

# **Advanced DevOps Lab**

## **Experiment 1**

Roll No.	24
Name	Iyer Sreekesh Subramanian
Class	D15-A
Subject	Advanced DevOps Lab

**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 programs together. While collaborating, your team members can see each other 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

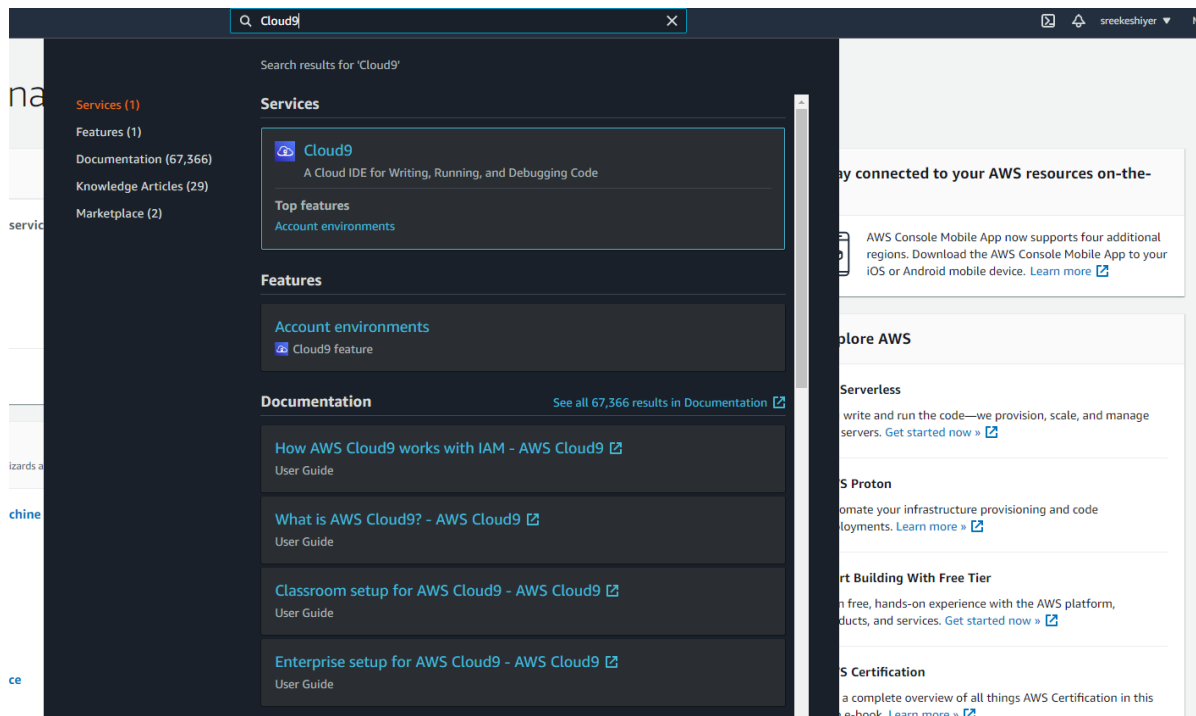
AWS Cloud9 comes with a terminal that includes sudo privileges to the managed Amazon EC2 instance that is hosting your development environment and a preauthenticated AWS Command Line Interface. This makes it easy for you to quickly run commands and directly access AWS services.

## START NEW PROJECTS QUICKLY

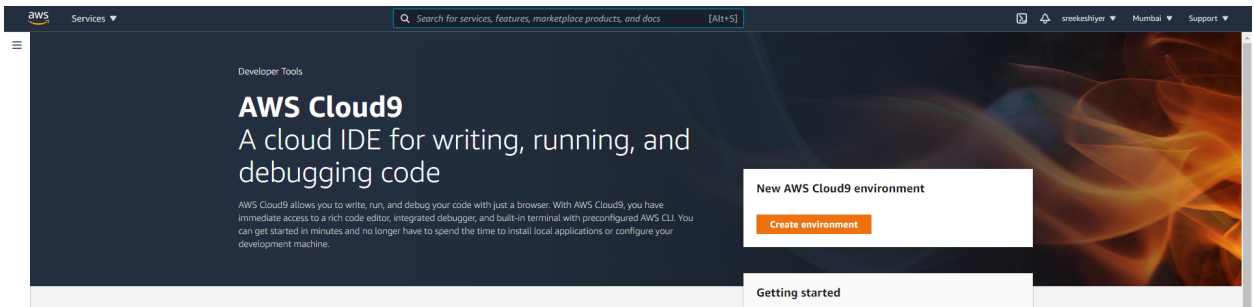
AWS Cloud9 makes it easy for you to start new projects. Cloud9's development environment comes prepackaged with tooling for over 40 programming languages, including Node.js, JavaScript, Python, PHP, Ruby, Go, and C++. This enables you to start writing code for popular application stacks within minutes 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.
2. Search for *Cloud9* under the list of services in the search menu.



