

Start and enable httpd server using **systemctl start httpd && systemctl enable httpd**.

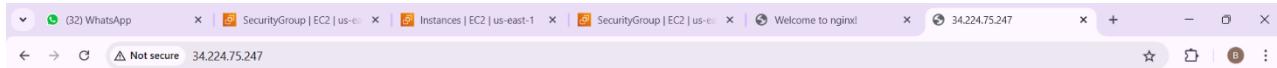
```
root@ip-172-31-22-170:~#
[6/12]: httpd-core-2.4.62-1.amzn2023.x86_64.rpm
[7/12]: apr-util-1.6.3-1.amzn2023.0.1.x86_64.rpm
[8/12]: httpd-tools-2.4.62-1.amzn2023.x86_64.rpm
[9/12]: mailcap-2.1.49-3.amzn2023.0.3.noarch.rpm
[10/12]: libbrotli-1.0.9-4.amzn2023.0.2.x86_64.rpm
[11/12]: mod http2-2.0.27-1.amzn2023.0.3.x86_64.rpm
[12/12]: mod_lua-2.4.62-1.amzn2023.x86_64.rpm

Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing :
Installing : apr-1.7.2-2.amzn2023.0.2.x86_64
Installing : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
Installing : apr-util-1.6.3-1.amzn2023.0.1.x86_64
Installing : httpd-filesystem-2.4.62-1.amzn2023.noarch
Installing : httpd-tools-2.4.62-1.amzn2023.x86_64
Installing : libbrotli-1.0.9-4.amzn2023.0.2.x86_64
Running scriptlet: httpd-filesystem-2.4.62-1.amzn2023.noarch
Installing : httpd-filesystem-2.4.62-1.amzn2023.x86_64
Installing : httpd-core-2.4.62-1.amzn2023.x86_64
Installing : mod http2-2.0.27-1.amzn2023.0.3.x86_64
Installing : mod_lua-2.4.62-1.amzn2023.x86_64
Installing : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
Installing : httpd-2.4.62-1.amzn2023.x86_64
Running scriptlet: httpd-2.4.62-1.amzn2023.x86_64
Verifying : apr-1.7.2-2.amzn2023.0.2.x86_64
Verifying : apr-util-1.6.3-1.amzn2023.0.1.x86_64
Verifying : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
Verifying : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
Verifying : httpd-2.4.62-1.amzn2023.x86_64
Verifying : httpd-core-2.4.62-1.amzn2023.x86_64
Verifying : httpd-tools-2.4.62-1.amzn2023.x86_64
Verifying : libbrotli-1.0.9-4.amzn2023.0.2.x86_64
Verifying : mailcap-2.1.49-3.amzn2023.0.3.noarch
Verifying : mod http2-2.0.27-1.amzn2023.0.3.x86_64
Verifying : mod_lua-2.4.62-1.amzn2023.x86_64

Installed:
  apr-1.7.2-2.amzn2023.0.2.x86_64      apr-util-1.6.3-1.amzn2023.0.1.x86_64      apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64      generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
  httpd-2.4.62-1.amzn2023.x86_64        httpd-core-2.4.62-1.amzn2023.x86_64        httpd-filesystem-2.4.62-1.amzn2023.noarch      httpd-tools-2.4.62-1.amzn2023.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64  mailcap-2.1.49-3.amzn2023.0.3.noarch      mod http2-2.0.27-1.amzn2023.0.3.x86_64      mod_lua-2.4.62-1.amzn2023.x86_64

Complete!
[root@ip-172-31-22-170 ~]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-22-170 ~]# systemctl start httpd
[root@ip-172-31-22-170 ~]#
```

Copy the public IP of your instance and paster it in our browser so that we can able to see like a below image if it is like this then the server is connected properly.



**It works!**

---

### Deploying one application through Apache (httpd):

Setting Up the Apache (HTTP) Server with "Antique Cafe" Template Connect to the Apache EC2 instance using SSH (ec2-user). Open Antique Cafe and click and copy the link and go to PuTTY and paste using wget <link> we will get one zip file that is **antique-cafe.zip** unzip the file using **unzip <filename>** after that we will get another file and go to that file and check through ll we will find some files move all files for the httpd path usually /var/www/html/ this is the default path for httpd. After that copy the public IP and paster it our browser so that we can able to see the application as shown in the below images.

```
wget: missing URL
Usage: wget [OPTION]... [URL]...
Try 'wget --help' for more options.
[root@ip-172-31-22-170 ~]# wget https://www.free-css.com/assets/files/free-css-templates/download/page295/antique-cafe.zip
--2024-10-29 11:25:16-- https://www.free-css.com/assets/files/free-css-templates/download/page295/antique-cafe.zip
Resolving www.free-css.com (www.free-css.com)... 217.160.0.242, 2001:8d8:100f:f000::20f
Connecting to www.free-css.com (www.free-css.com)|217.160.0.242|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2217176 (2.1M) [application/zip]
Saving to: 'antique-cafe.zip'

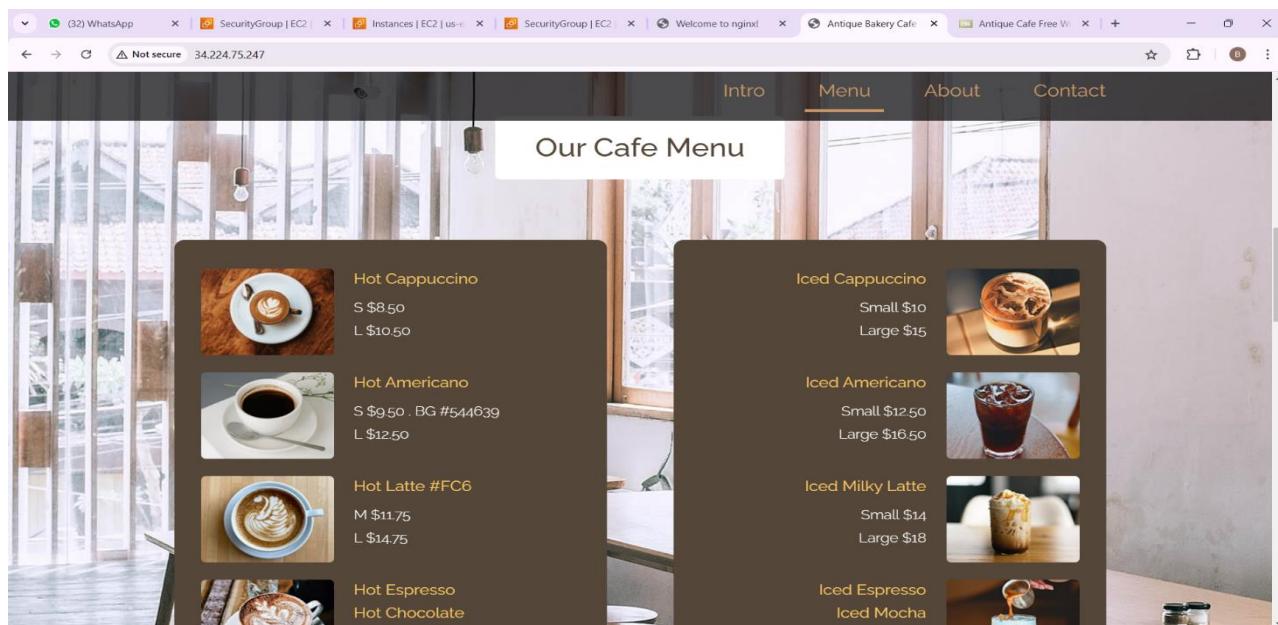
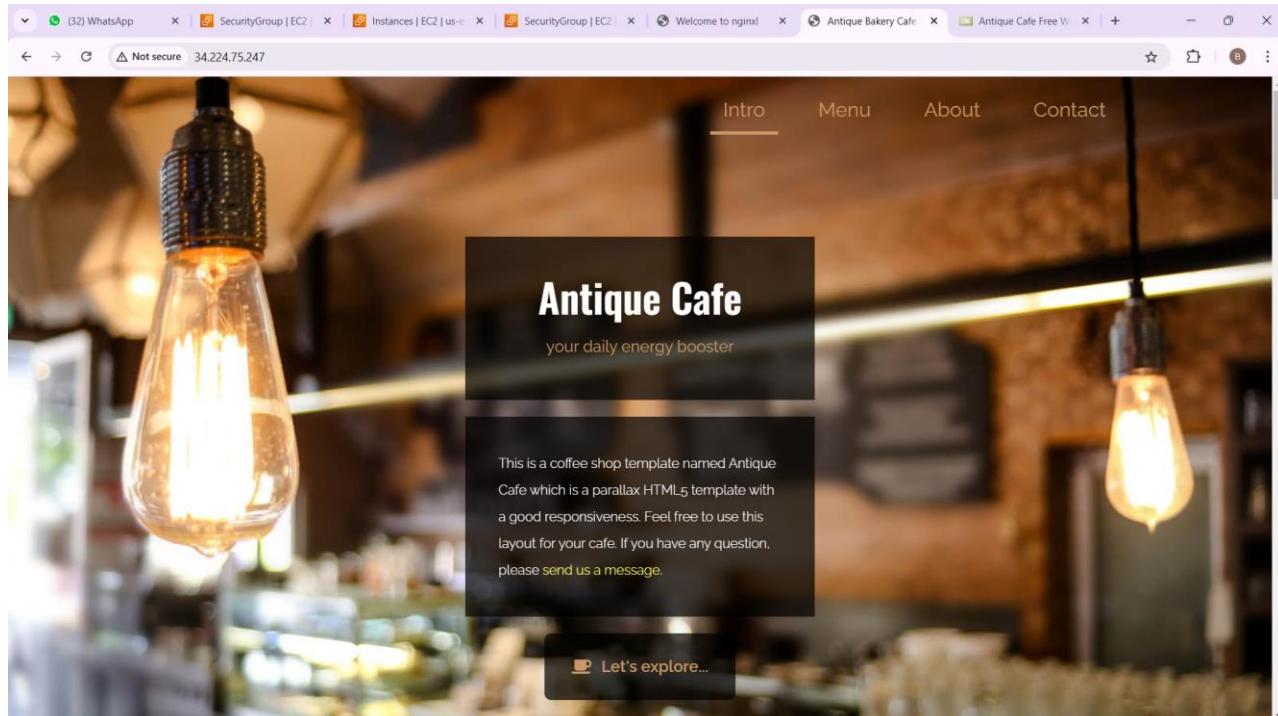
antique-cafe.zip          100%[=====]  2.11M  2.86MB/s    in 0.7s

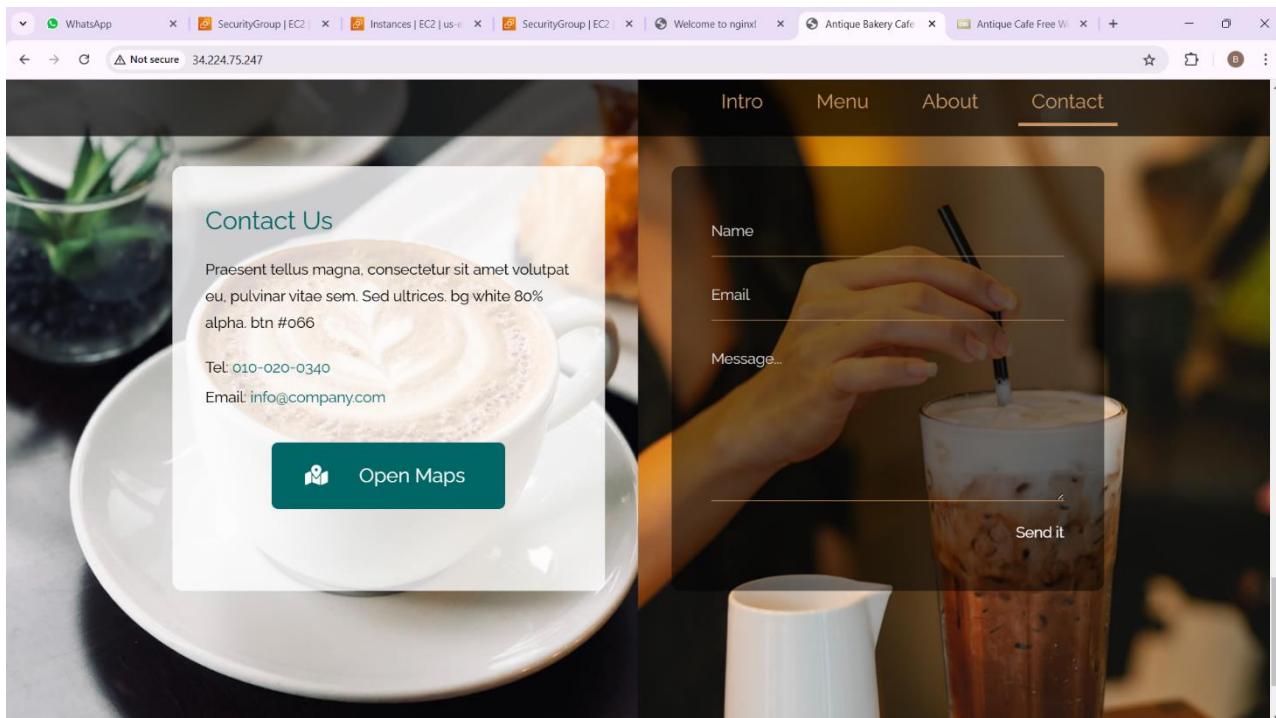
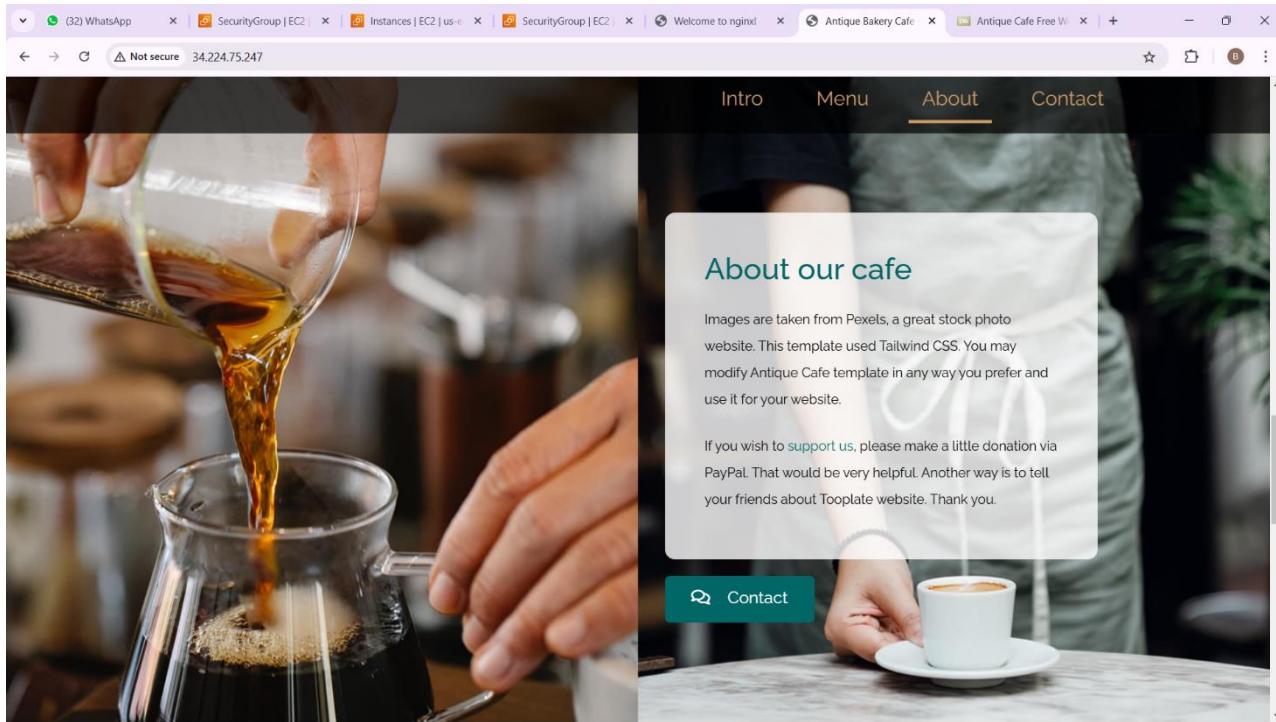
2024-10-29 11:25:17 (2.06 MB/s) - 'antique-cafe.zip' saved [2217176/2217176]

[root@ip-172-31-22-170 ~]# ll
total 2168
-rw-r--r-- 1 root root 2217176 Jan 26 2022 antique-cafe.zip
[root@ip-172-31-22-170 ~]# unzip antique-cafe.zip
Archive:  antique-cafe.zip
  creating: 2126_antique_cafe/
  creating: 2126_antique_cafe/css/
  inflating: 2126_antique_cafe/css/all.min.css
  inflating: 2126_antique_cafe/css/tailwind.css
  inflating: 2126_antique_cafe/css/tooplate-antique-cafe.css
  creating: 2126_antique_cafe/img/
  creating: 2126_antique_cafe/img/antique-cafe-128x128px.png
```

```
[root@ip-172-31-22-170 ~]# ll
total 2168
drwxr-xr-x. 6 root root    103 Sep 30  2021 2126_antique_cafe
-rw-r--r--. 1 root root 2217176 Jan 26  2022 antique-cafe.zip
[root@ip-172-31-22-170 ~]# cd 2126_antique_cafe/
[root@ip-172-31-22-170 2126_antique_cafe]# ll
total 52
-rw-r--r--. 1 root root   510 Jul 30  2019 'ABOUT THIS TEMPLATE.txt'
drwxr-xr-x. 2 root root    78 Sep 30  2021 css
drwxr-xr-x. 2 root root 16384 Sep 29  2021 img
-rw-r--r--. 1 root root 15522 Sep 30  2021 index.html
drwxr-xr-x. 2 root root   91 Sep 29  2021 js
drwxr-xr-x. 2 root root 16384 Sep 29  2021 webfonts
[root@ip-172-31-22-170 2126_antique_cafe]# mv * /var/www/html
[root@ip-172-31-22-170 2126_antique_cafe]# ll
total 0
[root@ip-172-31-22-170 2126_antique_cafe]#
```

