Name: Hrishikesh Kumbhar

Div: D15A

Roll no: 32

Sub: Advanced DevOps

Experiment No: 1

Date: 30/07/2022

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:-

What is AWS:-

AWS is a Cloud Computing Platform, which helps you build your application over the cloud. It offers various services like a combination of infrastructure and software services, along with the computing power, scalability, reliability and secure database storage.

The top 5 services provided by AWS are:-

- Amazon Elastic Cloud Compute (EC2)
- Amazon Simple Storage Service (S3)
- Amazon Virtual Private Cloud (VPC)
- Amazon CloudFront
- Amazon Relational Database Service (RDS).

Advantages of AWS:-

Companies and individuals prefer AWS as their cloud provider because of the numerous AWS benefits.

1) User Friendly:-

AWS is easy to use as the platform is specially designed for quick and secure access. User can modify their data whenever they want.

2) Flexible:-

Flexibility is the reason many companies prefer AWS. It lets your choose operating systems, programing languages and web application platforms that you are comfortable with. With a service like EC2, you can build your virtual computing environment by setting up your preferable operating systems and applications.

3) Secure:-

AWS provides a highly secure infrastructure to ensure the privacy of your data. Security professionals at AWS follow different layers of data serveillance such as:-

- Data Protection.
- Identity and access management.
- Infrastructure protection.
- Threat detection and continuous monitoring.
- Compliance and data privacy.

4) **Cost-effective:-**

AWS offers a pay-as-you-go pricing method, which means that a company will only pay for the services that it needs and has used for a period of time. AWS services are unique and cheaper than the traditional computing methods.

5) **Reliable**

Amazon offers the highest reliability for its customers. Services such as Amazon DynamoDB and amazon S3 store the data in three different availability zones so that even if two of them fail to work, the users will still have their data intact.

6) **Scalable and Elastic:-**

AWS is scalable because the AWS Auto Scaling services automatically increases the capacity of constrained resources as per requirements. so that application is always reliable. Spinning up servers is easy in AWS.

Elasticity is one of the AWS advantages. The upsizing and downsizing of the resources take place as per your requirement. Also, AWS always lets you know how many resources you are using at the moment.

What is AWS Cloud9 IDE?

AWS Cloud 9 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 languges including Javascript, Python, PHP and more, so you don't need to install files or configure your development machine to start a new project. Cloud9 IDE is cloud based, you can work on your project from anywhere. 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 application.

Benefits:-

1) 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. Provides browserbased shell experience with ability of Git.

2) Code together in real time:-

AWS Cloud9 makes collaborating on code easy. While collaborating, your team members can see each other type in real time, and instantly chat with one another from within the IDE.

3) 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.

4) Direct Terminal access to AWS:-

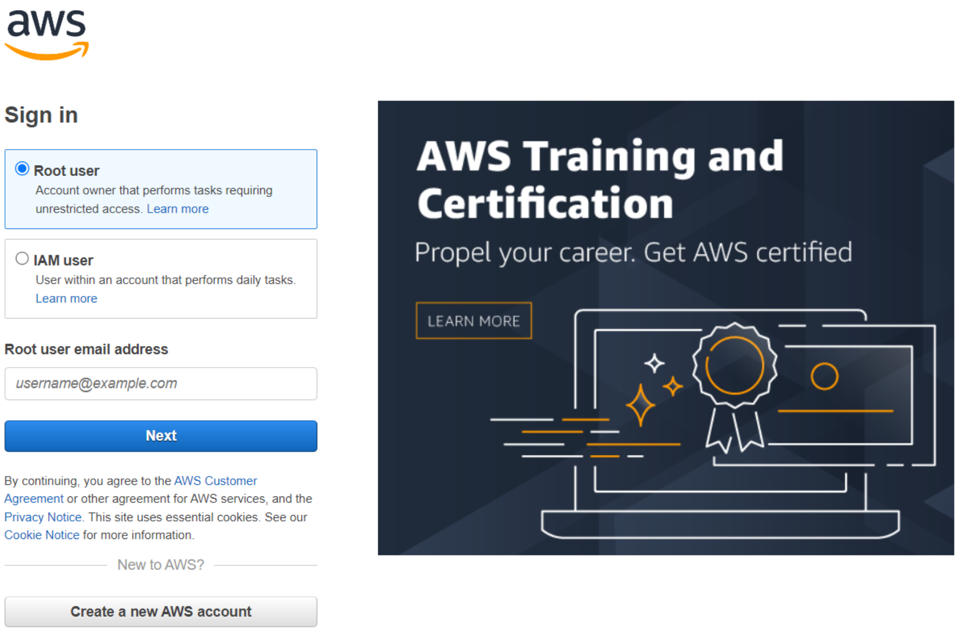
AWS cloud9 comes with a terminal that that includes sudo privileges to the managed Amazon EC2 instance that is hosting your development environment and a preauthenticated AWS command line interface.

5) Start New projects quickly:-

Cloud9's development environment comes prepackaged with tooling for over 40 programming languages. It elemenates the need of configuring files and installing ~~IDEs~~. SDKs.