3. In the Cloud9 Console, click on **Create Environment**.



4. Proceed after entering a name for your environment.

AWS Cloud9 > Environments > Create environment

Step 1  
Name environment

Step 2  
Configure settings

Step 3  
Review

## 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. Proceed with default 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)**

No tags associated with the resource.

**Add new tag**

You can add 50 more tags.

Cancel Previous step Next step

- Review the settings and create the environment.

## Review

### Environment name and settings

Name  
WebAppIDE

Description  
No description provided

Environment type  
EC2


Instance type  
t2.micro



Subnet

Platform  
Amazon Linux 2 (recommended)

Cost-saving settings  
After 30 minutes (default)

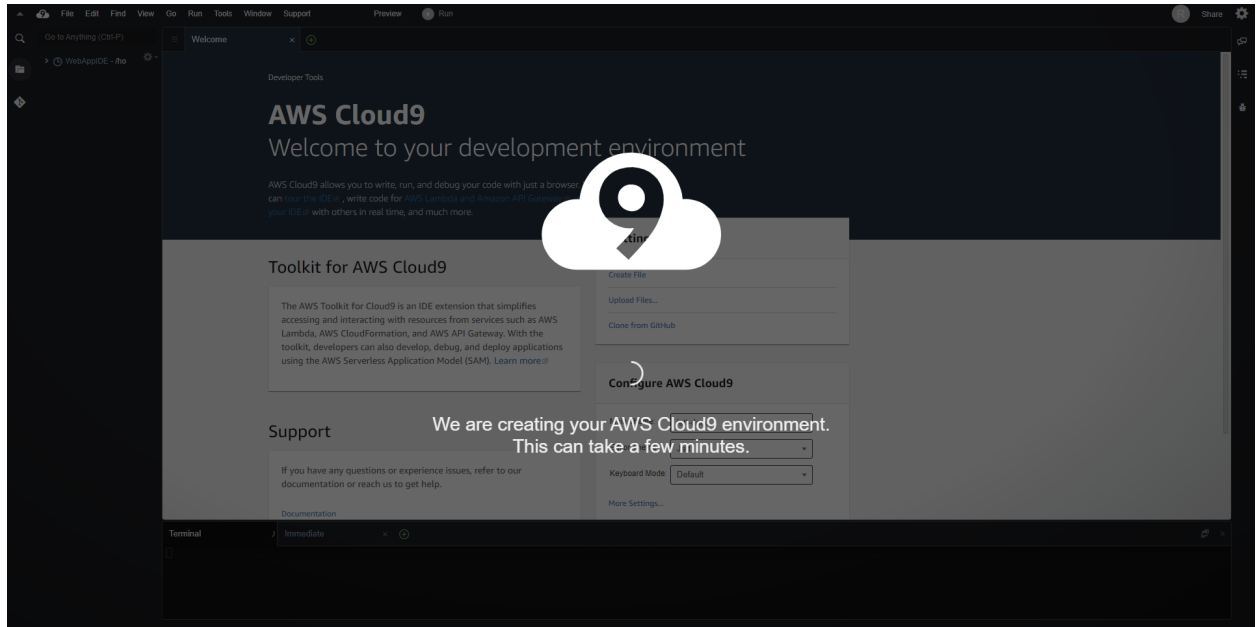
IAM role  
AWSServiceRoleForAWSCloud9 (generated)