This is my Antique Café application using httpd



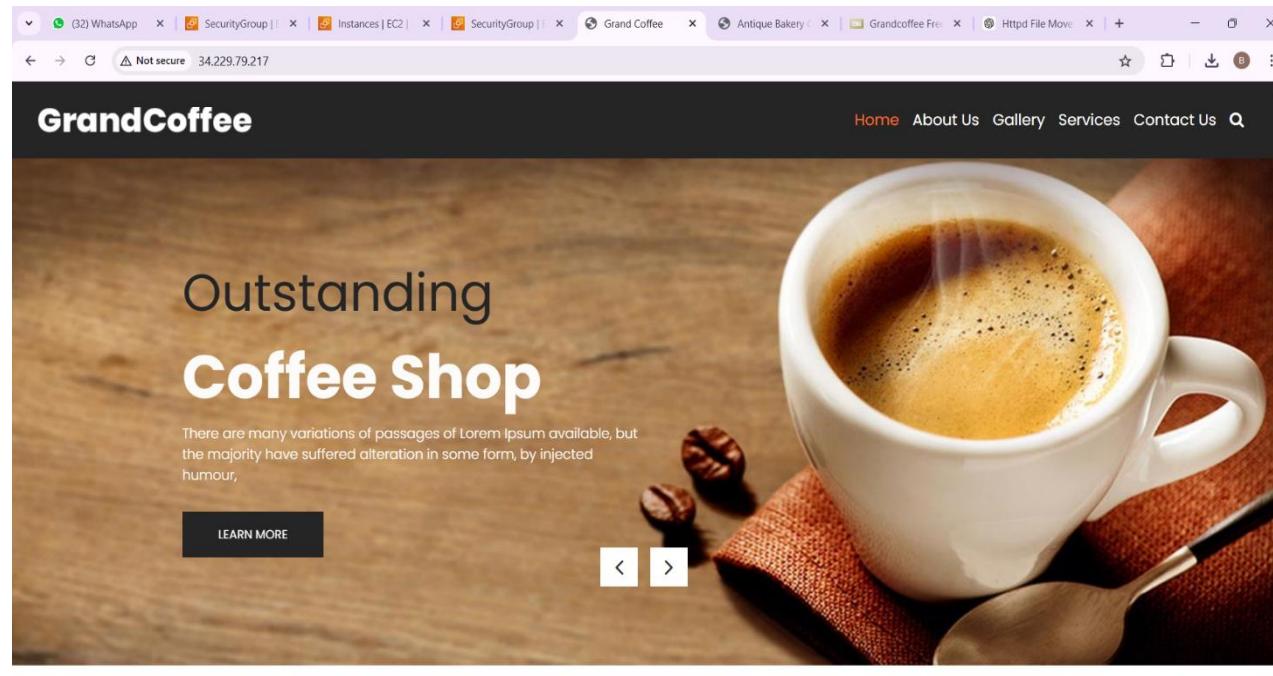


## Setting Up the Nginx Server with "Grand Coffee" Application.

Setting Up the nginx Server with "Grand Coffee" Template Connect to the nginx EC2 instance using SSH (ec2-user).

Open Grand Coffee and click and copy the link and go to PuTTY and paste using wget <link> we will get one zip file that is **grandcoffee.zip** unzip the file using **unzip <filename>** after that we will get another file and go to that file and check through ll we will find some files move all files for the nginx path usually /var/www/html/ or /usr/share/nginx/html this is the default path for nginx. After that copy the public IP and paster it our browser so that we can able to see the application as shown in the below images.

```
root@ip-172-31-19-221:~/html
inflating: html/index.html
creating: html/js/
inflating: html/js/bootstrap.js
inflating: html/js/bootstrap.bundle.js
inflating: html/js/bootstrap.bundle.js.map
inflating: html/js/bootstrap.bundle.min.js
inflating: html/js/bootstrap.bundle.min.js.map
inflating: html/js/bootstrap.js
inflating: html/js/bootstrap.js.map
inflating: html/js/bootstrap.min.js
inflating: html/js/bootstrap.min.js.map
inflating: html/js/custome.js
inflating: html/js/customScrollbar.concat.min.js
inflating: html/js/jquery.min.js
inflating: html/js/modernizr.js
inflating: html/js/jquery-3.0.0.min.js
inflating: html/js/modernizer.js
inflating: html/js/plugin.js
inflating: html/js/services.js
inflating: html/js/footer-setting.js
inflating: html/services.html
[root@ip-172-31-19-221 ~]# ll
total 22
-rw-r--r-- 1 root root 6749163 Aug 20 2021 grandcoffee.zip
drwxr-xr-x 5 root root 138 Oct 29 11:43 html
[root@ip-172-31-19-221 html]# ll
total 120
-rw-r--r-- 1 root root 7480 Mar 30 2020 about.html
-rw-r--r-- 1 root root 8417 Mar 30 2020 contact.html
drwxr-xr-x 2 root root 16384 Mar 30 2020 css
-rw-r--r-- 1 root root 11830 Mar 30 2020 gallery.html
drwxr-xr-x 2 root root 16384 Mar 30 2020 images
-rw-r--r-- 1 root root 24668 Mar 31 2020 index.html
drwxr-xr-x 2 root root 16384 Mar 30 2020 js
-rw-r--r-- 1 root root 9092 Mar 30 2020 services.html
[root@ip-172-31-19-221 html]# mv * /var/www/html
mv: cannot move '/var/www/html': Is not a directory
[root@ip-172-31-19-221 html]# mv * /var/www/html
[root@ip-172-31-19-221 html]# ll
total 120
-rw-r--r-- 1 root root 7480 Mar 30 2020 about.html
-rw-r--r-- 1 root root 8417 Mar 30 2020 contact.html
drwxr-xr-x 2 root root 16384 Mar 30 2020 css
-rw-r--r-- 1 root root 11830 Mar 30 2020 gallery.html
drwxr-xr-x 2 root root 16384 Mar 30 2020 images
-rw-r--r-- 1 root root 24668 Mar 31 2020 index.html
drwxr-xr-x 2 root root 16384 Mar 30 2020 js
-rw-r--r-- 1 root root 9092 Mar 30 2020 services.html
[root@ip-172-31-19-221 html]# mv * /usr/share/nginx/html
mv: cannot move '/usr/share/nginx/html/index.html': Is not a directory
[root@ip-172-31-19-221 html]#
```



Not secure 34.229.79.217

## ABOUT US

Full cleaning and housekeeping services for companies and households.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

[Read More](#)



## Our Gallery

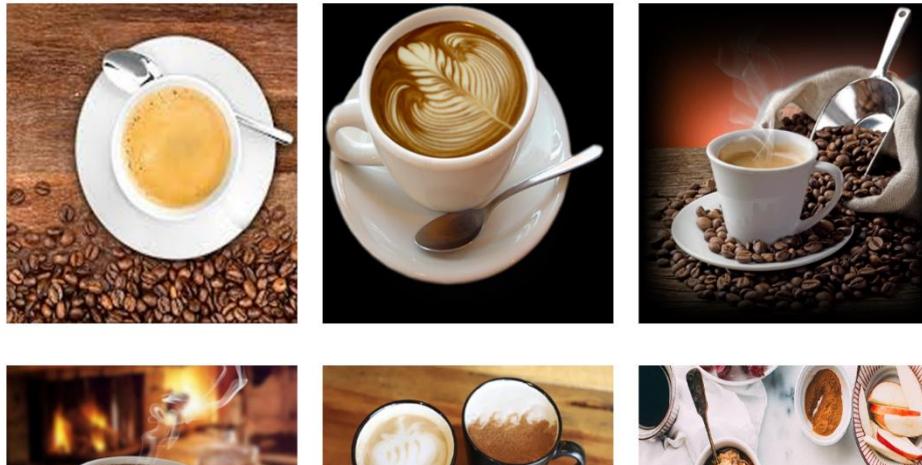
Not secure 34.229.79.217

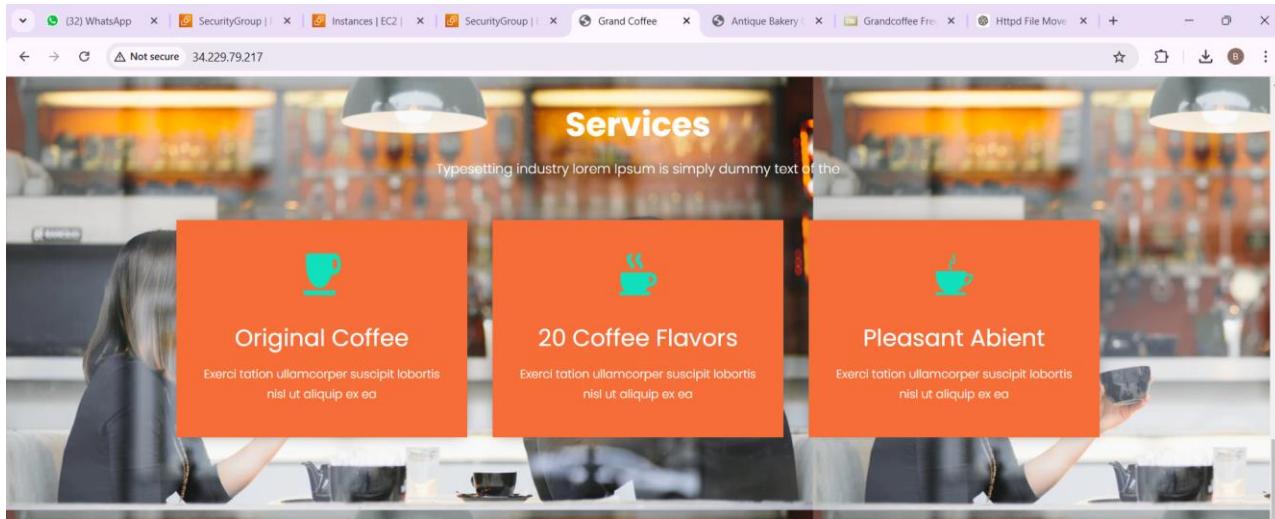
Lorem Ipsum is simply dummy text of printing typesetting systems. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

## Our Gallery

Not secure 34.229.79.217

Lorem Ipsum is simply dummy text of printing typesetting systems. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.





## Testimonial

Even slightly believable. If you are going to use a passage of Lorem ipsum, you need to

### ATTACHING LOAD BALACER TO SERVERS:

Go to EC2 Dashboard > Load Balancers and click Create Load Balancer. Choose Application Load Balancer or Classic Load Balancer .

Load balancer types		
<b>Application Load Balancer</b>	<b>Network Load Balancer</b>	<b>Gateway Load Balancer</b>
<p>Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.</p> <p><a href="#">Create</a></p>	<p>Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.</p> <p><a href="#">Create</a></p>	<p>Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.</p> <p><a href="#">Create</a></p>

Configure the load balancer give name for the load balancer Scheme we have to select **Internet-facing** IP Address Type is IPv4. Listeners Set the listener protocol to HTTP and port to 80.

The screenshot shows the 'Create Application Load Balancer' wizard on the AWS Management Console. In the 'Basic configuration' section, the 'Load balancer name' is set to 'MyWebLB'. The 'Scheme' dropdown is set to 'Internet-facing', which is described as routing requests from clients over the internet to targets. The 'Load balancer IP address type' dropdown is set to 'IPv4', which includes only IPv4 addresses. In the 'Network mapping' section, under 'VPC', a single subnet 'subnet-03d26f5ac0060abff' is selected. Under 'Availability Zones', 'us-east-1a (use1-az2)' and 'us-east-1b (use1-az4)' are checked, while others are unchecked. The bottom section, 'Security groups', is partially visible.

**Availability Zones** Select the availability zone where your EC2 instances are located and select one extra availability zone. If not it will through an error.

The screenshot continues the 'Create Application Load Balancer' wizard. In the 'Network mapping' section, under 'VPC', a single subnet 'subnet-03d26f5ac0060abff' is listed. Under 'Availability Zones', 'us-east-1a (use1-az2)' and 'us-east-1b (use1-az4)' are checked. In the 'Security groups' section, there is a note about selecting at least two security groups. The bottom section, 'CloudShell Feedback', is partially visible.

Configure Security Groups for the Load Balancer create a security group for the load balancer, allowing inbound HTTP traffic on port 80. If we have already then we can able to select it. Before creating Load balancer we need to create a target group if not we can able to create a target group directly. I am creating directly from configurations page after creating target group selecting target group.

The screenshot shows the AWS CloudFormation Create ALB Wizard - Step 1: Set up security groups. In the 'Listeners and routing' section, there is a new listener for port 80. The 'Default action' dropdown is set to 'Forward to' and has a placeholder 'Select a target group'. Below this, there is a 'Create target group' button. The 'Load balancer tags - optional' section is also visible.

## WE NEED TO SELECT THE TARGET GROUP

In the Registry targets we need to select our instances and we need to click on create target group which is shown in below images.

The screenshot shows the AWS CloudFormation Step 2: Create target group. In the 'Register targets' section, there is a table titled 'Available instances (2)' showing two instances: 'instance-2' and 'instance-1'. Both instances are marked as 'Running'. The 'Ports for the selected instances' field contains '80'. In the 'Targets' section, the registered targets are listed.

The screenshot shows the AWS EC2 Instances page. Two instances, "instance-2" and "instance-1", are selected. In the "Ports for the selected instances" section, ports 80 and 1-65535 are listed. Below this, a "Review targets" section displays two targets:

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID	Launch time
i-0eba5fea75ee41d1e	instance-2	80	Running	launch-wizard-16	us-east-1b	172.31.22.170	subnet-0c85ac199c0740b9f	October 29, 2024, 16:43 (UTC+05:30)
i-0e717b30d2b37db53	instance-1	80	Running	launch-wizard-15	us-east-1b	172.31.19.221	subnet-0c85ac199c0740b9f	October 29, 2024, 16:37 (UTC+05:30)

Successfully created Target group we can able to see in the below image.

The screenshot shows the AWS EC2 Target Groups page. A success message indicates the target group "MyWebApp-TG" was successfully created. The "Details" section shows the target type as "Instance", protocol as "HTTP: 80", and VPC as "vpc-014e267882d4c37b4". The "Targets" section shows 2 total targets, all healthy (0 healthy, 0 unhealthy, 0 anomalous, 0 unused, 0 initial, 0 draining). The "Registered targets" section lists 2 targets, both healthy, with an "Anomaly mitigation: Not applicable" status. The "Targets" tab is selected.

Click on target group go to targets and check our two instances are Healthy so again go to the configuration page.

The screenshot shows the AWS EC2 Target Groups page. On the left, there's a sidebar with various navigation options like EC2 Dashboard, Instances, Images, and Network & Security. The main area displays a table titled "Target groups (1/1) Info" with one entry: "MyWebApp-TG". Below this, a detailed view for "Target group: MyWebApp-TG" is shown, specifically the "Targets" tab. It lists two registered targets: "instance-2" and "instance-1", both marked as "Healthy".

### We need to assign a target group at the configuration of load balancer.

In the below image I have assigned a target group for load balancer. After that click on create Load balancer.

