



**Xavier Institute of Engineering**  
Mahim, Mumbai 400016

**Department of Information Technology**

(Affiliated to University of Mumbai)

(Academic Year: 2022-23)

Semester: V)

Class: TEIT

Subject: Advance DevOps Lab (ITL504)

Experiment No.	1
Title	To explain the benefits of Cloud Infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration.
Lab Outcome Achieved	1
Date of Performance	02/08/2022
Date of submission	10/08/2022
Roll No.	13
Name of the Student	Harsh Dalvi

Rubrics used for Laboratory Evaluation:

	Below Expectation	Average	Good
Knowledge (3)	1	2	3
Performance (3)	1	2	3
Content & Neatness of Documentation(2)	0	1	2
Punctuality & Submission on Time (2)	0	1	2

Marks Obtained: \_\_\_\_\_

Signature of Faculty: \_\_\_\_\_

## Experiment No. 1

**Aim:** To explain the benefits of Cloud Infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration. (LO1)

### Theory:

#### Definition of Cloud Computing?

The cloud enables users to access the same files and applications from almost any device, because the computing and storage takes place on servers in a data centre, instead of locally on the user device. Cloud computing is the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the Internet (“the cloud”) to offer faster innovation, flexible resources, and economies of scale. You typically pay only for cloud services you use, helping lower your operating costs, run your infrastructure more efficiently and scale as your business needs change.

#### Five Essential Characteristics of Cloud Computing:

- 1. On-demand self-service:** The Cloud computing services does not require any human administrators, user themselves are able to provision, monitor and manage computing resources as needed. Cloud computing resources can be provisioned without human interaction from the service provider.
- 2. Broad network access:** The Computing services are generally provided over standard networks and heterogeneous devices. Cloud computing resources are available over the network and can be accessed by diverse customer platforms. In other words, cloud services are available over a network—ideally high broadband communication link—such as the internet, or in the case of a private clouds it could be a local area network (LAN).
- 3. Multi-tenancy and resource pooling:** Cloud computing resources are designed to support a multi-tenant model. Multi-tenancy allows multiple customers to share the same applications or the same physical infrastructure while retaining privacy and security over their information. Resource pooling means that multiple customers are serviced from the same physical resources. Providers’ resource pool should be very large and flexible enough to service multiple client requirements and to provide for economy of scale.
- 4. Rapid elasticity and scalability:** One of the great things about cloud computing is the ability to quickly provision resources in the cloud as manufacturing organizations need them. And then to remove them when they don’t need them. Cloud computing resources can scale up or down rapidly and, in some cases, automatically, in response to business demands. Elasticity is a landmark of cloud computing and it implies that manufacturing organizations can rapidly provision and deprovision any of the cloud computing resources.
- 5. Measured service:** Cloud computing resources usage is metered and manufacturing organizations pay accordingly for what they have used. Resource utilization can be optimized by leveraging charge-per-use capabilities. This means that cloud resource usage—whether virtual server instances that are running or storage in the cloud—gets monitored, measured and reported by the cloud service provider.

### **Cloud Deployment Models:**

The cloud deployment model identifies the specific type of cloud environment based on ownership, scale, and access, as well as the cloud's nature and purpose. The location of the servers you're utilizing and who controls them are defined by a cloud deployment model.

Different types of cloud computing deployment models are:

1. **Public Cloud:** The public cloud makes it possible for anybody to access systems and services. The public cloud may be less secure as it is open for everyone. The public cloud is one in which cloud infrastructure services are provided over the internet to the general people or major industry groups. Example: Google App Engine etc.
2. **Private Cloud:** The private cloud deployment model is the exact opposite of the public cloud deployment model. It's a one-on-one environment for a single user (customer). There is no need to share your hardware with anyone else. The private cloud gives the greater flexibility of control over cloud resources
3. **Hybrid cloud:** By bridging the public and private worlds with a layer of proprietary software, hybrid cloud computing gives the best of both worlds. With a hybrid solution, you may host the app in a safe environment while taking advantage of the public cloud's cost savings. Organizations can move data and applications between different clouds using a combination of two or more cloud deployment methods, depending on their needs.
4. **Community cloud:** It allows systems and services to be accessible by a group of organizations. It is a distributed system that is created by integrating the services of different clouds to address the specific needs of a community, industry, or business. The infrastructure of the community could be shared between the organization which has shared concerns or tasks.

### **Cloud computing services and different cloud services providers.**

The three main types of cloud computing include Infrastructure as a Service, Platform as a Service, and Software as a Service. Each type of cloud computing provides different levels of control, flexibility, and management so that you can select the right set of services for your needs.

1. **Infrastructure as a Service (IaaS):** IaaS contains the basic building blocks for cloud IT. It typically provides access to networking features, computers (virtual or on dedicated hardware), and data storage space
2. **Platform as a Service (PaaS):** PaaS removes the need for you to manage underlying infrastructure (usually hardware and operating systems), and allows you to focus on the deployment and management of your applications.
3. **Software as a Service (SaaS):** SaaS provides you with a complete product that is run and managed by the service provider. In most cases, people referring to SaaS are referring to end-user applications (such as web-based email).

### **There are many Cloud Service providers in the market:**

1. **Amazon Web Services (AWS):** Amazon Web Services is a cloud computing platform that provides services such as compute power, database storage, content delivery and many other functions which will help to integrate a business.
2. **Server Space:** These are Cloud Servers with Windows & Linux OS. At Server Space Cloud Servers, you can choose your own custom configurations, spin up your VM in 40 sec, and

change the configuration at any time and pay as you go. Unlimited traffic, High-end performance and 24/7/365 human tech support.

3. **Microsoft Azure:** Microsoft Azure is a cloud computing service which is used for building testing deploying and managing the application. It is private as well as a public cloud platform. It uses virtualization which differentiates the coupling between the operating system and CPU with the help of an abstraction layer known as a hypervisor.
4. **Google Cloud Platform:** Google cloud platform is one of the leading Cloud Computing services which are offered by Google and it runs on the same infrastructure that Google uses for its end-user products. There are various services offered by Google Cloud such as data analysis, machine learning, and data storage.
5. **IBM Cloud Services:** IBM cloud offers services such as platform as a service and infrastructure as a service. This cloud organization can deploy and access its resources such as storage networking and compute power with the help of internet. There are several tools which help the customer to draw on deep industry expertise.

#### **Uses of DevOps:**

1. The DevOps model relies on effective tooling to help teams rapidly and reliably deploy and innovate for their customers.
2. These tools automate manual tasks, help teams manage complex environments at scale, and keep engineers in control of the high velocity that is enabled by DevOps.
3. It is used to build security into your software development life cycle at speed and scale without sacrificing safety, while minimizing risks, ensuring compliance, and reducing friction and costs.

