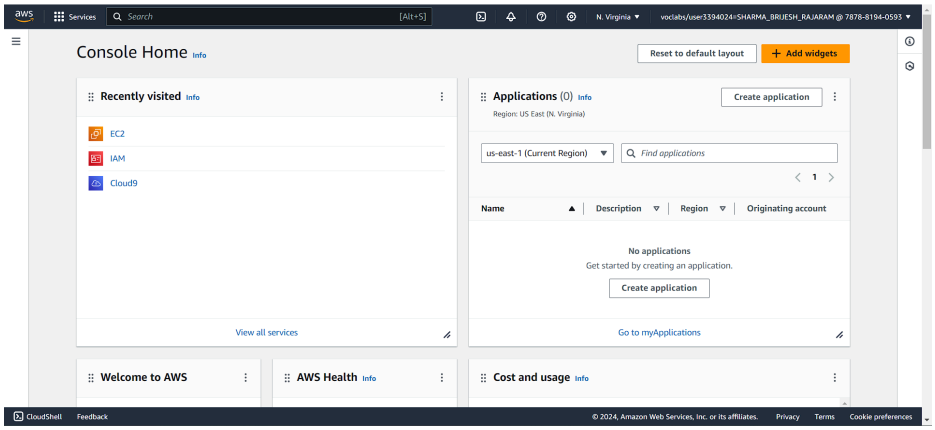
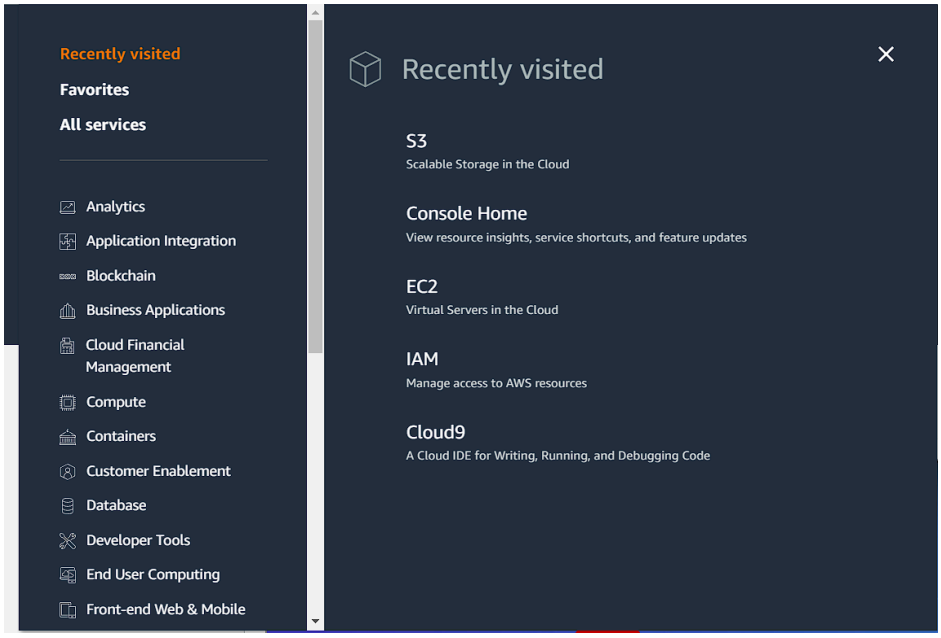
Hosting a static website on Amazon Web Services (S3)

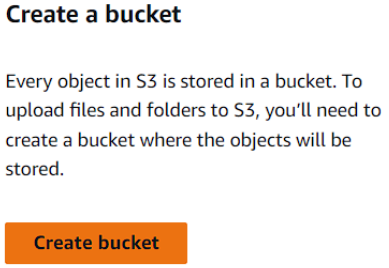
1) Open the AWS console home



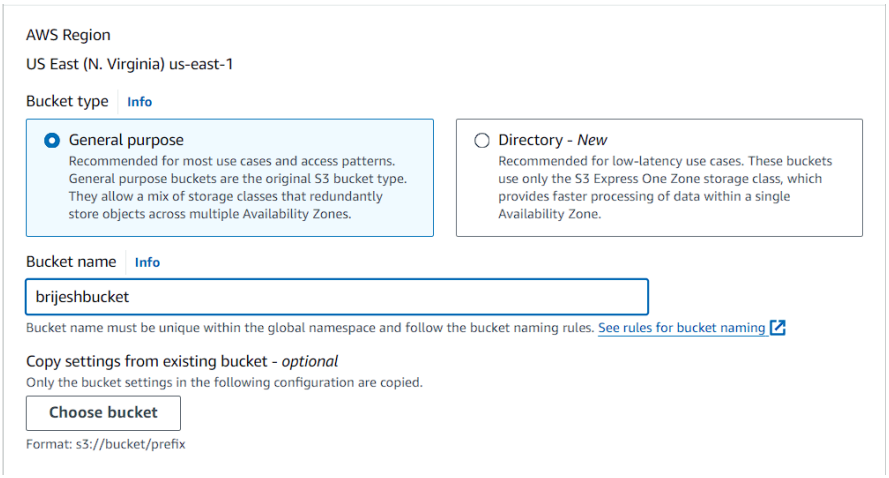
2) Navigate to the S3 to host the website



3) On S3, click on create bucket



4) Click on Bucket type as General Purpose and name the bucket.



5) Keep the default settings intact, checkingfor bucket versioning as disbale and bucket key enabled.

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

- (•) ACLs disabled (recommended)
  All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.
- ( ) ACLs enabled
  Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

Bucket owner enforced

Block Public Access settings for this bucket

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 this bucket and its 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 this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

- [ ] **Block all public access**
  Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
  - [ ] **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
    S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
  - [ ] **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
    S3 will ignore all ACLs that grant public access to buckets and objects.
  - [ ] **Block public access to buckets and objects granted through *new* public bucket or access point policies**
    S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
  - [ ] **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
    S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

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

- (•) Disable
- ( ) Enable

Tags - *optional* (0)

You can use bucket tags to track storage costs and organize buckets. Learn more

No tags associated with this bucket.

Add tag

Default encryption Info

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type | Info

- (•) Server-side encryption with Amazon S3 managed keys (SSE-S3)
- ( ) Server-side encryption with AWS Key Management Service keys (SSE-KMS)
- ( ) Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
  Secure your objects with two separate layers of encryption. For details on pricing, see DSSE-KMS pricing on the Storage tab of the Amazon S3 pricing page.

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. Learn more

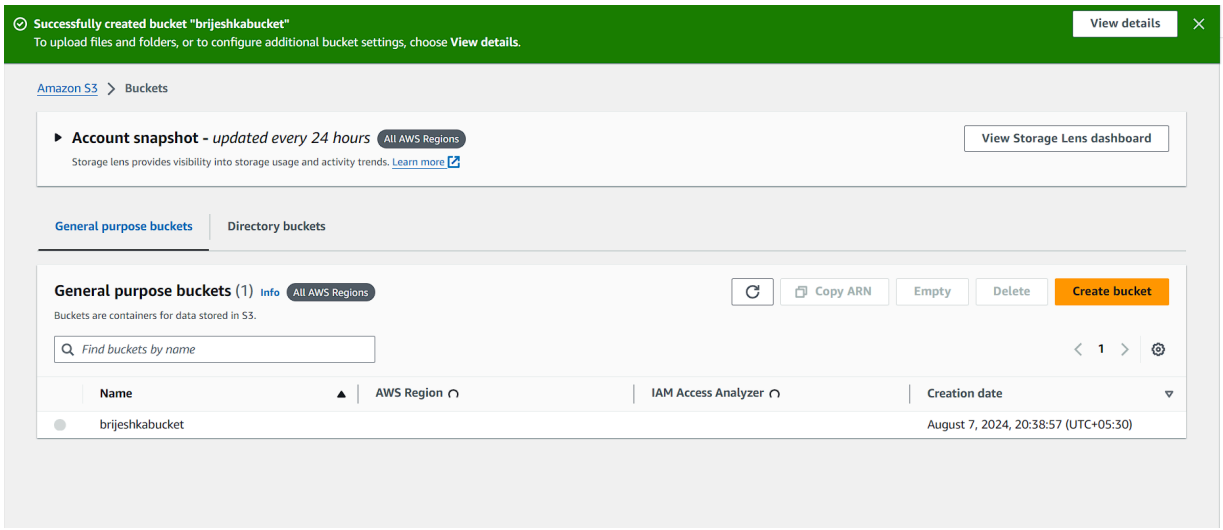
- ( ) Disable
- (•) Enable

▸ **Advanced settings**

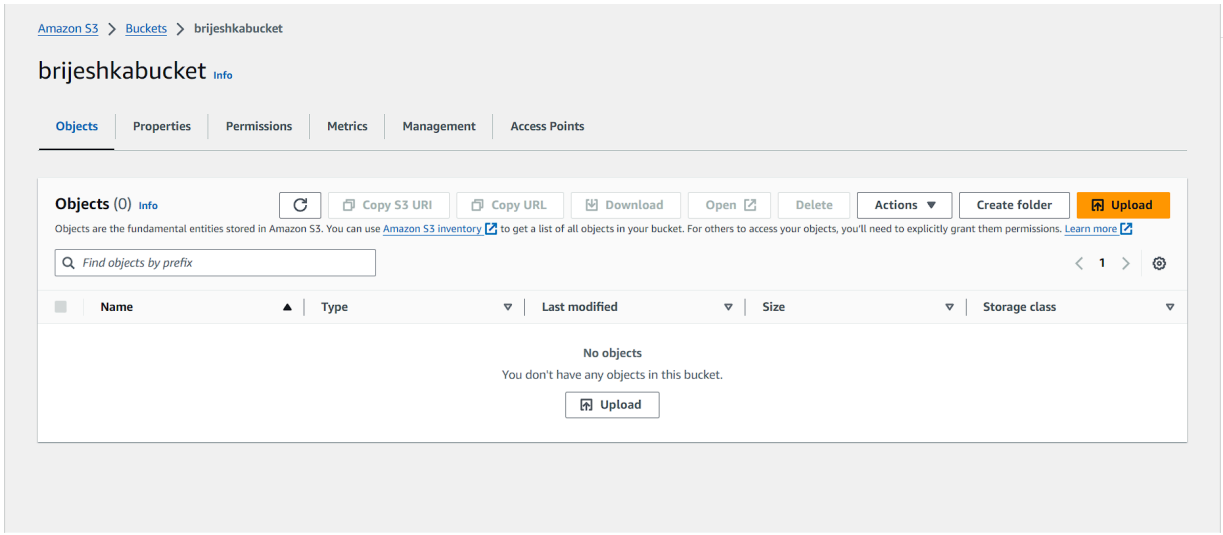
After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel | Create bucket

6) After successfully creating the bucket, click on bucket name to change the settings to host the website.