Steps:

1. Login with your AWS account.



Account owner that performs tasks requiring unrestricted access. Learn more

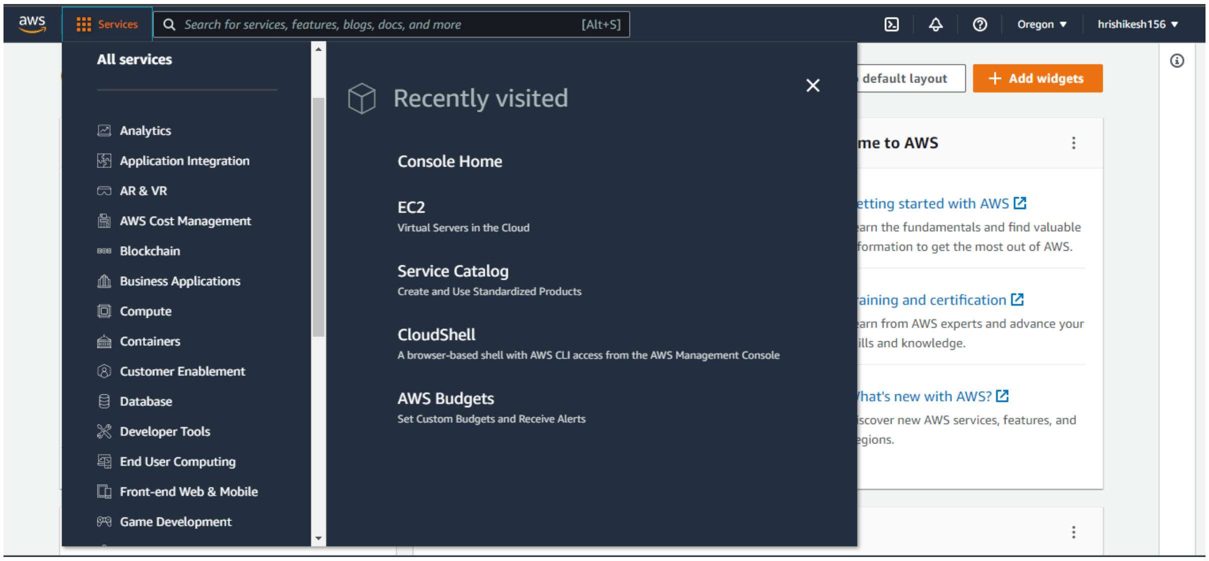
User within an account that performs daily tasks. Learn more

Root user email address: username@example.com

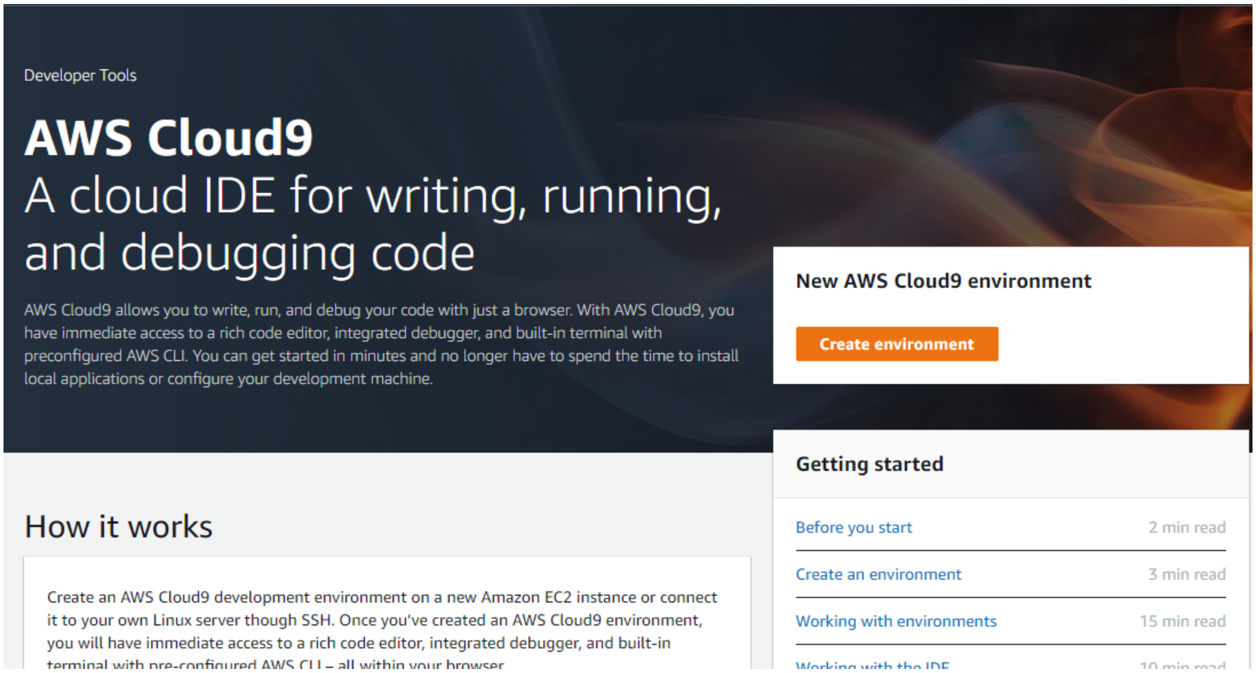
By continuing, you agree to the AWS Customer Agreement or other agreement for AWS services, and the Privacy Notice. This site uses essential cookies. See our Cookie Notice for more information.

New to AWS?
Create a new AWS account
AWS Training and Certification
Propel your career. Get AWS certified
LEARN MORE

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



3. Go to Developer tools and then click cloud 9 and click on create environment.



Developer Tools

AWS Cloud9

A cloud IDE for writing, running, and debugging code

AWS Cloud9 allows you to write, run, and debug your code with just a browser. With AWS Cloud9, you have immediate access to a rich code editor, integrated debugger, and built-in terminal with preconfigured AWS CLI. You can get started in minutes and no longer have to spend the time to install local applications or configure your development machine.