**We recommend the following best practices for using your AWS Cloud9 environment**

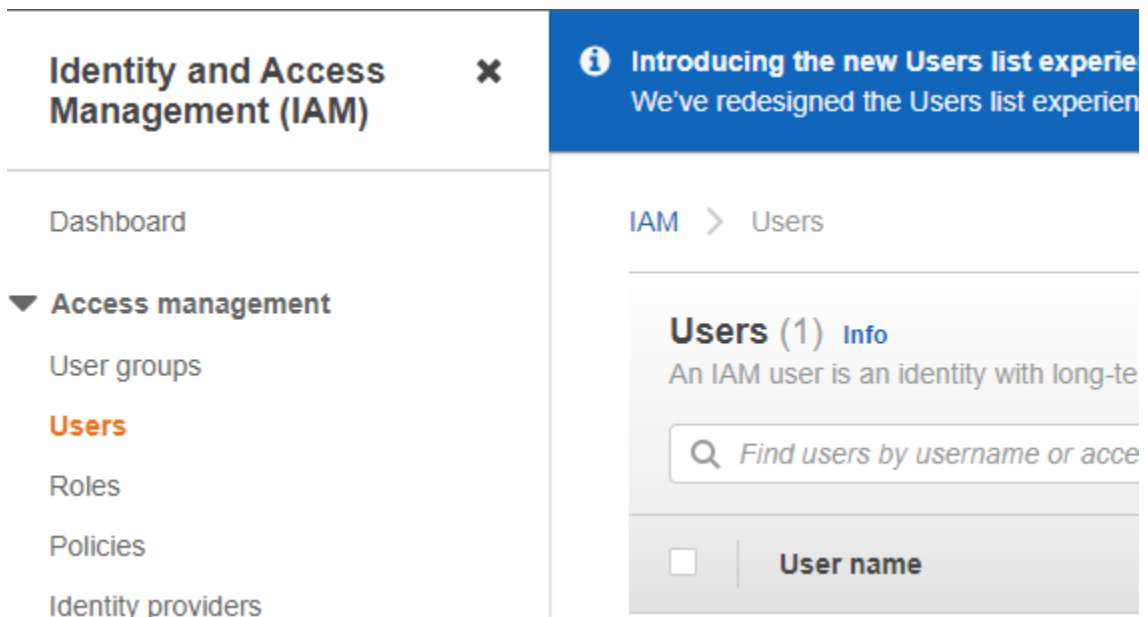
- Use **source control and backup** your environment frequently. AWS Cloud9 does not perform automatic backups.
- Perform regular **updates of software** on your environment. AWS Cloud9 does not perform automatic updates on your behalf.
- **Turn on AWS CloudTrail in your AWS account** to track activity in your environment. [Learn more](#) 
- Only share your environment with **trusted users**. Sharing your environment may put your AWS access credentials at risk. [Learn more](#) 

Cancel Previous step Create environment

- The process of creating your IDE has started. It will take some time and while that happens, we can create our IAM user for sharing our code.



8. Click on the AWS nav logo or open up the AWS console in a new tab and search for IAM. The IAM console opens up.



## 9. Create a new user.

## 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\*

apsit

+

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



## 10. Go to User Groups and create a new User Group

## WebAppGroup

## Summary

User group name WebAppGroup	Creation time August 28, 2021, 10:46 (UTC+05:30)
--------------------------------	---

Users

Permissions

Access advisor

**Users in this group (1)** [Info](#)  
An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

<input type="checkbox"/>	User name <a href="#">↗</a>	Groups
<input type="checkbox"/>	zenon	