The screenshot shows the "Create ALB Wizard" step. In the "Listeners and routing" section, a new listener for "HTTP:80" is being configured. Under the "Forward to:" dropdown, "MyWebApp-TG" is selected as the target group. The "Protocol" is set to "HTTP" and the "Port" is "80". The "Default action" is set to "HTTP".

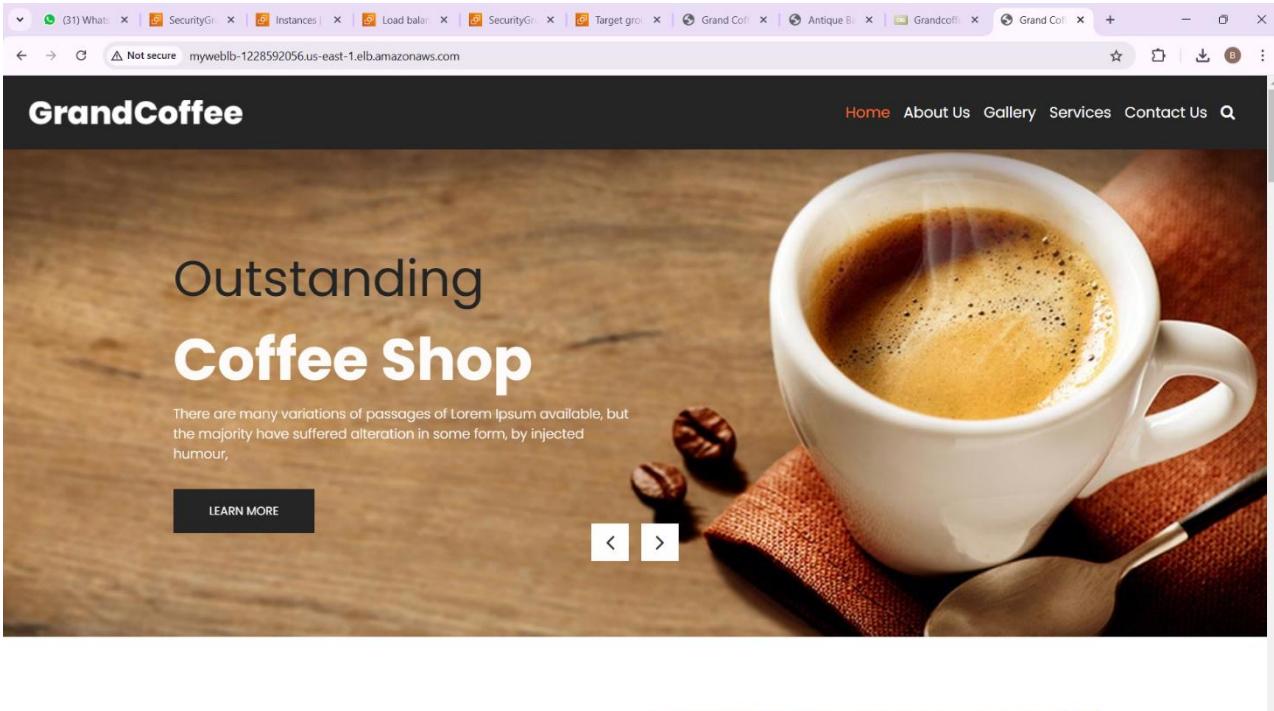
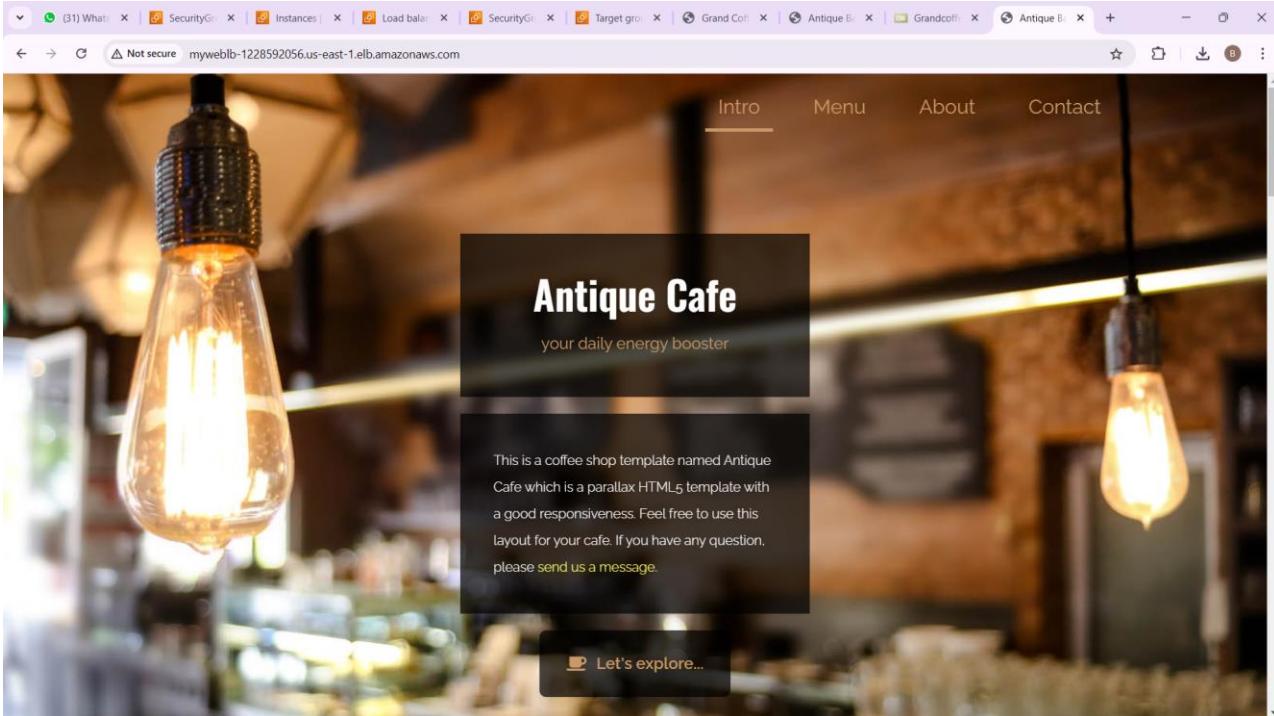
In the below image we can able to see our load balancer has been created Successfully.

The screenshot shows the AWS CloudWatch Metrics console with a green header bar indicating a successful metric creation. The main area displays a single metric named 'MyWebLB' with a value of 1. A detailed view of the metric's configuration is shown, including its namespace ('aws/lambda'), metric name ('MyWebLB'), unit ('None'), and dimensions ('FunctionName: MyWebLB'). The metric is described as tracking the number of requests processed by the Lambda function. The 'Metrics' tab is selected, showing the metric's value over time.

We need to copy DNS url and paste it in browser we can able to see Grand Coffee first and click on reload option provided at the top left corner.

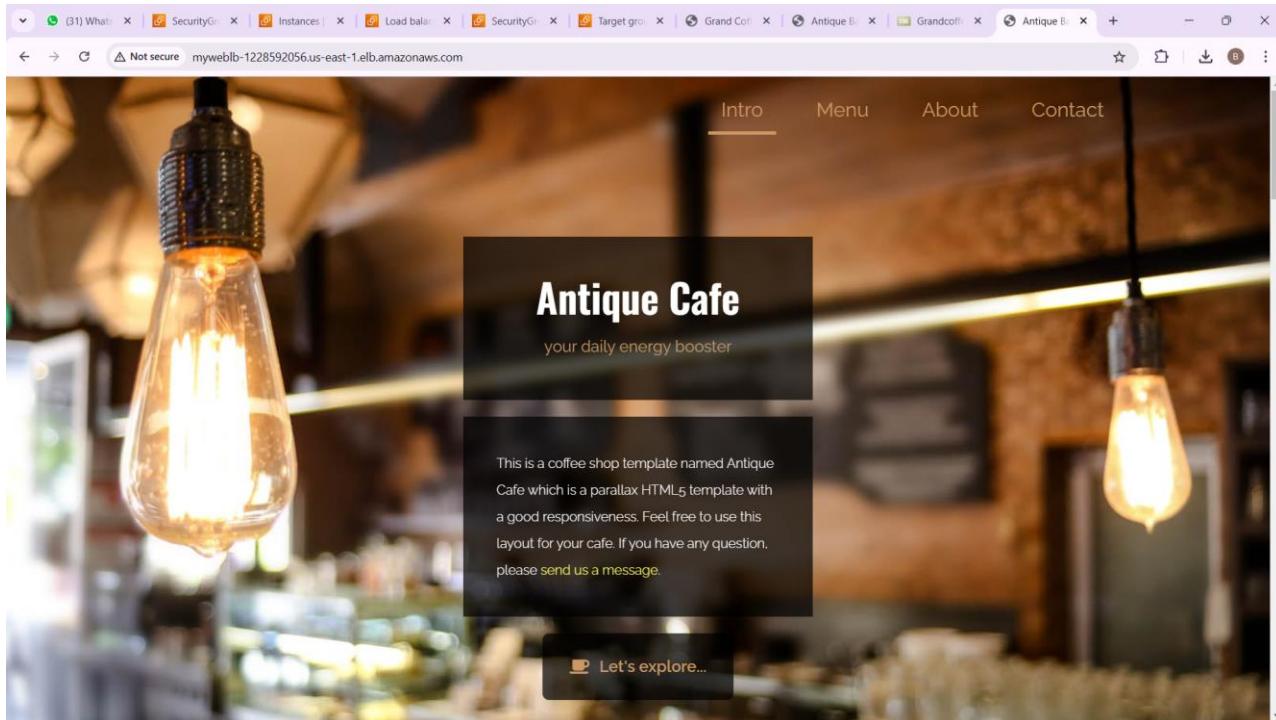
The screenshot shows a web browser displaying the 'GrandCoffee' website. The URL in the address bar is 'myweblb-1228592056.us-east-1.elb.amazonaws.com'. The page features a large image of a coffee cup with steam rising from it, accompanied by the text 'Outstanding Coffee Shop'. Below the image is a 'LEARN MORE' button and navigation arrows. The top navigation bar includes links for Home, About Us, Gallery, Services, and Contact Us.

After reloading we can able to see the Antique Cafe so that repeat it two or three types load is fluctuate and we can able to see two applications alternatively. From this we can able to see Load is distributing.



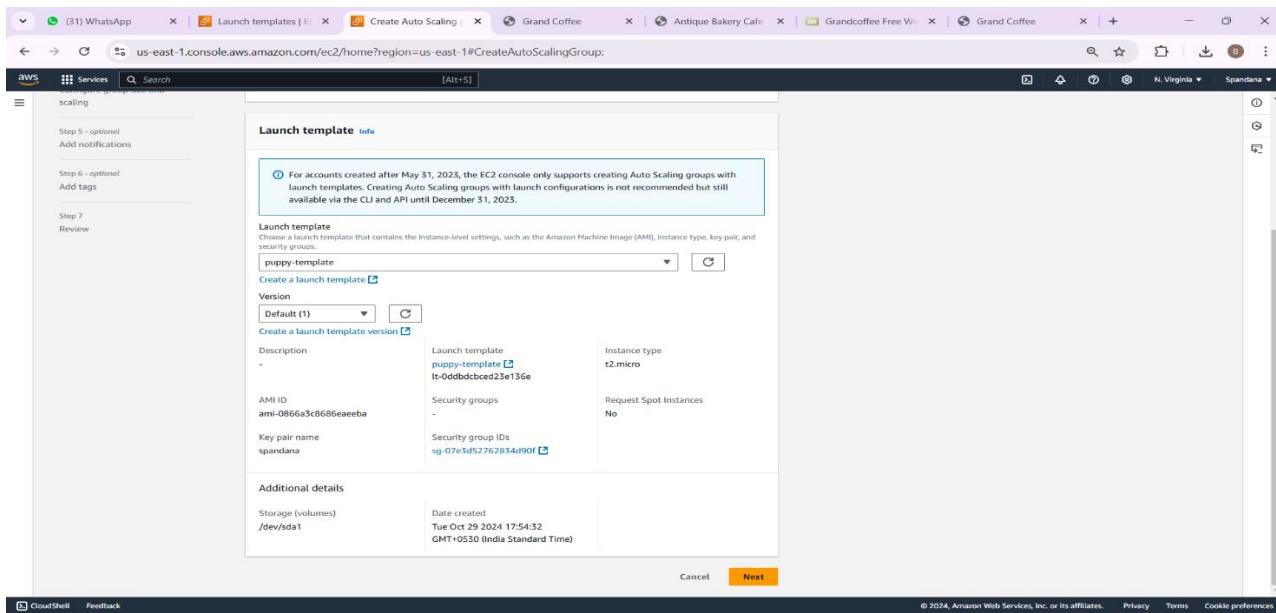
Screenshot of a web browser showing the "Antique Cafe" website. The page has a dark background with a central white content area. At the top, there is a navigation bar with links for "Intro", "Menu", "About", and "Contact". The "Intro" link is underlined, indicating it is the active page. Below the navigation, there is a header section with the title "Antique Cafe" and the subtitle "your daily energy booster". A text box contains the following text: "This is a coffee shop template named Antique Cafe which is a parallax HTML5 template with a good responsiveness. Feel free to use this layout for your cafe. If you have any question, please [send us a message](#)". At the bottom of the content area is a button labeled "Let's explore..." with a coffee cup icon.

Screenshot of a web browser showing the "GrandCoffee" website. The page has a dark background with a large image of a cup of coffee and coffee beans on the right side. On the left, there is a text overlay with the title "Outstanding Coffee Shop". Below the title, there is a short paragraph of placeholder text: "There are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration in some form, by injected humour," followed by a "LEARN MORE" button. At the top, there is a navigation bar with links for "Home", "About Us", "Gallery", "Services", "Contact Us", and a search icon. The "Home" link is highlighted in orange, indicating it is the active page. There are also "Stop loading this page" and "Not secure" icons in the browser's address bar.



## LAB 9 – AUTO SCALING GROUPS (ASG)

Create an Auto Scaling Group (ASG) go to Auto Scaling Groups in the EC2 Dashboard. Click Create Auto Scaling group. Select Launch template and choose the template you created earlier. Auto Scaling group name: Give it a descriptive name, like UbuntuAutoScalingGroup. VPC and Subnet Select the VPC and the subnets where you want the instances to launch.



Successfully template has been created.

The screenshot shows a browser window with multiple tabs open, including WhatsApp, Create template from, Create Auto Scaling, Grand Coffee, Antique Bakery Cafe, Grandcoffee Free W..., and Grand Coffee. The main content area is titled 'Create template from instance' and displays a green success message: 'Successfully created puppy-template(t-0ddbdcbcd23e136e)'. Below this, there's a 'Next Steps' section with links to 'Launch an instance', 'Launch instance from this template', 'Create an Auto Scaling group from your template', 'Create Auto Scaling group', and 'Create Spot Fleet'. At the bottom right is a yellow button labeled 'View launch templates'.

Configure ASG Size and Scaling Policies Set the Desired Capacity:

- Minimum capacity: 1
- Desired capacity: 2
- Maximum capacity: 3

The screenshot shows the 'Create Auto Scaling group' wizard at Step 3: 'Configure group size and scaling - optional'. It includes sections for 'Group size', 'Desired capacity', 'Scaling', and 'Scaling limits'. Under 'Group size', the 'Units (number of instances)' dropdown is set to '2'. Under 'Desired capacity', the input field shows '2'. Under 'Scaling', it says 'You can resize your Auto Scaling group manually or automatically to meet changes in demand.' Under 'Scaling limits', the 'Min desired capacity' is '1' and the 'Max desired capacity' is '3'. At the bottom, there's an 'Automatic scaling - optional' section with a note about target tracking scaling policies. The left sidebar lists steps 1 through 7.

Scaling policies for simplicity, you can use the default setting, which keeps instances at the desired capacity. If you want more control, you can set policies based on CPU utilization or other metrics.

**Automatic scaling - optional**

Choose whether to use a target tracking policy [Info](#)  
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

**No scaling policies**  
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

**Target tracking scaling policy**  
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

**Instance maintenance policy [Info](#)**

Control your Auto Scaling group's availability during instance replacement events. This includes health checks, instance refreshes, maximum instance lifetime features and events that happen automatically to keep your group balanced, called rebalancing events.