New AWS Cloud9 environment

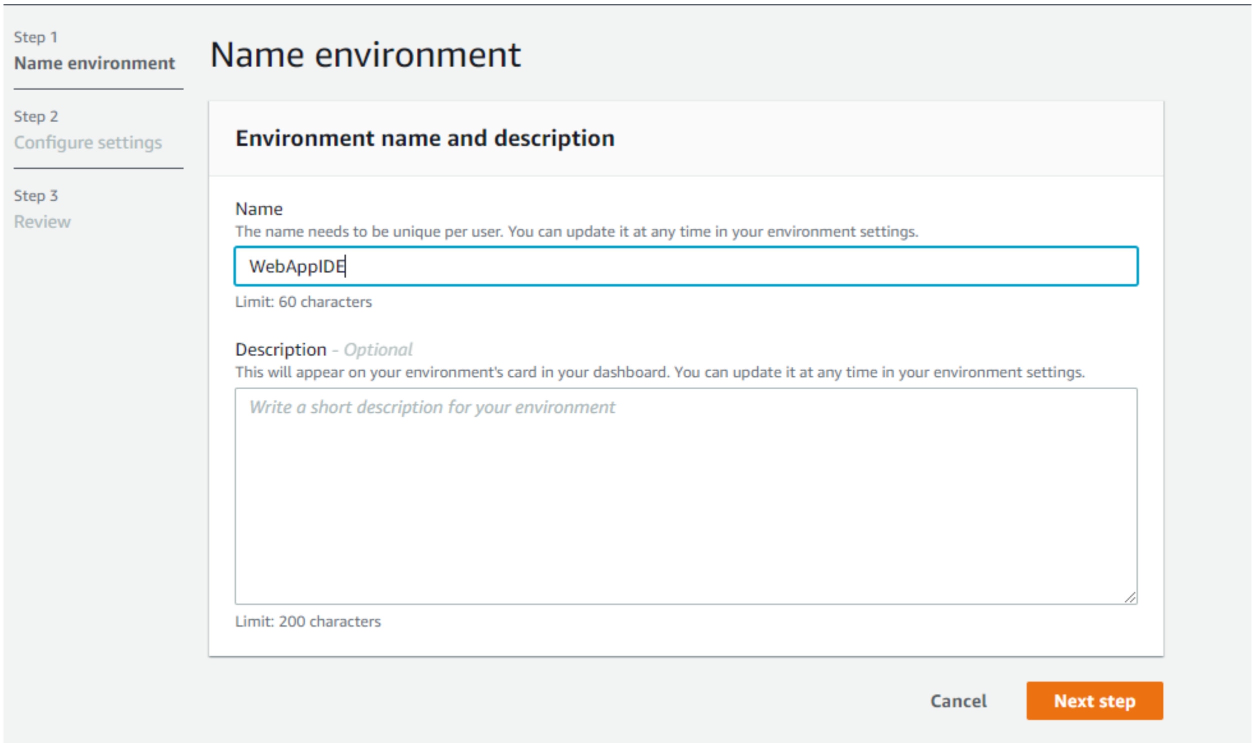
How it works

Create an AWS Cloud9 development environment on a new Amazon EC2 instance or connect it to your own Linux server though SSH. Once you've created an AWS Cloud9 environment, you will have immediate access to a rich code editor, integrated debugger, and built-in terminal with pre-configured AWS CLI – all within your browser

Getting started

Before you start	2 min read
Create an environment	3 min read
Working with environments	15 min read
Working with the IDE	10 min read