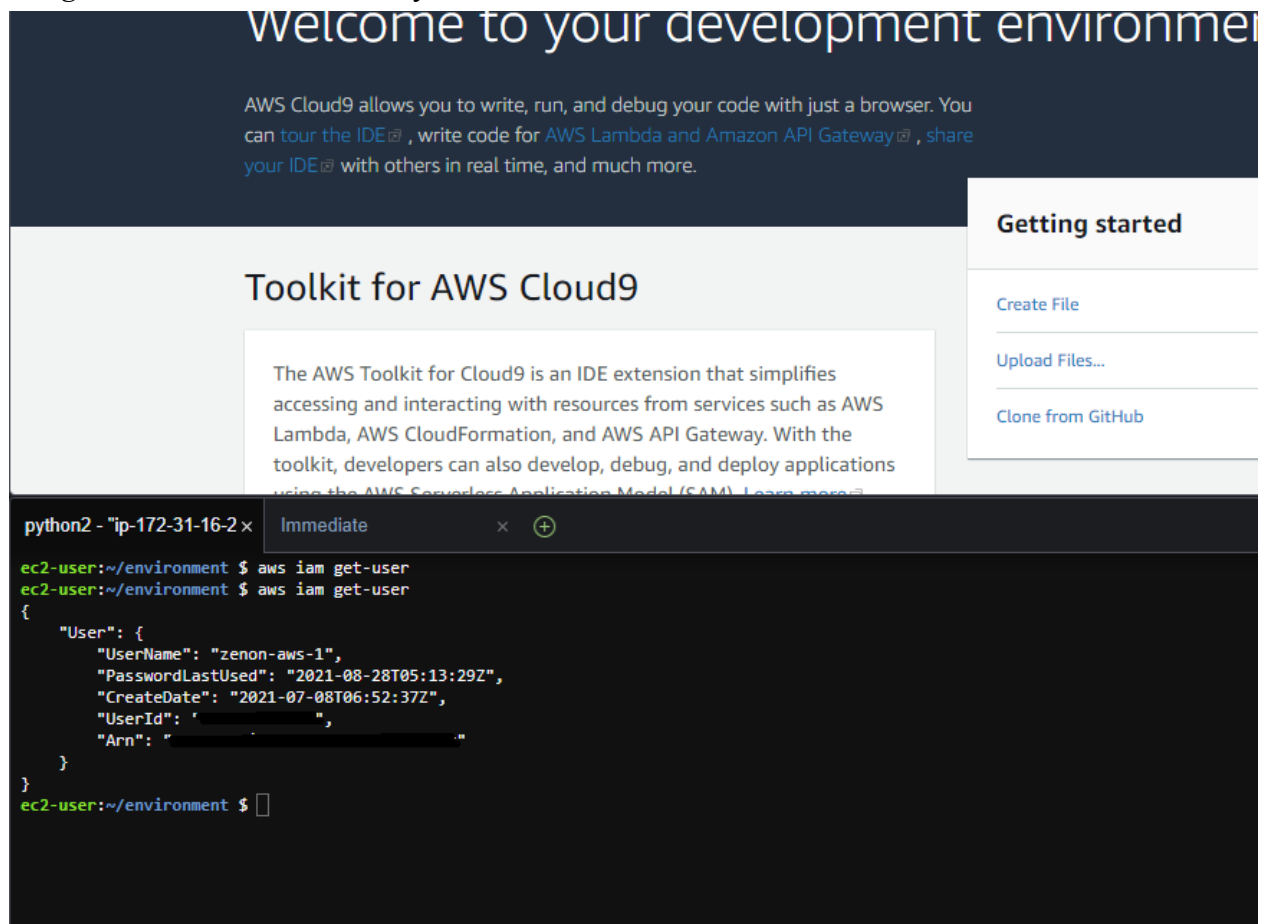
11. Under Permissions, add the *AWSCloud9EnvironmentMember* Permission.