#### **Use of advance DevOps:**

1. Provisioning, configuring and managing the infrastructure on AWS, Azure, and GCP for our customers
2. Using Docker containers for continuous apps delivery
3. Using Kubernetes for fast and reliable container management
4. Using Jenkins, Gitlab CI and CircleCI for building continuous software delivery pipelines
5. Using Prometheus & Grafana, Sumologic and ELK Stack for in-depth service monitoring and logging

#### **Cloud9:**

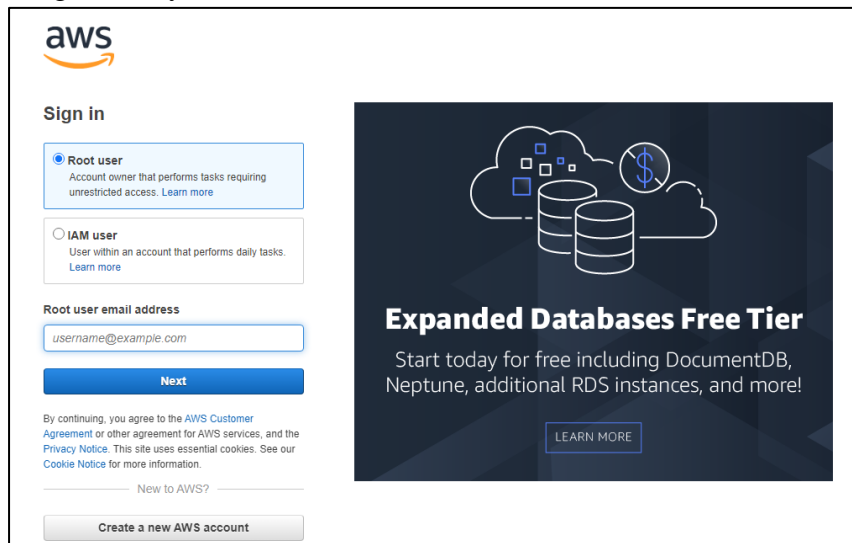
AWS Cloud9 is a cloud-based integrated development environment (IDE) that lets you write, run, and debug your code with just a browser. It includes a code editor, debugger, and terminal. Cloud9 comes pre-packaged with essential tools for popular programming languages, including JavaScript, Python, PHP, and more, so you don't need to install files or configure your development machine to start new projects. Since your Cloud9 IDE is cloud-based, you can work on your projects from your office, home, or anywhere using an internet-connected machine. Cloud9 also provides a seamless experience for developing server less applications enabling you to easily define resources, debug, and switch between local and remote execution of server less applications. With Cloud9, you can quickly share your development environment with your team, enabling you to pair program and track each other's inputs in real time.

Name: Harsh Dalvi  
Roll No: 13

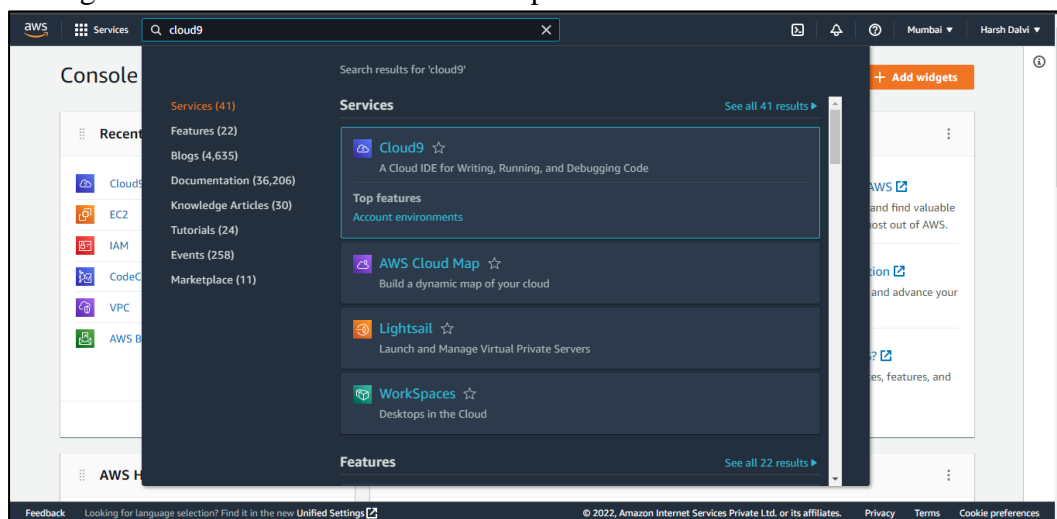
Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B

## Steps to perform this experiment:

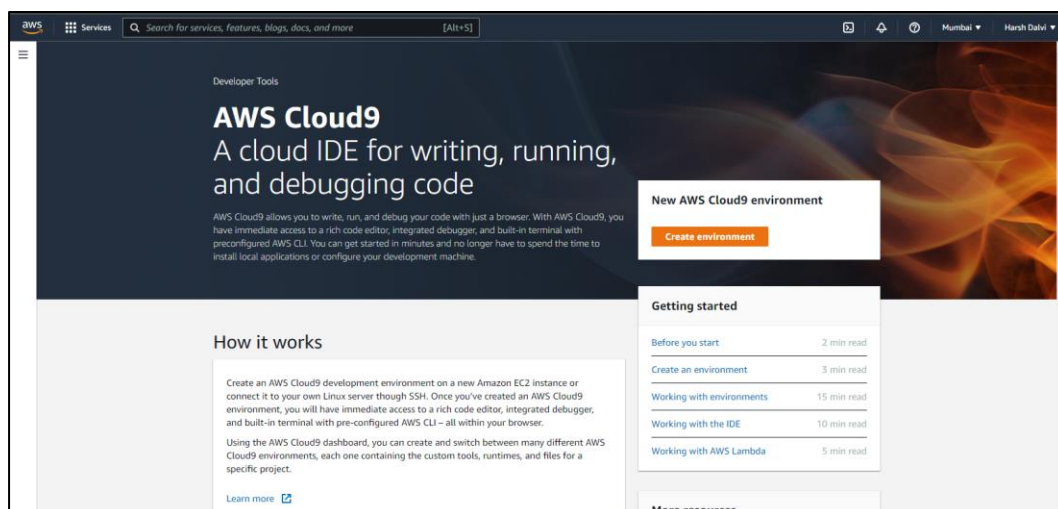
1. Log in into your AWS account. (IAM/Root).



2. Navigate to Cloud9 service from developer tool section as shown below.



3. Click on create environment.



Name: Harsh Dalvi  
Roll No: 13

Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B

4. Provide name for the environment (environment name), click on next.

### Name environment

#### Environment name and description

**Name**  
The name needs to be unique per user. You can update it at any time in your environment settings.

Limit: 60 characters

**Description - *Optional***  
This will appear on your environment's card in your dashboard. You can update it at any time in your environment settings.

Write a short description for your environment

Limit: 200 characters

Cancel Next step

5. Keep all the default setting as shown below.

### Configure settings

#### Environment settings

**Environment type** [Info](#)  
Run your environment in a new EC2 instance or an existing server. With EC2 instances, you can connect directly through Secure Shell (SSH) or connect via AWS Systems Manager (without opening inbound ports).

☒ **Create a new EC2 instance for environment (direct access)**  
Launch a new instance in this region that your environment can access directly via SSH.

☐ **Create a new no-ingress EC2 instance for environment (access via Systems Manager)**  
Launch a new instance in this region that your environment can access through Systems Manager.

☐ **Create and run in remote server (SSH connection)**  
Configure the secure connection to the remote server for your environment.

**Instance type**

☒ **t2.micro (1 GiB RAM + 1 vCPU)**  
Free-tier eligible. Ideal for educational users and exploration.

☐ **t3.small (2 GiB RAM + 2 vCPU)**  
Recommended for small-sized web projects.

☐ **m5.large (8 GiB RAM + 2 vCPU)**  
Recommended for production and general-purpose development.