4. Provide the name for the Environment (WebAppIDE) and 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.

WebAppIDE

Limit: 60 characters

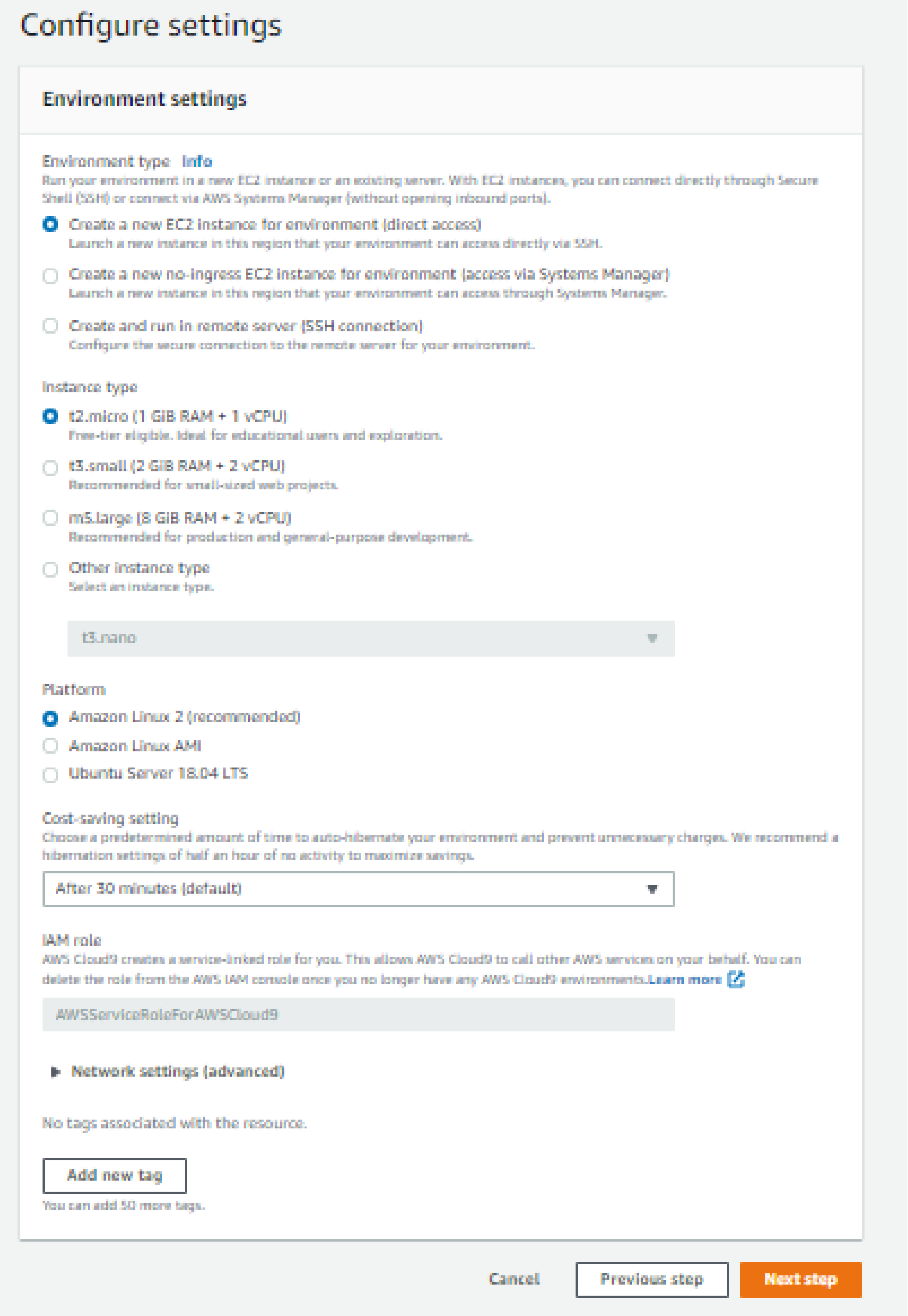
Description - *Optional*

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

5. Keep all the Default settings as shown in 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

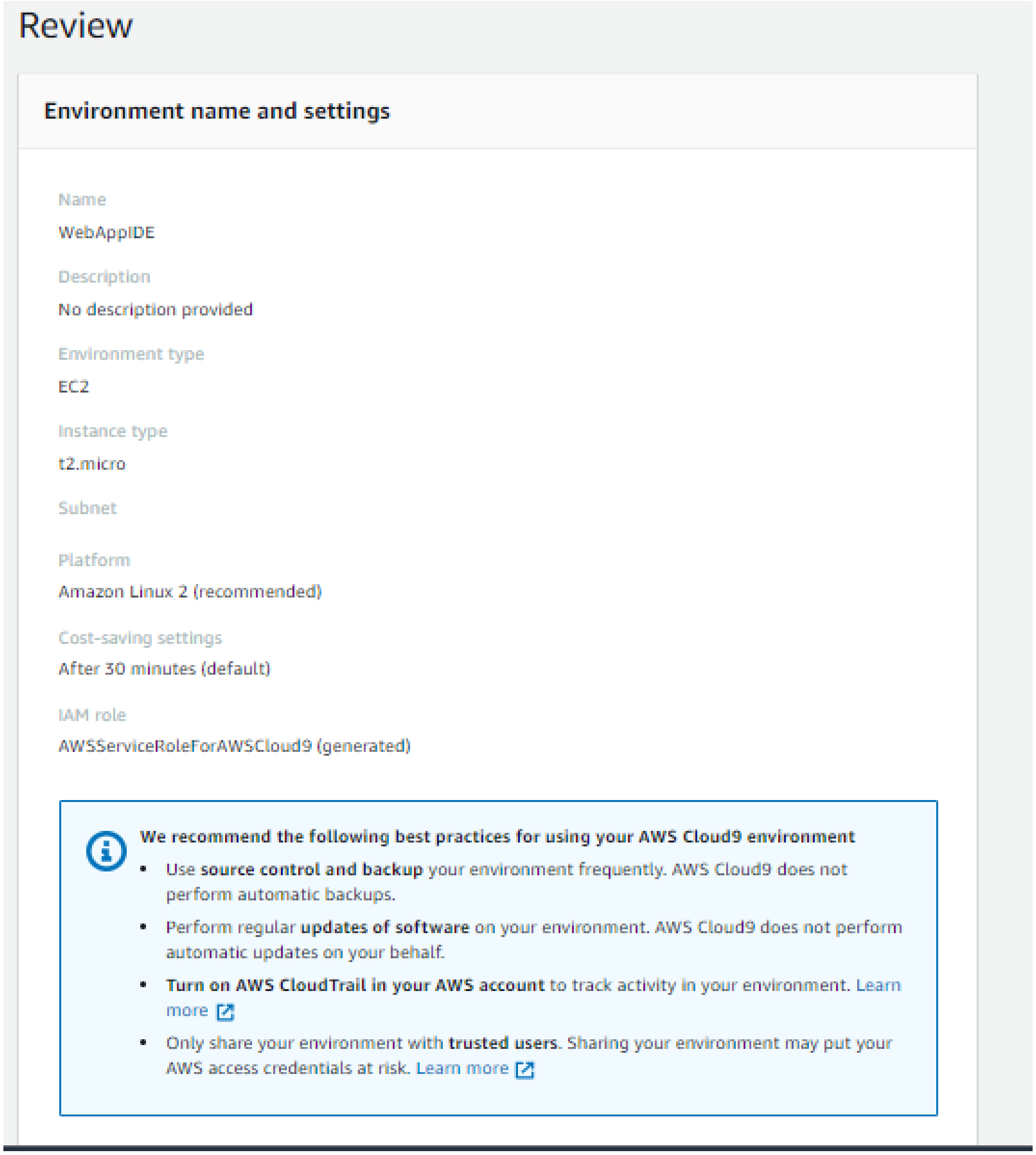
▶ Network settings (advanced)

No tags associated with the resource.

Add new tag

You can add 50 more tags.