**Permissions policies (1)** [Info](#)  
You can attach up to 10 managed policies.

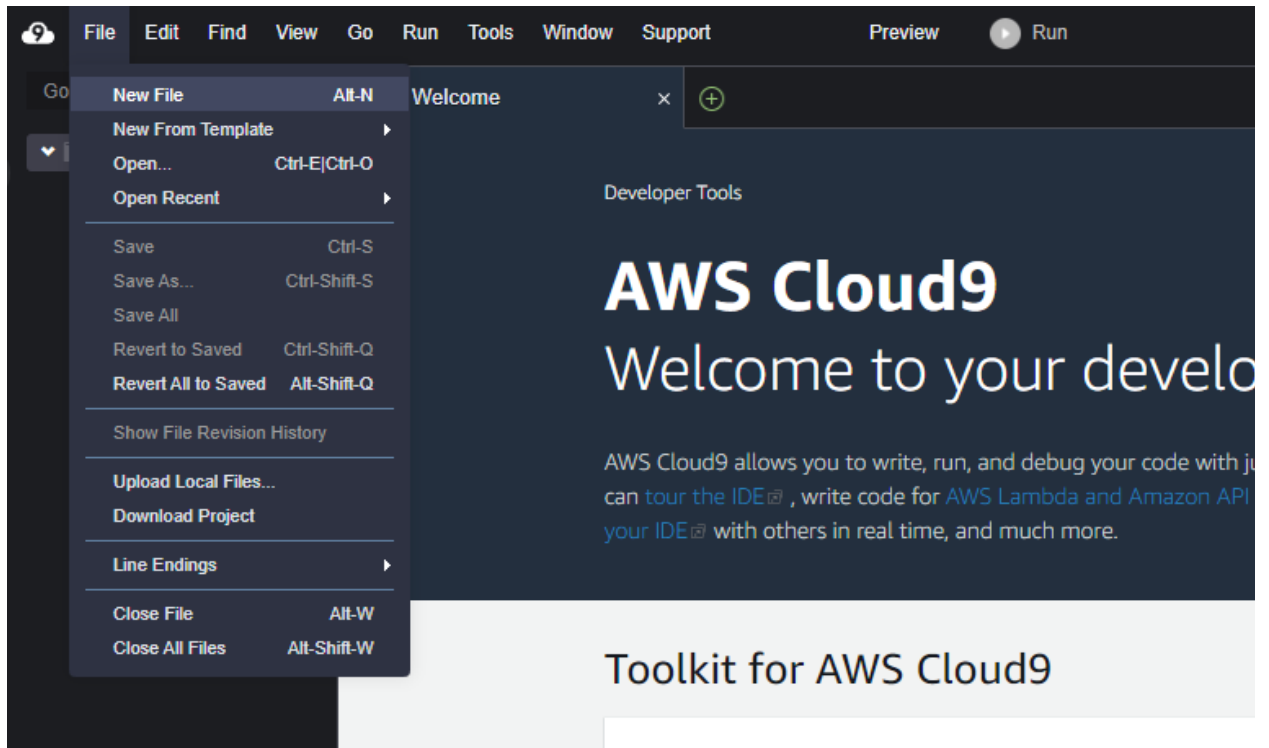
<input type="checkbox"/>	Policy Name <a href="#">↗</a>	Type
<input type="checkbox"/>	 AWSCloud9EnvironmentMember	AWS managed

## 12. Now, once that's done, come back to the Cloud9IDE

13. You can use the command-line here, it also has the AWS CLI built in. You can use the `aws get-user` command and see your IAM users.