Choose a replacement behavior depending on your availability requirements

- No policy**  
For rebalancing events, new instances will launch before terminating others. For health events, instances terminate and launch at the same time.
- Prioritize availability**  
Launch before terminating others  
Launch new instances and wait for them to be ready before terminating others. This allows you to go above your desired capacity by a given percentage and may temporarily increase costs.
- Control costs**  
Launch and launch instances at the same time. This allows you to go below your desired capacity by a given percentage and may temporarily reduce availability.
- Flexible**  
Custom behavior  
Set custom values for the minimum and maximum amount of available capacity. This provides greater flexibility in setting how far below and over your desired capacity your Auto Scaling goes when replacing instances.

**Instance scale-in protection**

Scale-in protection prevents newly launched instances from being terminated by scaling activities. Make sure to remove scale-in protection for the group or individual instances when instances are ready to be terminated.

Enable instance scale-in protection

[Cancel](#) [Skip to review](#) [Previous](#) [Next](#)

**Step 1**  
[Choose launch template](#)

**Step 2**  
[Choose instance launch options](#)

**Step 3 - optional**  
[Configure advanced options](#)

**Step 4 - optional**  
[Configure group size and scaling](#)

**Step 5 - optional**  
[Add notifications](#)

**Step 6 - optional**  
[Add tags](#)

**Step 7**  
[Review](#)

**Add notifications - optional** [Info](#)

Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.

[Add notification](#)

[Cancel](#) [Skip to review](#) [Previous](#) [Next](#)

Review and create Auto Scaling Group below we can see that Auto Scaling Group has been created.

The screenshot shows the AWS EC2 Auto Scaling Groups page. At the top, there is a search bar and a navigation bar with tabs for Launch configurations, Launch templates, Actions, and Create Auto Scaling group. Below the search bar, there is a table with columns for Name, Launch template/configuration, Instances, Status, Desired capacity, Min, Max, and Availability Zones. One row is visible for 'spandana-ASG' with a status of 'Updating capacity...'. At the bottom of the table, it says '0 Auto Scaling groups selected'.

The screenshot shows the AWS EC2 Auto Scaling Groups page with the 'spandana-ASG' group selected. The table at the top now has a checked checkbox next to the 'Name' column. Below the table, a detailed view of the 'spandana-ASG' group is shown. The 'Group details' section includes fields for Auto Scaling group name (spandana-ASG), Desired capacity (2), Desired capacity type (Units (number of instances)), Date created (Tue Oct 29 2024 18:00:56 GMT+0530 (India Standard Time)), Minimum capacity (1), Status (Updating capacity...), Maximum capacity (3), and Amazon Resource Name (ARN) (arn:aws:autoscaling:us-east-1:637423550105:autoScalingGroup:5c128e99-8806-44e6-bba0-2e0085beb4bf:autoScalingGroupName/spandana-ASG). There are tabs for Details, Activity, Automatic scaling, Instance management, Monitoring, and Instance refresh.

Verify Auto Scaling Behaviour after the ASG is created WS will automatically launch instances based on the desired capacity.

The screenshot shows the AWS EC2 Instances page with the following details:

- Instances (1/5) Info:**
  - Instance ID: i-03d579fb403e28ce2 (Lab-9)
  - Instance state: Running
  - Instance type: t2.micro
  - Public IPv4 DNS: ec2-3-94-64-191.compute-1.amazonaws.com
  - Private IPv4 address: 172.31.26.137
  - Public IPv4 address: 3.94.64.191
  - Subnet ID: vpc-014e267882d4c37b4
- Actions:** Launch instances

Delete existing instances and monitor.

The screenshot shows the AWS EC2 Instances page with the following details:

- Instances (3/6) Info:**
  - Instance ID: i-0d77cde0b55d36517, i-04a0b0a934d31742a, i-03d579fb403e28ce2
  - Instance state: Terminated
  - Instance type: t2.micro
  - Public IPv4 DNS: ec2-3-94-64-191.compute-1.amazonaws.com
  - Private IPv4 address: 172.31.26.137
  - Public IPv4 address: 3.94.64.191
  - Subnet ID: vpc-014e267882d4c37b4
- Notifications:** 3 instances selected
- Monitoring:**
  - Alarms: CPU utilization (%), Network in (bytes), Network out (bytes), Network packets in (count), Network packets out (count), CPU credit usage (count), CPU credit balance (count)

## Auto Scaling Group start launching additional instances to reach the capacity.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with links like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancing, CloudShell, and Feedback. The main area displays a table of instances with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, Public IPv4, and Elastic IP. Two instances are selected: Lab-9 (running) and Lab-9 (initializing). Below the table, a section titled '2 instances selected' shows monitoring metrics for CPU utilization, Network in (bytes), Network out (bytes), Network packets in (count), Network packets out (count), CPU credit usage (count), and CPU credit balance (count). A 'Monitoring' tab is selected. At the top right, there are buttons for 'Launch instances', 'Actions', and 'Launch CloudWatch agent'.

## LAB-10 RDS

Amazon RDS is a managed database service supporting multiple engines (e.g., MySQL, PostgreSQL). It automates backups, patching, and scaling, while offering high availability with Multi-AZ deployment and robust security features.

Firstly we have to create one EC2 instance.

The screenshot shows the AWS EC2 Instances page. The left sidebar is identical to the previous one. The main area shows a table with one instance named 'RDS'. The instance details page for 'i-099260ca339c8af5d' (RDS) is shown below. It includes tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. Under Details, it shows the instance summary info: Instance ID (i-099260ca339c8af5d), Public IPv4 address (100.29.16.99), Instance state (Running), Instance type (t2.micro), Status check (Initializing), Availability Zone (us-east-1c), Public IPv4 DNS (ec2-100-29-16-99.com...), Public IPv4 (100.29.16.99), and Subnet ID. It also lists Private IP addresses (172.31.32.217), Private IPv4 DNS (ec2-100-29-16-99.compute-1.amazonaws.com), and Elastic IP addresses. A note at the bottom says 'AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations.' The page footer includes links for CloudShell, Feedback, and the AWS logo.

In the console search for RDS service and click on Databases and click on create databases.

The screenshot shows the AWS RDS Management console with the 'Databases' tab selected. A modal window titled 'Consider creating a Blue/Green Deployment to minimize downtime during upgrades' is open, providing information and links to User Guide and Aurora User Guide. The main 'Databases (0)' table has no instances found. The left sidebar includes sections for Dashboard, Databases (selected), Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, Event subscriptions, Recommendations (6), and Certificate update. The bottom navigation bar includes CloudShell, Feedback, and links to AWS Privacy, Terms, and Cookie preferences.

We have to give configurations for databases choose data creation method and choose Database Engine as MYSQL .

The screenshot shows the 'Create database' page. Under 'Choose a database creation method', the 'Easy create' option is selected, described as using recommended best-practice configurations. Under 'Configuration', the 'MySQL' engine type is chosen, highlighted with a blue border. Other options shown include Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MariaDB, PostgreSQL, Oracle, and Microsoft SQL Server. On the right side, a detailed description of MySQL is provided, stating it's the most popular open source database and listing its features: supports database size up to 64 TiB, General Purpose, Memory Optimized, and Burstable Performance instance classes, automated backup and point-in-time recovery, up to 15 Read Replicas per instance, and 5 read replicas cross-region. The bottom navigation bar includes CloudShell, Feedback, and links to AWS Privacy, Terms, and Cookie preferences.

Select DB instance specifications and provide DB instance identifier optional and Provide Master username and Master password for security purpose.

The screenshot shows the AWS RDS MySQL creation wizard. On the left, under 'DB instance size', three options are listed: Production (db.r6g.xlarge, 4 vCPUs, 32 GB RAM, 500 GB, 1.017 USD/hour), Dev/Test (db.r6g.large, 2 vCPUs, 16 GB RAM, 100 GB, 0.231 USD/hour), and Free tier (db.t4g.micro, 2 vCPUs, 1 GB RAM, 20 GB, 0.019 USD/hour). The 'Free tier' option is selected. Under 'DB instance identifier', the value 'puppyDB' is entered. In the 'Master username' section, 'admin' is specified. The 'Master password' field contains a strong password ('\*\*\*\*\*'). The 'Password strength' bar is labeled 'Very strong'. To the right, a sidebar titled 'MySQL' provides a brief overview of the database and lists several features: supports database sizes up to 64 TiB, general purpose, memory optimized, and burstable performance instance classes, automated backup and point-in-time recovery, and up to 15 read replicas per instance. At the bottom, there are links for CloudShell, Feedback, and a footer with copyright information.

Choose EC2 instance and computer resource as Connect to an EC2 Compute resource after that click on create database.

The screenshot shows the continuation of the AWS RDS MySQL creation wizard. In the 'Set up EC2 connection - optional' section, the 'Auto generate password' checkbox is checked. Below it, the 'Master password' field contains a password ('\*\*\*\*\*') and the 'Password strength' bar is 'Very strong'. In the 'Compute resource' section, the 'Connect to an EC2 compute resource' checkbox is selected. A note states that connecting to a compute resource will automatically change connectivity settings. The 'EC2 instance' dropdown shows 'i-099260ca339c8af5d' and 'RDS'. A note at the bottom of this section explains that VPC settings cannot be changed when a compute resource is added. The right sidebar and footer are identical to the previous screenshot.

Database has been creating in the below image check the status. The status of the below image is creating.

The screenshot shows the AWS RDS Management console with the URL [us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#databases](https://us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#databases). The left sidebar is titled "Amazon RDS" and includes sections for Dashboard, Databases (selected), Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, Event subscriptions, Recommendations (6), and Certificate update. The main content area is titled "Creating database puppydb" with the sub-instruction "Your database might take a few minutes to launch. You can use settings from puppydb to simplify configuration of suggested database add-ons while we finish creating your DB for you." Below this is a "Databases (1)" table with one row for "puppydb". The table columns are DB identifier, Status, Role, Engine, Region ..., Size, Recommendations, CPU, Current..., and Maintenance. The "Status" column shows "Creating". Other details include "Instance MySQL Co...", "Region us-east-1c", "Size db.t4g.micro", "Recommendations -", "CPU none", and "Current...". A "Create database" button is visible at the top right of the table. The bottom of the screen shows standard AWS navigation links: CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

The status of the below image is Backing-up.

This screenshot is identical to the previous one, showing the AWS RDS Management console with the same URL and sidebar. The main difference is the database status: the "Status" column for the "puppydb" row now shows "Backing-up". The progress bar indicates "23.18%". All other details (Instance, Engine, Region, Size, Recommendations, CPU, and Maintenance) remain the same as in the first screenshot. The bottom of the screen shows standard AWS navigation links: CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

The status of the below image is Modifying.

This screenshot shows the AWS RDS Management console. The left sidebar is titled 'Amazon RDS' and includes links for Dashboard, Databases, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, Event subscriptions, Recommendations, and Certificate update. The main content area is titled 'Databases (1)' and shows a table with one row. The row details a database named 'puppydb' with the following information: Status: Modifying, Role: Instance, Engine: MySQL Co..., Region: us-east-1c, and Size: db.t4g.micro. A progress bar indicates 3.86% completion. At the top of the main area, there is a message: 'Consider creating a Blue/Green Deployment to minimize downtime during upgrades. You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases.' Below the message are buttons for Group resources, Modify, Actions, Restore from S3, and Create database. The bottom of the screen shows standard AWS navigation links for CloudShell, Feedback, and copyright information.

At last the database status is Available.

This screenshot shows the same AWS RDS Management console interface as the previous one, but the database status has changed. The 'puppydb' entry in the table now shows 'Status: Available'. All other details (Role: Instance, Engine: MySQL Co..., Region: us-east-1c, Size: db.t4g.micro) remain the same. The progress bar at the bottom of the table row is now at 100%. The rest of the interface, including the sidebar, message bar, and footer, is identical to the first screenshot.

Click in the database it will open the below page and copy the Endpoint & port and paste it in PuTTY or terminal using command.

The screenshot shows the AWS RDS Database Details page for a database named 'puppydb'. The top navigation bar includes tabs for Instances, Databases, and Database Details. The left sidebar has sections for Dashboard, Databases, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, Event subscriptions, Recommendations, and Certificate update. The main content area displays the 'Summary' of the database, showing details like DB identifier (puppydb), Status (Available), Role (Instance), Engine (MySQL Community), and Region & AZ (us-east-1c). Below the summary, there are tabs for Connectivity & security, Monitoring, Logs & events, Configuration, Zero-ETL integrations, Maintenance & backups, Tags, and Recommendations. The 'Connectivity & security' tab is selected, showing information about the endpoint and port (puppydb.ctsumykgq23d.us-east-1.rds.amazonaws.com, Port 3306), Networking (Availability Zone us-east-1c, VPC vpc-014e267882d4c37b4, Subnet group rds-ec2-db-subnet-group-1, Subnets subnet-0a365131ab2ae9b2b, subnet-08240e4830a10fbcd, subnet-01e662ee48c69c1, subnet-025ae671df6ff8a6f, subnet-0a6a0af64985f9c32), and Security (VPC security groups rds-ec2-1 (sg-0f42c24debafsc997) Active, Publicly accessible No, Certificate authority rds-ca-rsa2048-g1, Certificate authority date May 26, 2061, 05:04 (UTC+05:30), DB instance certificate expiration date October 30, 2025, 18:37 (UTC+05:30)). A green callout box highlights the 'Endpoint copied' message next to the endpoint URL.

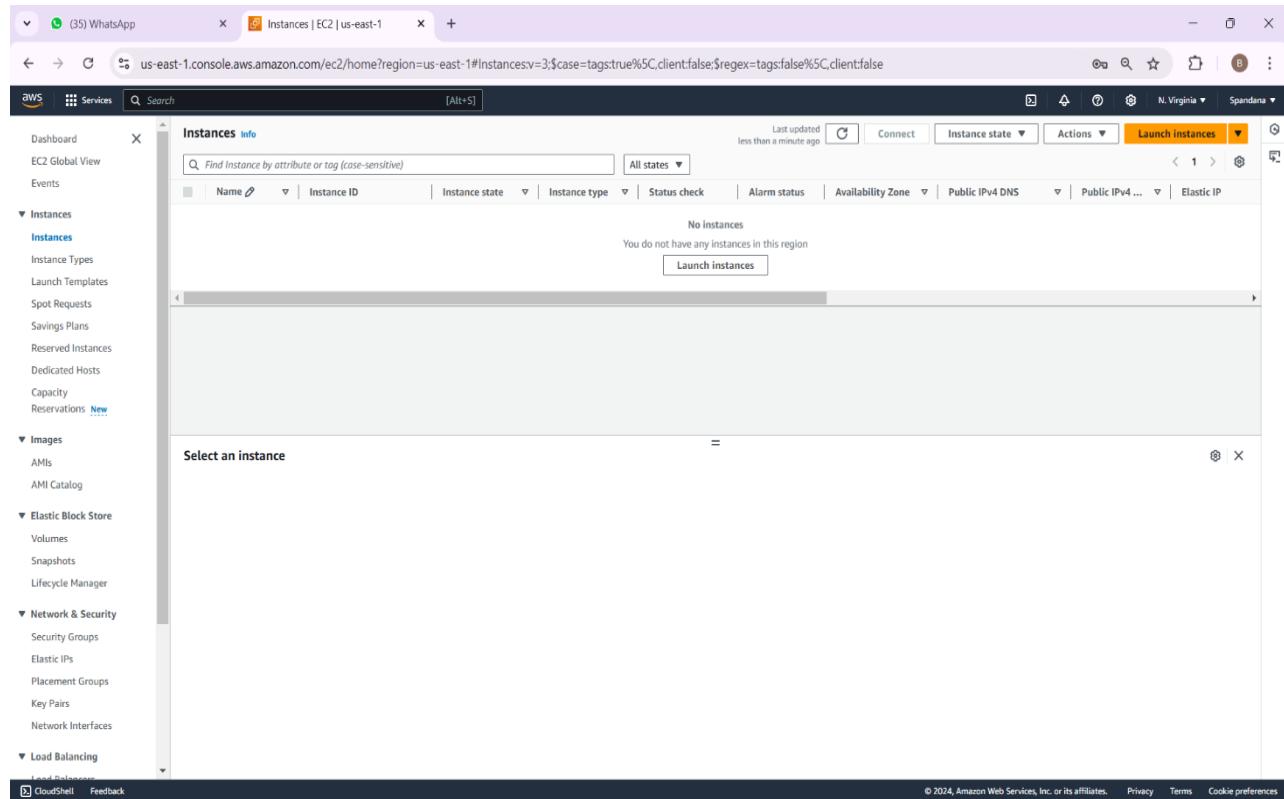
In Terminal install MySQL and using mysql -h <RDS-Endpoint> -u admin -P and provide password so we can able to access this RDS database from EC2 instance.