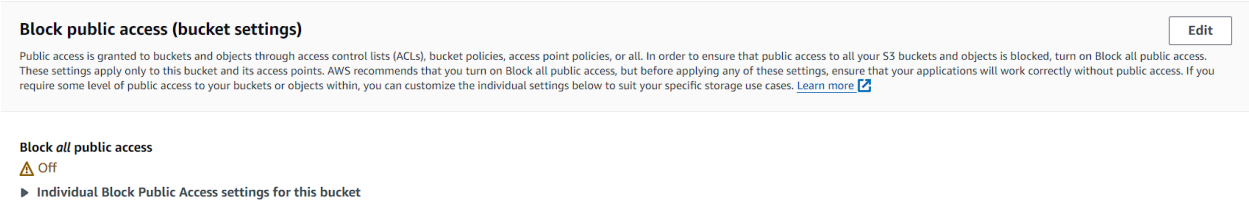


7) Go on Permissions tab and check for Block public access

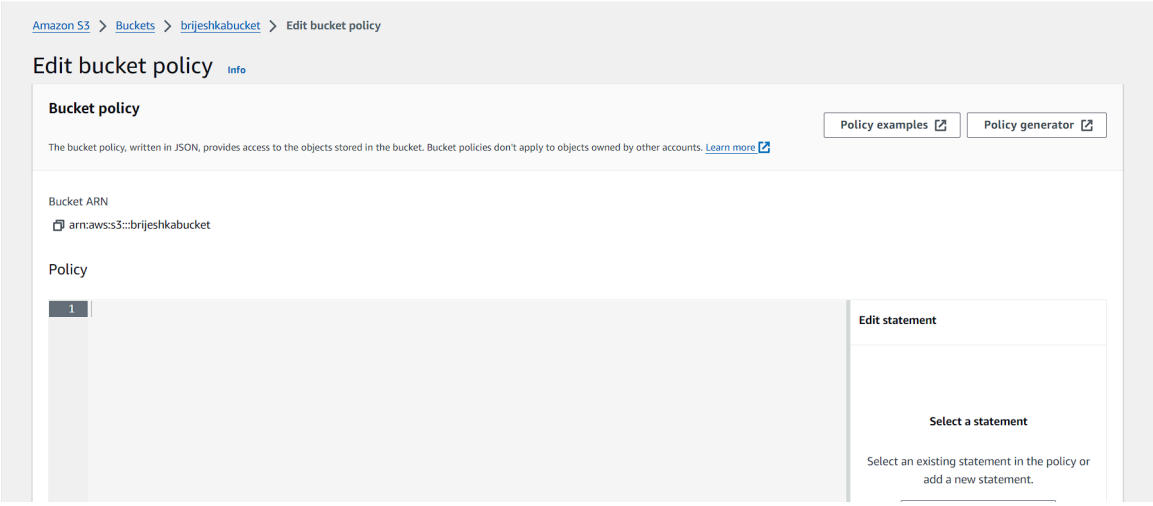


8) Block public access is default on, we need to uncheck it to ensure the hosted website is public.

9) Now the block public access option is unchecked and hence the website can be hosted successfully.



10) Now, navigate to the edit bucket policy in Properties tab to provide access to the services.

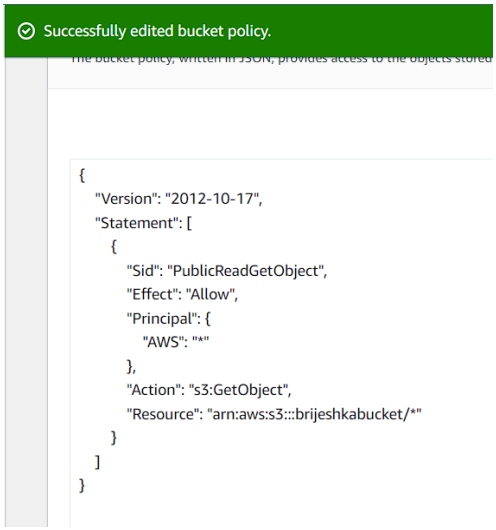


11) Fill the following policy in the empty policy space. Ensure that you change the name of the bucket in Resource with the name of your bucket.