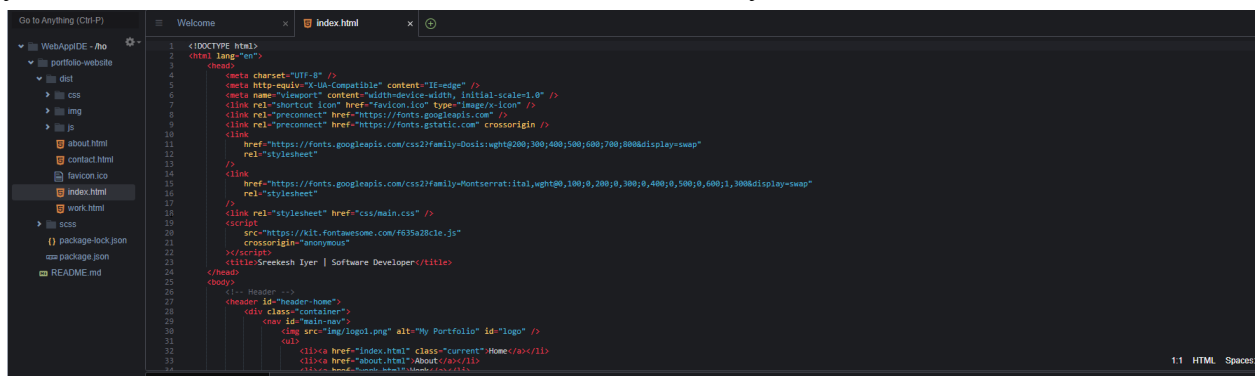


14. We can now create a new file and start working on it. Instead, I'm going to clone my portfolio GitHub Repository.

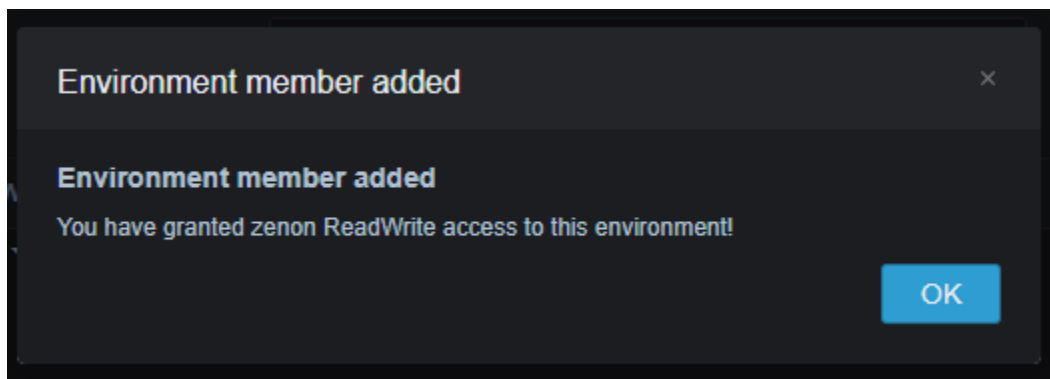
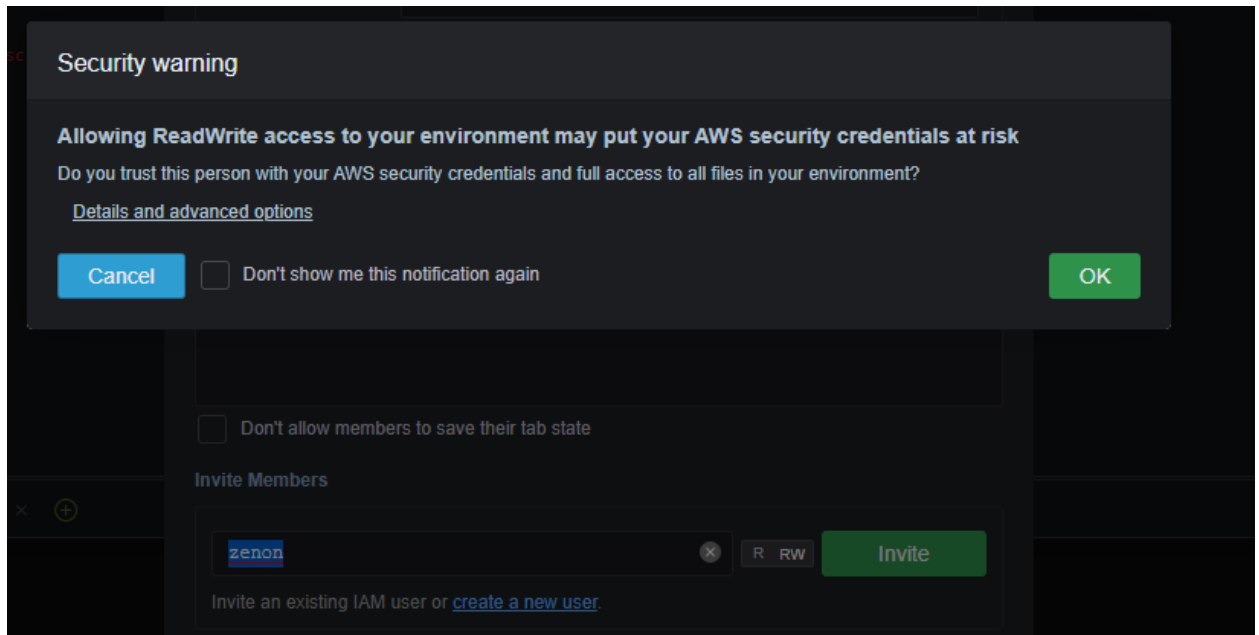


```
ec2-user:~/environment $ git clone https://github.com/sreekeshiyer/portfolio-website.git
Cloning into 'portfolio-website'...
Username for 'https://github.com/sreekeshiyer/portfolio-website.git': sreekeshiyer
Password for 'https://sreekeshiyer@github.com/sreekeshiyer/portfolio-website.git':
remote: Enumerating objects: 89, done.
remote: Counting objects: 100% (89/89), done.
remote: Compressing objects: 100% (59/59), done.
remote: Total 89 (delta 29), reused 84 (delta 24), pack-reused 0
Receiving objects: 100% (89/89), 20.98 MiB | 32.70 MiB/s, done.
Resolving deltas: 100% (29/29), done.
```

