

EXPERIMENT NO. 01

Aim: To understand the benefits of Cloud Infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration.

Theory:

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 prepackaged 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 serverless applications enabling you to easily define resources, debug, and switch between local and remote execution of serverless 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.

Benefits:

CODE WITH JUST A BROWSER

AWS Cloud9 gives you the flexibility to run your development environment on a managed Amazon EC2 instance or any existing Linux server that supports SSH. This means that you can write, run, and debug applications with just a browser, without needing to install or maintain a local IDE. The Cloud9 code editor and integrated debugger include helpful, time-saving features such as code hinting, code completion, and step-through debugging. The Cloud9 terminal provides a browser-based shell experience enabling you to install additional software, do a git push, or enter commands.

CODE TOGETHER IN REAL TIME

AWS Cloud9 makes collaborating on code easy. You can share your development environment with your team in just a few clicks and pair program together. While collaborating, your team members can see each other type in real time, and instantly chat with one another from within the IDE.

BUILD SERVERLESS APPLICATIONS WITH EASE

AWS Cloud9 makes it easy to write, run, and debug serverless applications. It preconfigures the development environment with all the SDKs, libraries, and plug-ins needed for serverless development. Cloud9 also provides an environment for locally testing and debugging AWS Lambda functions. This allows you to iterate on your code directly, saving you time and improving the quality of your code.

DIRECT TERMINAL ACCESS TO AWS

AWS Cloud9 comes with a terminal that includes sudo privileges to the managed Amazon EC2 instance that is hosting your development environment and a preauthenticated AWS Command

Line

Interface. This makes it easy for you to quickly run commands and directly access AWS services

START NEW PROJECTS QUICKLY

AWS Cloud9 makes it easy for you to start new projects. Cloud9's development environment comes prepackaged with tooling for over 40 programming languages, including Node.js, JavaScript, Python, PHP, Ruby, Go, and C++. This enables you to start writing code for popular application stacks within minutes by eliminating the need to install or configure files, SDKs, and plug-ins for your development machine. Because Cloud9 is cloud-based, you can easily maintain multiple development environments to isolate your project's resources.

Steps:

- 1. Login with your AWS account.**



Sign in

☒ Root user

Account owner that performs tasks requiring unrestricted access. [Learn more](#)

☐ IAM user

User within an account that performs daily tasks. [Learn more](#)

Root user email address

username@example.com

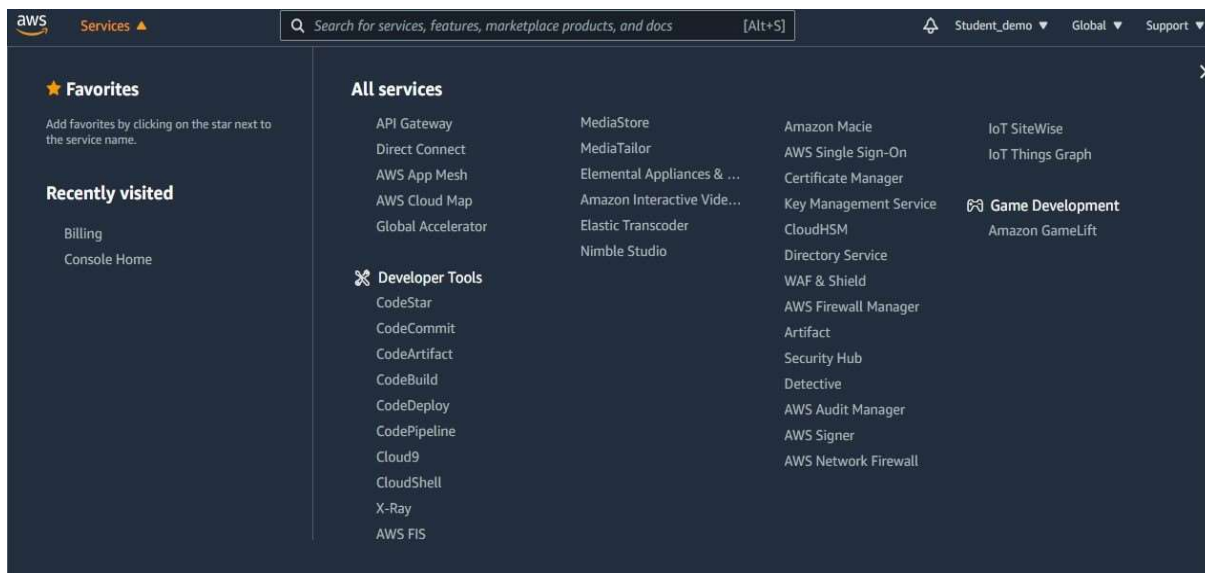
Next

By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.