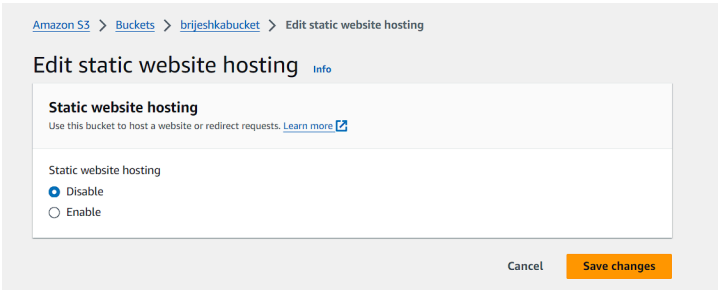
Policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PublicReadGetObject",
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::brijeshkabucket/*"
        }
    ]
}
```

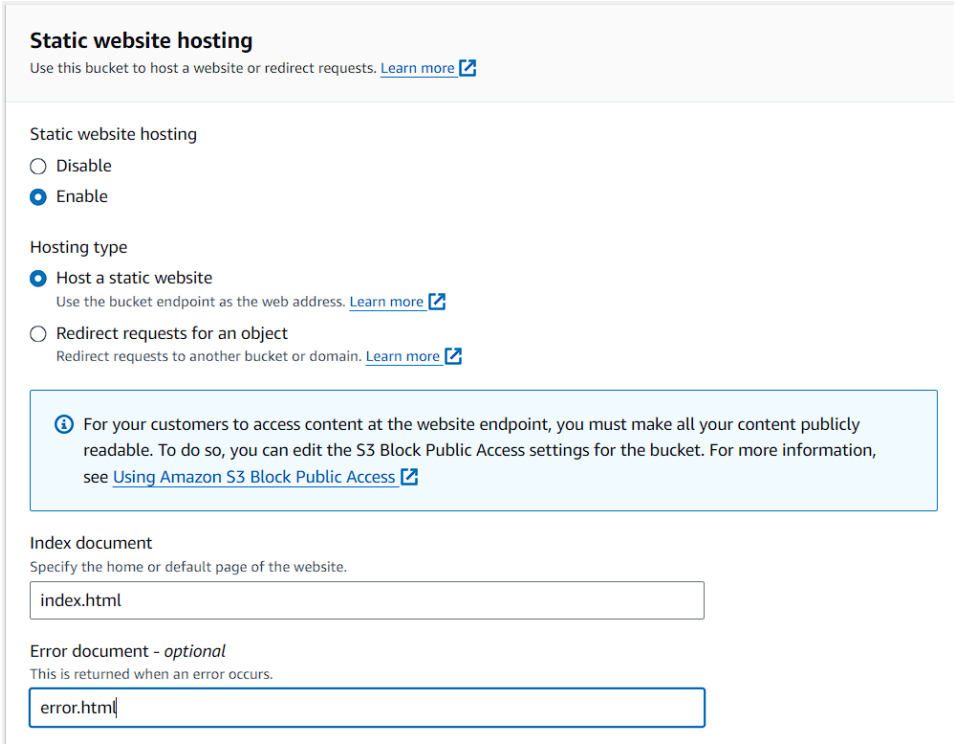
12) After saving the changes, you will see a message.



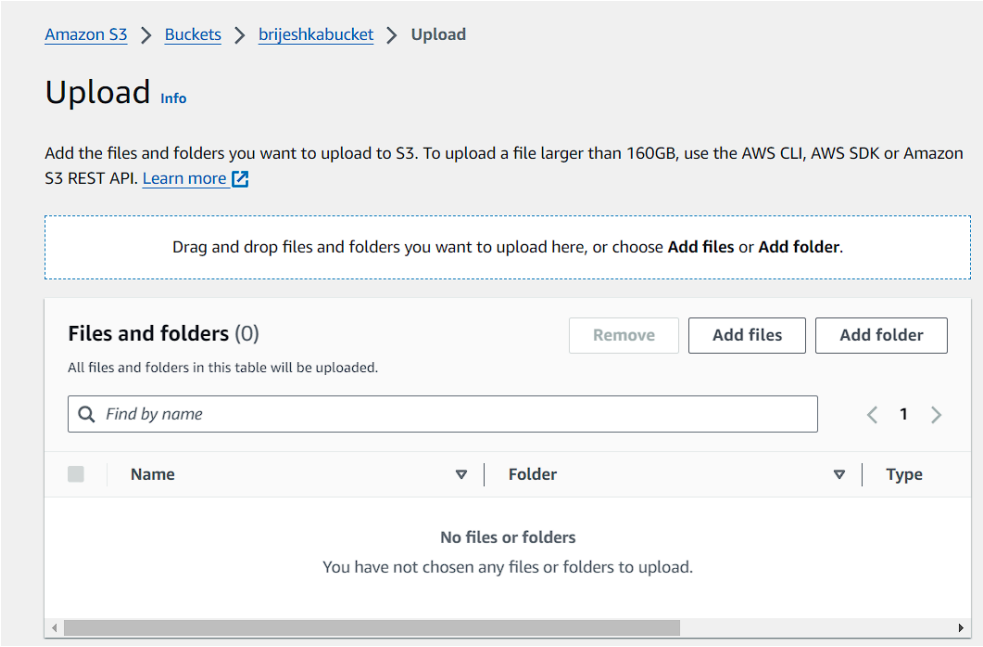
13) Go to the edit static website hosting in the properties tab to use bucket to host websites.



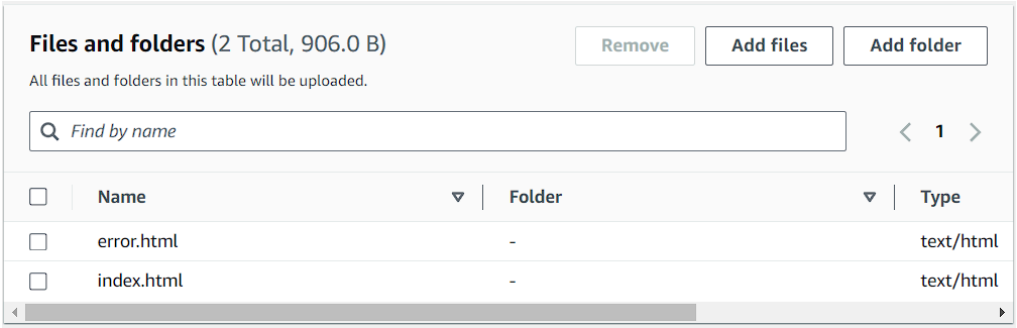
14) Check the options as shown below, and add the names of the file.



15) Navigate to the Upload section and upload the documents with the name as mentioned in the previous section.



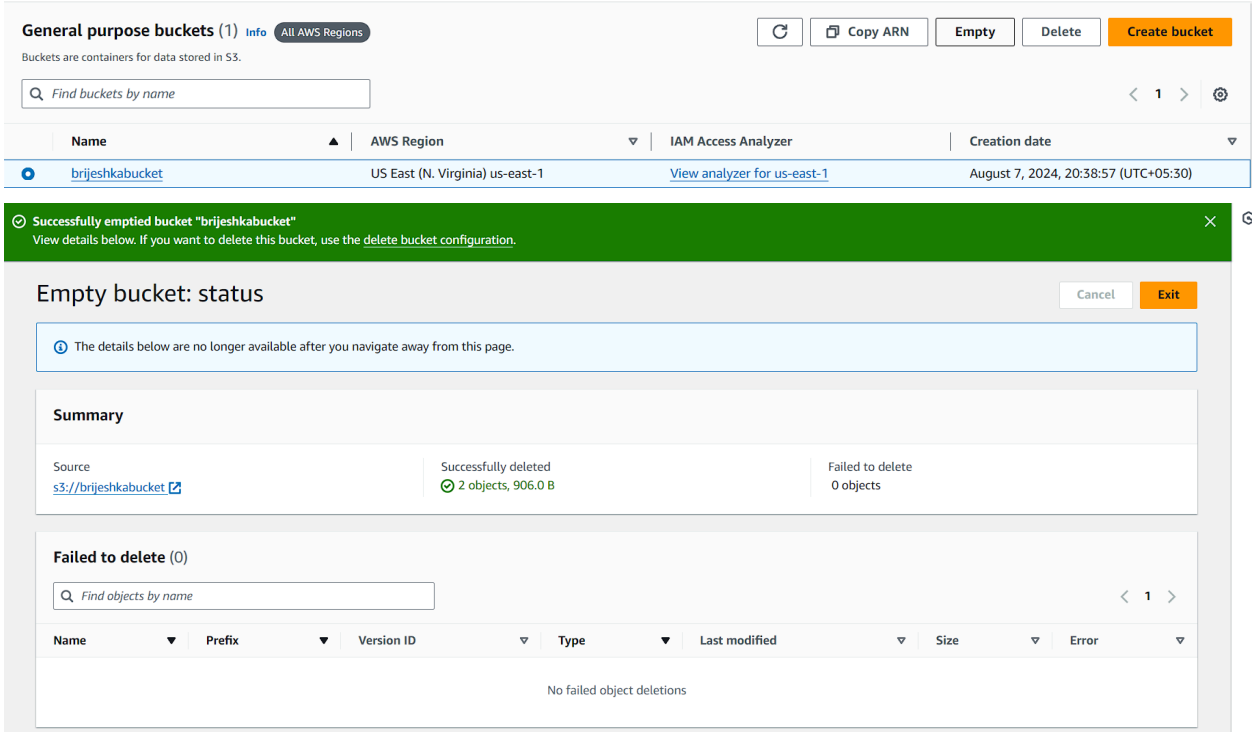
16) Uploaded files will be visible after successful upload.



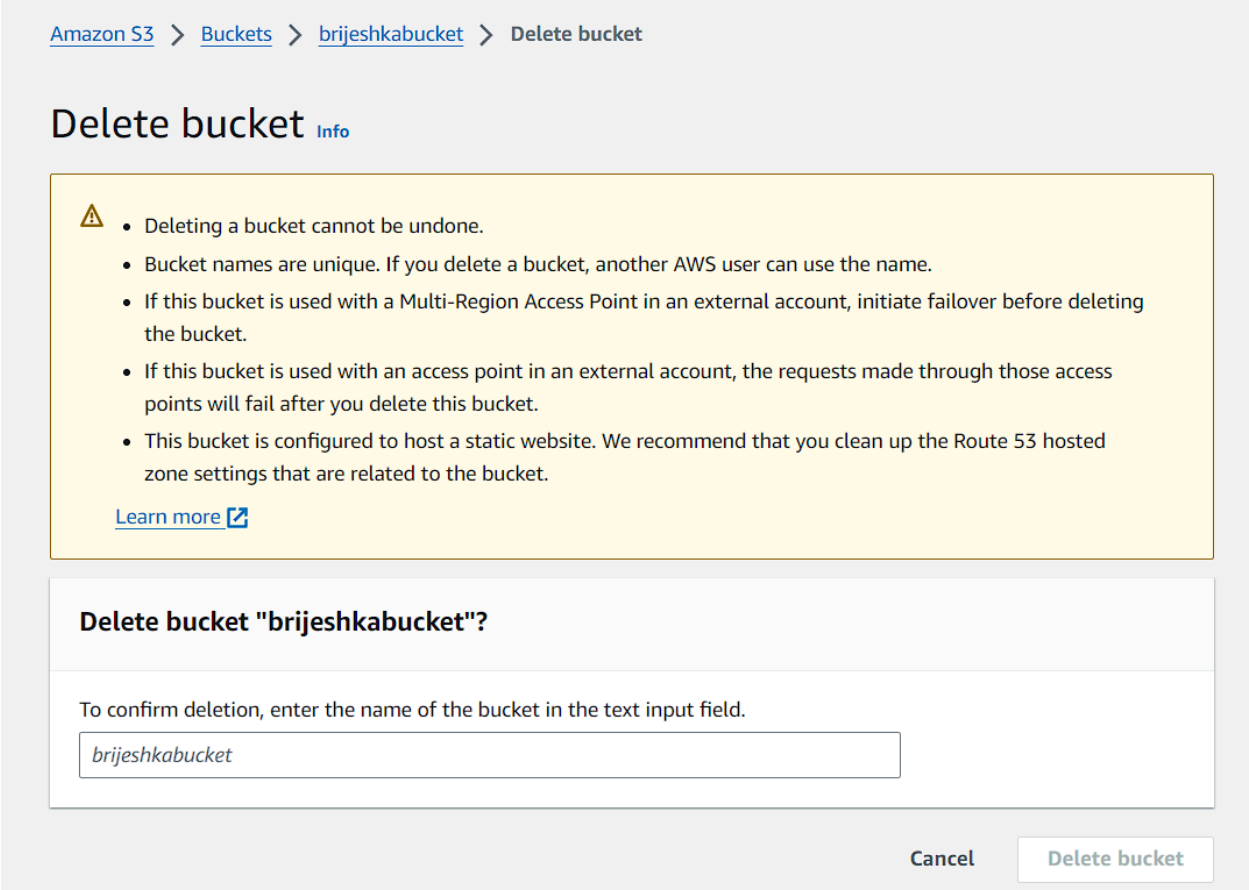
17) Get the link for the hosted website in the properties tab at the bottom.

18) The hosted website using AWS S3.

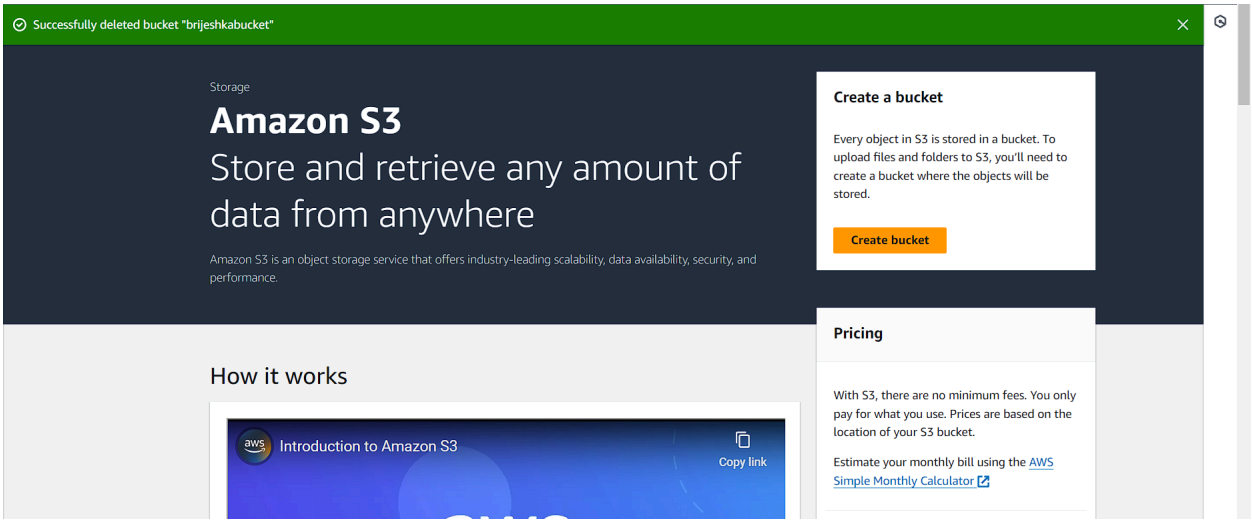
19) To terminate the S3 bucket, first empty the bucket by selecting the files and clicking on Empty.



20) Then navigate to the Delete bucket option and enter the name of the bucket and delete the bucket.

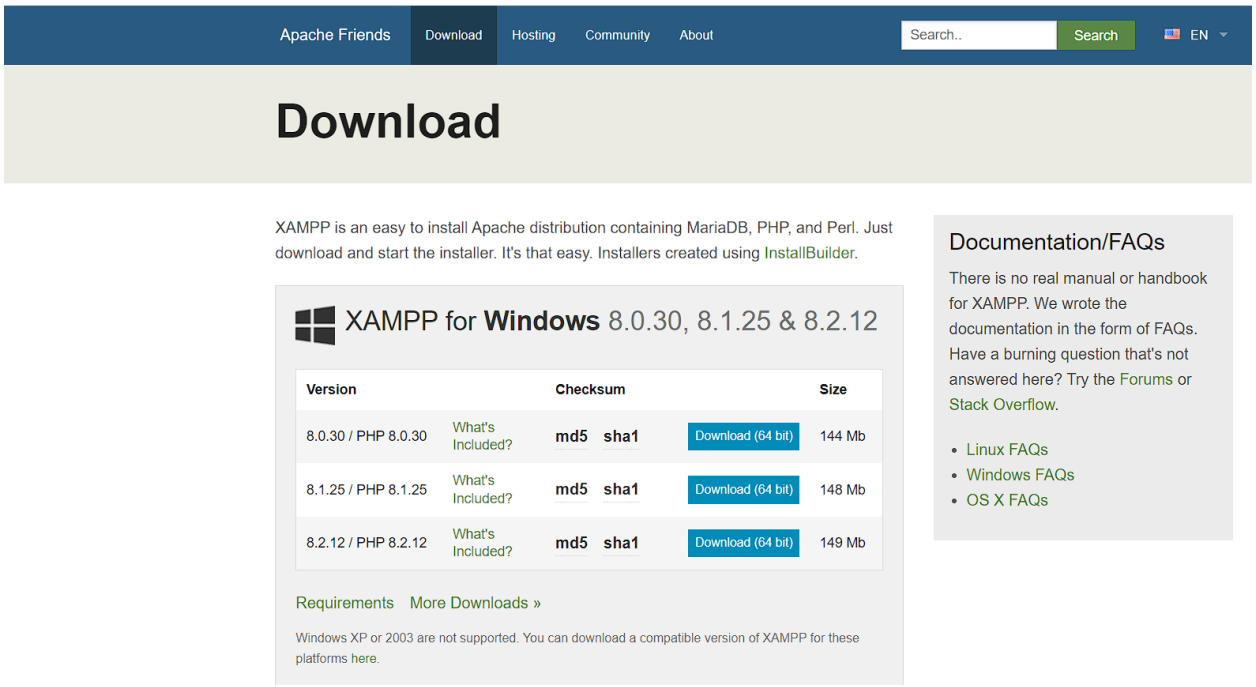


21) After deleting the bucket, a message will appear.

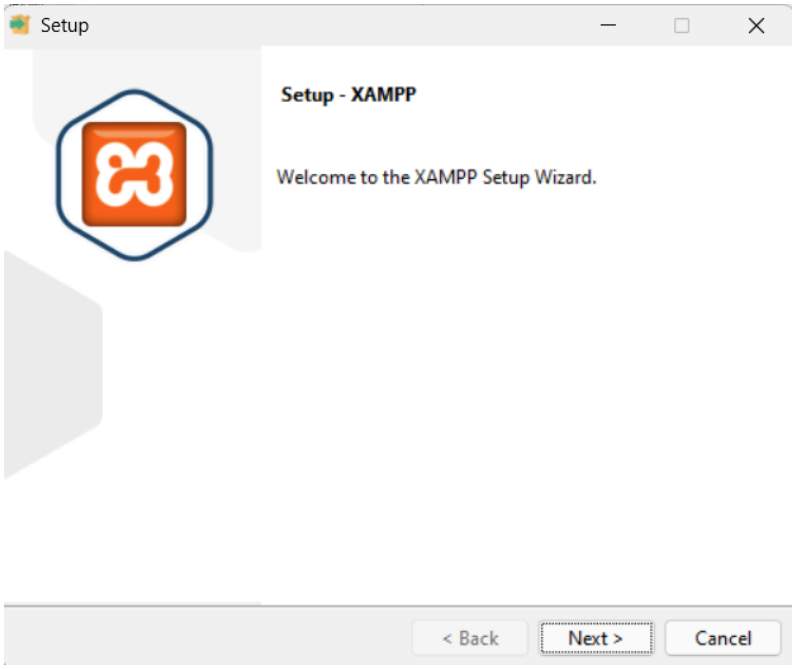


XAMPP Hosting

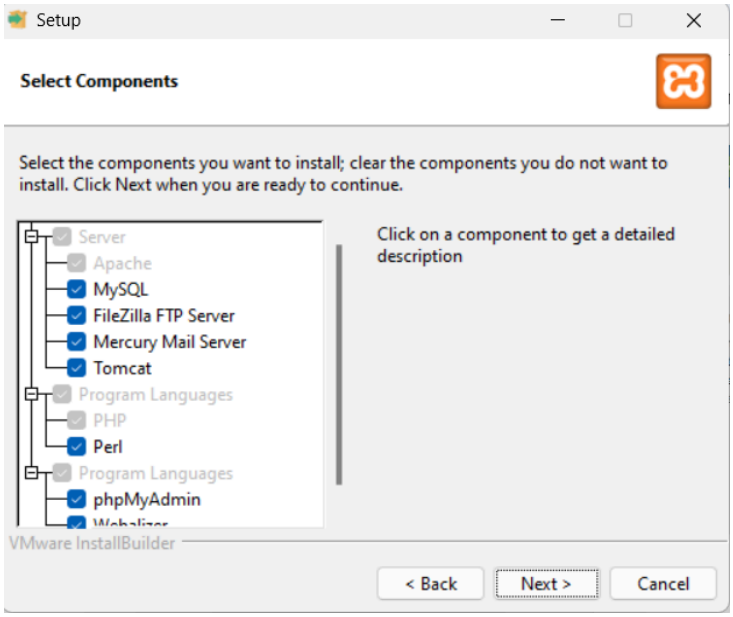
1) Search for XAMPP download and navigate to the xampp official website and click on download as per your system.



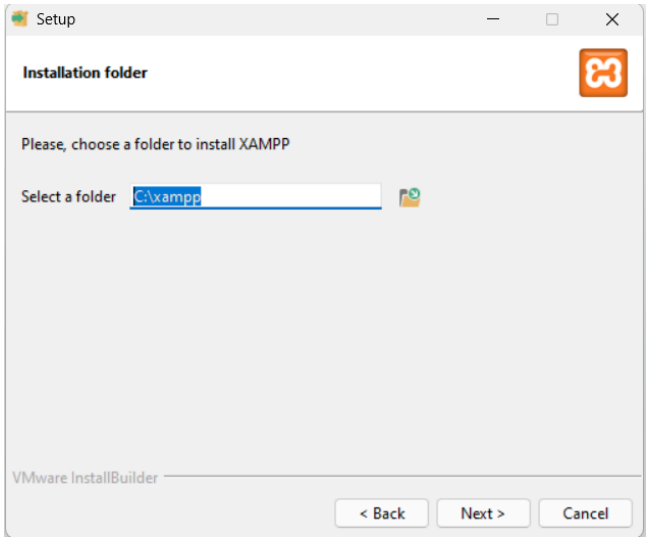
2) Xampp installer will be installed in the system.
3) Window asking to give permission will appear. Click pn ‘Yes’.
4) A window will appear for setup. Click on ‘Next’.



5) Keep the default settings and click on Next.



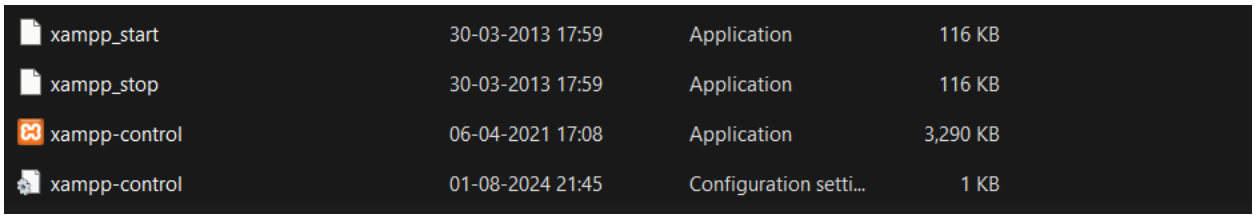
6) Choose the folder path.



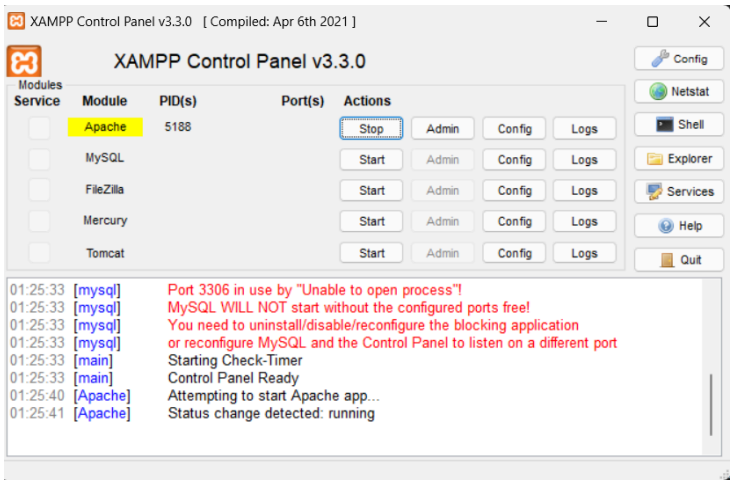
7) Then the XAMPP installation will be done.
8) Locate the folder and then locate the ‘htdocs’ folder in the xampp folder.

9) Create a test.php file in the htdocs folder and write php code.