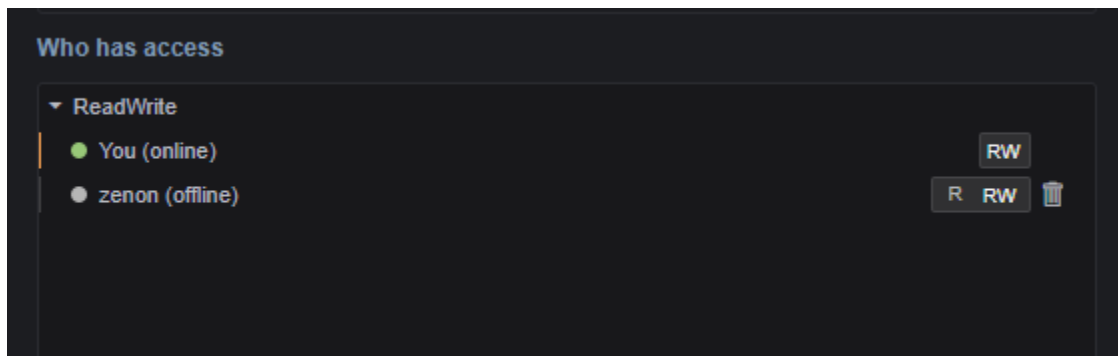
15. Double click on any of the files and you should see them open up. Congrats, this is now your IDE on the web, inside a web-browser with absolutely no software installed.



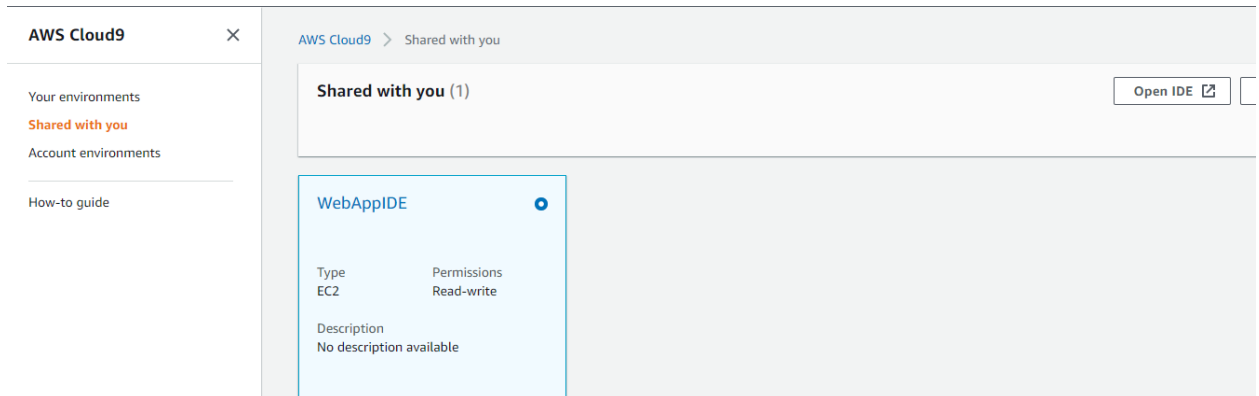
16. To share this IDE with your IAM user, click on **Share** on the top right and add your IAM username there. You can choose if you want to give the IAM user Read Only permissions or Read/Write permissions in the little boxes on the right.



17. You can see the status of collaborators here too.

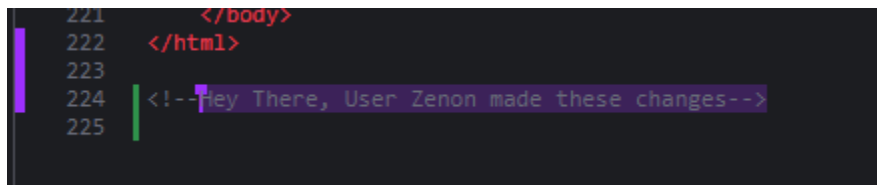


18. Now, open up an incognito window or any browser in private mode and login to AWS from your IAM user account. Go ahead, open Cloud9 and switch to the “Shared with you” panel.