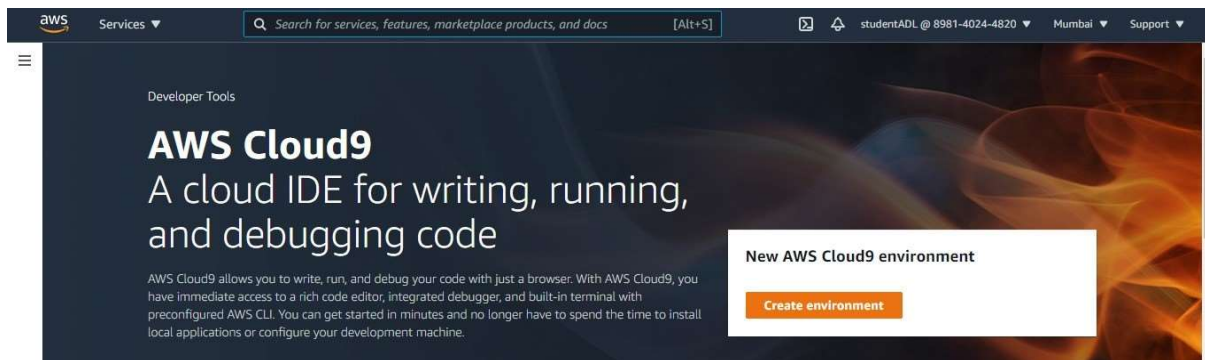
☐ New to AWS?

Create a new AWS account

2. Navigate to Cloud 9 service from Developer tools section as below:



3. Click on Create Environment:



4. Provide name for the Environment (WebAppIDE) and click on next.

This screenshot shows the 'Name environment' step in the AWS Cloud9 console. The left sidebar indicates 'Step 1 Name environment' is selected, with 'Step 2 Configure settings' and 'Step 3 Review' below it. The main area is titled 'Name environment' and contains a form titled 'Environment name and description'. The form has two sections: 'Name' and 'Description'. The 'Name' section has a text input field containing 'WebAppIde', a note that the name must be unique per user, and a 'Limit: 60 characters' warning. The 'Description' section is optional, with a text area for a short description and a 'Limit: 200 characters' warning. At the bottom right of the form, there are 'Cancel' and 'Next step' buttons.

5. Keep all the Default settings as shown in below:

aws

Services

Search for services, features, marketplace products, and docs

[Alt+S]

Student_demo

Mumbai

Support

AWS Cloud9 > Environments > Create environment

Step 1
Name environment

Step 2
Configure settings

Step 3
Review

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.

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)

No tags associated with the resource.

Add new tag

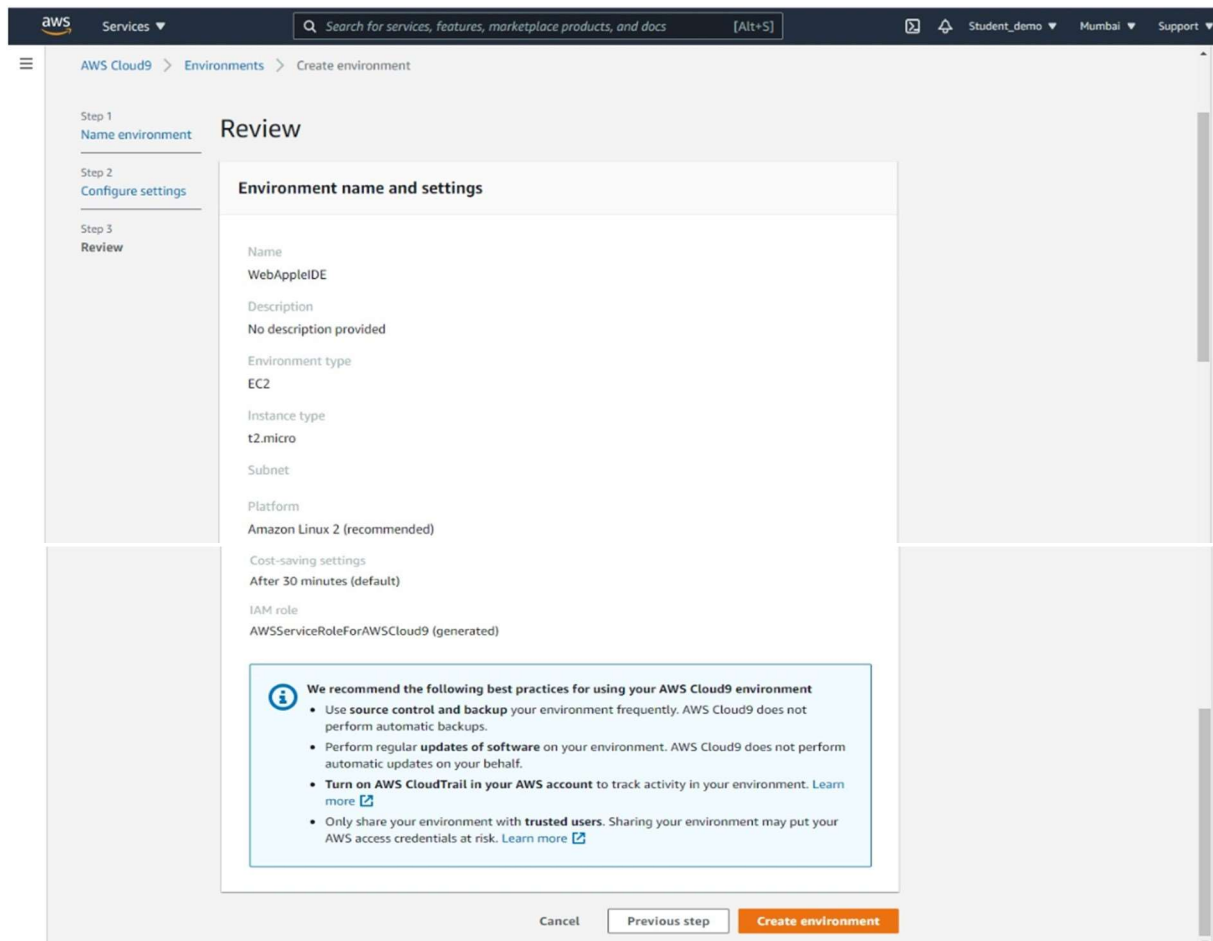
You can add 50 more tags.