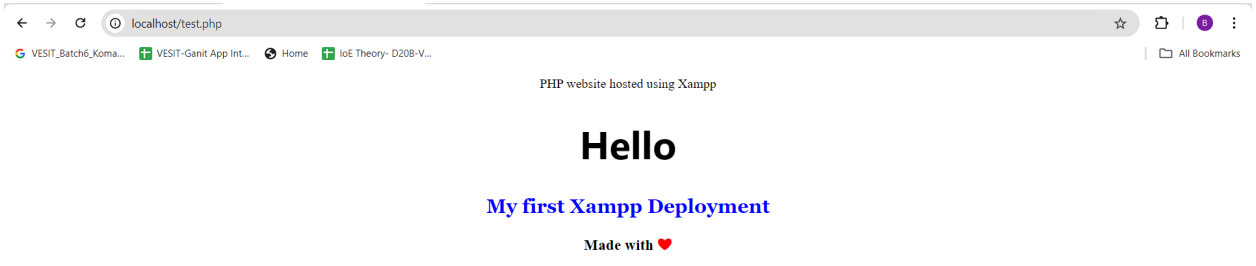
10) Now go to the xampp contol panel in the xampp folder.



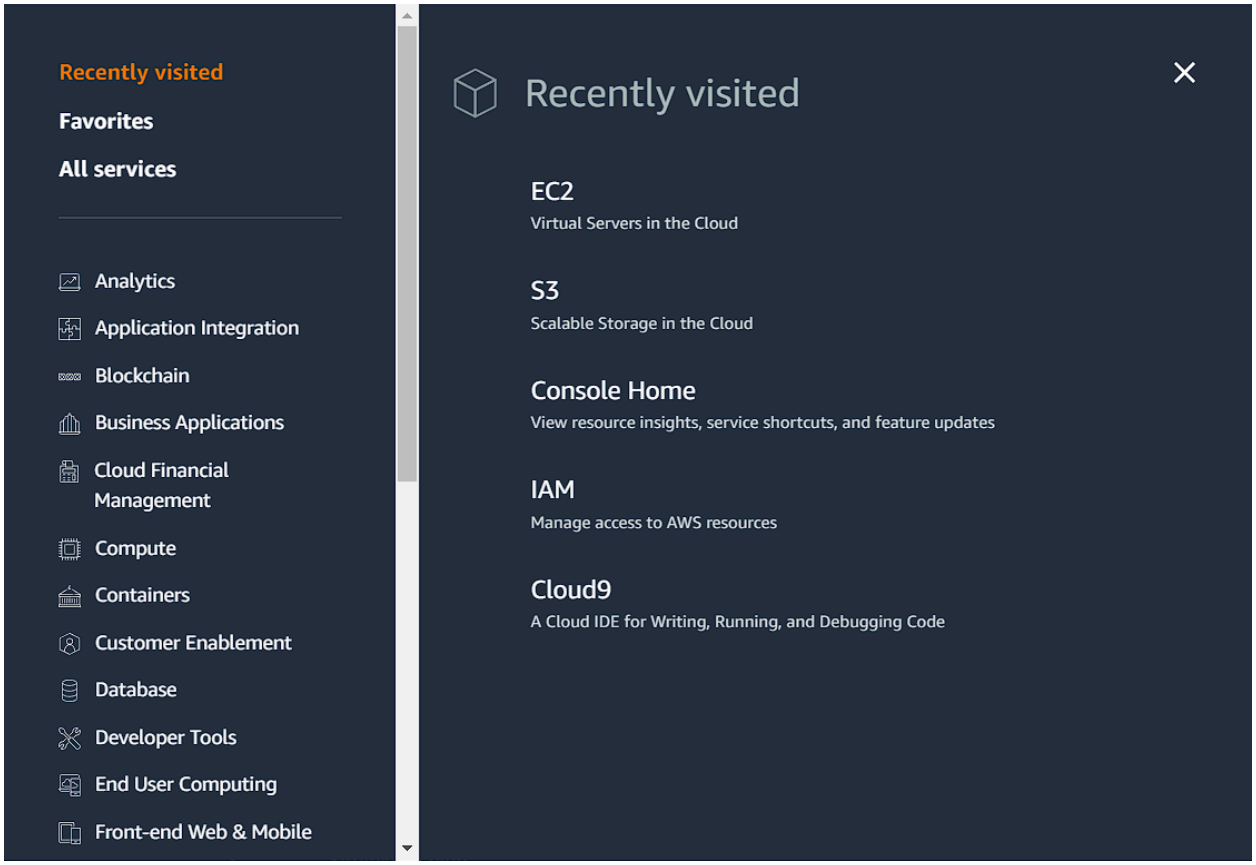
11) Start the Apache server.

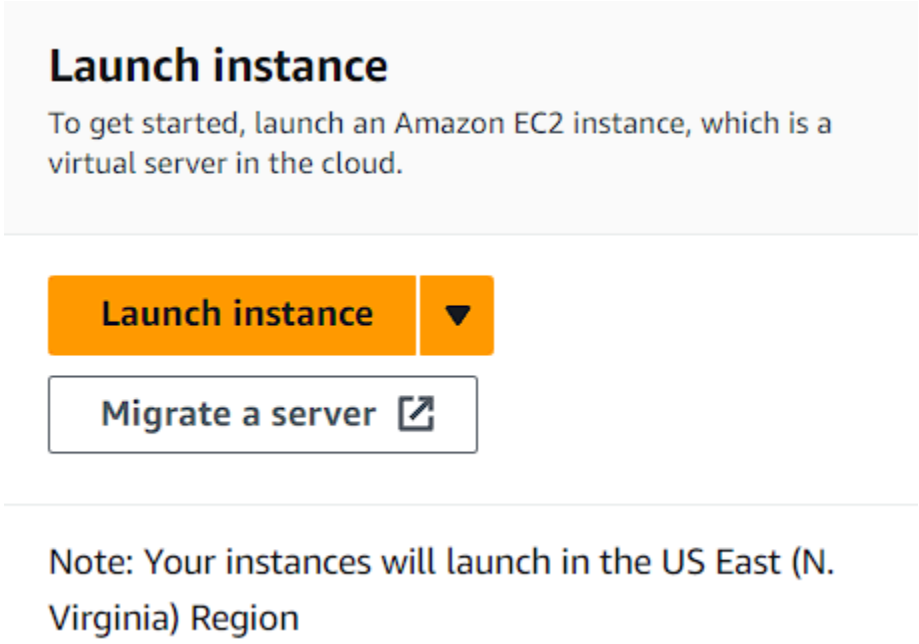


12) After strating the service, got to “localhost/file_name”, then the output window will appear.



EC2 Instance





Launch an instance Info

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

Name and tags Info

Name

AWS Test Server

Add additional tags

Application and OS Images (Amazon Machine Image) Info

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

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

Recents | Quick Start

Amazon Linux | macOS | Ubuntu | Windows | Red Hat | SUSE Li

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type — Free tier eligible

ami-04a81a99f5ec58529 (64-bit (x86)) / ami-0c14ff330901e49ff (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Architecture: 64-bit (x86)

AMI ID: ami-04a81a99f5ec58529 Verified provider

Instance type Info | Get advice

Instance type

t2.micro — Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

Key pair (login) Info

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

Key pair name - *required*

vockey

Create new key pair

Network settings Info

Edit

Network | Info

vpc-0531204c9e29f6332

Subnet | Info

No preference (Default subnet in any availability zone)

Auto-assign public IP | Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) | Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

- (•) Create security group
- ( ) Select existing security group

We'll create a new security group called '**launch-wizard-2**' with the following rules:

- [x] Allow SSH traffic from
  Helps you connect to your instance
  Anywhere
  0.0.0.0/0
- [ ] Allow HTTPS traffic from the internet
  To set up an endpoint, for example when creating a web server
- [ ] Allow HTTP traffic from the internet
  To set up an endpoint, for example when creating a web server

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

Configure storage Info

Advanced