☐ **Other instance type**  
Select an instance type.

t3.nano

### Platform

☒ **Amazon Linux 2 (recommended)**

☐ Amazon Linux AMI

☐ Ubuntu Server 18.04 LTS

**Cost-saving setting**  
Choose a predetermined amount of time to auto-hibernate your environment and prevent unnecessary charges. We recommend a hibernation settings of half an hour of no activity to maximize savings.

After 30 minutes (default)

**IAM role**  
AWS Cloud9 creates a service-linked role for you. This allows AWS Cloud9 to call other AWS services on your behalf. You can delete the role from the AWS IAM console once you no longer have any AWS Cloud9 environments. [Learn more](#)

AWSServiceRoleForAWSCloud9

**▼ Network settings (advanced)**

**Network (VPC)**  
Launch your EC2 instance into an existing Amazon Virtual Private Cloud (VPC) or create a new one. To allow the AWS Cloud9 environment to connect to its EC2 instance, attach an internet gateway (IGW) to your new VPC.

vpc-09d41b4578f6a5941 (default)

↻ Create new VPC

**Subnet**  
Select a public subnet in which the EC2 instance is created. (For a private subnet, you must create an environment that connects to its instance via Systems Manager.)

subnet-05af1ba4e2430056b | Default in ap-south-1a

↻ Create new subnet

No tags associated with the resource.

Add new tag

Name: Harsh Dalvi  
Roll No: 13

Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B

6. Review the environment and setting and click on create environment.

### Review

#### Environment name and settings

Name  
WebAppEnv

Description  
No description provided

Environment type  
EC2

Instance type  
t2.micro

Subnet  
subnet-05af1ba4e2430056b

Platform  
Amazon Linux 2 (recommended)

Cost-saving settings  
After 30 minutes (default)

IAM role  
AWSServiceRoleForAWSCloud9 (generated)

7. Till that time open the IAM Identity and access management in order to add user in other tab.

The screenshot shows the AWS IAM dashboard. On the left is a navigation menu with options like Dashboard, Access management, Users, Roles, Policies, Identity providers, Account settings, Access reports, and Related consoles. The main content area displays 'IAM dashboard' with a 'Security recommendations' section showing 'Add MFA for root user' and 'Root user has no active access keys'. Below this is a summary of IAM resources: 0 User groups, 0 Users, 3 Roles, 0 Policies, and 0 Identity providers. A 'What's new' section lists recent updates. On the right, the 'AWS Account' section shows the Account ID (052585512669) and Account Alias (052585512669). There are also 'Quick Links' and 'Tools' sections.

The screenshot shows the 'Users' page in the AWS IAM console. At the top, there's a notification banner about the new Users list experience. Below the navigation menu, the 'Users (0) Info' section explains that an IAM user is an identity with long-term credentials. A search bar is present with the placeholder 'Find users by username or access key'. Below the search bar is a table with columns: User name, Groups, Last activity, MFA, Password age, and Active key age. The table is currently empty, displaying 'No resources to display'.

Name: Harsh Dalvi  
Roll No: 13

Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B

8. Add User provide with password selection with auto generate password and click on auto generate tab.

**Add user**

1 2 3 4 5

**Set user details**

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name\*

[Add another user](#)

**Select AWS access type**

Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

**Select AWS credential type\***

☐ Access key - Programmatic access  
Enables an access key ID and secret access key for the AWS API, CLI, SDK, and other development tools.

☒ Password - AWS Management Console access  
Enables a password that allows users to sign-in to the AWS Management Console.

**Console password\***

☒ Autogenerated password

☐ Custom password

**Require password reset** ☐ User must create a new password at next sign-in  
Users automatically get the IAMUserChangePassword policy to allow them to change

\* Required

[Cancel](#) [Next: Permissions](#)

9. Click on create group.

**Add user**

1 2 3 4 5

**Set permissions**

[Add user to group](#) [Copy permissions from existing user](#) [Attach existing policies directly](#)

**Get started with groups**

You haven't created any groups yet. Using groups is a best-practice way to manage users' permissions by job functions, AWS service access, or your custom permissions. [Get started by creating a group.](#) [Learn more](#)

[Create group](#)

**Set permissions boundary**

[Cancel](#) [Previous](#) [Next: Tags](#)

10. Provide group name and click create group by selecting current policy.

**Add user**

1 2 3 4 5

**Create group**

Create a group and select the policies to be attached to the group. Using groups is a best-practice way to manage users' permissions by job functions, AWS service access, or your custom permissions. [Learn more](#)

Group name

[Create policy](#) [Refresh](#)

**Filter policies**  Showing 4 results

	Policy name	Type	Used as	Description
<input type="checkbox"/>	AWSCloud9Administrator	AWS managed	None	Provides administrator access to AWS Cloud9.
<input checked="" type="checkbox"/>	AWSCloud9EnvironmentMember	AWS managed	None	Provides the ability to be invited into AWS Cloud9 shared development environments.
<input type="checkbox"/>	AWSCloud9SSMInstanceProfile	AWS managed	None	This policy will be used to attach a role on a InstanceProfile which will allow Cloud9 to ...
<input type="checkbox"/>	AWSCloud9User	AWS managed	None	Provides permission to create AWS Cloud9 development environments and to manag...

[Cancel](#) [Create group](#)

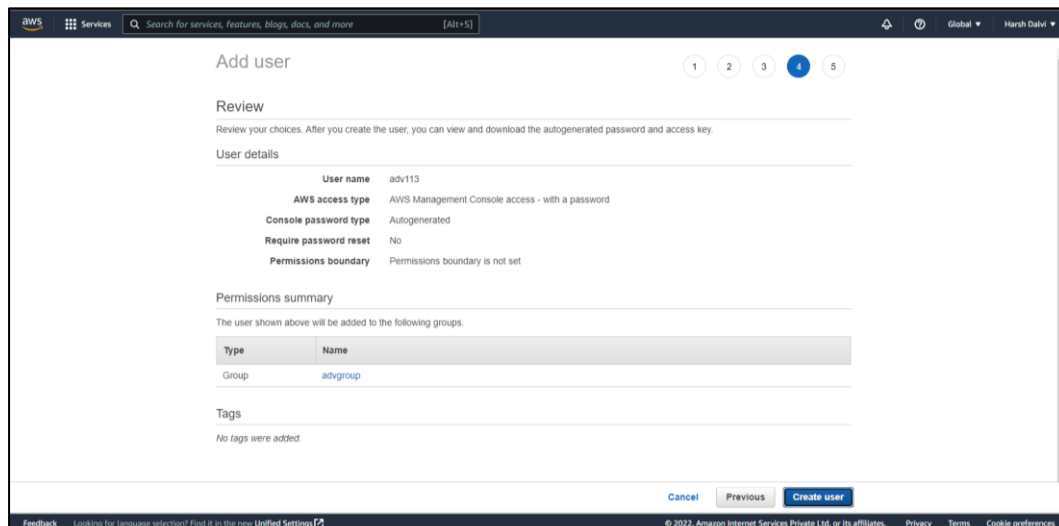
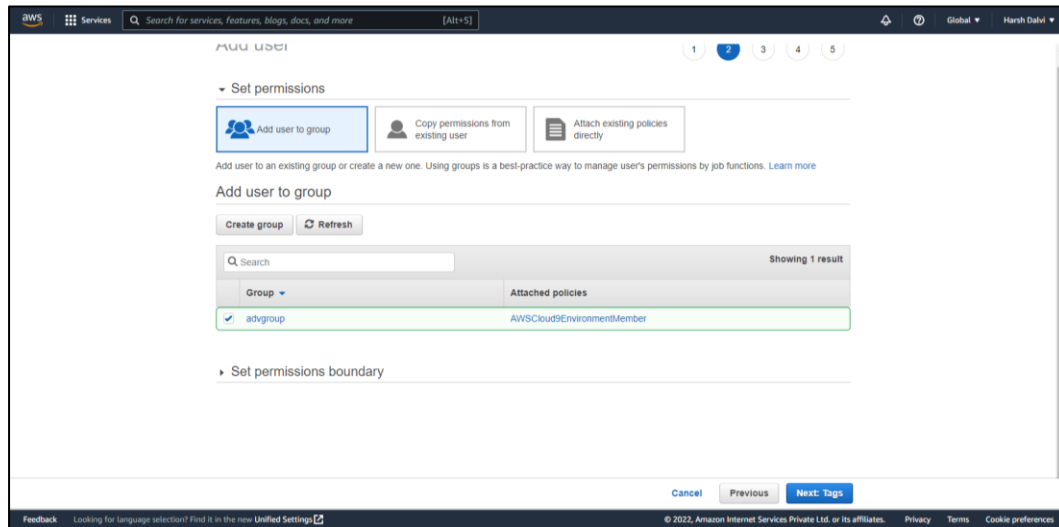
[Cancel](#) [Previous](#) [Next: Tags](#)