Cancel

Previous step

Next step

6. **Review the Environment name and Settings and click on Create Environment:**



It will take few minutes to create aws instance for your Cloud 9 Environment.

7. Till that time open IAM Identity and Access Management in order to Add user in other tab.

- ▼ Access management
 - User groups
 - Users**
 - Roles
 - Policies
 - Identity providers
 - Account settings
- ▼ Access reports
 - Access analyzer
 - Archive rules
 - Analyzers
 - Settings
 - Credential report
 - Organization activity
 - Service control policies (SCPs)

Users (4) Info Refresh Delete Add users

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

<input type="checkbox"/>	User name	Path	Groups	Last activity	MFA	Password a...
--------------------------	-----------	------	--------	---------------	-----	---------------

8. Add user provide manual password if you want and click on Next permission tab.

aws Services [Alt+S] Student_demo Global Support

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 access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

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

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

Console password* ☐ Autogenerated password
☒ Custom password
 ☐ Show 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 their own password.

* Required Cancel Next: Permissions

9. Click on Create group

aws Services [Alt+S] Student_demo Global Support

▼ Set permissions

Add user to group

Copy permissions from existing user

Attach existing policies directly

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Add user to group

Create group Refresh

Showing 1 result

Group	Attached policies
<input type="checkbox"/> TEIT	Billing and 1 more

► Set permissions boundary

Cancel Previous Next: Tags

10. Provide group name and click on create group.

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 668 results

	Policy name	Type	Used as	Description
<input type="checkbox"/>	AdministratorAccess	Job function	Permissions policy (2)	Provides full access to AWS services and resources.
<input type="checkbox"/>	AdministratorAccess-A...	AWS managed	None	Grants account administrative permissions while explicitly a...
<input type="checkbox"/>	AdministratorAccess-A...	AWS managed	None	Grants account administrative permissions. Explicitly allow...
<input type="checkbox"/>	AlexaForBusinessDevic...	AWS managed	None	Provide device setup access to AlexaForBusiness services
<input type="checkbox"/>	AlexaForBusinessFullAc...	AWS managed	None	Grants full access to AlexaForBusiness resources and acc...
<input type="checkbox"/>	AlexaForBusinessGate...	AWS managed	None	Provide gateway execution access to AlexaForBusiness se...

Cancel Create group

11. After that group is created click on next if u want to provide tag else click on Review for user settings and click on create user as shown in fig.

aws

Services

Search for services, features, marketplace products, and docs

[Alt+S]

Student_demo

Global

Support

Add user

12345

Review

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

User details

User name

studentADL

AWS access type

AWS Management Console access - with a password

Console password type

Custom

Require password reset

Yes

Permissions boundary

Permissions boundary is not set

Permissions summary

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

Type	Name
Group	WebAppADLGroup

Cancel

Previous

Create user

aws

Services

Search for services, features, marketplace products, and docs

[Alt+S]

Student_demo

Global

Support

Set permissions

Add user to group

Copy permissions from existing user

Attach existing policies directly

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Add user to group

Create group

Refresh

Search

Showing 2 results

Group	Attached policies
<input checked="" type="checkbox"/> WebAppADLGroup	None
<input type="checkbox"/> TEIT	Billing and 1 more