## MINI PROJECT – 2

### GIT-VERSION CONTROL SYSTEM(VCS) BY USING AMAZON WEB SERVICES

#### LAB-1 CREATING INSTANCE

Go to AWS Management console and navigate to EC2 in the AWS console search for EC2 and select it click on launch instance and give instance name select IAM based on preference choose instance type t2.micro key pair security group allow SSH access from IP address once configured click on launch instance.



Copy public IP of our instance and open PuTTY and paste public IP and go to SSH > Auth > Connection and browse for .ppk and connect through PuTTY and login as ec2-user because I have taken amazon machine image as Amazon Linux 2. We can able to see this process in below image.

The screenshot shows the AWS Management Console with the following details:

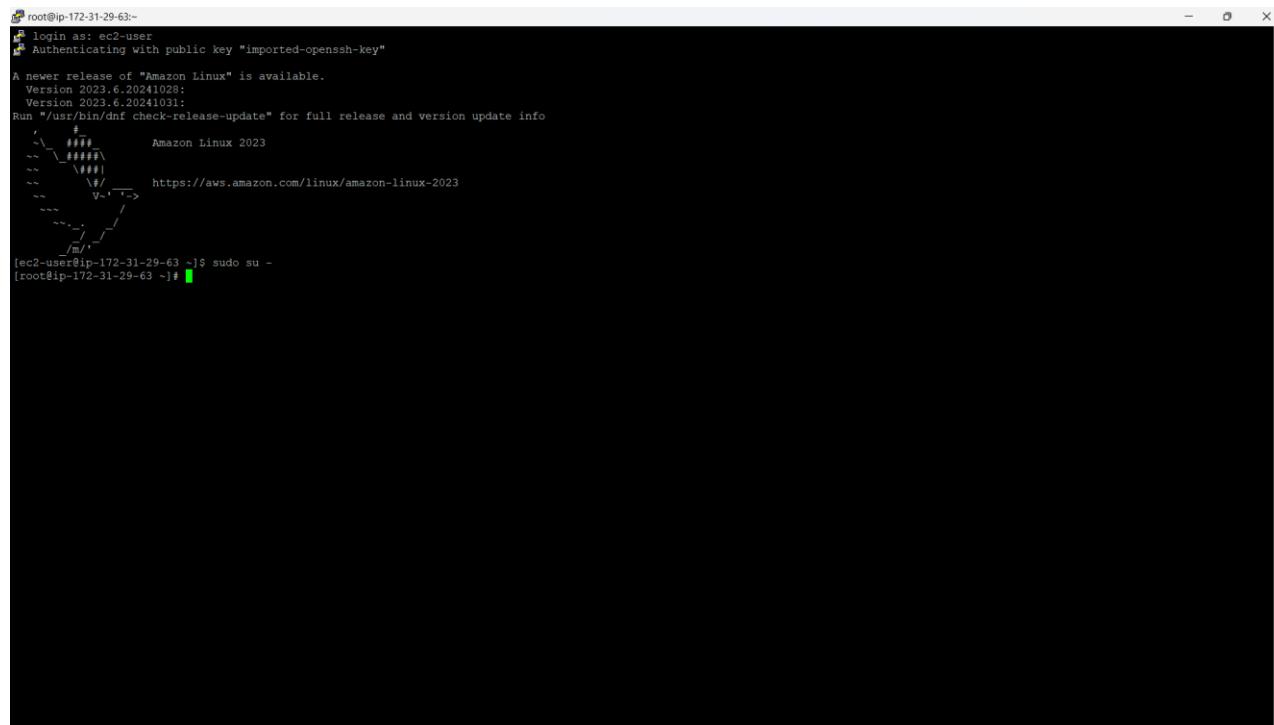
- Instances (1/1) Info**: Shows an instance named "miniproject-2" (Instance ID: i-0361f70856bb0a59c) in the "Running" state.
- Putty Configuration Window**: Overlays the main interface. It shows the host name as "107.20.105.72" and the port as "22". The connection type is set to "SSH".
- Instance Summary**: Provides detailed information about the instance, including its IP address (107.20.105.72), VPC ID (vpc-014e267882d4c37b4), and Subnet ID.
- Bottom Bar**: Includes links for CloudShell, Feedback, and navigation icons.

## Connection through PuTTY.

The screenshot shows the AWS Management Console with the following details:

- PuTTY Security Alert Dialog**: A modal window titled "PuTTY Security Alert" displays a warning message about an untrusted host key fingerprint for the server at 107.20.105.72 (port 22). It provides options to "Accept", "Connect Once", or "Cancel".
- Instances (1/1) Info**: Shows the same instance "miniproject-2" (i-0361f70856bb0a59c) in the "Running" state.
- Bottom Bar**: Includes links for CloudShell, Feedback, and navigation icons.

This is the terminal open when we connect through PuTTY.

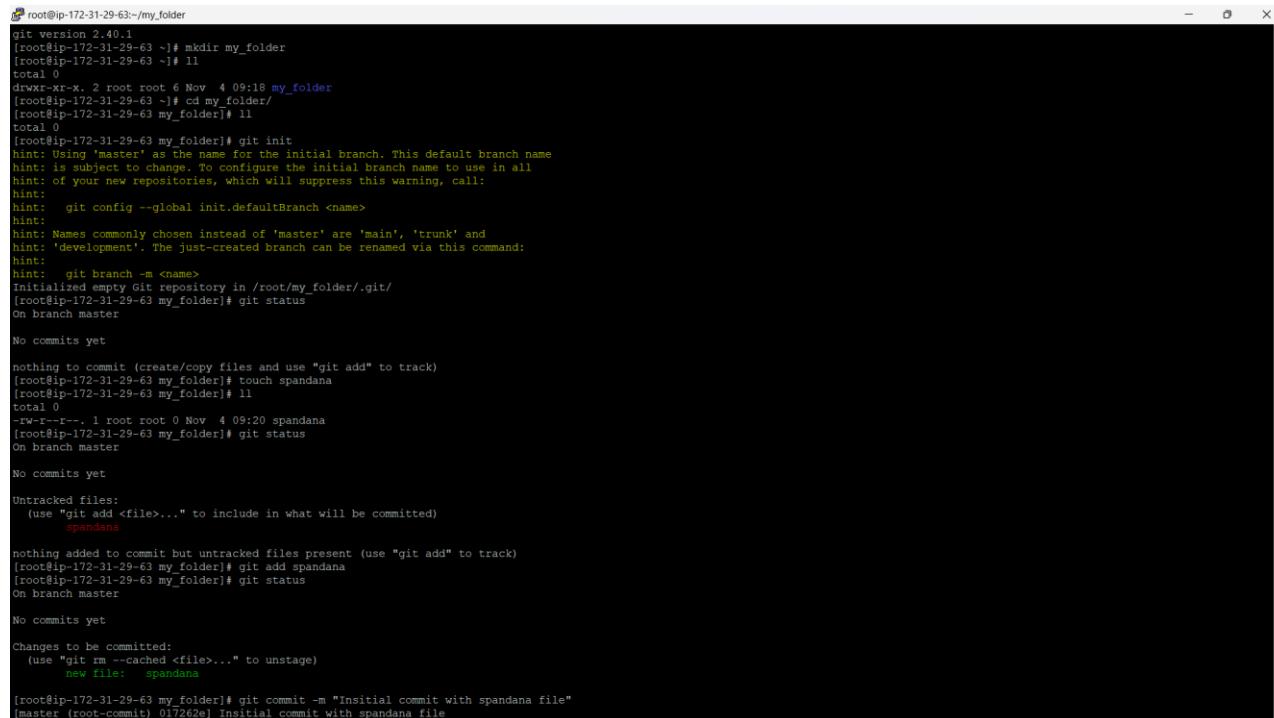


```
root@ip-172-31-29-63:~#
login as: ec2-user
Authenticating with public key "imported-openssh-key"

A newer release of "Amazon Linux" is available.
 Version 2023.6.20241028;
 Version 2023.6.20241031;
Run "/usr/bin/dnf check-release-update" for full release and version update info
      #          Amazon Linux 2023
      / \        https://aws.amazon.com/linux/amazon-linux-2023
     / \  V-->
     \ /  m
      \/
[ec2-user@ip-172-31-29-63 ~]$ sudo su -
[ec2-user@ip-172-31-29-63 ~]#
```

## LAB-2 CREATING REPO IN LOCAL MACHINE

First install git in our local machine using yum install git -y and create one folder on local machine and navigate to that folder inside the folder initialize it using **git init**. This will create a hidden .git folder and set up the directory run the **git status** command to check the status.



```
root@ip-172-31-29-63:~/my_folder
git version 2.40.1
[root@ip-172-31-29-63 ~]# mkdir my_folder
[root@ip-172-31-29-63 ~]# ll
total 0
drwxr-xr-x. 2 root root 6 Nov  4 09:18 my_folder
[root@ip-172-31-29-63 my_folder]# cd my_folder/
[root@ip-172-31-29-63 my_folder]# ll
total 0
[root@ip-172-31-29-63 my_folder]# git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint: git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint: git branch -m <name>
Initialized empty Git repository in /root/my_folder/.git/
[root@ip-172-31-29-63 my_folder]# git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
[root@ip-172-31-29-63 my_folder]# touch spandana
[root@ip-172-31-29-63 my_folder]# ll
total 0
-rw-r--r--. 1 root root 0 Nov  4 09:20 spandana
[root@ip-172-31-29-63 my_folder]# git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    spandana

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-29-63 my_folder]# git add spandana
[root@ip-172-31-29-63 my_folder]# git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   spandana

[root@ip-172-31-29-63 my_folder]# git commit -m "Initial commit with spandana file"
[master (root-commit) 017262e] Initial commit with spandana file
```

Create one empty file using touch command **touch <filename>** run git status and see we will see that file as a untracked file. Now we need to stage the file for that we need to **git add <filename>** running this command the file will be moved to the staging area and start tracking. Again check the **git status** now the file will be appear in green including it's staged and ready to be committed. Now commit the changes using **git commit -m "message"** to save the changes to local repo and run **git status** it will show working tree is clean. Check **git log** to check the commits have been done or not.

```
nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-29-63 my_folder]# git add spandana
[root@ip-172-31-29-63 my_folder]# git status
On branch master
No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   spandana

[root@ip-172-31-29-63 my_folder]# git commit -m "Initial commit with spandana file"
[master (root-commit) 017262e] Initial commit with spandana file
  Committer: root <root@ip-172-31-29-63.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

  git config --global --edit

After doing this, you may fix the identity used for this commit with:

  git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 spandana
[root@ip-172-31-29-63 my_folder]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-29-63 my_folder]# git log
commit 017262e059ea92b678006ce390b0724cac2dac (HEAD -> master)
Author: root <root@ip-172-31-29-63.ec2.internal>
Date:  Mon Nov 4 09:22:08 2024 +0000

  Initial commit with spandana file
[root@ip-172-31-29-63 my_folder]#
```

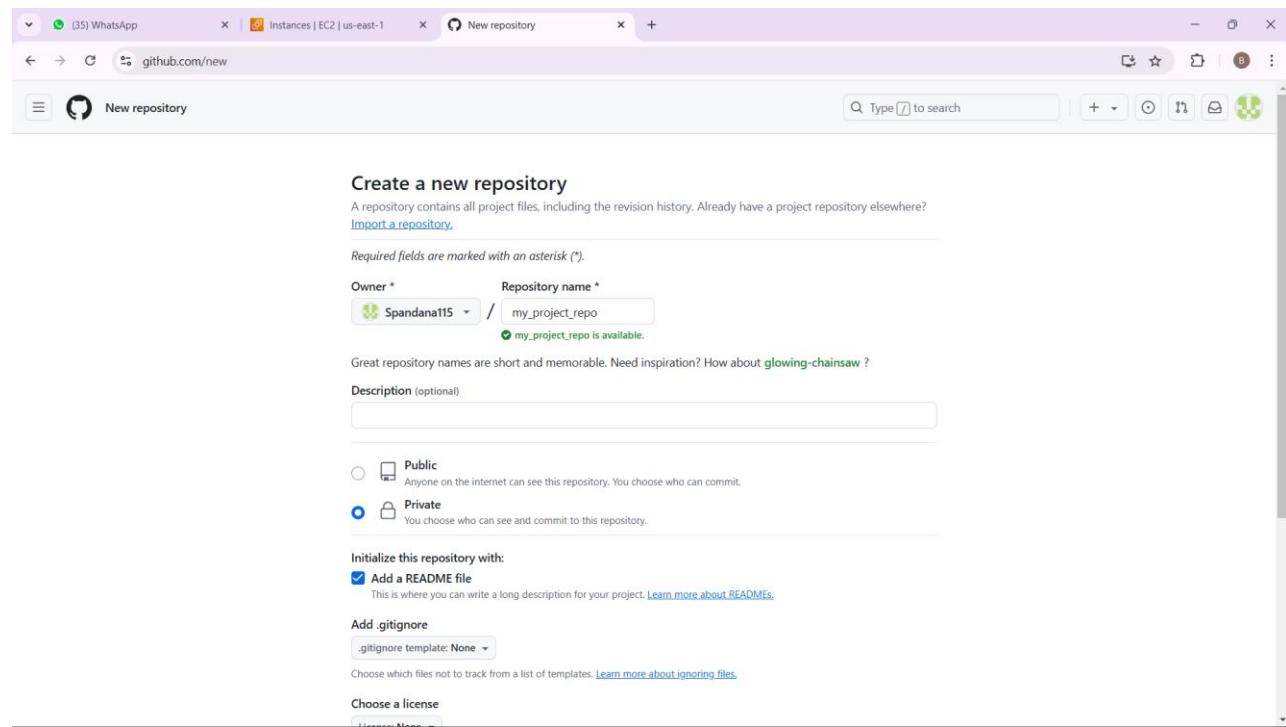
## LAB-3 CREATING REPO IN REMOTE LOCATION- GITHUB

Open GitHub account we can able to see the below page after opening GitHub in the upper left corner we have a new click on that and create new repository.

The screenshot shows the GitHub Home page with the following elements:

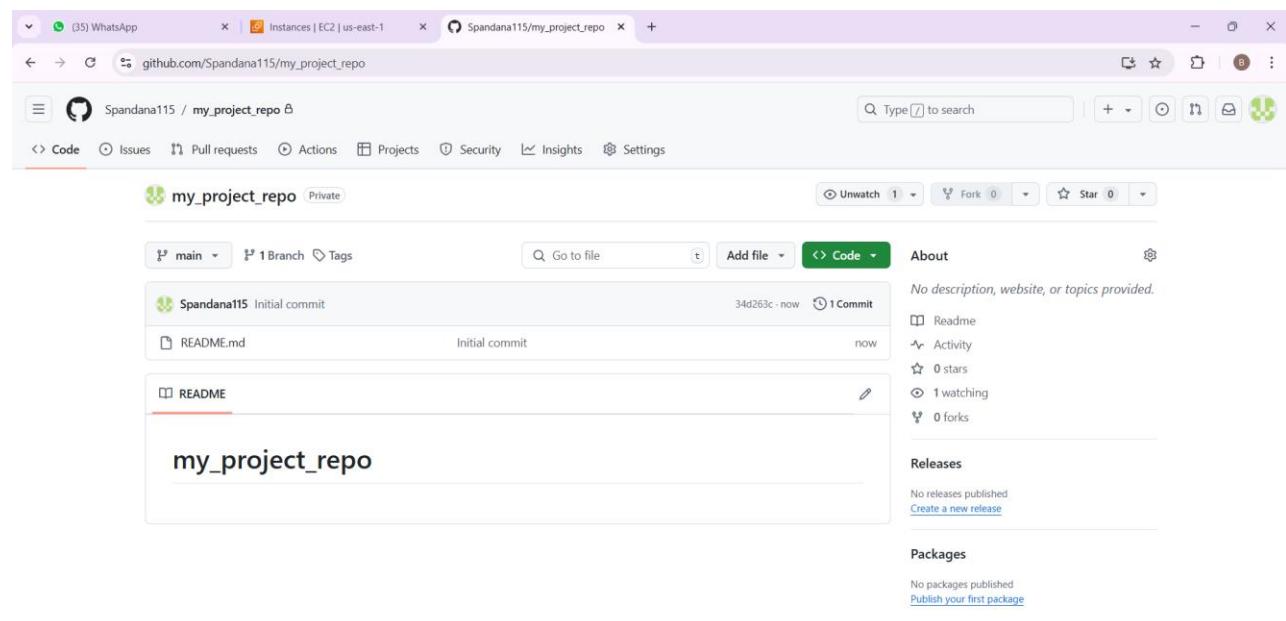
- Top repositories:** A sidebar listing repositories such as Spandana115/Devops\_Project, Spandana115/Web\_Cafe, Spandana115/my-html-site, Spandana115/docker-project, Spandana115/nodejs-project-docker, Spandana115/deployment-jar-app, and Spandana115/New-War-Application.
- Start writing code:** A search bar labeled "Start writing code".
- Start a new repository for Spandana115:** A card with the text "A repository contains all of your project's files, revision history, and collaborator discussion." and a "Create" button.
- Repository name \***: A text input field with placeholder "name your new repository..." and a "Create" button.
- Visibility:** Options for "Public" (anyone can see) and "Private" (choose who can see).
- Create a new repository**: A green button.
- Introduce yourself with a profile README:** A card with the text "Share information about yourself by creating a profile README, which appears at the top of your profile page." and a sample README content.
- Latest changes:** A sidebar showing recent activity:
  - 3 days ago: Claude 3.5 Sonnet is now available to all Copilot users in public preview
  - 4 days ago: SAST vulnerabilities summary now available on the security overview dashboard
  - 4 days ago: Actions Performance Metrics in public preview
  - 4 days ago: GitHub Copilot Metrics API GA release now available
- Explore repositories:** A sidebar showing popular repositories:
  - cgeo / cgeo
  - Rdatatable / data.table
  - inertiajs / inertia

Setup our new repository enter a name for repository in the repository field name choose whether the repository should be public or private I have chosen private repository. Initialize it with README.md this file is a place to describe our project and easy to get started and initializing.