1x 8 GiB gp3 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

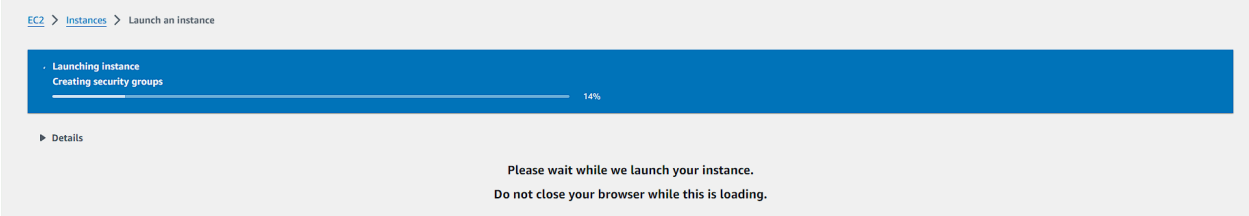
Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

Click refresh to view backup information

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems Edit



EC2 > Instances > i-033e8bf30d3d5ed91 > Connect to instance

Connect to instance Info

Connect to your instance i-033e8bf30d3d5ed91 (AWS Test Server) using any of these options

EC2 Instance Connect | Session Manager | SSH client | EC2 serial console

Port 22 (SSH) is open to all IPv4 addresses
Port 22 (SSH) is currently open to all IPv4 addresses, indicated by **0.0.0.0/0** in the inbound rule in your security group. For increased security, consider restricting access to only the EC2 Instance Connect service IP addresses for your Region: 18.206.107.24/29. Learn more.

Instance ID
i-033e8bf30d3d5ed91 (AWS Test Server)

Connection Type

- (selected) Connect using EC2 Instance Connect
  Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.
- ( ) Connect using EC2 Instance Connect Endpoint
  Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IP address
35.171.89.89

Username
Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ubuntu.

ubuntu

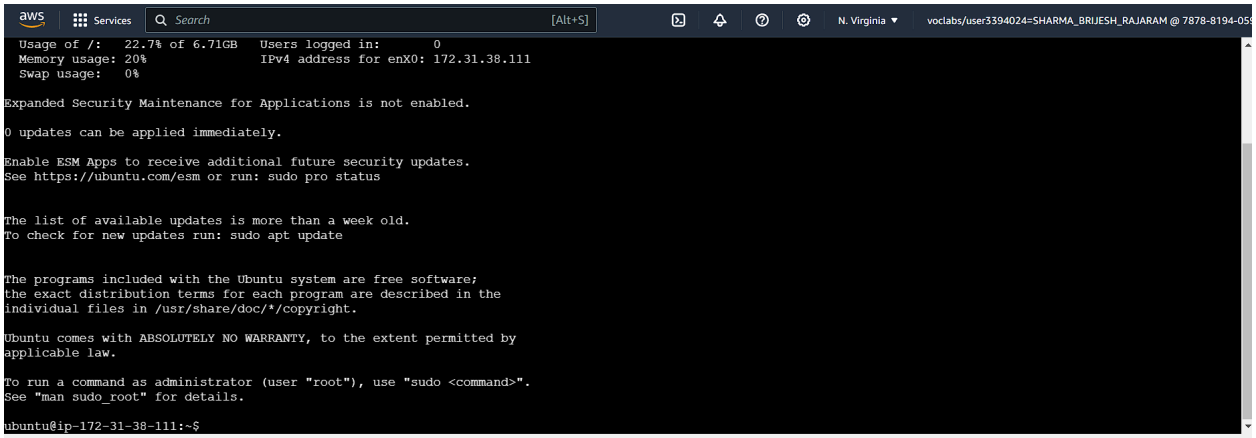
Note: In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

You have insufficient IAM permissions to connect to an instance using EC2 Instance Connect
To connect to an instance via EC2 Instance Connect, you must have an attached IAM policy that grants the following permissions:

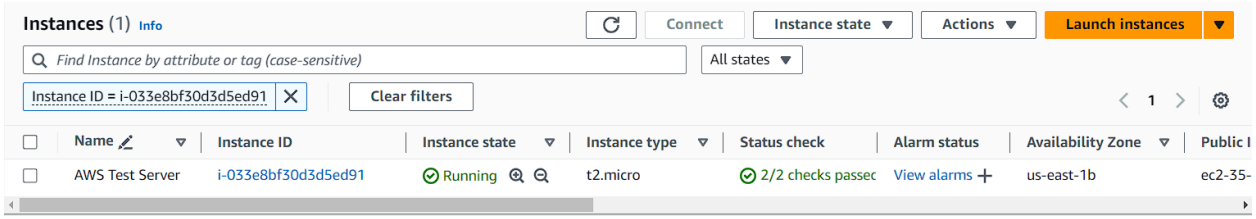
- `ec2-instance-connect:SendSSHPublicKey`
- `ec2:DescribeInstances`

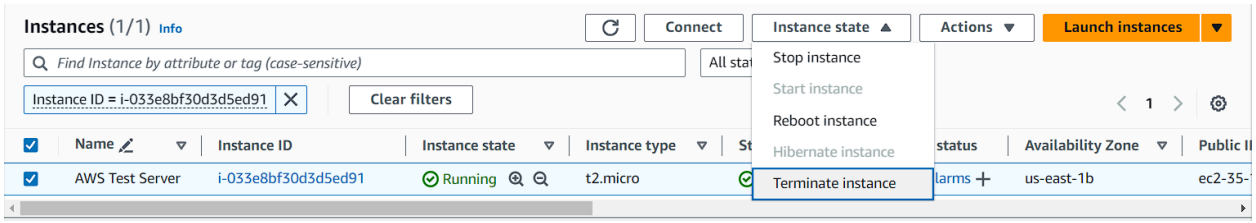
Consider restricting access to specific EC2 instances using `ec2:osuser` condition, or specific resource tag.
Visit IAM Console to verify if you have above permissions.
For more information about IAM policy examples, see Grant IAM permissions for EC2 Instance Connect.

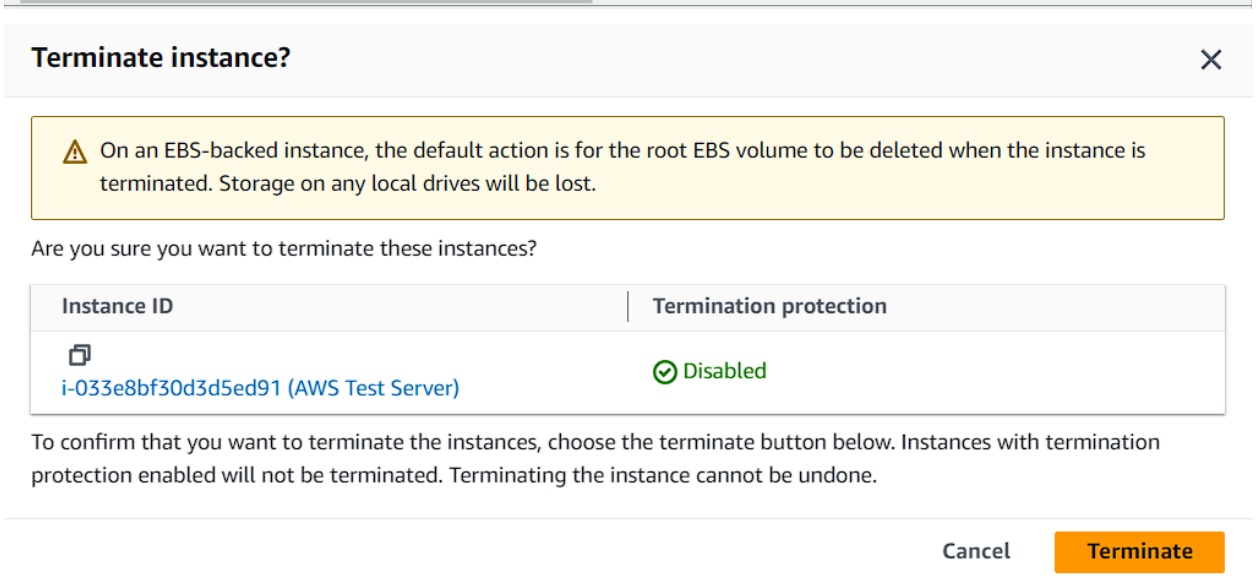
Cancel | Connect

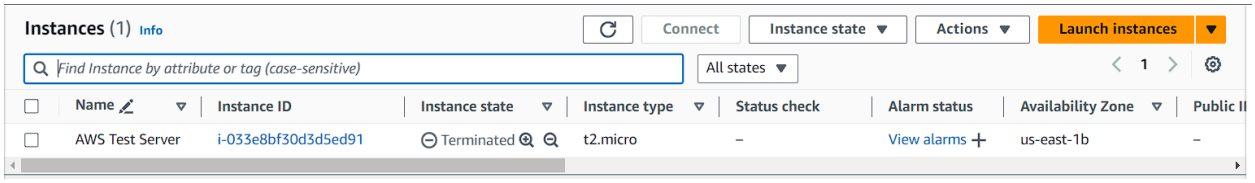