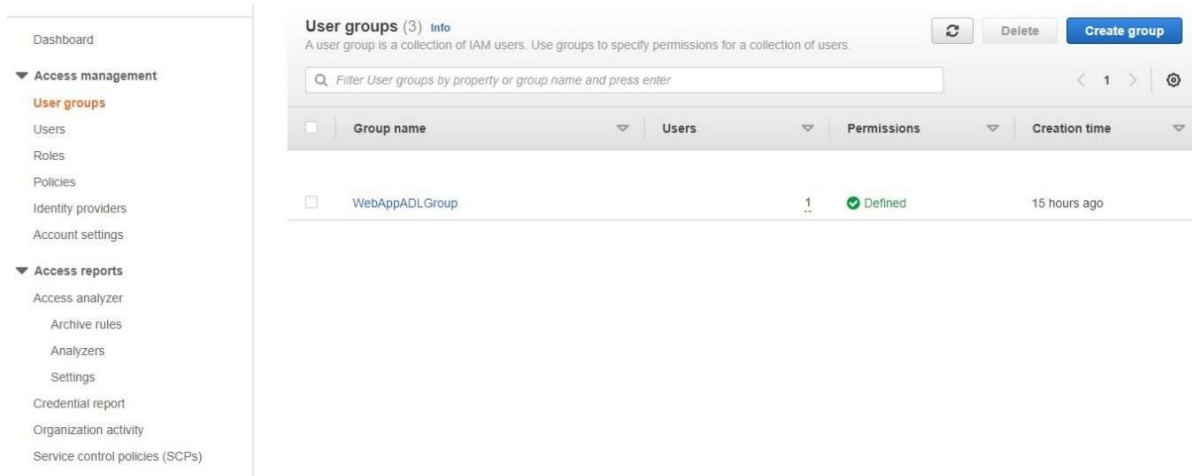
Set permissions boundary

Cancel

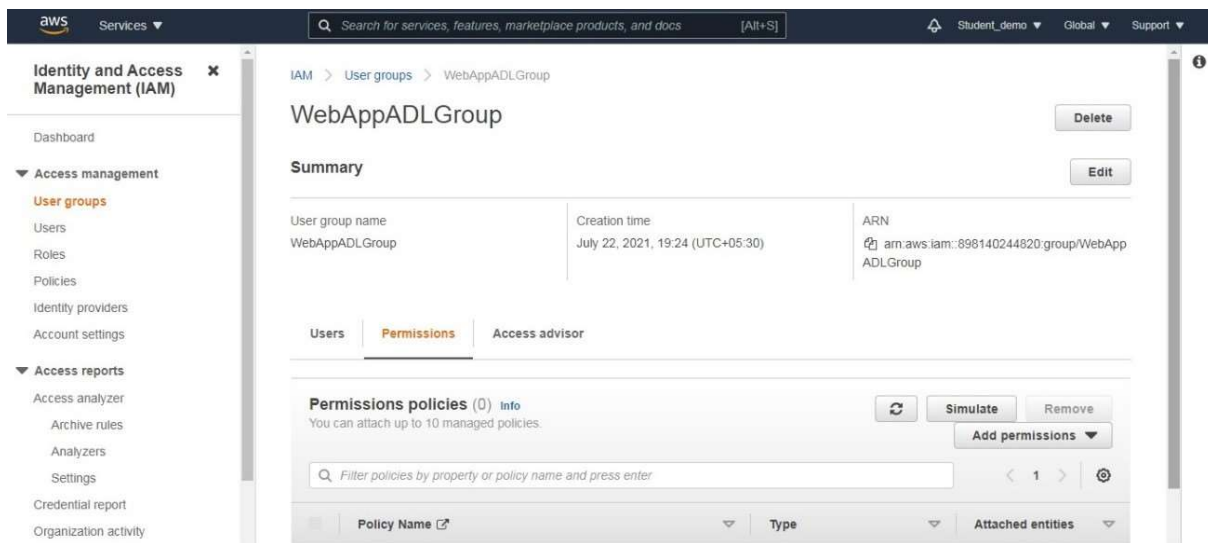
Previous

Next: Tags

12. Now close that window and navigate to user Groups from left pane in IAM.



13. Click on your group name which you have created and navigate to permission tab as shown:



14. Now click on Add permission and select Attach Policy after that search for Cloud9 related policy and select Awscloud9EnviornmentMember policy and add it.

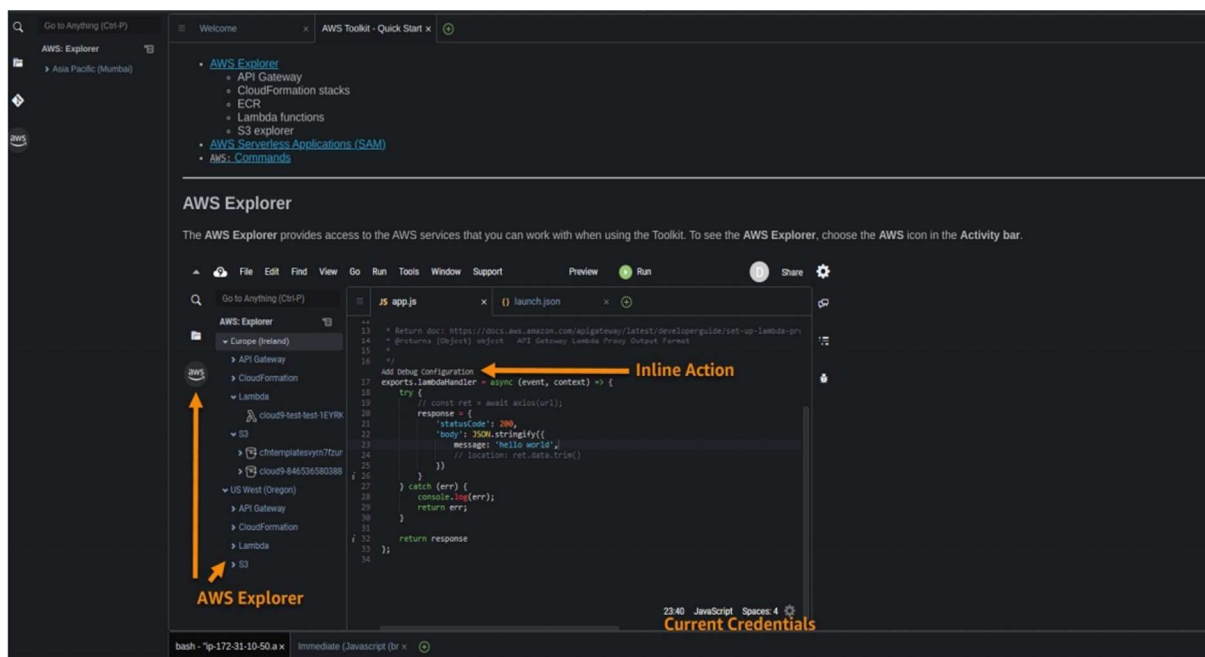
Other permission policies (Selected 1/668) [Info](#)

You can attach up to 10 managed policies to this user group. All of the users in this group inherit the attached permissions.