The screenshot shows the GitHub interface for creating a new repository. The title bar says "New repository". The main section is titled "Create a new repository". It asks for the "Repository name" which is "my\_project\_repo". The "Owner" dropdown is set to "Spandana115". Below the repository name, there's a note that "my\_project\_repo is available". There's a link to "Import a repository". The "Description (optional)" field is empty. Under "Visibility", the "Private" option is selected. The "Initialize this repository with:" section has "Add a README file" checked. A note says "This is where you can write a long description for your project. Learn more about READMEs.". The "Add .gitignore" section shows ".gitignore template: None". A note says "Choose which files not to track from a list of templates. Learn more about ignoring files.". The "Choose a license" section has "Unset" selected. At the bottom right, there's a "Create repository" button.

At the last we have Create repository button so click on that button then new repository has been created it will directly taken to the repository main page.



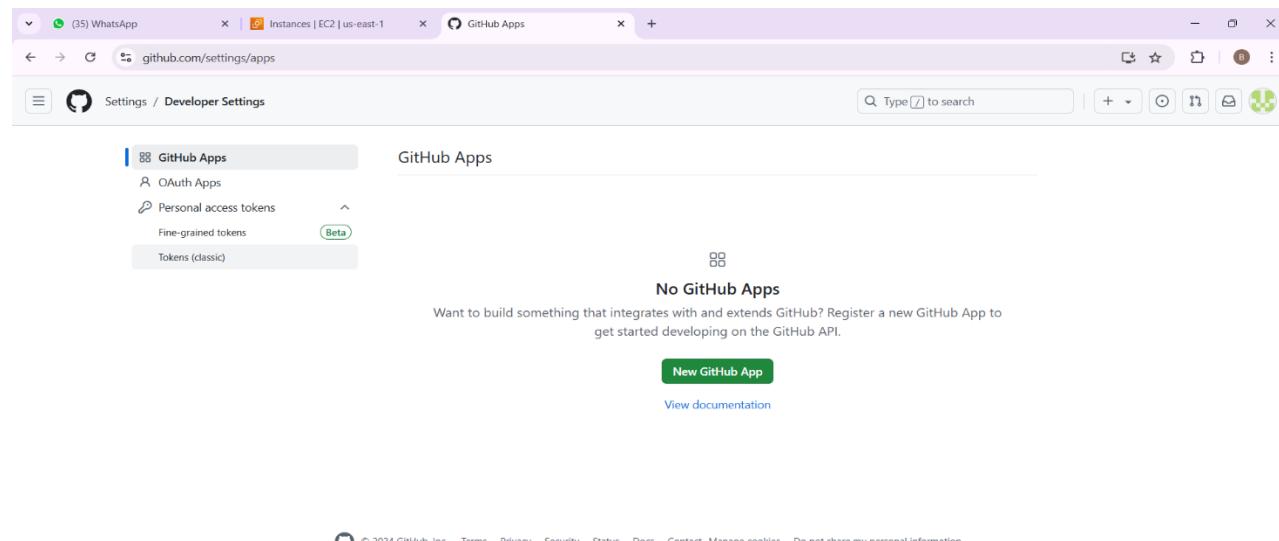
The screenshot shows the GitHub repository main page for "my\_project\_repo". The title bar says "Spandana115 / my\_project\_repo". The top navigation bar includes "Code", "Issues", "Pull requests", "Actions", "Projects", "Security", "Insights", and "Settings". The repository name "my\_project\_repo" is shown with a "Private" label. The "Code" tab is selected, showing the "main" branch, 1 branch, and 0 tags. A commit by "Spandana115" is listed: "Initial commit" at "34d263c · now" with "1 Commit". A file named "README.md" is shown with the content "Initial commit" at "now". The "About" section notes "No description, website, or topics provided." It lists "Readme", "Activity", "0 stars", "1 watching", and "0 forks". The "Releases" section says "No releases published" and "Create a new release". The "Packages" section says "No packages published" and "Publish your first package".

## LAB-4 WORKING WITH REMOTE REPOSITORY

Go to repository page on GitHub and click on code button and copy the repository url use HTTPS. Open the terminal of our local machine run the command **git clone <repo url>** it will ask for username and password in the place of password it will throw an error.so we need to create personal access token for that.

```
root@ip-172-31-29-63:~/my_folder
No commits yet
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    spandana
nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-29-63 my_folder]# git add spandana
[root@ip-172-31-29-63 my_folder]# git status
On branch master
No commits yet
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:  spandana
[root@ip-172-31-29-63 my_folder]# git commit -m "Initial commit with spandana file"
[master (root-commit) 017262e] Initial commit with spandana file
  Committer: root <root@ip-172-31-29-63.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:
  git config --global --edit
After doing this, you may fix the identity used for this commit with:
  git commit --amend --reset-author
  1 file changed, 0 insertions(+), 0 deletions(-)
  create mode 100644 spandana
[root@ip-172-31-29-63 my_folder]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-29-63 my_folder]# git log
commit 017262e5059ea92b67806ce390b0724cac2dac (HEAD -> master)
Author: root <root@ip-172-31-29-63.ec2.internal>
Date:  Mon Nov 4 09:22:08 2024 +0000
  Initial commit with spandana file
[root@ip-172-31-29-63 my_folder]# git clone https://github.com/Spandanall15/my_project_repo.git
Cloning into 'my_project_repo'...
Username for 'https://github.com': spandanall15
Password for 'https://spandanall15@github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls for information on currently recommended modes of authentication.
fatal: Authentication failed for 'https://github.com/Spandanall15/my_project_repo.git'
[root@ip-172-31-29-63 my_folder]#
```

To create personal access token first we need to click on profile and go to settings in that it we have to select Developer settings after that click on personal access token (classic) as shown below image.



Click on generate new token provide a Note to label token we have to provide expiration date scope select repo check box for full access and then click on generate token.

The screenshot shows the GitHub 'New personal access token (classic)' creation interface. On the left, a sidebar lists 'GitHub Apps', 'OAuth Apps', and 'Personal access tokens' (selected), with sub-options 'Fine-grained tokens' and 'Tokens (classic)'. A 'Beta' badge is visible next to 'Personal access tokens'. The main area is titled 'New personal access token (classic)' and contains a note about personal access tokens. A 'Note' input field contains 'mini\_project\_token'. Below it, a 'What's this token for?' input field is empty. The 'Expiration' dropdown is set to '30 days', with a note that the token will expire on Wed, Dec 4 2024. The 'Select scopes' section allows selecting from various GitHub API scopes. The 'repo' scope is selected (indicated by a checked checkbox). Other available scopes include 'workflow', 'write:packages', 'delete:packages', and 'admin:org'. Each scope has a description and a checkbox. The 'repo' scope includes sub-scopes: 'repository\_status', 'repository\_deployment\_status', 'public\_repo', 'repo\_invite', and 'security\_events'. The 'repo' scope is described as providing 'Full control of private repositories'.

Now the token has been created successfully copy that token immediately and paste it in one place because we won't be able to view it again later.

The screenshot shows the GitHub 'Personal access tokens (classic)' list page. The sidebar is identical to the previous screen. The main area displays a message: 'Some of the scopes you've selected are included in other scopes. Only the minimum set of necessary scopes has been saved.' Below this, a 'Personal access tokens (classic)' heading and a 'Generate new token' button are shown. A note at the top of the token list says 'Make sure to copy your personal access token now. You won't be able to see it again!'. Three tokens are listed: 'Task — repo' (last used Oct 20 2024, delete), 'ansible — repo' (last used Oct 12 2024, delete), and 'Project\_token — repo' (last used Aug 30 2024, delete). At the bottom, a note states: 'Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.'

Now we again we need to clone it asks for username and password at the username we have to provide our GitHub account username and at the place of password we have to provide a token that we are generated previously.

```
[root@ip-172-31-29-63:~/my_folder]# git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:  spandana

[root@ip-172-31-29-63 my_folder]# git commit -m "Initial commit with spandana file"
[master (root-commit) 017262e] Initial commit with spandana file
  Committer: root <root@ip-172-31-29-63.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file!

  git config --global --edit

After doing this, you may fix the identity used for this commit with:

  git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 spandana
[root@ip-172-31-29-63 my_folder]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-29-63 my_folder]# git log
commit 017262e059ea92b67806ce390b0724ac2dac (HEAD -> master)
Author: root <root@ip-172-31-29-63.ec2.internal>
Date:  Mon Nov 4 09:22:08 2024 +0000

  Initial commit with spandana file
[root@ip-172-31-29-63 my_folder]# git clone https://github.com/Spandanall15/my_project_repo.git
Cloning into 'my_project_repo'...
Username for 'https://github.com': spandanall15
Password for 'https://spandanall15@github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls for information on currently recommended modes of authentication.
fatal: Authentication failed for 'https://github.com/Spandanall15/my_project_repo.git'
[root@ip-172-31-29-63 my_folder]# git clone https://github.com/Spandanall15/my_project_repo.git
Cloning into 'my_project_repo'...
Username for 'https://github.com': Spandanall15
Password for 'https://Spandanall15@github.com':
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
[root@ip-172-31-29-63 my_folder]#
```

Go to new cloned repository folder use **touch** command and create some empty files. Add the new files to the staging area using **git add <file names>** after staging commit the changes using **git commit -m “message”**

```
[root@ip-172-31-29-63:~/my_folder/my_project_repo]# git clone https://github.com/Spandanall15/my_project_repo.git
Cloning into 'my_project_repo'...
Username for 'https://github.com': spandanall15
Password for 'https://spandanall15@github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls for information on currently recommended modes of authentication.
fatal: Authentication failed for 'https://github.com/Spandanall15/my_project_repo.git'
[root@ip-172-31-29-63 my_folder]# git clone https://github.com/Spandanall15/my_project_repo.git
Cloning into 'my_project_repo'...
Username for 'https://github.com': Spandanall15
Password for 'https://Spandanall15@github.com':
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
[root@ip-172-31-29-63 my_folder]# ll
total 4
drwxr-xr-x. 3 root root 35 Nov 4 09:37 my_project_repo
-rw-r--r--. 1 root root 0 Nov 4 09:20 spandana
[root@ip-172-31-29-63 my_folder]# cd my_project_repo/
[root@ip-172-31-29-63 my_project_repo]# ll
total 4
-rw-r--r--. 1 root root 17 Nov 4 09:37 README.md
[root@ip-172-31-29-63 my_project_repo]# touch puppy file1 file2
[root@ip-172-31-29-63 my_project_repo]# ll
total 4
-rw-r--r--. 1 root root 17 Nov 4 09:37 README.md
-rw-r--r--. 1 root root 0 Nov 4 09:39 file1
-rw-r--r--. 1 root root 0 Nov 4 09:39 file2
-rw-r--r--. 1 root root 0 Nov 4 09:39 puppy
[root@ip-172-31-29-63 my_project_repo]# git add puppy file1 file2
[root@ip-172-31-29-63 my_project_repo]# git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:  file1
    new file:  file2
    new file:  puppy

[root@ip-172-31-29-63 my_project_repo]# git commit -m "committing multiple files"
[master 35ef30e] committing multiple files
  Committer: root <root@ip-172-31-29-63.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:
```

After committing changes we need to push changes to the Remote Repository using **git push** while pushing it asks for username and password so provide username and token in the place of password and push. The new files should be uploaded in remote GitHub repository.

```
root@ip-172-31-29-63:~/my.folder/my.project.repo
(use "git restore --staged <file>..." to unstage)
  new file:  file1
  new file:  file2
  new file:  puppy

[root@ip-172-31-29-63 my.project.repo]# git commit -m "committing multiple files"
[main 35ef30e] committing multiple files
 Committer: root <root@ip-172-31-29-63.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

  git config --global --edit

After doing this, you may fix the identity used for this commit with:

  git commit --amend --reset-author

3 files changed, 0 insertions(+), 0 deletions(-)
Create mode 100644 file1
Create mode 100644 file2
Create mode 100644 puppy
[root@ip-172-31-29-63 my.project.repo]# git log
commit 35ef30e75652a38ed35ec8c1474bbbaef14215a1 (HEAD -> main)
Author: root <root@ip-172-31-29-63.ec2.internal>
Date:  Mon Nov 4 09:40:46 2024 +0000

  committing multiple files

commit 34d263c25c3530f24998065489cdf900124df061 (origin/main, origin/HEAD)
Author: Spandana115 <baindaspandana0115@gmail.com>
Date:  Mon Nov 4 14:59:03 2024 +0530

  Initial commit
[root@ip-172-31-29-63 my.project.repo]# git push
Username for 'https://github.com': spandana115
Password for 'https://spandana115@github.com':
error: Invalid username or password.
fatal: Authentication failed for 'https://github.com/Spandana115/my_project_repo.git'
[root@ip-172-31-29-63 my.project.repo]# git push
Username for 'https://github.com': Spandana115
Password for 'https://Spandana115@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 296 bytes | 296.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Spandana115/my_project_repo.git
  34d263c..35ef30e main -> main
[root@ip-172-31-29-63 my.project.repo]#
```

Go to GitHub account and open our newly created repository and check the repository the new files have been pushed to remote server we will find a new changes.

The screenshot shows a GitHub repository named 'my\_project\_repo'. The repository has 1 branch (main) and 2 commits. The first commit was made 4 minutes ago by 'root' with the message 'committing multiple files'. The second commit was made 15 minutes ago by 'Spandana115' with the message 'Initial commit'. The repository has 0 stars, 1 watching, and 0 forks. It also shows a README file with the text 'my\_project\_repo'.