```
ubuntu@ip-172-31-38-111:~$ ping www.google.com
PING www.google.com (142.251.167.106) 56(84) bytes of data.
64 bytes from ww-in-f106.1e100.net (142.251.167.106): icmp_seq=1 ttl=58 time=2.22 ms
64 bytes from ww-in-f106.1e100.net (142.251.167.106): icmp_seq=2 ttl=58 time=2.24 ms
64 bytes from ww-in-f106.1e100.net (142.251.167.106): icmp_seq=3 ttl=58 time=2.26 ms
64 bytes from ww-in-f106.1e100.net (142.251.167.106): icmp_seq=4 ttl=58 time=2.38 ms
64 bytes from ww-in-f106.1e100.net (142.251.167.106): icmp_seq=5 ttl=58 time=2.31 ms
64 bytes from ww-in-f106.1e100.net (142.251.167.106): icmp_seq=6 ttl=58 time=2.26 ms
^C
--- www.google.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5007ms
rtt min/avg/max/mdev = 2.220/2.276/2.376/0.052 ms
ubuntu@ip-172-31-38-111:~$ vmstat
procs -----------memory---------- ---swap-- -----io---- -system-- -------cpu-------
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st gu
 2  0      0 504660  18052 288972    0    0   381   298  141    1  2  1 95  1  2  0
ubuntu@ip-172-31-38-111:~$ df
Filesystem     1K-blocks    Used Available Use% Mounted on
/dev/root        7034376 1609612   5408380  23% /
tmpfs             490212       0    490212   0% /dev/shm
tmpfs             196088     868    195220   1% /run
tmpfs               5120       0      5120   0% /run/lock
/dev/xvda16       901520   76972    761420  10% /boot
/dev/xvda15       106832    6246    100586   6% /boot/efi
tmpfs              98040      12     98028   1% /run/user/1000