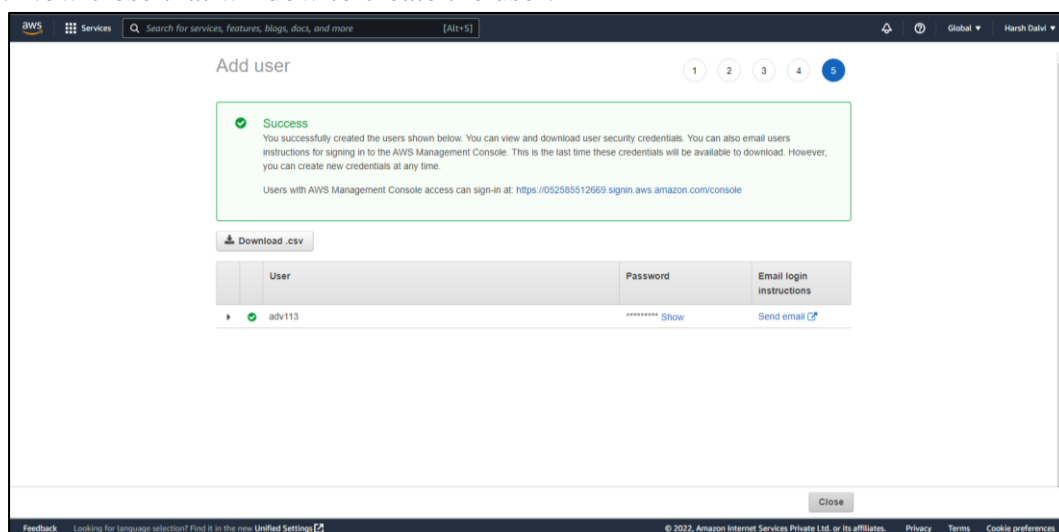
Name: Harsh Dalvi  
Roll No: 13

Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B

11. After the group is created click on next if you want provide tag else click on review for user setting and click on create user as shown below.



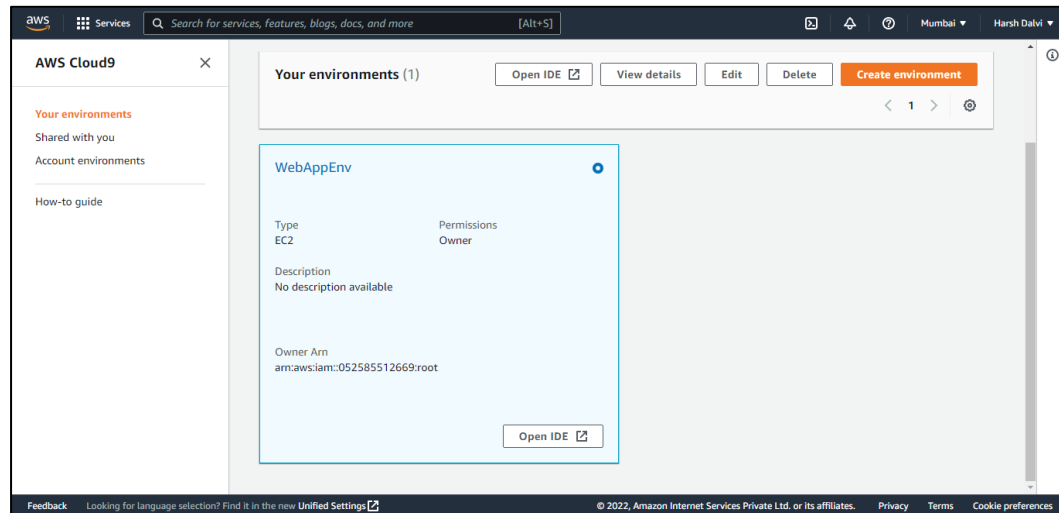
12. Now close that window to create the user.



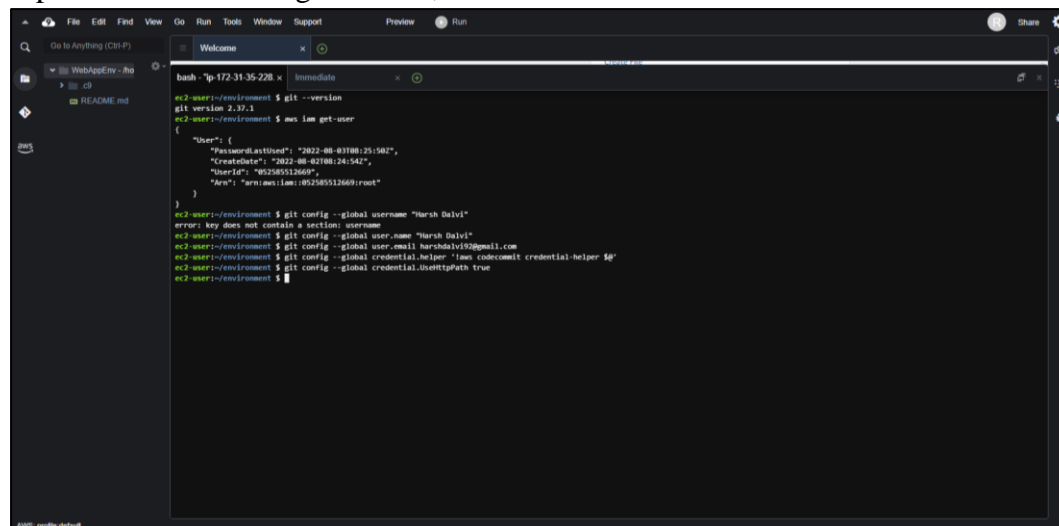
Name: Harsh Dalvi  
Roll No: 13

Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B

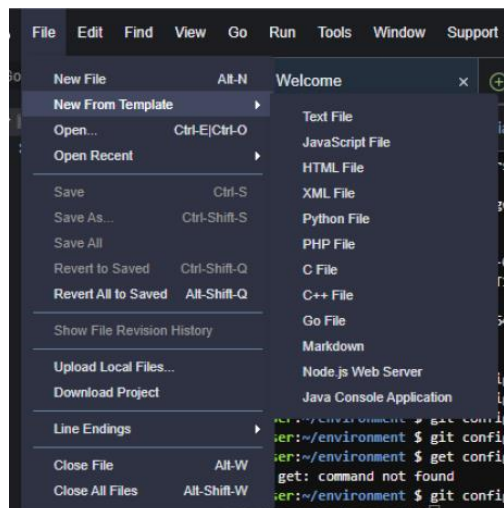
13. Now move towards Cloud9 IDE environment tab as shown below.