6. Review the Environment name and Settings and click on Create 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)

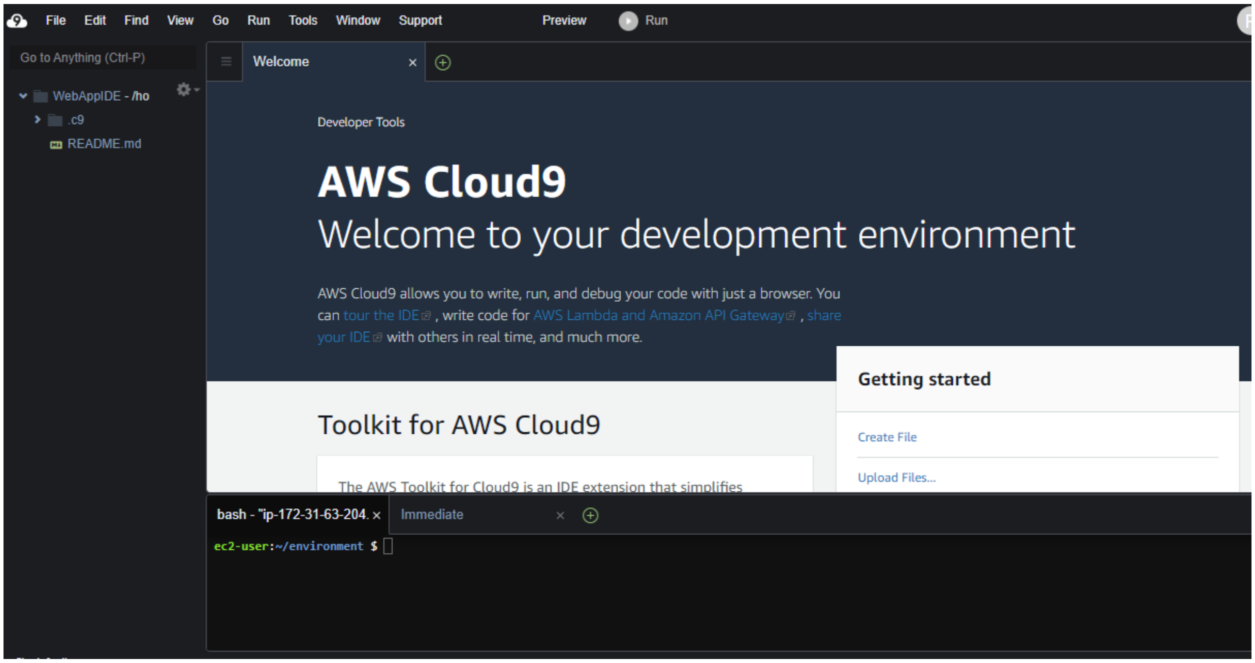
Cost-saving settings
After 30 minutes (default)

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

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



Developer Tools

AWS Cloud9

Welcome to your development environment

AWS Cloud9 allows you to write, run, and debug your code with just a browser. You can tour the IDE, write code for AWS Lambda and Amazon API Gateway, share your IDE with others in real time, and much more.

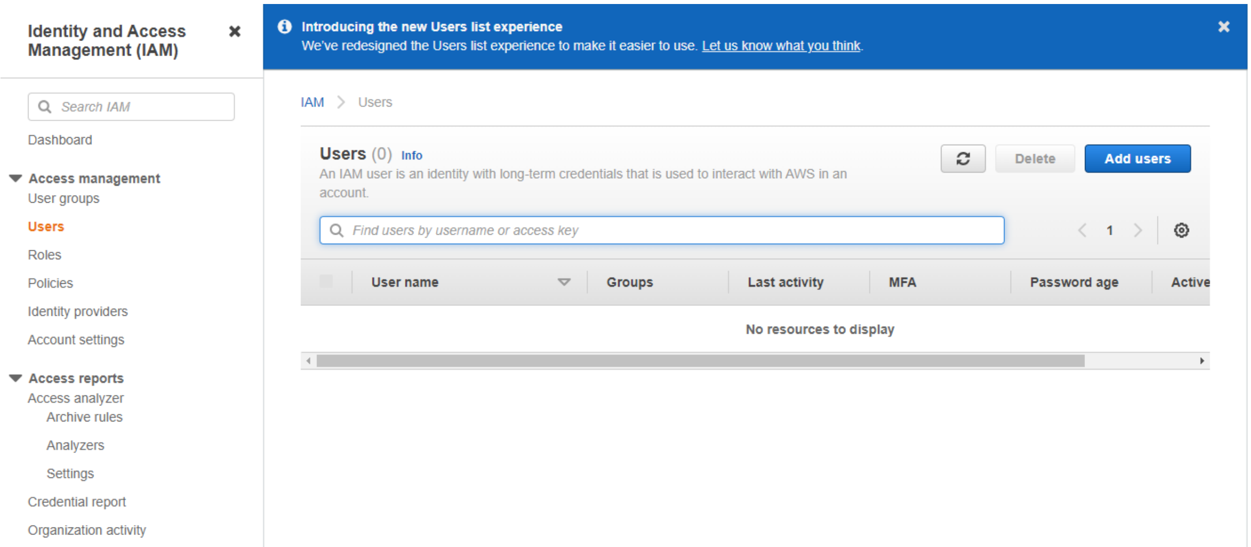
Getting started

Toolkit for AWS Cloud9

The AWS Toolkit for Cloud9 is an IDE extension that simplifies

```
ec2-user:~/environment $
```

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



Introducing the new Users list experience
We've redesigned the Users list experience to make it easier to use. Let us know what you think.