ubuntu@ip-172-31-38-111:~$ mkdir test
ubuntu@ip-172-31-38-111:~$ touch e.txt
ubuntu@ip-172-31-38-111:~$ ls
e.txt  test
ubuntu@ip-172-31-38-111:~$ history
    1  ping
    2  ping www.google.com
    3  vmstat
    4  df
    5  mkdir test
    6  touch e.txt
    7  ls
    8  history
```


Instances (1) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance ID = i-033e8bf30d3d5ed91
Clear filters
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public I
AWS Test Server
i-033e8bf30d3d5ed91
Running
t2.micro
2/2 checks passec
View alarms
us-east-1b
ec2-35-

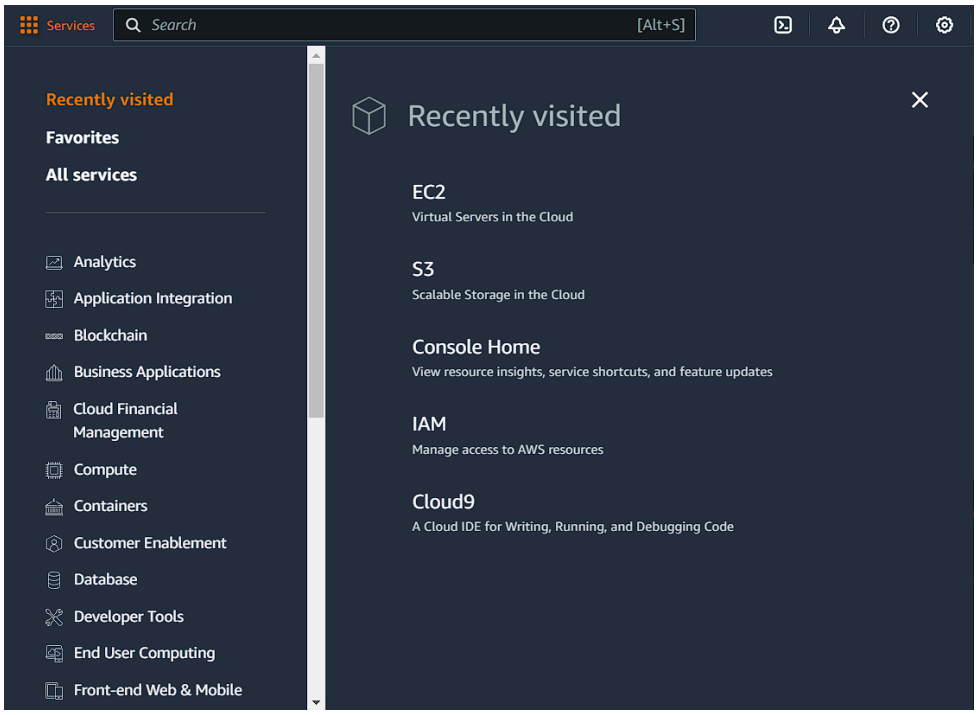

Instances (1/1) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
Instance ID = i-033e8bf30d3d5ed91
Clear filters
1
Stop instance
Start instance
Reboot instance
Hibernate instance
Terminate instance
Name
Instance ID
Instance state
Instance type
Availability Zone
AWS Test Server
i-033e8bf30d3d5ed91
Running
t2.micro
us-east-1b


Terminate instance?
On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.
Are you sure you want to terminate these instances?
Instance ID
Termination protection
i-033e8bf30d3d5ed91 (AWS Test Server)
Disabled
To confirm that you want to terminate the instances, choose the terminate button below. Instances with termination protection enabled will not be terminated. Terminating the instance cannot be undone.
Cancel
Terminate

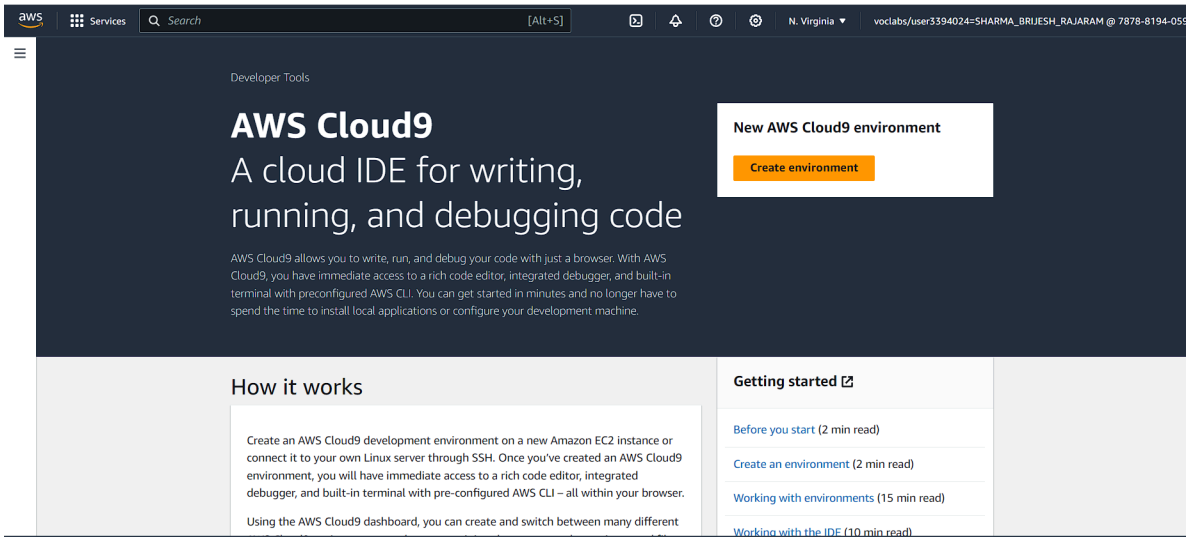

Instances (1) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public I
AWS Test Server
i-033e8bf30d3d5ed91