14. If you check at bottom side cloud9 IDE also giving you an AWS CLI for command operations. To check git version, IAM user details and so on.

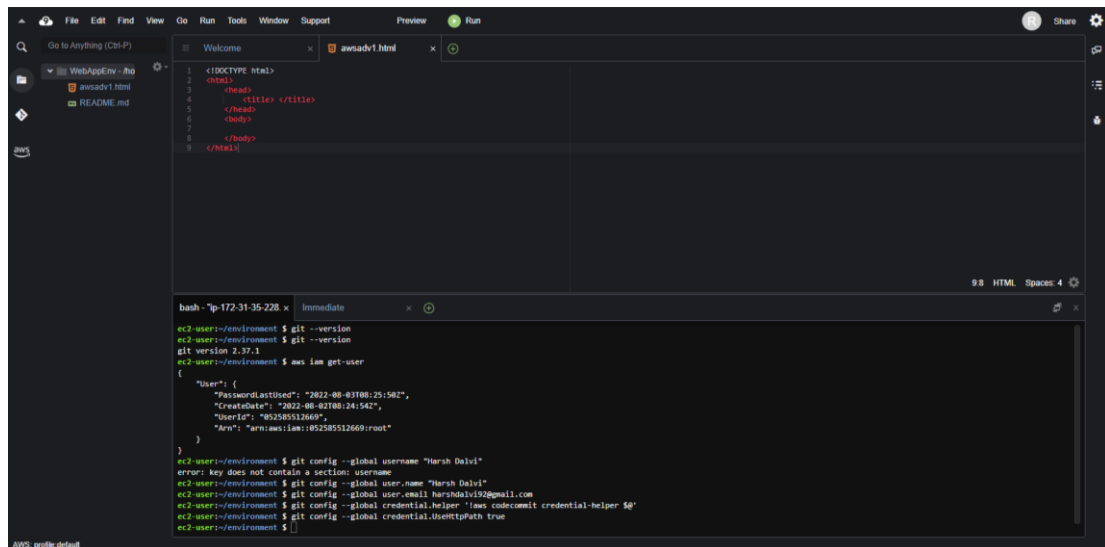


15. Now setup collaborative environment. Create HTML file from the file tab. Edit HTML file and save it.



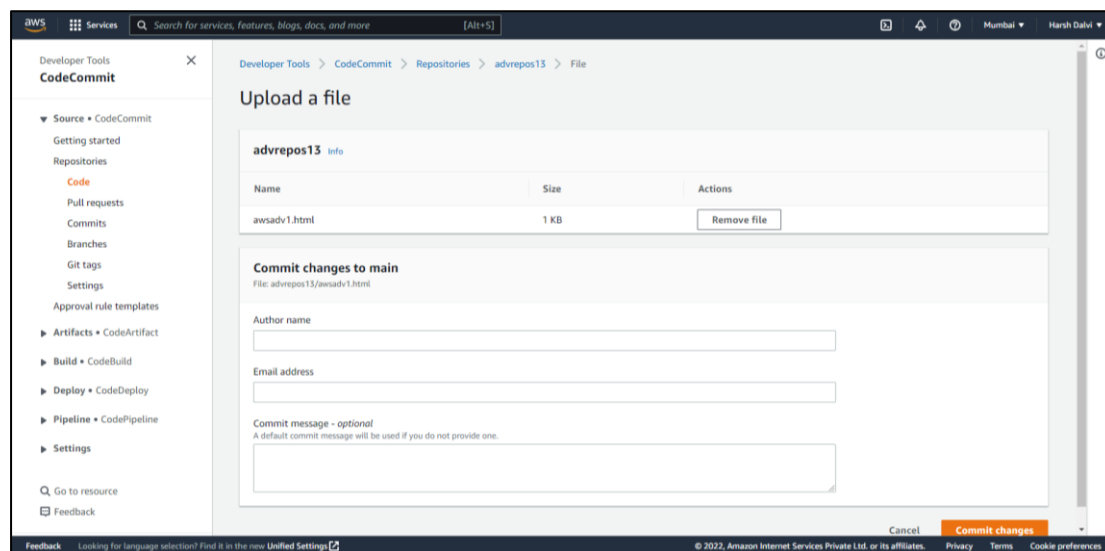
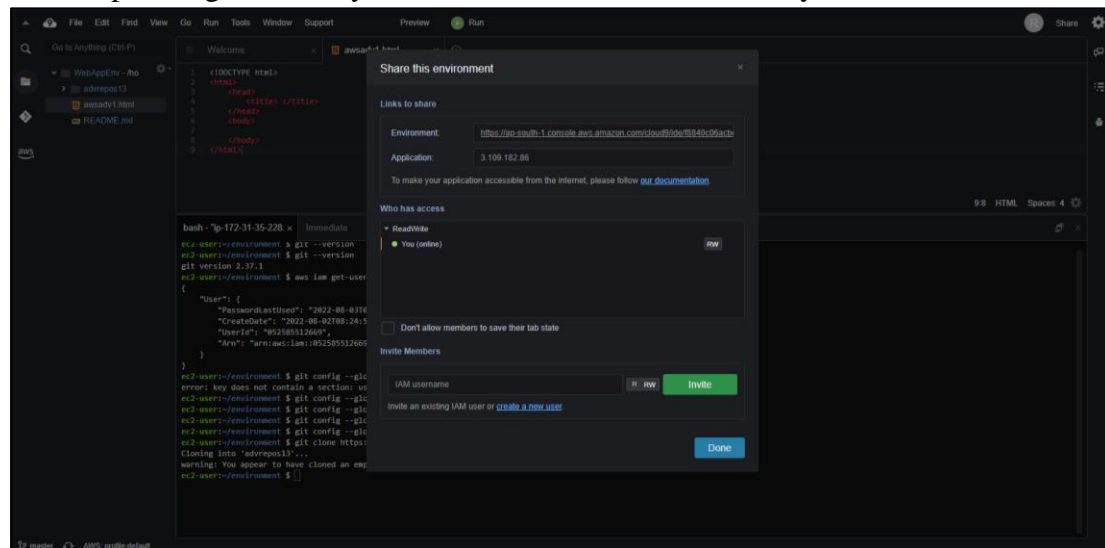
Name: Harsh Dalvi  
Roll No: 13

Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B



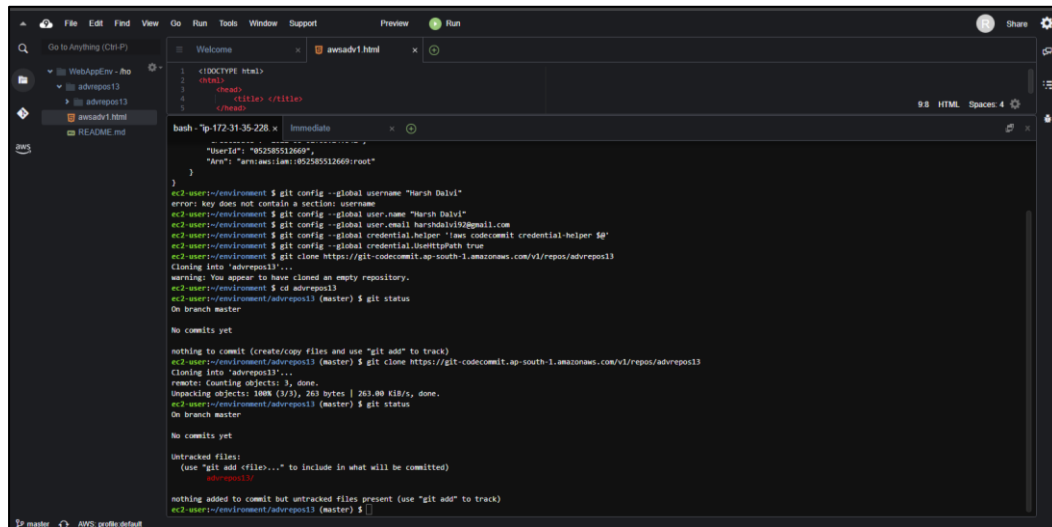
```
bash - "ip-172-31-35-228" x Immediate x
ec2-user:/environment $ git --version
ec2-user:/environment $ git --version
git version 2.37.1
ec2-user:/environment $ aws iam get-user
{
  "User": {
    "UserName": "Harsh Dalvi",
    "CreateDate": "2022-08-02T08:25:56Z",
    "UserId": "A52585512669",
    "Arn": "arn:aws:iam::A52585512669:root"
  }
}
ec2-user:/environment $ git config --global user.name "Harsh Dalvi"
error: key does not contain a section: username
ec2-user:/environment $ git config --global user.name "Harsh Dalvi"
ec2-user:/environment $ git config --global user.email harshdalvi92@gmail.com
ec2-user:/environment $ git config --global credential.helper !aws codecommit credential-helper 5p
ec2-user:/environment $ git config --global credential.useHttpPath true
ec2-user:/environment $
```

16. Now, in order to share this file to collaborate with other members of your team click on share option right-side of your window and username which you created in IAM.



Name: Harsh Dalvi  
Roll No: 13

Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B



```
bash -p-172-31-35-228 x Immediate
"UserId": "852585512669",
"Arn": "arn:aws:iam:852585512669:root"
}
ec2-user:/environment $ git config --global user.name "Harsh Dalvi"
error: key does not contain a section: username
ec2-user:/environment $ git config --global user.name "Harsh Dalvi"
ec2-user:/environment $ git config --global user.email harshdali13@gmail.com
ec2-user:/environment $ git config --global credential.helper !aws codecommit credential-helper $@
ec2-user:/environment $ git config --global credential.useHttpPath true
ec2-user:/environment $ git clone https://git-codecommit.ap-south-1.amazonaws.com/v1/repos/adreposit3
Cloning into 'adreposit3'...
warning: You appear to have cloned an empty repository.
ec2-user:/environment $ cd adreposit3
ec2-user:/environment/adreposit3 (master) $ git status
On branch master

No commits yet

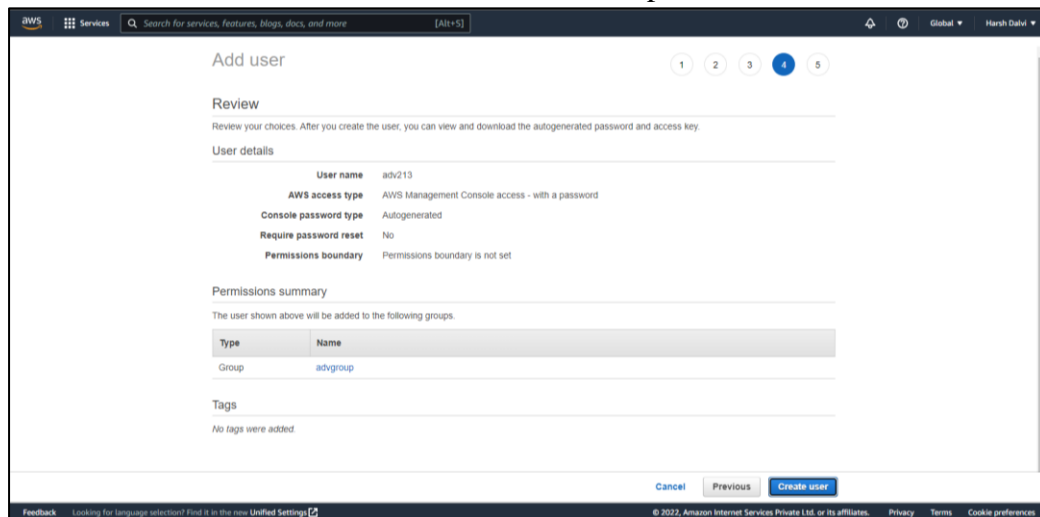
nothing to commit (create/clone files and use "git add" to track)
ec2-user:/environment/adreposit3 (master) $ git clone https://git-codecommit.ap-south-1.amazonaws.com/v1/repos/adreposit3
Cloning into 'adreposit3'...
remote: Counting objects: 3, done.
remote: Unpacking objects: 3, done.
remote: Total 3 (delta 0), reused 0 (delta 0), compressed 0 (delta 0)
ec2-user:/environment/adreposit3 (master) $ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  adreposit3/

nothing added to commit but untracked files present (use "git add" to track)
ec2-user:/environment/adreposit3 (master) $
```

17. Click on create New User and follow the same steps to create the user.



**Add user**

1 2 3 4 5

**Review**

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

**User details**

User name	adv213
AWS access type	AWS Management Console access - with a password
Console password type	Autogenerated
Require password reset	No
Permissions boundary	Permissions boundary is not set

**Permissions summary**

The user shown above will be added to the following groups.

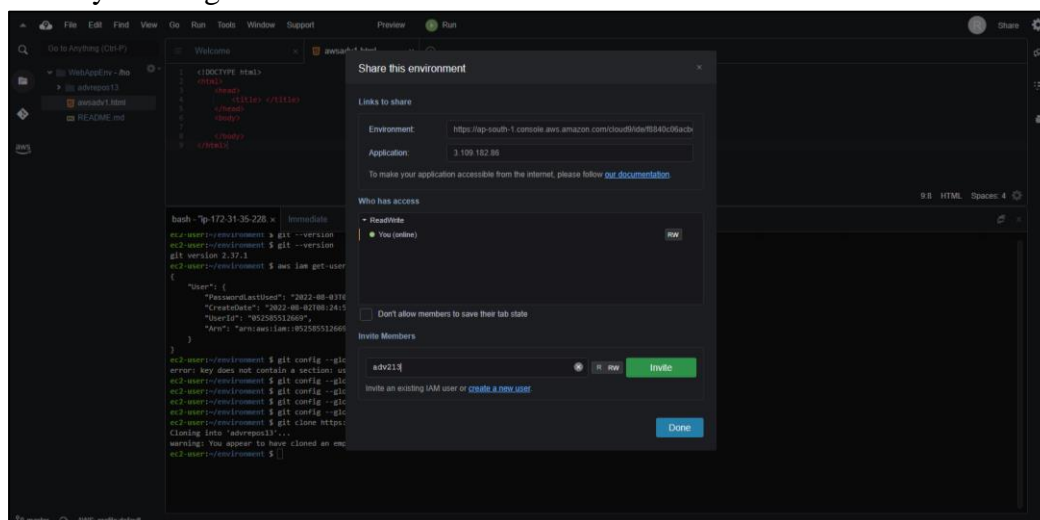
Type	Name
Group	advgroup

**Tags**

No tags were added

Cancel Previous **Create user**

18. Invite the new user by giving read and write permission and click on done. Click OK for security warning.



**Share this environment**

Links to share

Environment: <https://ap-south-1.console.aws.amazon.com/cloudhsm/home?region=ap-south-1>