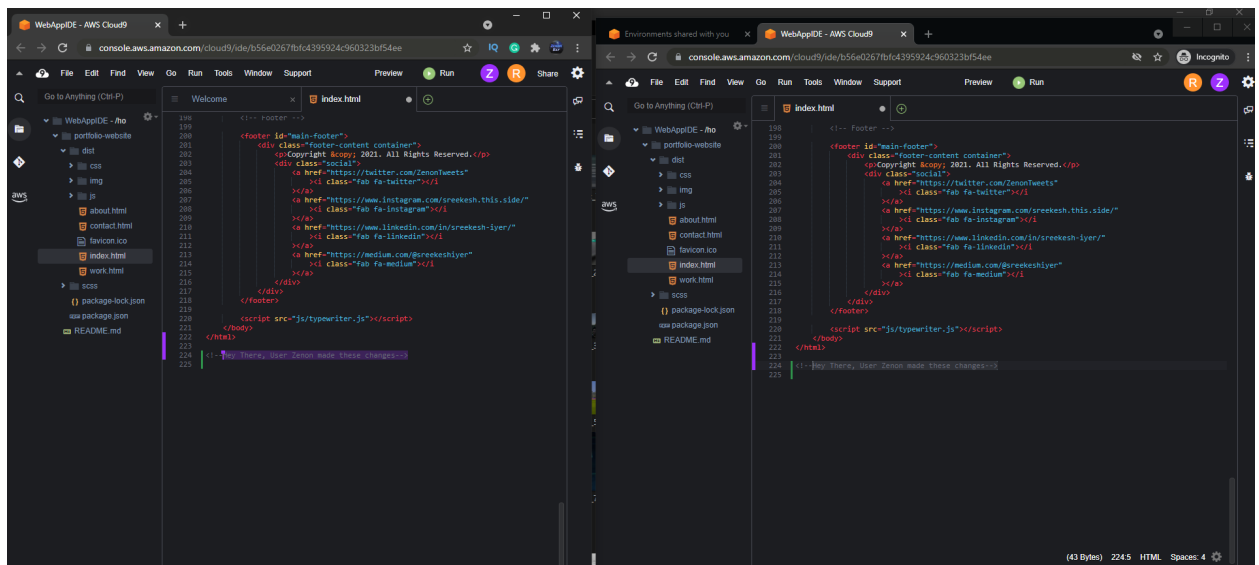


19. You should be able to see your environment. Click on Open IDE and you'll have almost the same UI as the root user, depending on your permissions and access options.

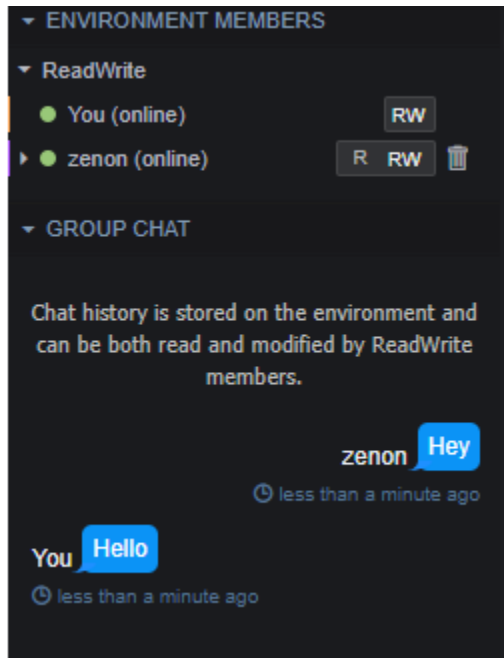
20. Go ahead and make a change to any of your files.



21. Both users now should be able to see real-time changes.



22. You can also open up the chat window on the right to communicate!



**For more info related to AWS-Cloud 9 you all can refer to the following documentation -**  
<https://docs.aws.amazon.com/cloud9/latest/user-guide/aws-cloud9-ug.pdf>

### **Conclusion:**

In this experiment, we learned how to use AWS Cloud9 to create an IDE and code in a collaborative environment, creating and managing IAM users, creating user groups, setting permissions, etc.