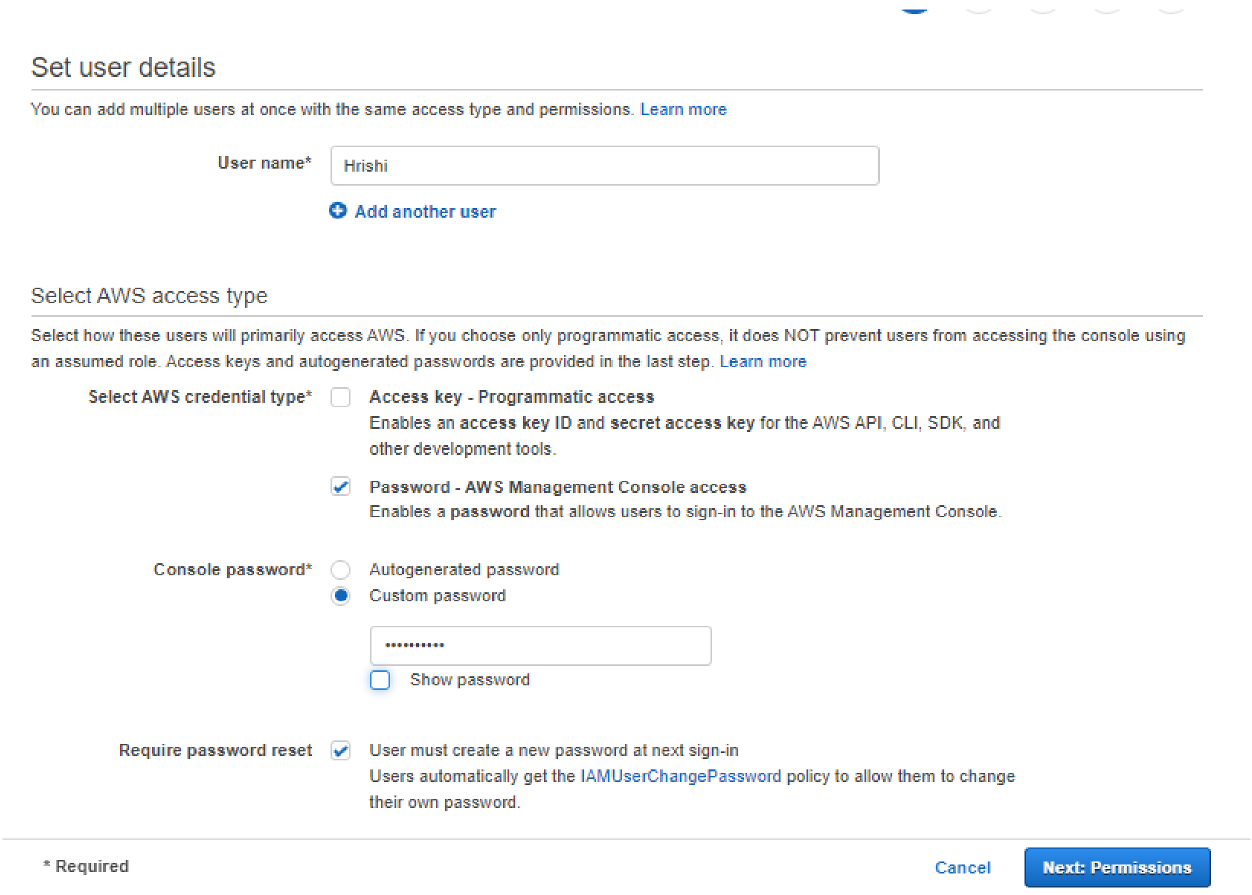
Users (0)

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

User name	Groups	Last activity	MFA	Password age	Active

No resources to display

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



Set user details

You can add multiple users at once with the same access type and permissions. Learn more