Terminated
t2.micro
View alarms
us-east-1b

Cloud 9 - IDE

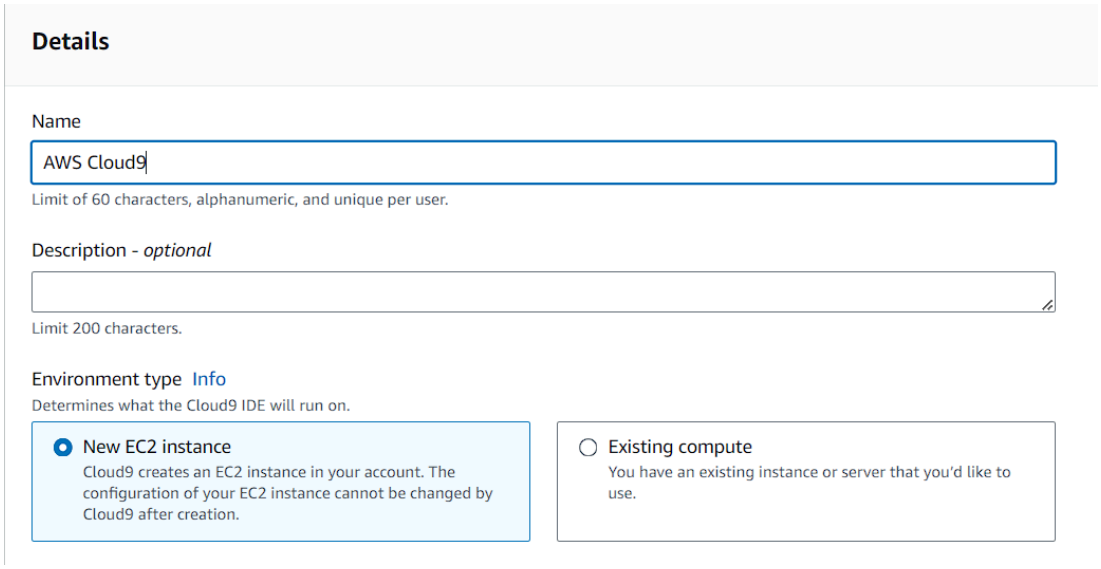
1) Navigate to developer tools -> Cloud9 and start creating Cloud9 environment.



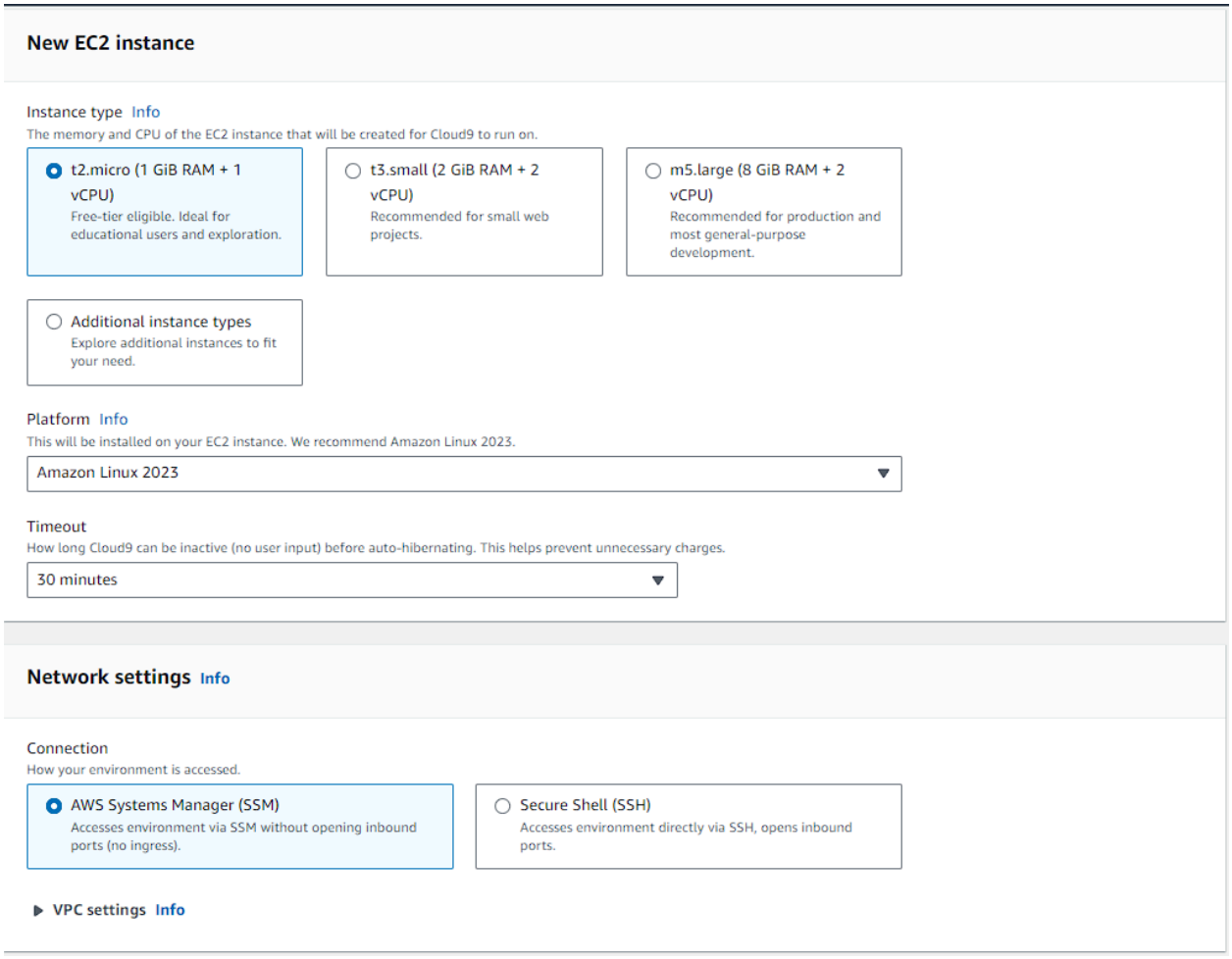
2) Click on Create Enviornment and start creating the environment



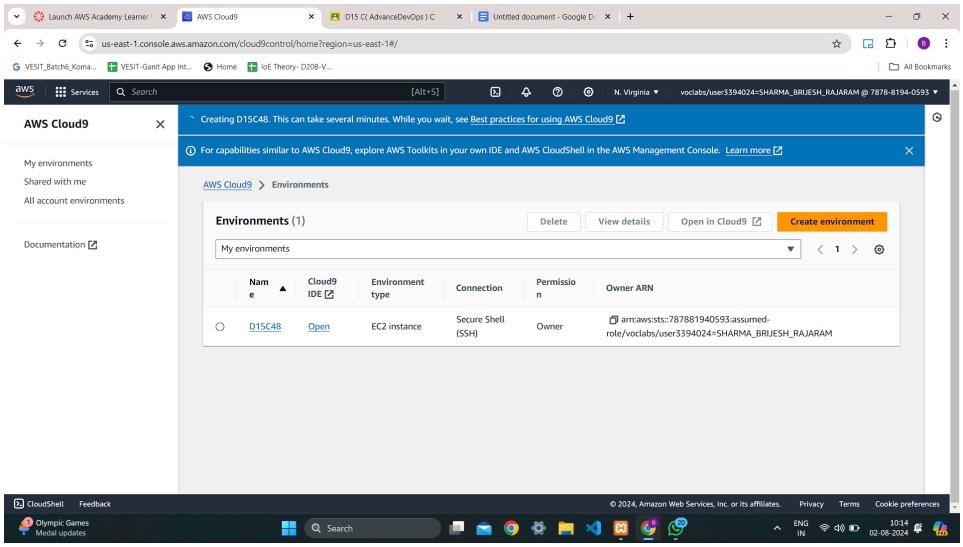
3) Name the environment and select new EC2 instance.

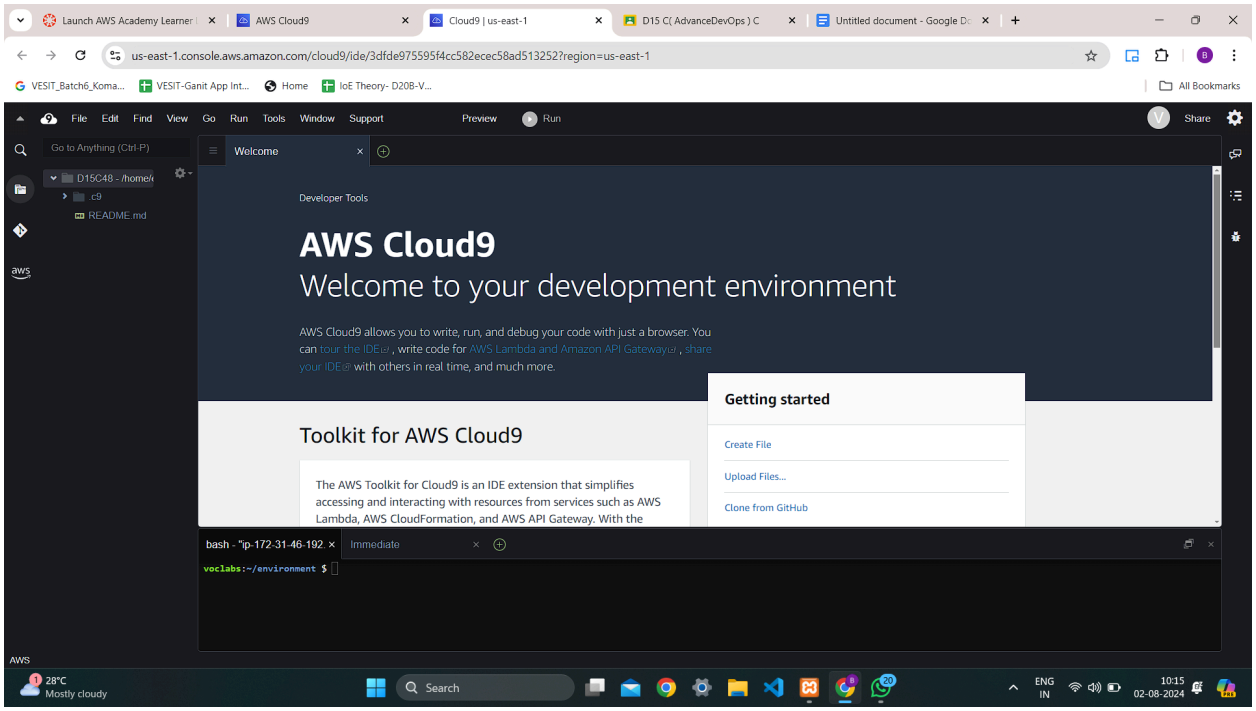


4) Keep the options default and proceed

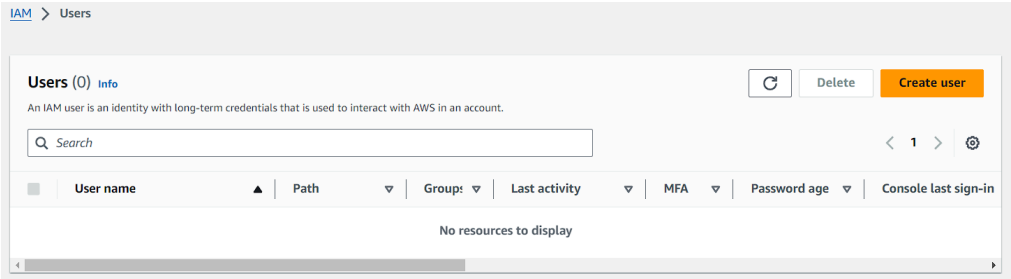


5) Environment created successfully.

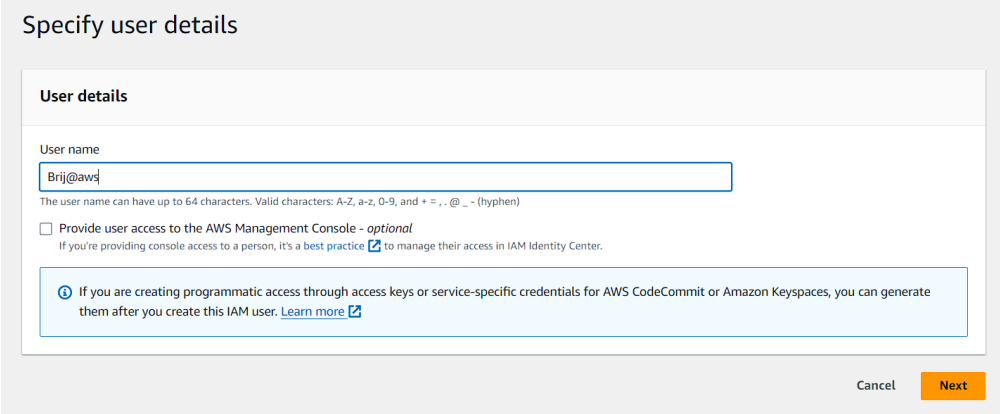




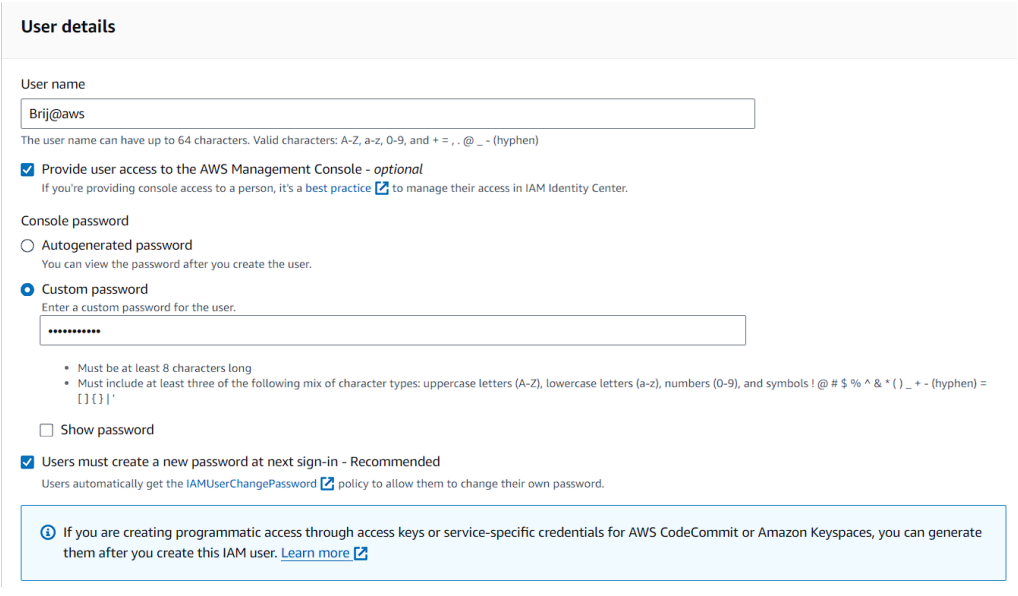
6) Create user using the IAM.



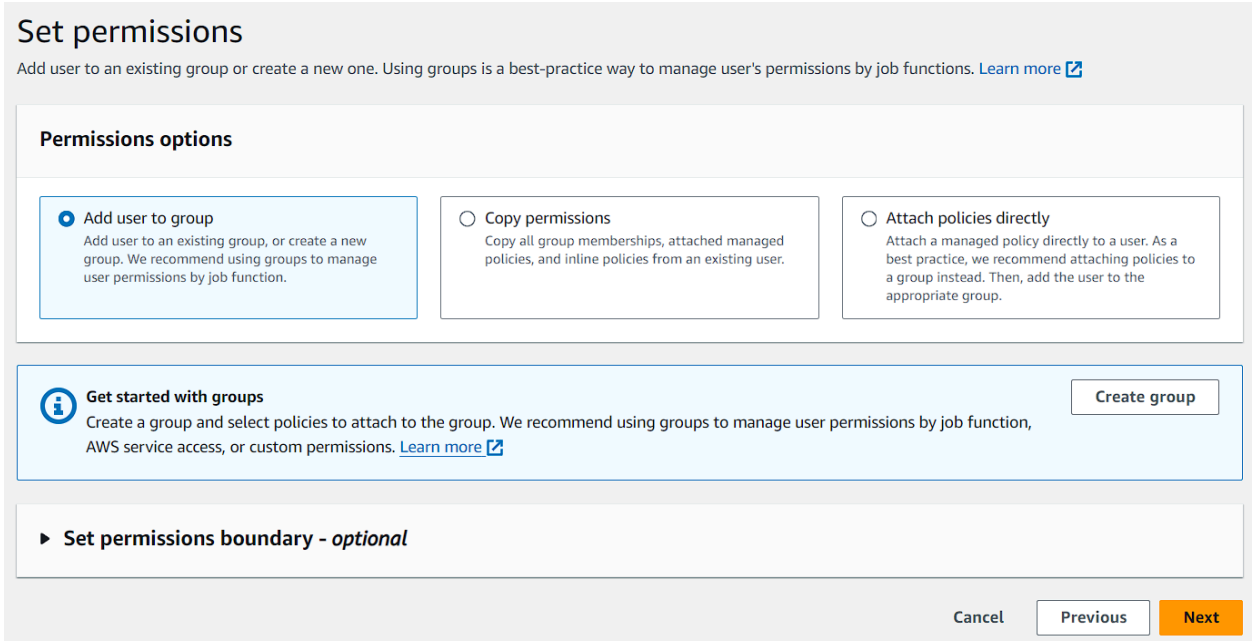
7) Add the username



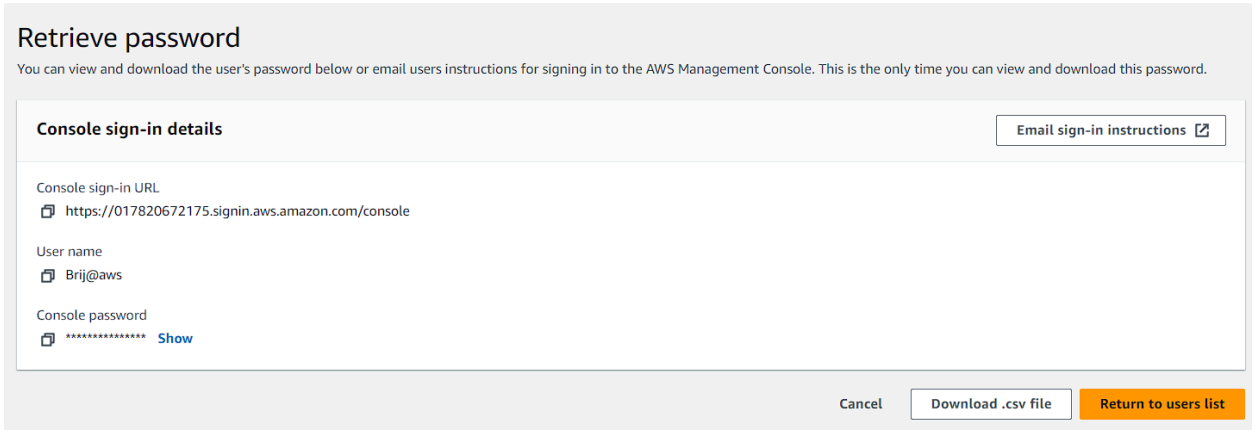
8) Add the remaining user details and provide access to the AWS Management Console



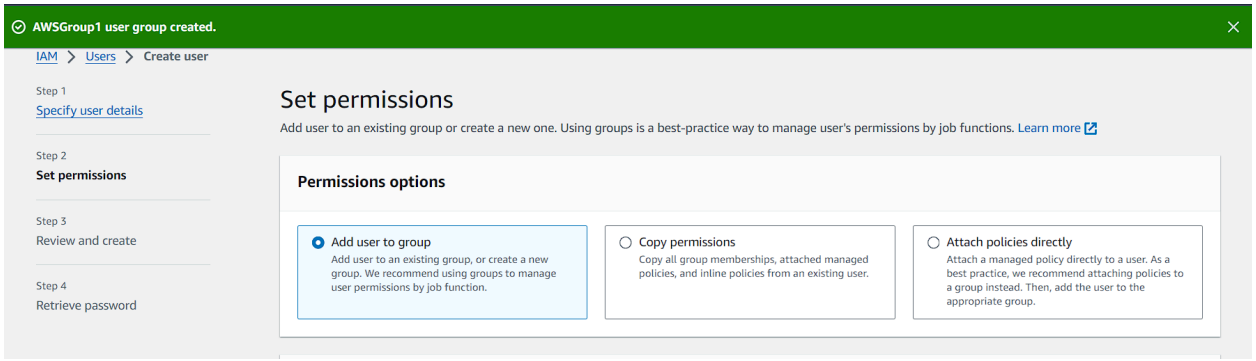
9) User created successfully and can be added to user groups.



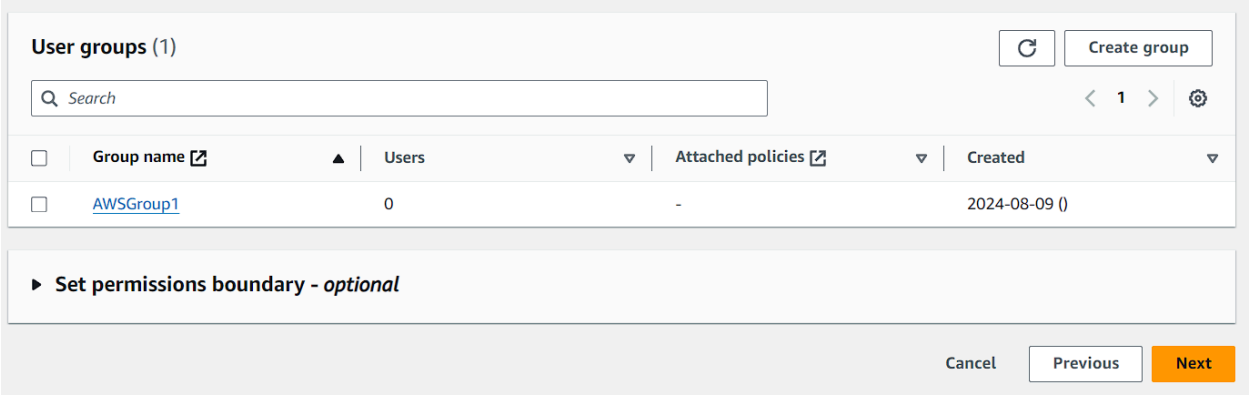
10) User credentials can be downloaded.



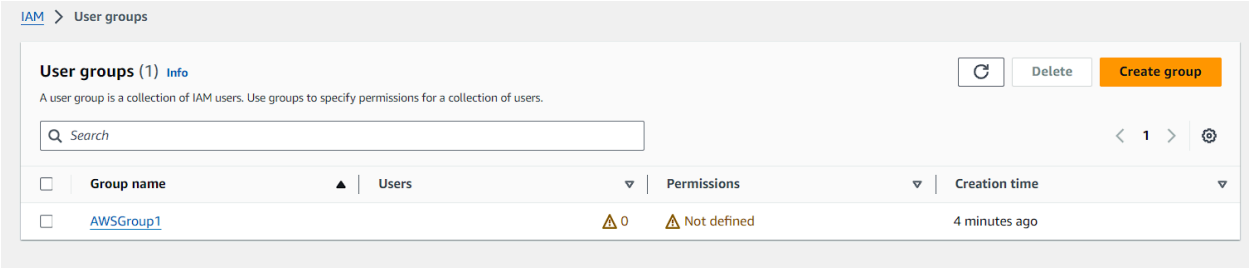
11) Add user to group if group exists else create a new group.



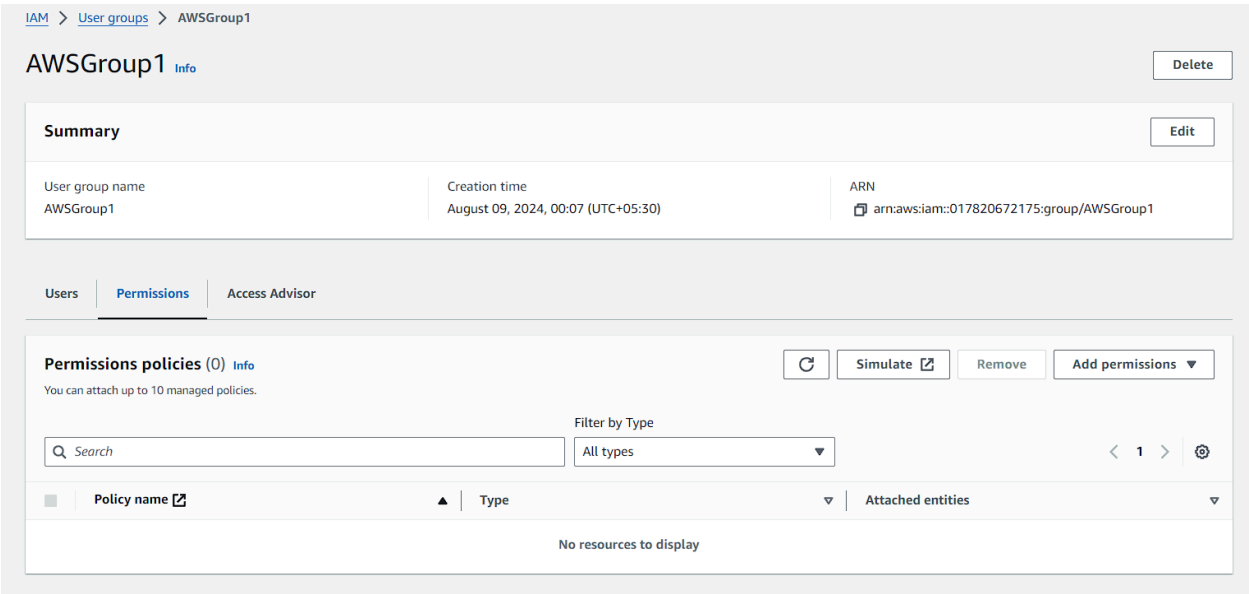
12) Write the user group name and proceed.



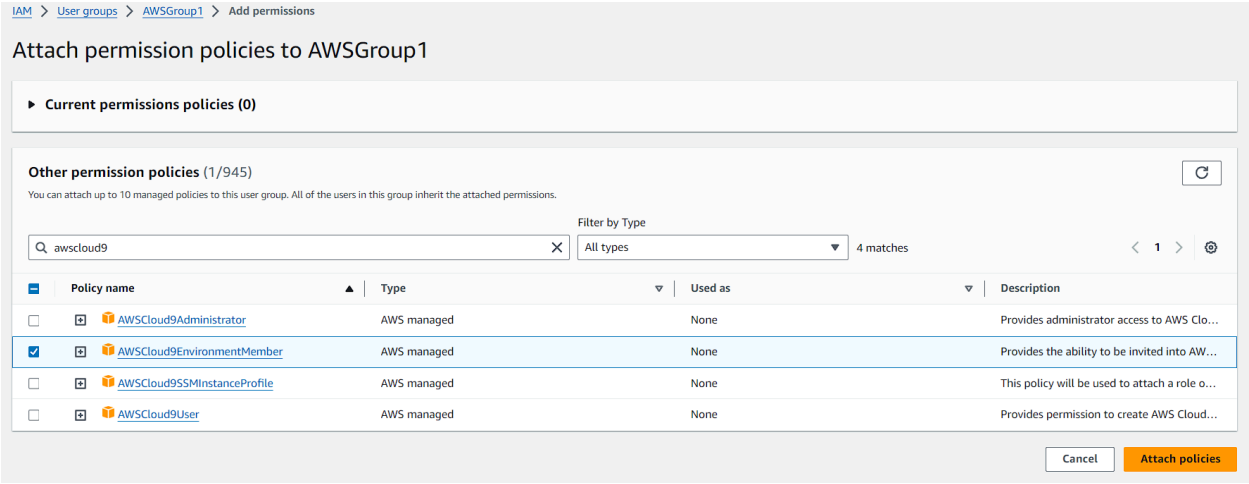
13) Click on *group_name.*



14) Go to Add permissions and click on Add Permissions



15) On attach policies, select AWSCloud9EnvironmentMember and click on Attach policies.



16) User group is created successfully.