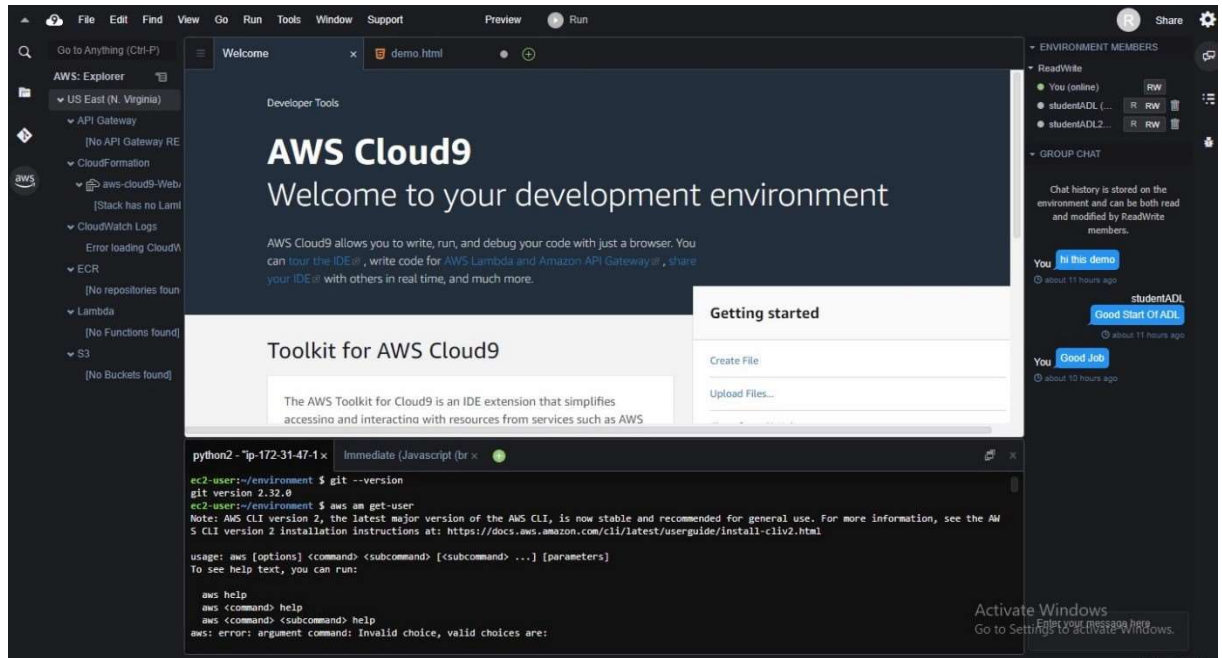
Filter policies by property or policy name and press enter

	Policy Name	Type	Attached entities
<input type="checkbox"/>	AWSQuicksightAthenaAccess	AWS managed	0
<input type="checkbox"/>	AWSCloudMapRegisterInstanceAccess	AWS managed	0
<input type="checkbox"/>	AWSMarketplaceImageBuildFullAccess	AWS managed	0
<input type="checkbox"/>	AWSCodeCommitPowerUser	AWS managed	0
<input type="checkbox"/>	AWSCodeCommitFullAccess	AWS managed	0
<input type="checkbox"/>	IAMSelfManageServiceSpecificCredentials	AWS managed	0
<input checked="" type="checkbox"/>	AWSCloud9EnvironmentMember	AWS managed	0

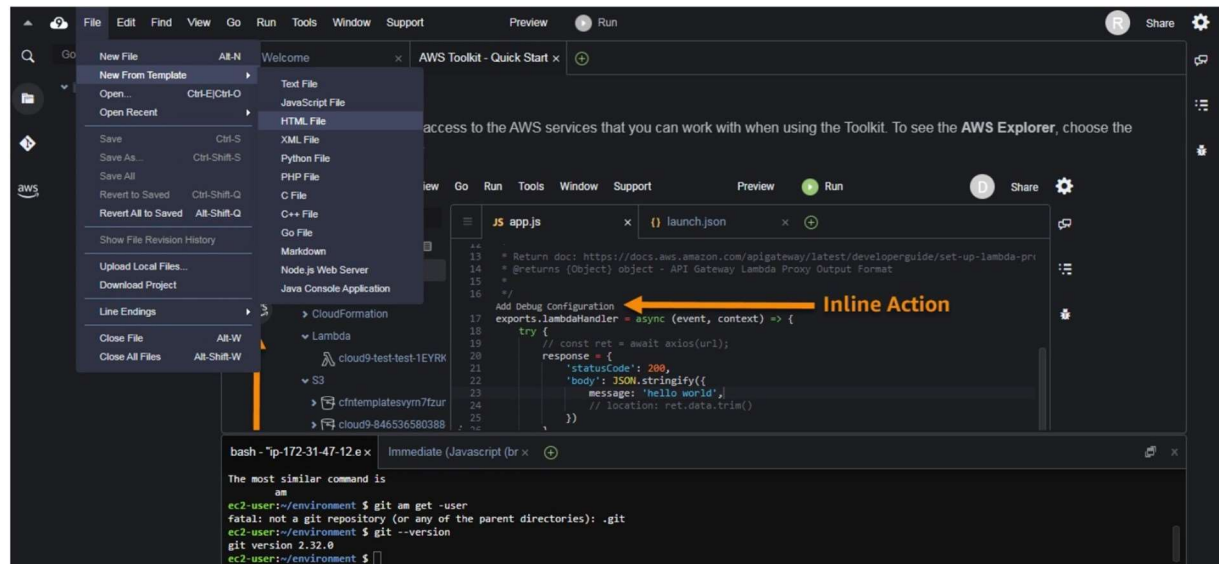
15. Now we move towards our cloud9 IDE Environment tab it shows as shown:



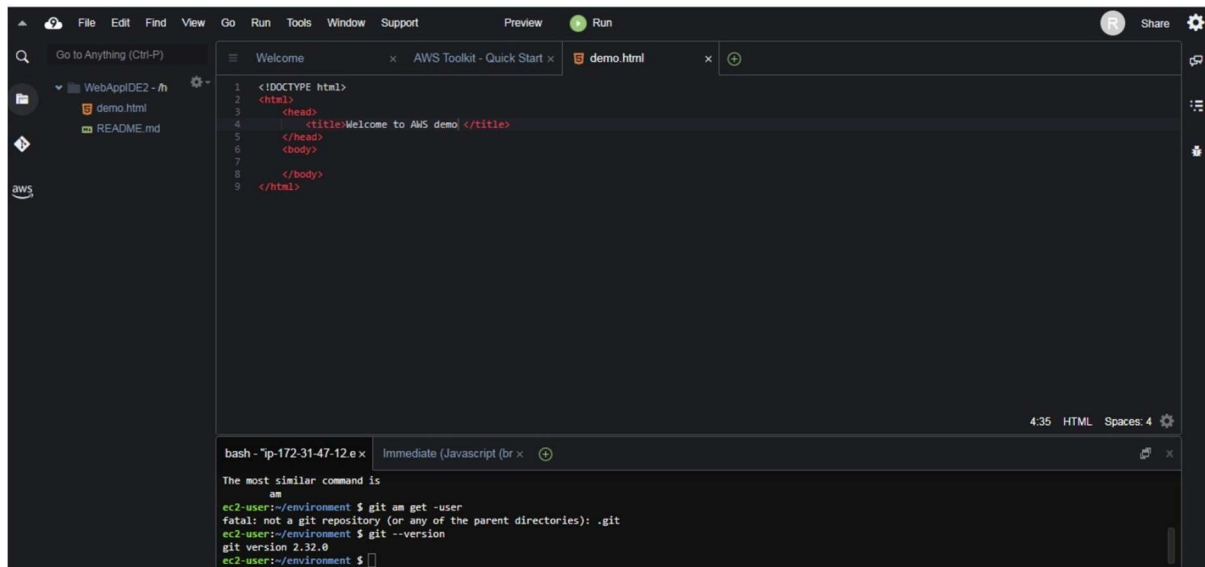
16. If you check at bottom side Cloud9 IDE also giving you and AWS CLI for command operations: as we here checked git version, I am user details and so on...



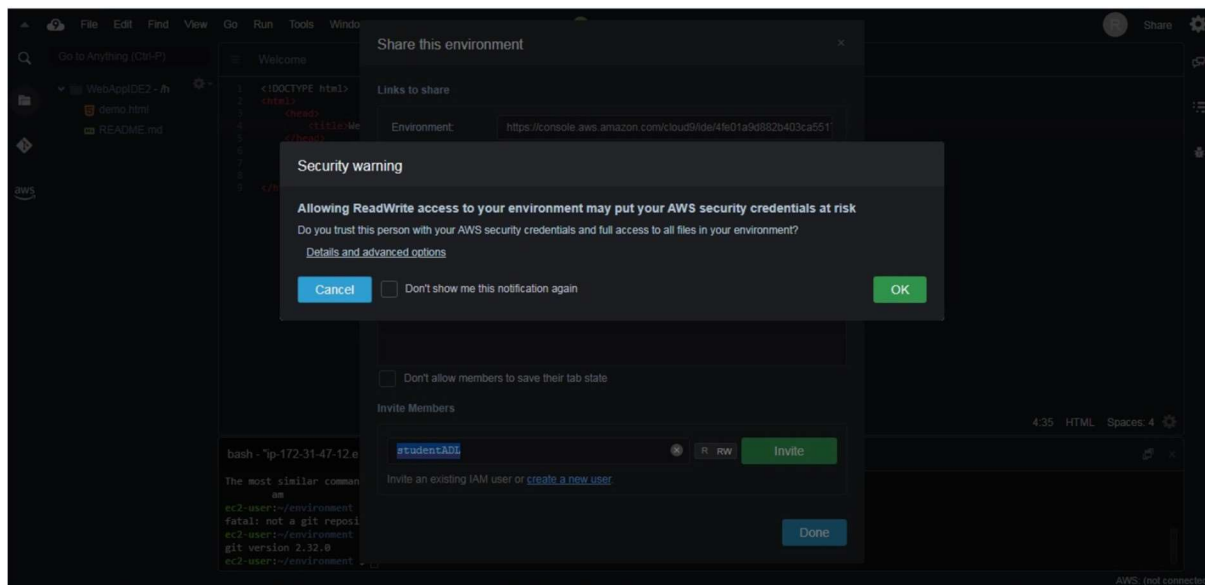
17. Now we will setup collaborative environment Click on File you can create new file or choose from template, here m opting html file to collaborate.