User name* Hrishi

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

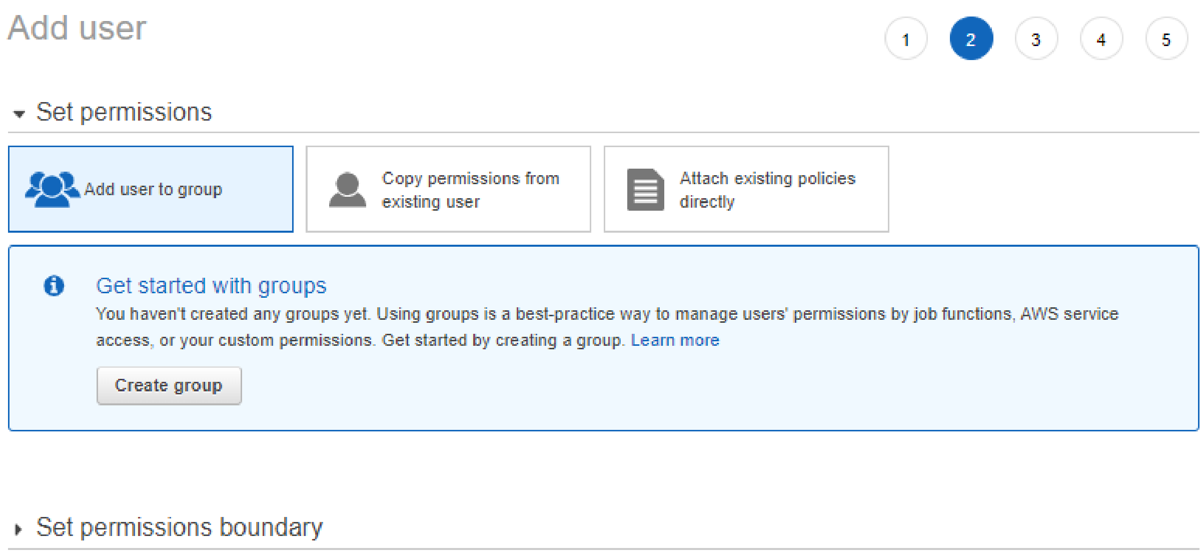
••••••••••

□ 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

9. Click on Create group



Add user

Set permissions

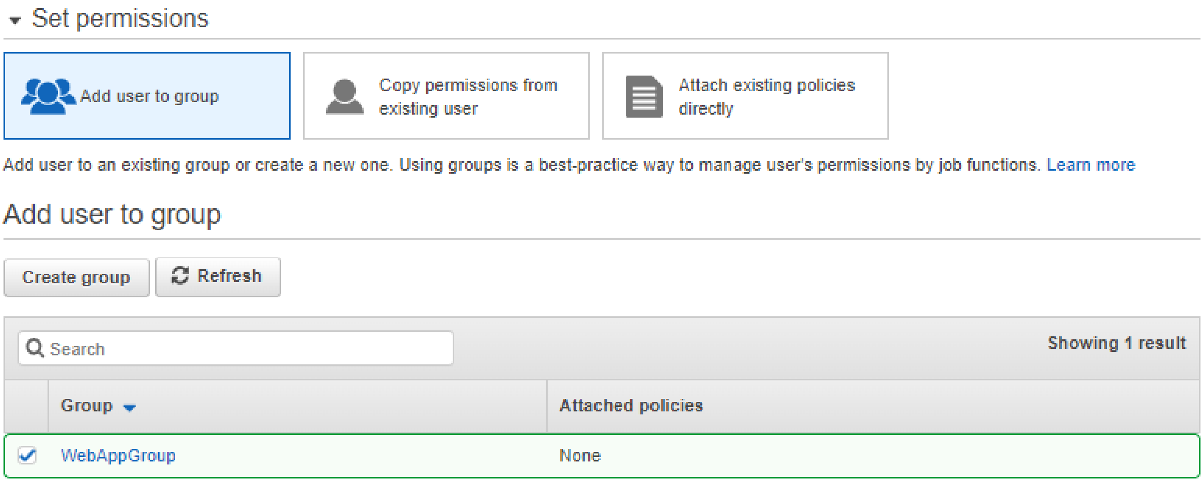
Add user to group
Copy permissions from existing user
Attach existing policies directly

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

Set permissions boundary

10. Provide group name and click on create group.



▾ Set permissions

Add user to group
Copy permissions from existing user
Attach existing policies directly

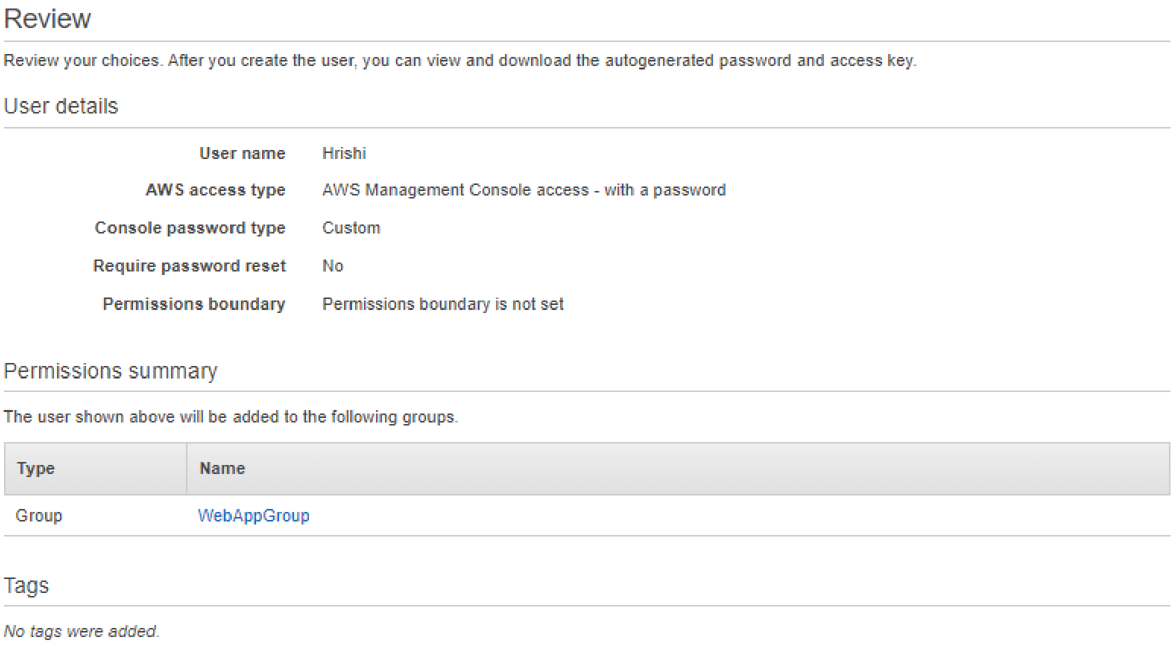
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more

Add user to group

Create group
Refresh
Search
Showing 1 result

	Group	Attached policies
☑	WebAppGroup	None

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.



Review

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

User details

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

Permissions summary

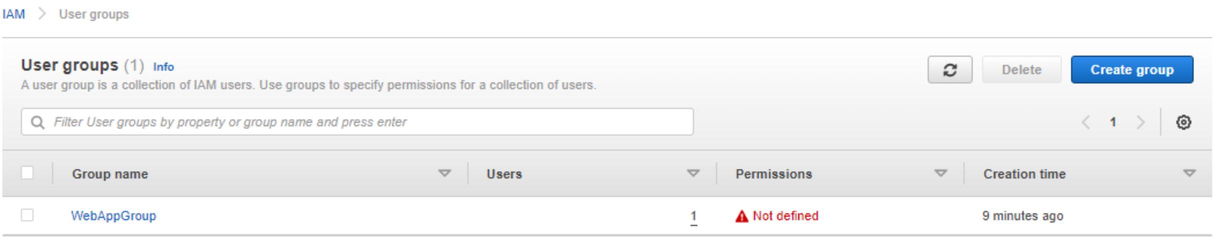
The user shown above will be added to the following groups.