Application: 3.100.182.88

To make your application accessible from the internet, please follow [our documentation](#)

**Who has access**

ReadWrite

Don't allow members to save their tab state

**Invite Members**

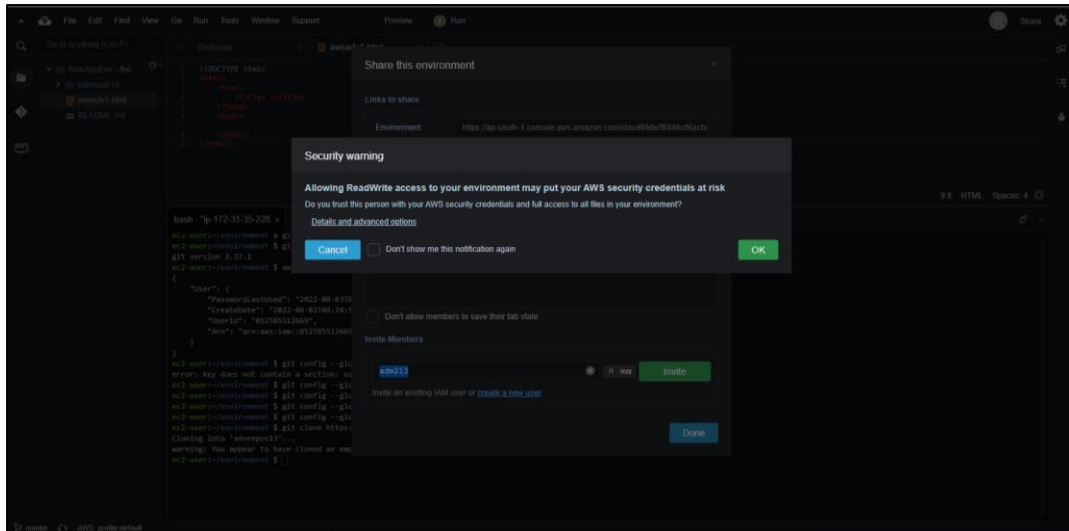
adv213	Invite
--------	--------

Invite an existing IAM user or [create a new user](#)

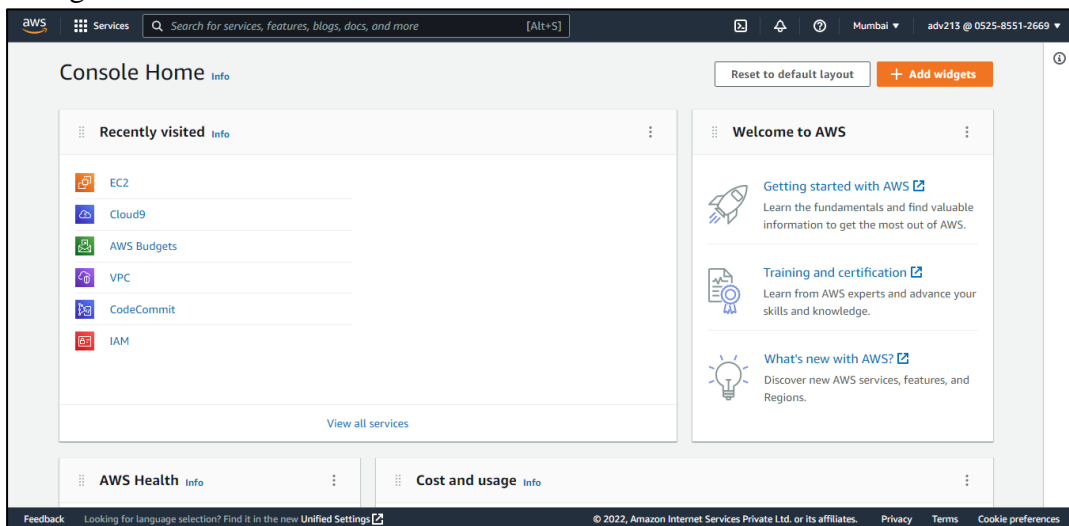
**Done**

Name: Harsh Dalvi  
Roll No: 13

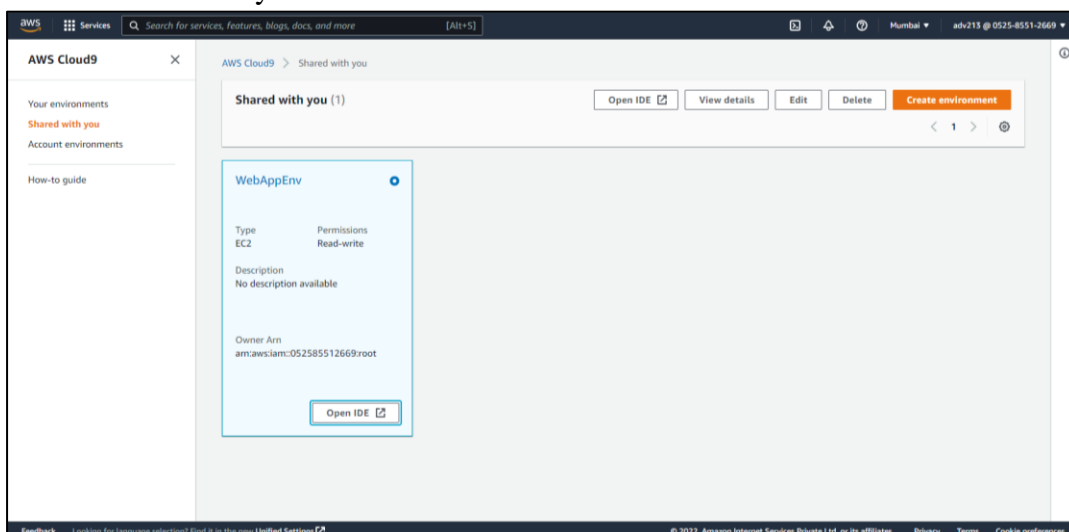
Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B