## LAB-5 PUSHING A LOCALLY CREATED REPO TO GITHUB

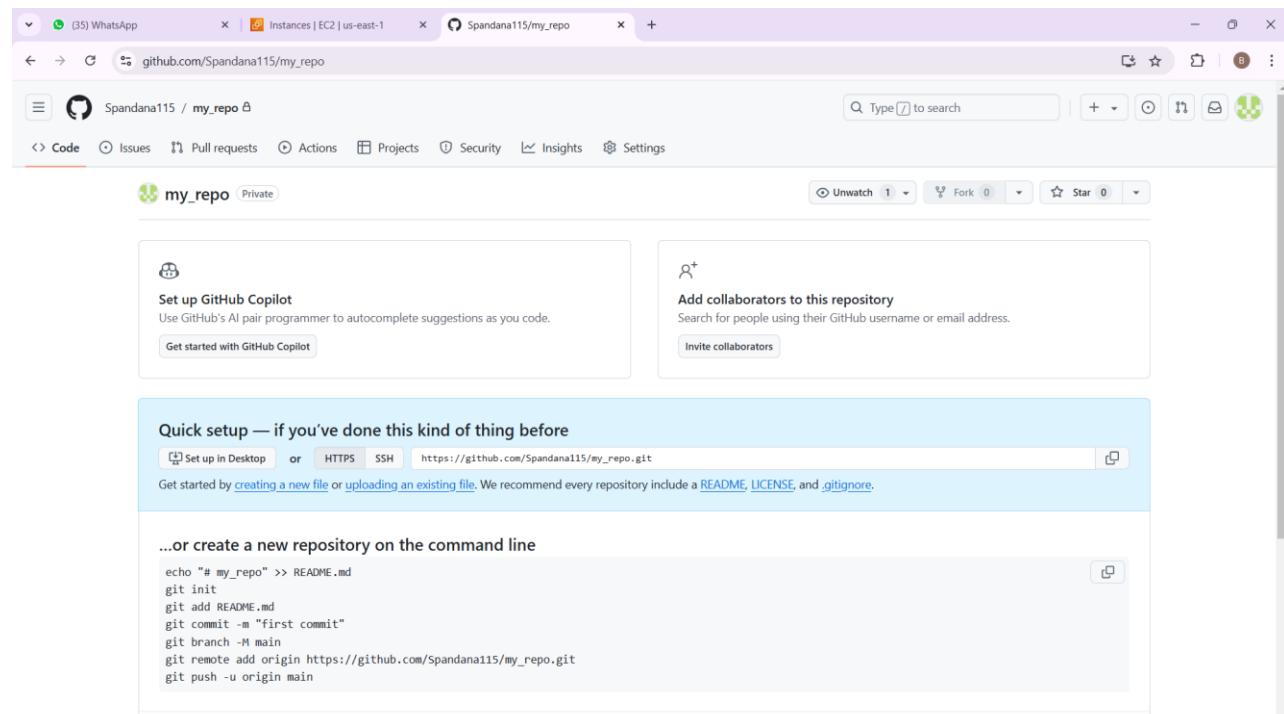
Create a repository in local machine and go to GitHub click on new repository name the repository the same as your local repository we do not initialize this repo with a README.is selected click on create repo.

```
root@ip-172-31-29-63:~/my_repo
git commit --amend --reset-author
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file1
create mode 100644 file2
create mode 100644 puppy
[root@ip-172-31-29-63 my_project_repo]# git log
commit 35ef30e75652a30e4d5ec8c1474bbbaef14215a1 [HEAD -> main]
Author: root <root@ip-172-31-29-63.ec2.internal>
Date: Mon Nov 4 09:40:46 2024 +0000

    committing multiple files

commit 34d63c25c3530f24998065489cdf900124df061 {origin/main, origin/HEAD}
Author: Spandanail15 <baindaspandana0115@gmail.com>
Date: Mon Nov 4 14:59:03 2024 +0530

Initial commit
[root@ip-172-31-29-63 my_project_repo]# git push
Username for 'https://github.com': spandanail15
Password for 'https://spandanail15@github.com':
remote: Invalid username or password.
fatal: Authentication failed for 'https://github.com/spandanail15/my_project_repo.git'
[root@ip-172-31-29-63 my_project_repo]# git push
Username for 'https://github.com': Spandanail15
Password for 'https://Spandanail15@github.com':
Enumerating objects: 4, done.
Counting objects: 100%, 4(4), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 296 bytes | 296.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Spandanail15/my_project_repo.git
  34d63c... main -> main
[root@ip-172-31-29-63 my_project_repo]# cd
[root@ip-172-31-29-63 ~]# mkdir my_repo
[root@ip-172-31-29-63 ~]# cd my_repo/
[root@ip-172-31-29-63 my_repo]# ll
total 0
[root@ip-172-31-29-63 my_repo]# git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /root/my_repo/.git/
[root@ip-172-31-29-63 my_repo]#
```



Rename the Default Branch to main **git branch -M main** and use the repository url get from GitHub creating remote repository run the following command to add the GitHub repository as the remote **git remote add origin <remote-repo-url>** we will get an error because we did not committed any files so after adding and committing any file it will be pushed.

```
root@ip-172-31-29-63:~/my_repo
commit 34d263c25c3530f24998065489cdf900124df061 (origin/main, origin/HEAD)
Author: Spandanall5 <bandalaspandana0115@gmail.com>
Date: Mon Nov 4 14:59:03 2024 +0530

Initial commit
[root@ip-172-31-29-63 my_project_repo]# git push
Username for 'https://github.com': spandanall5
Password for 'https://spandanall5@github.com':
remote: Invalid username or password.
fatal: Authentication failed for 'https://github.com/Spandanall5/my_project_repo.git'
[root@ip-172-31-29-63 my_project_repo]# git push
Username for 'https://github.com': Spandanall5
Password for 'https://Spandanall5@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 296 bytes | 296.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Spandanall5/my_project_repo.git
  34d263c..35ef30e main -> main
[root@ip-172-31-29-63 my_project_repo] cd
[Root@ip-172-31-29-63 ~]# mkdir my_repo
[Root@ip-172-31-29-63 ~]# cd my_repo/
[Root@ip-172-31-29-63 my_repo]# ll
total 0
[Root@ip-172-31-29-63 my_repo]# git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint: git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint: git branch -m <name>
Initialized empty Git repository in /root/my_repo/.git/
[Root@ip-172-31-29-63 my_repo]# git branch -M main
[Root@ip-172-31-29-63 my_repo]# git remote add origin https://github.com/Spandanall5/my_repo.git
[Root@ip-172-31-29-63 my_repo]# git push -u origin main
error: src refspec main does not match any
error: failed to push some refs to 'https://github.com/Spandanall5/my_repo.git'
[Root@ip-172-31-29-63 my_repo]# git push -u origin main
error: src refspec main does not match any
error: failed to push some refs to 'https://github.com/Spandanall5/my_repo.git'
[Root@ip-172-31-29-63 my_repo]# git push
fatal: The current branch main has no upstream branch.
To push the current branch and set the remote as upstream, use
git push --set-upstream origin main
```

Push your local branch(main) to GitHub **git push -u origin main -u** flag sets the upstream branch linking your local branch to the remote.

```
root@ip-172-31-29-63:~/my_repo
[Root@ip-172-31-29-63 my_repo]# git remote add origin https://github.com/Spandanall5/my_repo.git
[Root@ip-172-31-29-63 my_repo]# git push -u origin main
error: src refspec main does not match any
error: failed to push some refs to 'https://github.com/Spandanall5/my_repo.git'
[Root@ip-172-31-29-63 my_repo]# git push -u origin main
error: src refspec main does not match any
error: failed to push some refs to 'https://github.com/Spandanall5/my_repo.git'
[Root@ip-172-31-29-63 my_repo]# git push
fatal: The current branch main has no upstream branch.
To push the current branch and set the remote as upstream, use
git push --set-upstream origin main

To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetUpstream' in 'git help config'.

[Root@ip-172-31-29-63 my_repo]# echo "# My New Repository" > README.md
[Root@ip-172-31-29-63 my_repo]# git log
fatal: your current branch 'main' does not have any commits yet
[Root@ip-172-31-29-63 my_repo]# git add README.md
[Root@ip-172-31-29-63 my_repo]# ll
total 0
-rw-r--r-- 1 root root 20 Nov 4 09:56 README.md
[Root@ip-172-31-29-63 my_repo]# git commit -m "initial commit"
[main (root-commit) abc5ba9] initial commit
 Committer: root <root@ip-172-31-29-63.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:
 git config --global --edit

After doing this, you may fix the identity used for this commit with:
 git commit --amend --reset-author

 1 file changed, 1 insertion(+)
 create mode 100644 README.md
[Root@ip-172-31-29-63 my_repo]# git push -u origin main
Username for 'https://github.com': Spandanall5
Password for 'https://Spandanall5@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 237 bytes | 237.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Spandanall5/my_repo.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
[Root@ip-172-31-29-63 my_repo]#
```

Open GitHub repository click on main branch in the dropdown type the name of new branch select the option and create new branch from main branch.

The screenshot shows a GitHub repository page for 'Spandana115/my\_repo'. The 'Code' tab is selected. In the top left, there's a dropdown menu next to 'main' with the option 'My New Repository'. A tooltip or modal is visible, indicating the action of creating a new branch. The repository contains one file, 'README.md', which has the content 'My New Repository'. On the right side, there are sections for 'About', 'Releases', and 'Packages', all of which are currently empty.

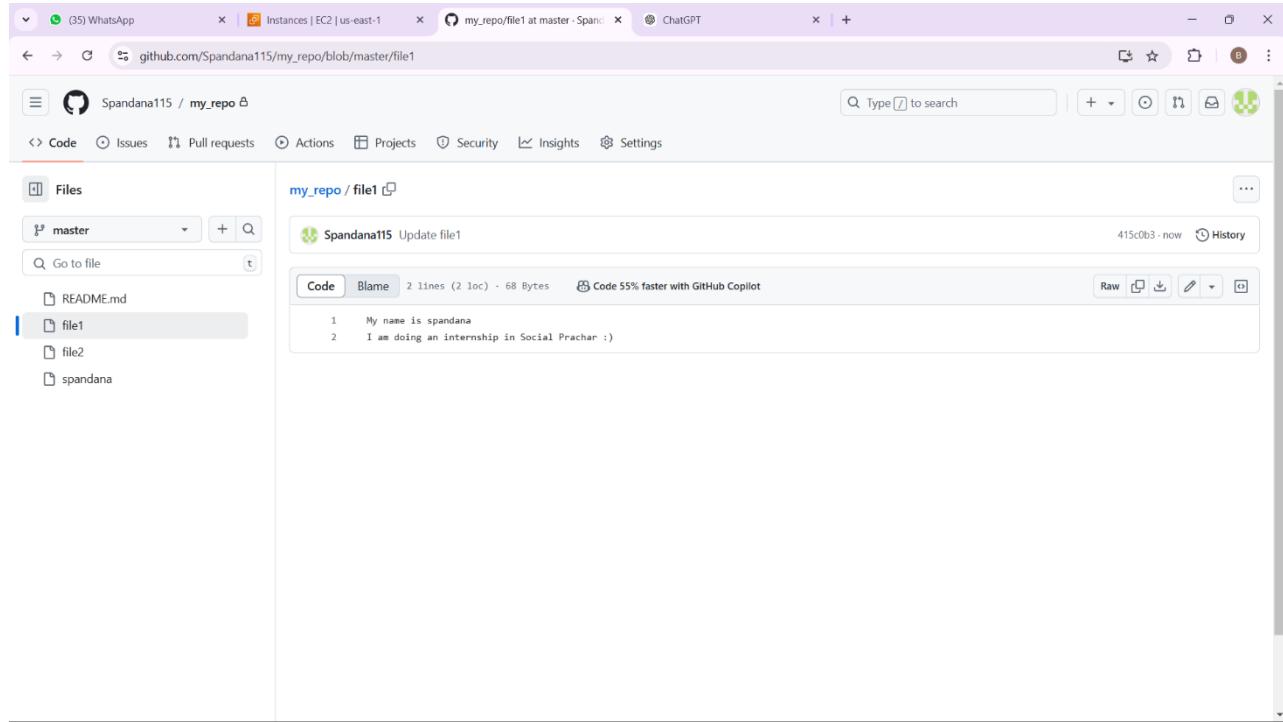
In the below image we can able to see that master branch has been created successfully.

The screenshot shows a GitHub repository page for 'Spandana115/my\_project\_repo'. The 'Branches' tab is selected. A green button labeled 'New branch' is visible in the top right. The 'Your branches' section shows a table with one row:

Branch	Updated	Check status	Behind	Ahead	Pull request
master	now	Green	0	0	...

Below this, there are sections for 'Default' (showing the 'main' branch) and 'Active branches' (also showing the 'master' branch).

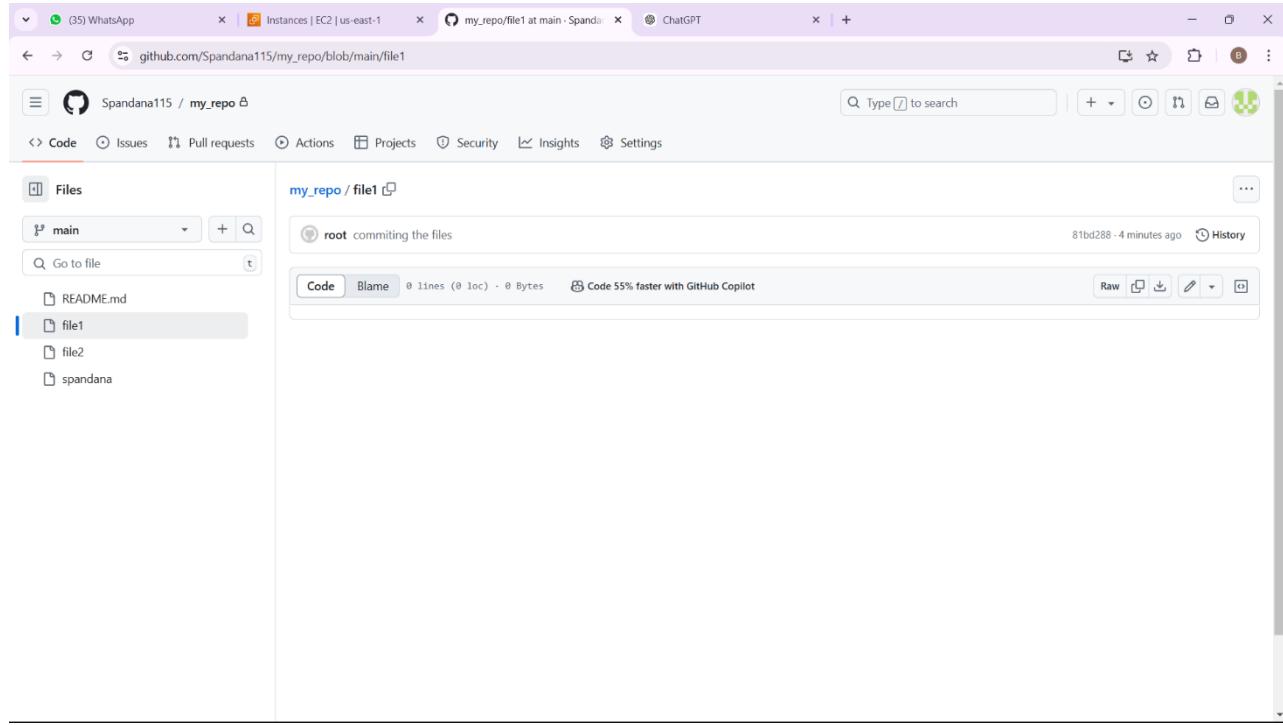
Switch to new branch and go to the any file edit that file or write a content in that file with in new repository make changes and click on commit changes.



The screenshot shows a browser window with several tabs open. The active tab is 'my\_repo / file1' on GitHub. The left sidebar shows a tree view of files: 'master' (selected), 'README.md', 'file1' (highlighted in blue), 'file2', and 'spandana'. The main content area displays the 'file1' content with the following code:

```
My name is spandana
I am doing an internship in Social Prachar :)
```

Switch back to main branch and using branch dropdown check the file that has been updated or modified we can see that remains unchanged in file1 in main branch in the below image. Go back to the previous image and see that stores some content in the file1.