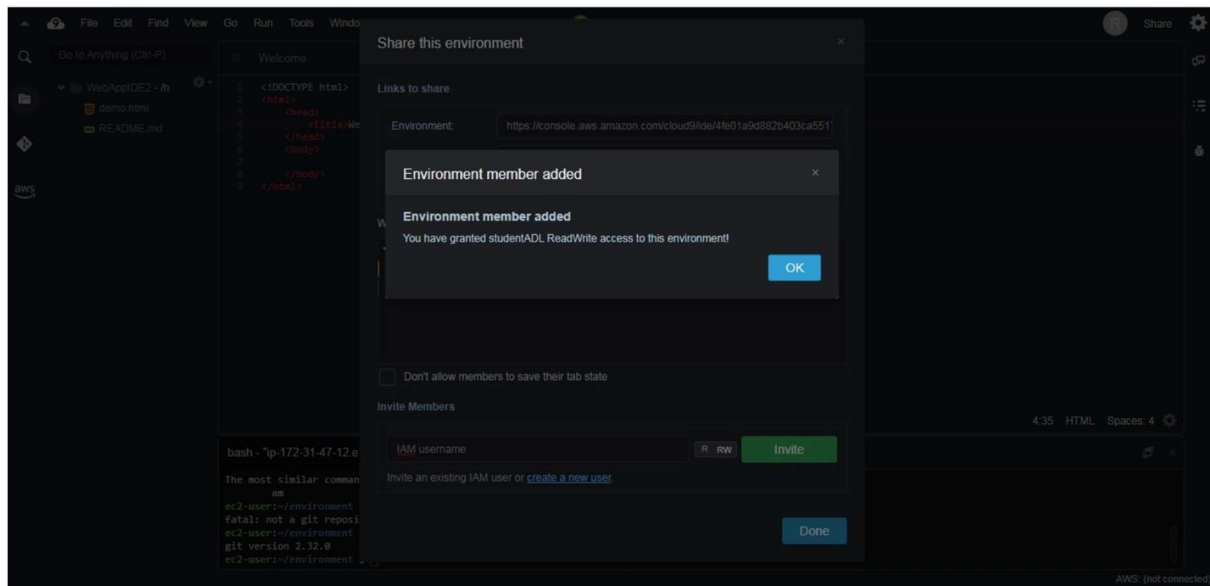


18. Edit html file and save it



19. Now in order to share this file to collaborate with other members of your team click on Share option on Right Pane and username which you created in IAM before into Invite members and enable permission as RW (Read and Write) and click on Done. Click OK for Security warning.





20. Now Open your Browsers Incognito Window and login with IAM user which you configured before.

aws

Sign in

☐ Root user
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

☒ IAM user
User within an account that performs daily tasks. [Learn more](#)

Account ID (12 digits) or account alias

898140244820

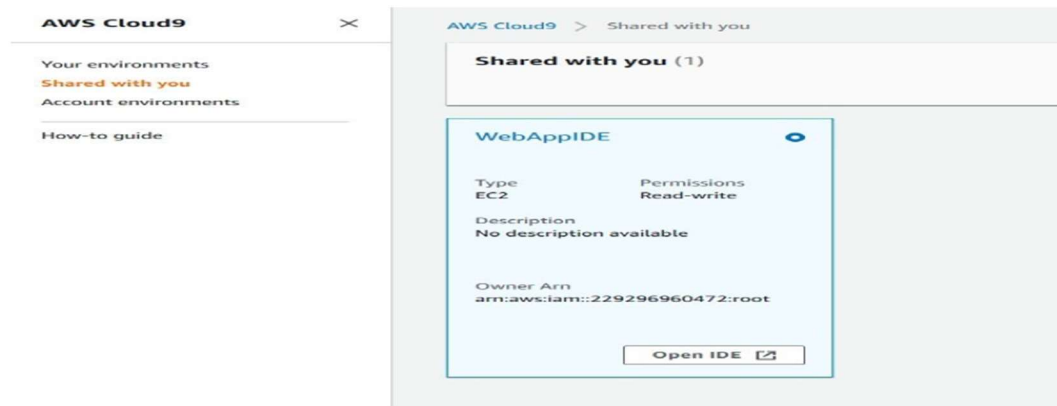
☐ Remember this account

Next

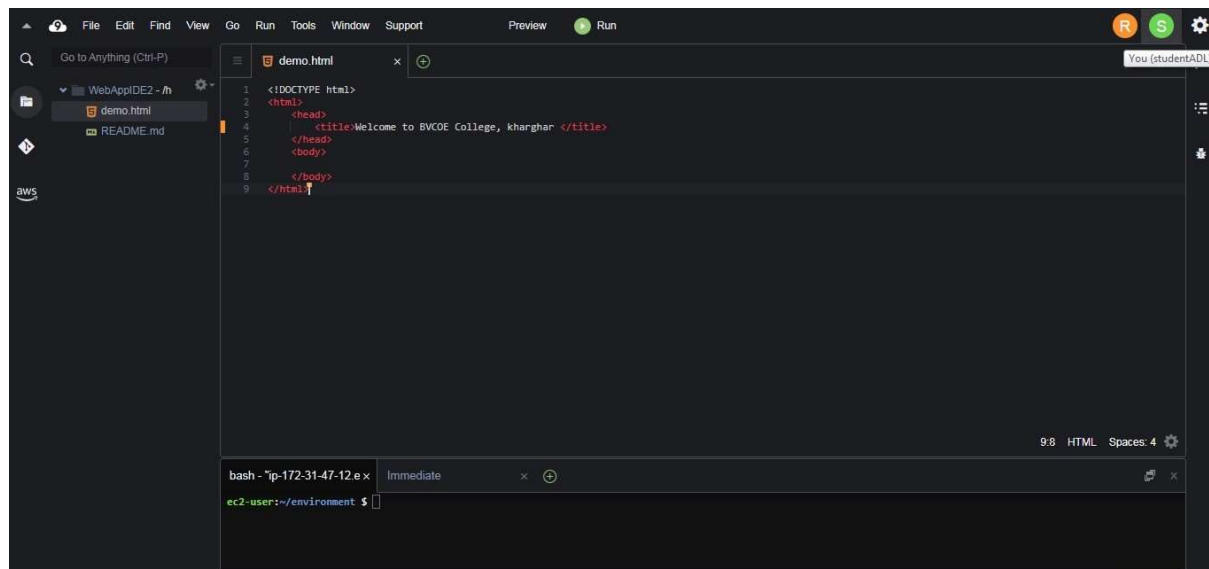
By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.