19. Now, open your browser incognito window and login with IAM user which you configured before.



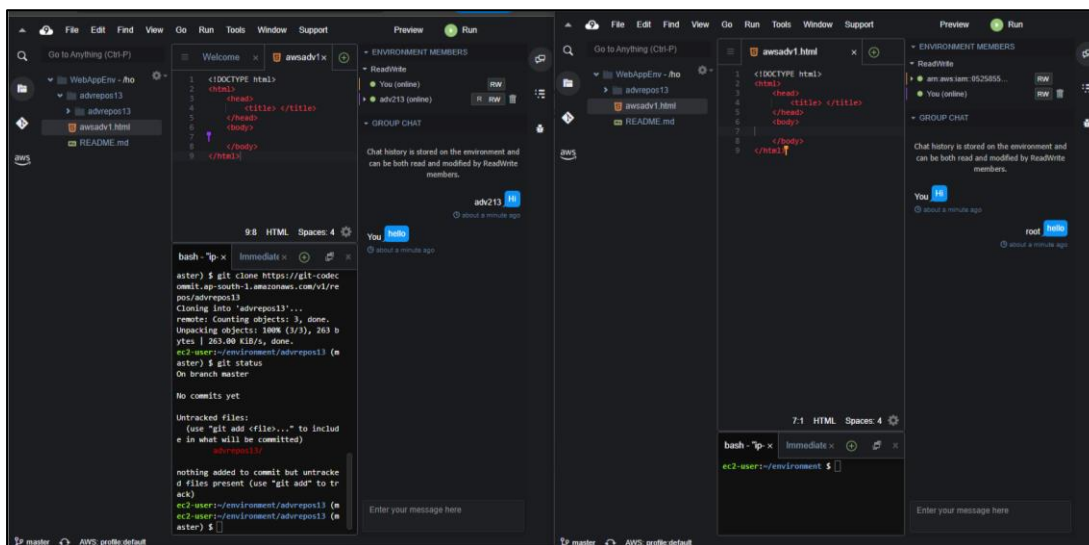
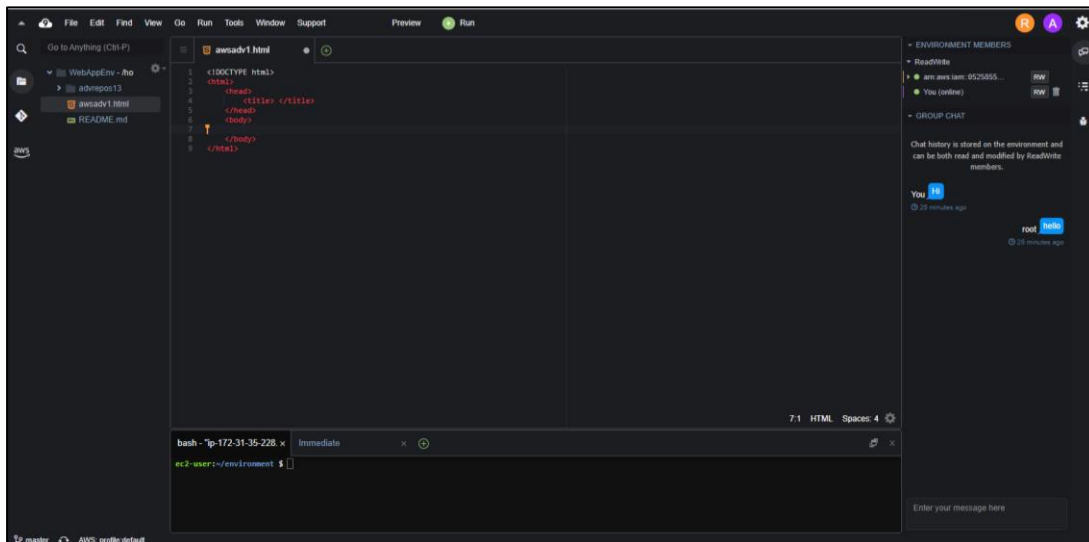
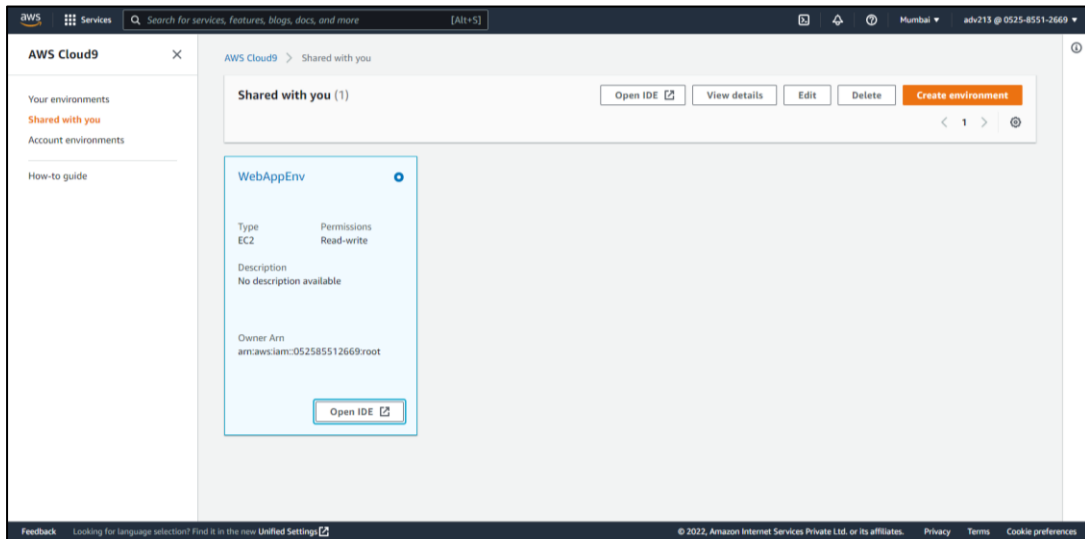
20. After successful login with IAM user. Open cloud9 service from dashboard services and click on share with you environment to collaborate.



Name: Harsh Dalvi  
Roll No: 13

Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B

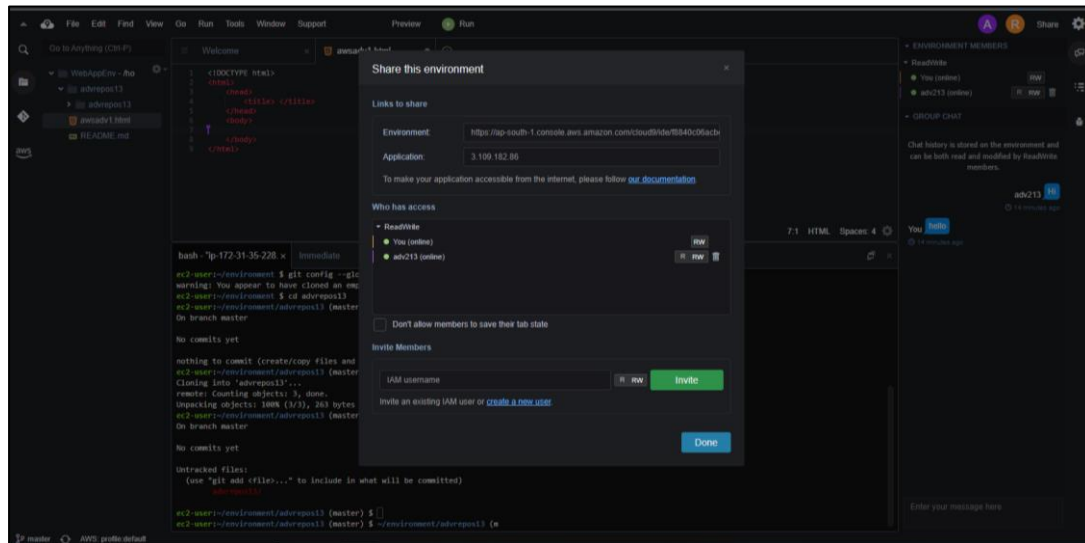
21. Click on open IDE you will get same interface as your other member have to collaborate in real time, also all you in team can do group chats as shown below.



Name: Harsh Dalvi  
Roll No: 13

Subject: Advance DevOps , Sem: SEM V  
Class / Batch: TE-IT / Batch B

22. You can also explore settings where you can update permission of your team members from read write to write only or and you can remove user too.



**Conclusion:** From this experiment, it is concluded that we have successfully learnt the concepts of Cloud9 and its benefits. We also demonstrated collaborative environment on AWS Cloud9 IDE. Thus, we have achieved the Lab Outcome 1(LO1).