## LAB -7 PULL ALL THE BRANCHES IN YOUR LOCAL MACHINES

Open PuTTY we have to go to where git repository located we have to run the following commands to fetch the branches from remote repository **git pull** list all branches using **git branch -a**.

```

root@ip-172-31-29-63:~/my_repo
git: 'branches' is not a git command. See 'git --help'.
[root@ip-172-31-29-63 my_repo]# git branchs
git: 'branchs' is not a git command. See 'git --help'.

The most similar command is
  branch
[root@ip-172-31-29-63 my_repo]# git branch
* main
[root@ip-172-31-29-63 my_repo]# cat spandana
[root@ip-172-31-29-63 my_repo]# ll
total 4
-rw-r--r-- 1 root root 20 Nov  4 09:56 README.md
-rw-r--r-- 1 root root 0 Nov  4 10:09 file1
-rw-r--r-- 1 root root 0 Nov  4 10:09 file2
-rw-r--r-- 1 root root 0 Nov  4 10:09 spandana
[root@ip-172-31-29-63 my_repo]# cat file1
[root@ip-172-31-29-63 my_repo]# git pull
Username for 'https://github.com': Spandanall5
Password for 'https://Spandanall5@github.com':
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0).
Unpacking objects: 100% (3/3), 1019 bytes | 1019.00 KiB/s, done.
From https://github.com/Spandanall5/my_repo
 * [new branch] master      -> origin/master
Already up to date.
[root@ip-172-31-29-63 my_repo]# git branch -a
* main
  remotes/origin/main
  remotes/origin/master
[root@ip-172-31-29-63 my_repo]# git checkout master
branch 'master' set up to track 'origin/master'.
Switched to a new branch 'master'
[root@ip-172-31-29-63 my_repo]# ll
total 8
-rw-r--r-- 1 root root 20 Nov  4 09:56 README.md
-rw-r--r-- 1 root root 68 Nov  4 10:17 file1
-rw-r--r-- 1 root root 0 Nov  4 10:09 file2
-rw-r--r-- 1 root root 0 Nov  4 10:09 spandana
[root@ip-172-31-29-63 my_repo]# git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
[root@ip-172-31-29-63 my_repo]# git branch
* main
* master
[root@ip-172-31-29-63 my_repo]# cat file1
My name is spandana
I am doing an internship in Social Prachar :)
[root@ip-172-31-29-63 my_repo]#

```

Switch to newly created branch using **git checkout <master>** confirm that we are on a new branch by using **git branch**

the active branch is marked with \*.

```
root@ip-172-31-29-63:~/my_repo
[root@ip-172-31-29-63 my_repo]# git branches
git: 'branches' is not a git command. See 'git --help'.

The most similar command is
  branch
[root@ip-172-31-29-63 my_repo]# git branch
* main
[root@ip-172-31-29-63 my_repo]# cat spandana
[root@ip-172-31-29-63 my_repo]# ll
total 4
-rw-r--r-- 1 root root 20 Nov  09:56 README.md
-rw-r--r-- 1 root root  0 Nov  4 10:09 file1
-rw-r--r-- 1 root root  0 Nov  4 10:09 file2
-rw-r--r-- 1 root root  0 Nov  4 10:09 spandana
[root@ip-172-31-29-63 my_repo]# cat file1
[root@ip-172-31-29-63 my_repo]# git pull
Username for 'https://github.com': Spandanall15
Password for 'https://Spandanall15@github.com':
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 1019 bytes | 1019.00 KiB/s, done.
From https://github.com/Spandanall15/my_repo
 * [new branch] master      -> origin/master
Already up to date.
[root@ip-172-31-29-63 my_repo]# git branch -
* main
* remotes/origin/main
* remotes/origin/master
[root@ip-172-31-29-63 my_repo]# git checkout master
branch 'master' set up to track 'origin/master'.
Switched to a new branch 'master'
[root@ip-172-31-29-63 my_repo]# ll
total 0
-rw-r--r-- 1 root root 20 Nov  09:56 README.md
-rw-r--r-- 1 root root 68 Nov  4 10:17 file1
-rw-r--r-- 1 root root  0 Nov  4 10:09 file2
-rw-r--r-- 1 root root  0 Nov  4 10:09 spandana
[root@ip-172-31-29-63 my_repo]# git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
[root@ip-172-31-29-63 my_repo]# git branch
* main
* master
[root@ip-172-31-29-63 my_repo]# cat file1
My name is spandana
I am doing an internship in Social Prachar :)
[root@ip-172-31-29-63 my_repo]# touch file3 file4
[root@ip-172-31-29-63 my_repo]# git status
```

Add some files to the branch you can create empty files with the touch command **touch file3.txt file4.txt** and stage the new files by **git add <file names>** and commit changes using **git commit -m “message”** after committing check logs if everything correct push the committed changes to remote feature branch.

```
root@ip-172-31-29-63:~/my_repo
your configuration file:
git config --global --edit

After doing this, you may fix the identity used for this commit with:
git commit --amend --reset-author

2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file3
create mode 100644 file4
[root@ip-172-31-29-63 my_repo]# git log
commit c17b59a71d4de57fe054c63b4e936ae0d5f61796 (HEAD -> master)
Author: root <root@ip-172-31-29-63.ec2.internal>
Date:   Mon Nov 4 10:20:26 2024 +0000

    Adding two more files and updating the master branch

commit 415c0b3ae9cc4b35074462aecfc4a923618a2c2 (origin/master)
Author: Spandanall15 <baindaspandana0115@gmail.com>
Date:   Mon Nov 4 15:42:10 2024 +0530

    Update file1

commit 81bd2881bea030d5da6d0b458802d9886964d71f (origin/main, main)
Author: root <root@ip-172-31-29-63.ec2.internal>
Date:   Mon Nov 4 10:09:44 2024 +0000

    committing the files

commit a8c5cba9bfe9ca4213d3f9528aaafee3cdab1fff3
Author: root <root@ip-172-31-29-63.ec2.internal>
Date:   Mon Nov 4 09:57:13 2024 +0000

    initial commit
[root@ip-172-31-29-63 my_repo]# git push
Username for 'https://github.com': Spandanall15
Password for 'https://Spandanall15@github.com':
error: Invalid username or password.
fatal: Authentication failed for 'https://github.com/Spandanall15/my_repo.git'
[root@ip-172-31-29-63 my_repo]# git push
Username for 'https://github.com': Spandanall15
Password for 'https://Spandanall15@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 294 bytes | 294.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/Spandanall15/my_repo.git
  415c0b3..c17b59a master -> master
[root@ip-172-31-29-63 my_repo]#
```

Confirm no changes in main branch switch back to the main branch in GitHub and verify that the new files are not present in the main branch.

The screenshot shows a GitHub repository named "my\_repo". The repository is private. It has two branches: "main" and "master". The "master" branch is selected. The repository has 2 commits:

- "initial commit" by "spandana" at 81bd288 · 12 minutes ago
- "committing the files" by "spandana" at 12 minutes ago

The "main" branch has 1 commit:

- "committing the files" by "spandana" at 12 minutes ago

The README file contains the text "My New Repository".

Switch to master branch to verify that the new files are presented.

The screenshot shows the same GitHub repository "my\_repo". Now the "master" branch is selected. The repository has 4 commits:

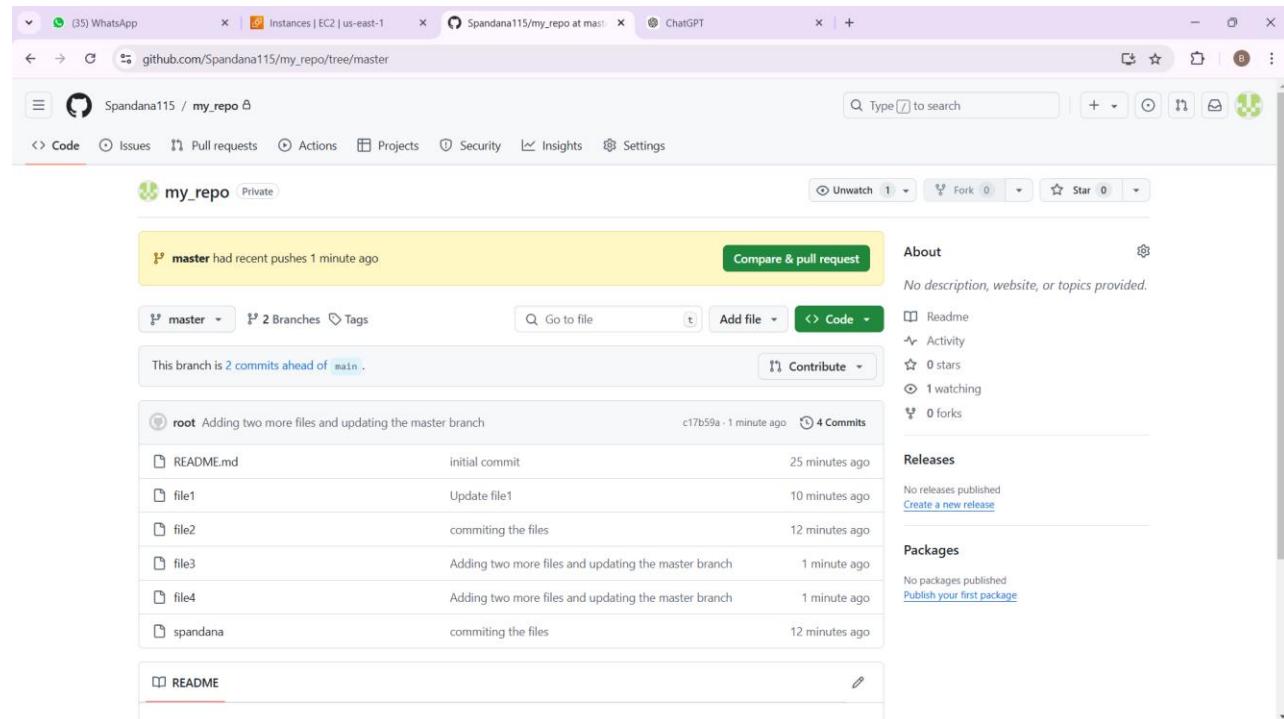
- "initial commit" by "spandana" at 25 minutes ago
- "Update file1" by "spandana" at 10 minutes ago
- "committing the files" by "spandana" at 12 minutes ago
- "Adding two more files and updating the master branch" by "spandana" at 1 minute ago

The "main" branch has 1 commit:

- "committing the files" by "spandana" at 12 minutes ago

## LAB-8 MERGE OUR FEATURE BRANCH WITH OUR MAIN BRANCH

Open your repository in GitHub where you have both the main and master branches. Click on the “Pull request” tab at the top of the repository page.



Click on the “New pull request” button to initiate a pull request.

The screenshot shows a GitHub interface with the URL [github.com/Spandana115/my\\_repo/pulls](https://github.com/Spandana115/my_repo/pulls). The repository name is Spandana115 / my\_repo. The 'Pull requests' tab is selected. A search bar contains the query 'ispr isopen'. Below the search bar, there are filters: '1 Open' (with 1 open pull request) and 'Master' (with one pull request labeled '#1 opened now by Spandana115'). A tip message says 'ProTip! Adding `nolabel` will show everything without a label.' At the bottom, there's a copyright notice: '© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information'.

Select branches for the Pull Request in the comparison options set base branch as main and set compare branch as a master click on create pull request and confirm.

The screenshot shows a GitHub interface with the URL [github.com/Spandana115/my\\_repo/compare/main...master](https://github.com/Spandana115/my_repo/compare/main...master). The repository name is Spandana115 / my\_repo. The 'Comparing changes' section shows a pull request between 'base: main' and 'compare: master'. The status is 'Able to merge'. The 'Master #1' branch has no description available. It shows 2 commits, 3 files changed, and 2 contributors. A commit titled 'Update file1' by Spandana115 was authored 14 minutes ago. Another commit titled 'Adding two more files and updating the master branch' by root was committed 5 minutes ago. The commit details are shown in a detailed view at the bottom. The bottom right corner of the screenshot has a small watermark: '90 | Page'.

In the pull request tab locate a newly created pull request. Click on the pull request to view details. Click on the “Merge pull request” button and confirm the merge by clicking on confirm merge.

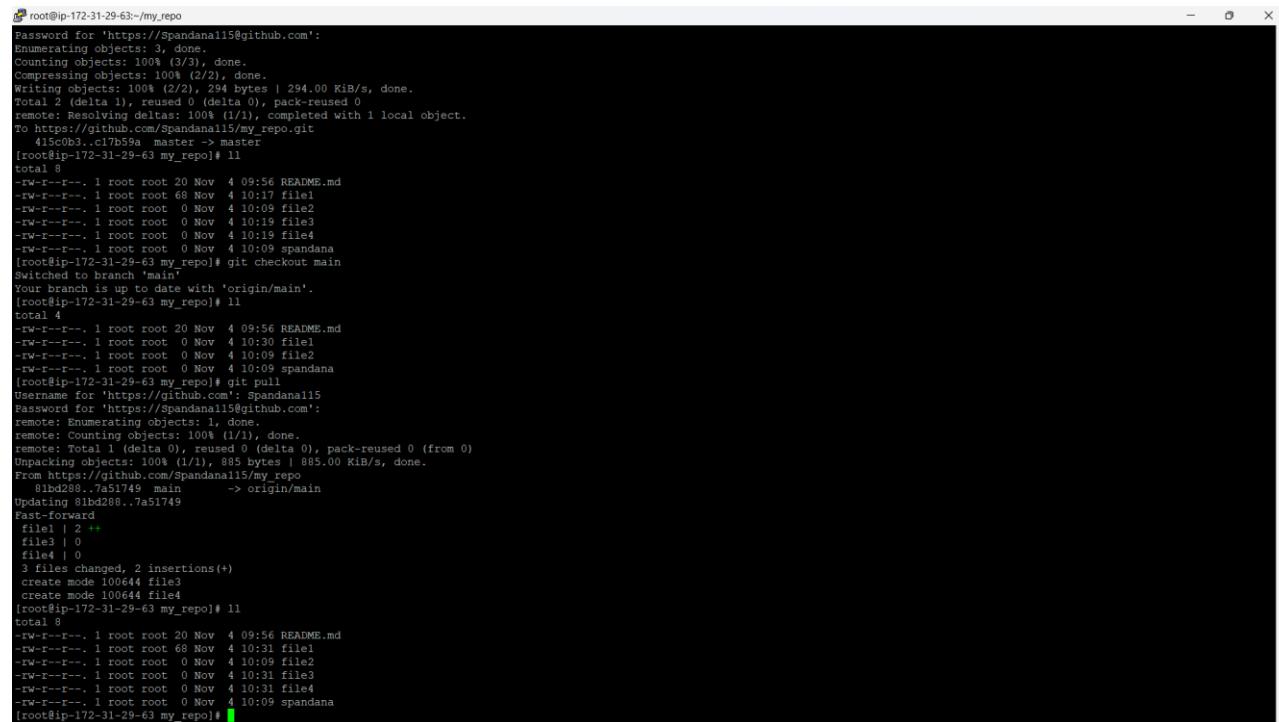
The screenshot shows a GitHub pull request page for a repository named "my\_repo". The pull request has been merged, as indicated by the "Merged" status and a message from "Spandana115" stating "Spandana115 merged 2 commits into main from master now". The commit history shows two commits from "Spandana115" merging changes from the "master" branch into the "main" branch. A message box at the bottom of the pull request details states "Pull request successfully merged and closed". The repository page also shows the merged pull request under the "Pull requests" tab.

We can able to see that the files are merged from master branch to main branch.

The screenshot shows the GitHub repository page for "my\_repo". The "main" branch is selected, and the commit history shows several commits from "Spandana115" merging changes from the "master" branch into the "main" branch. The commits include "initial commit", "Update file1", "committing the files", "Adding two more files and updating the master branch", and "committing the files". The repository page also displays the "About" section, which includes the repository's description ("No description, website, or topics provided."), activity metrics (0 stars, 1 watching, 0 forks), release information (no releases published), and package information (no packages published).

## LAB-9 GO TO LOCAL MACHINE

Open PuTTY on local machine where you have a cloned version of remote repository. Checkout to the main branch using **git checkout main** this ensures we are working with the main branch before pulling any changes. To pull the latest changes from the remote repository using **git pull** this will fetch and merge any updates from the remote repository into local main branch. After the pull command is executed main branch will be up to date with the remote repository run the following command to check the status and confirm using **git status** and **git log**.



```
root@ip-172-31-29-63:~/my_repo
Password for 'https://Spandanali5@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression objects: 100% (2/2), done.
Writing objects: 100% (2/2), 294 bytes | 294.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/Spandanali5/my_repo.git
   415c0b3..c17b59a master -> master
[root@ip-172-31-29-63 my_repo]# ll
total 8
-rw-r--r--. 1 root root 20 Nov 4 09:56 README.md
-rw-r--r--. 1 root root 68 Nov 4 10:17 file1
-rw-r--r--. 1 root root 0 Nov 4 10:09 file2
-rw-r--r--. 1 root root 0 Nov 4 10:19 file3
-rw-r--r--. 1 root root 0 Nov 4 10:19 file4
-rw-r--r--. 1 root root 0 Nov 4 10:09 spandana
[root@ip-172-31-29-63 my_repo]# git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
[root@ip-172-31-29-63 my_repo]# ll
total 4
-rw-r--r--. 1 root root 20 Nov 4 09:56 README.md
-rw-r--r--. 1 root root 0 Nov 4 10:30 file1
-rw-r--r--. 1 root root 0 Nov 4 10:09 file2
-rw-r--r--. 1 root root 0 Nov 4 10:09 spandana
[root@ip-172-31-29-63 my_repo]# git pull
Username for 'https://github.com': Spandanali5
Password for 'https://Spandanali5@github.com':
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (1/1), 885 bytes | 885.00 KiB/s, done.
From https://github.com/Spandanali5/my_repo
   61bd280..7a51749 main      -> origin/main
Updating 61bd280..7a51749
Fast-forward
 file1 | 2 ++
 file3 | 0
 file4 | 0
 3 files changed, 2 insertions(+)
 create mode 100644 file3
 create mode 100644 file4
[root@ip-172-31-29-63 my_repo]# ll
total 8
-rw-r--r--. 1 root root 20 Nov 4 09:56 README.md
-rw-r--r--. 1 root root 68 Nov 4 10:31 file1
-rw-r--r--. 1 root root 0 Nov 4 10:09 file2
-rw-r--r--. 1 root root 0 Nov 4 10:31 file3
-rw-r--r--. 1 root root 0 Nov 4 10:31 file4
-rw-r--r--. 1 root root 0 Nov 4 10:09 spandana
[root@ip-172-31-29-63 my_repo]#
```