[New to AWS?](#)

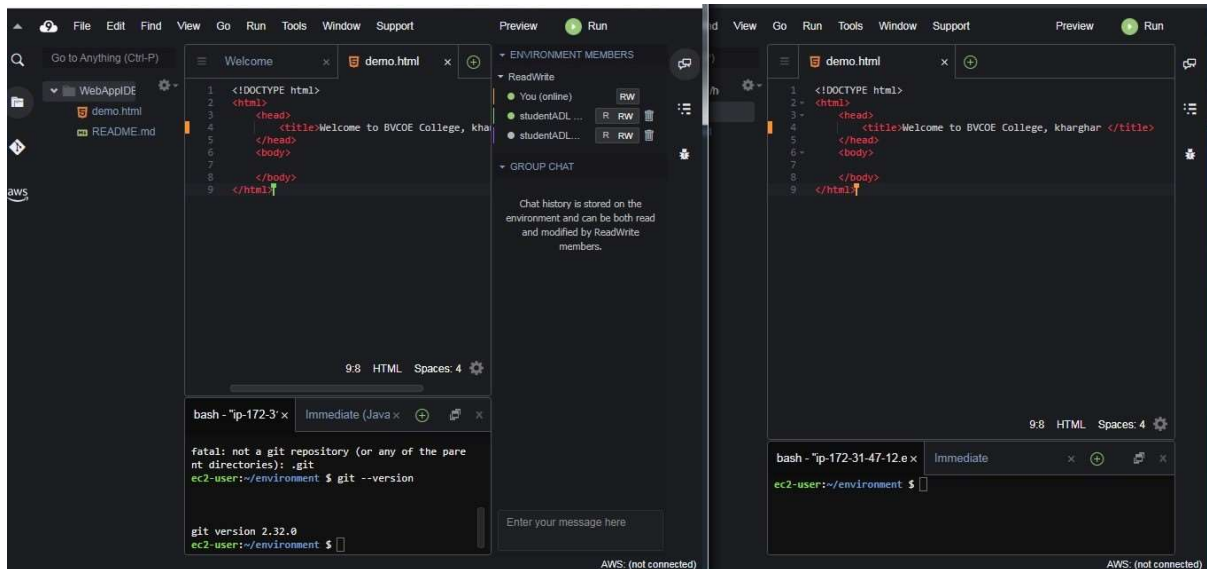
21. After Successful login with IAM user open Cloud9 service from dashboard services and click on shared with your environment to collaborate.



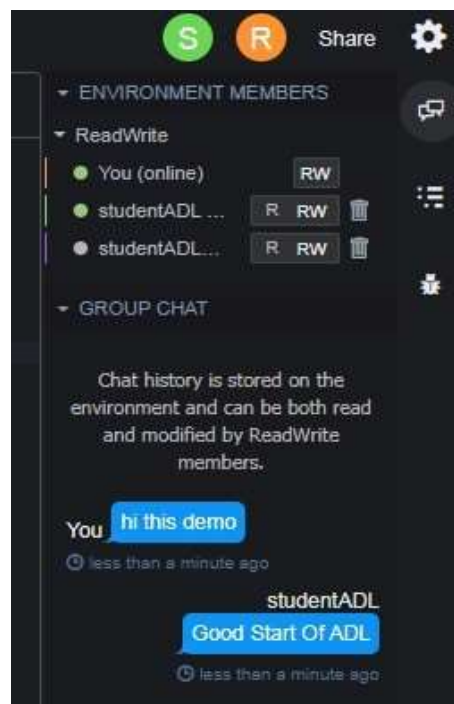
22. Click on Open IDE you will same interface as your other member have to collaborate in real time.



23. Also, you all within team can do group chats as shown below:



24. You can also explore settings where you can update permissions of your teammates as from RW to R only or you can remove user too.



Conclusion: After Successful setup of AWS account, Cloud9IDE environment service from the AWS installed successfully. The AWS Console web Application provided by Amazon Web Services helped me in viewing and managing a select set of resources to support incident response while on-the-go. I can use the Console Mobile Application to monitor resources through a dedicated dashboard and view configuration details, metrics, and alarms for select AWS services. AWS Cloud9 which is a cloud-based

integrated development environment (IDE) that helped me to write, run, and debug my code with just a browser. I do not have to wait for an environment or a device to be code-ready. By providing a code-ready climate, AWS Cloud9 brings collaboration to Dev Ops, improves productivity, and helps with better time management.