Type	Name
Group	WebAppGroup

Tags

No tags were added.

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

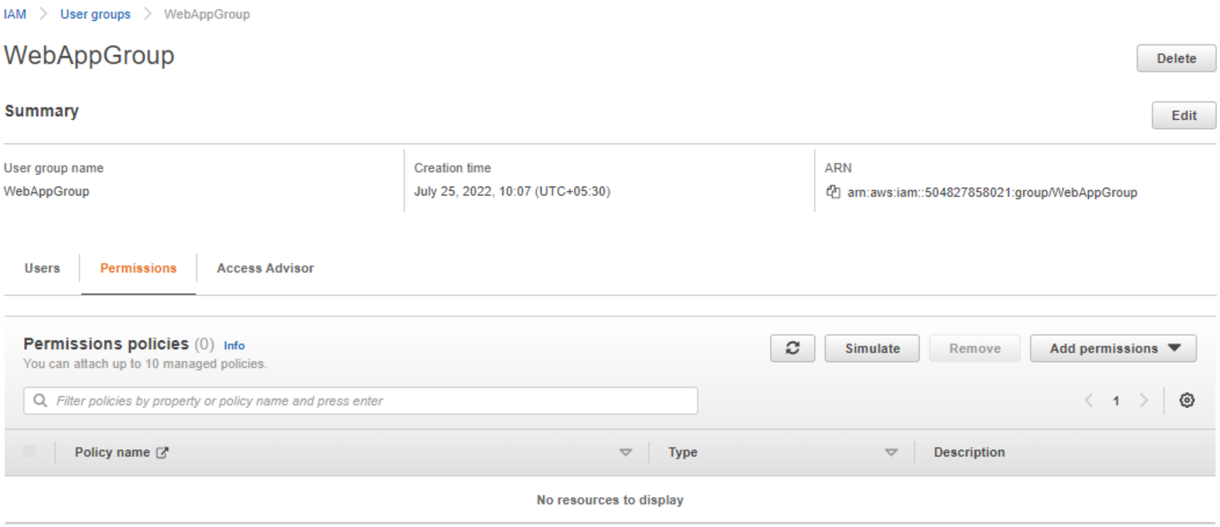

IAM > User groups
Delete
Create group
Filter User groups by property or group name and press enter
1

User groups (1) Info

A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.

Group name	Users	Permissions	Creation time
WebAppGroup	1	⚠ Not defined	9 minutes ago

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


IAM > User groups > WebAppGroup
Delete
Edit
Users
Permissions
Access Advisor
Simulate
Remove
Add permissions
Filter policies by property or policy name and press enter
1

WebAppGroup

Summary

User group name: WebAppGroup

Creation time: July 25, 2022, 10:07 (UTC+05:30)

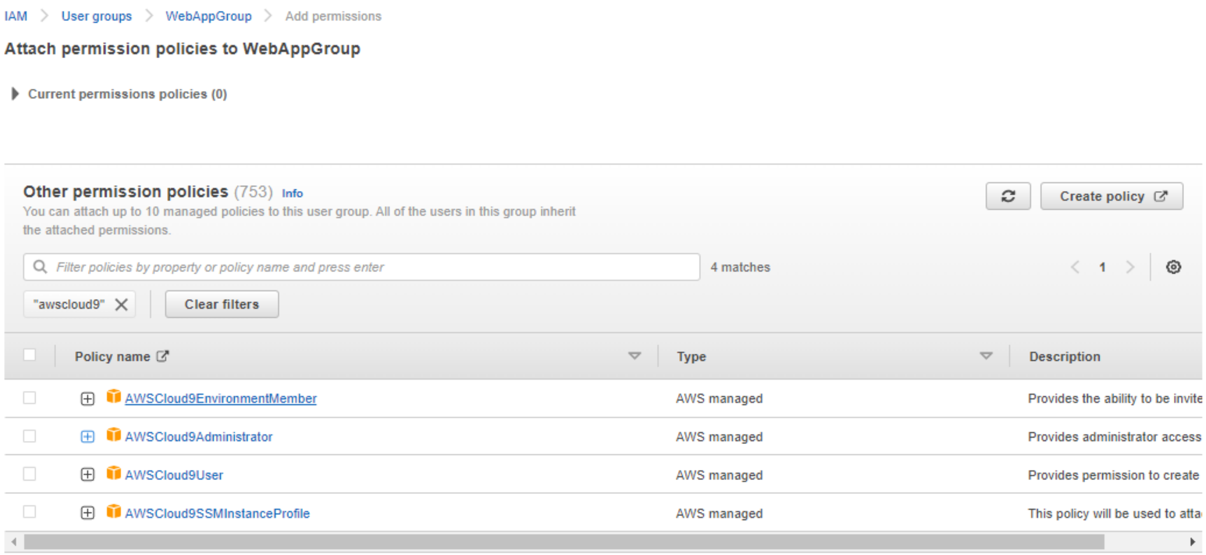
ARN: arn:aws:iam::504827858021:group/WebAppGroup

Permissions policies (0) Info

You can attach up to 10 managed policies.

Policy name	Type	Description
No resources to display		

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



IAM > User groups > WebAppGroup > Add permissions

Attach permission policies to WebAppGroup

Current permissions policies (0)

Other permission policies (753) Info

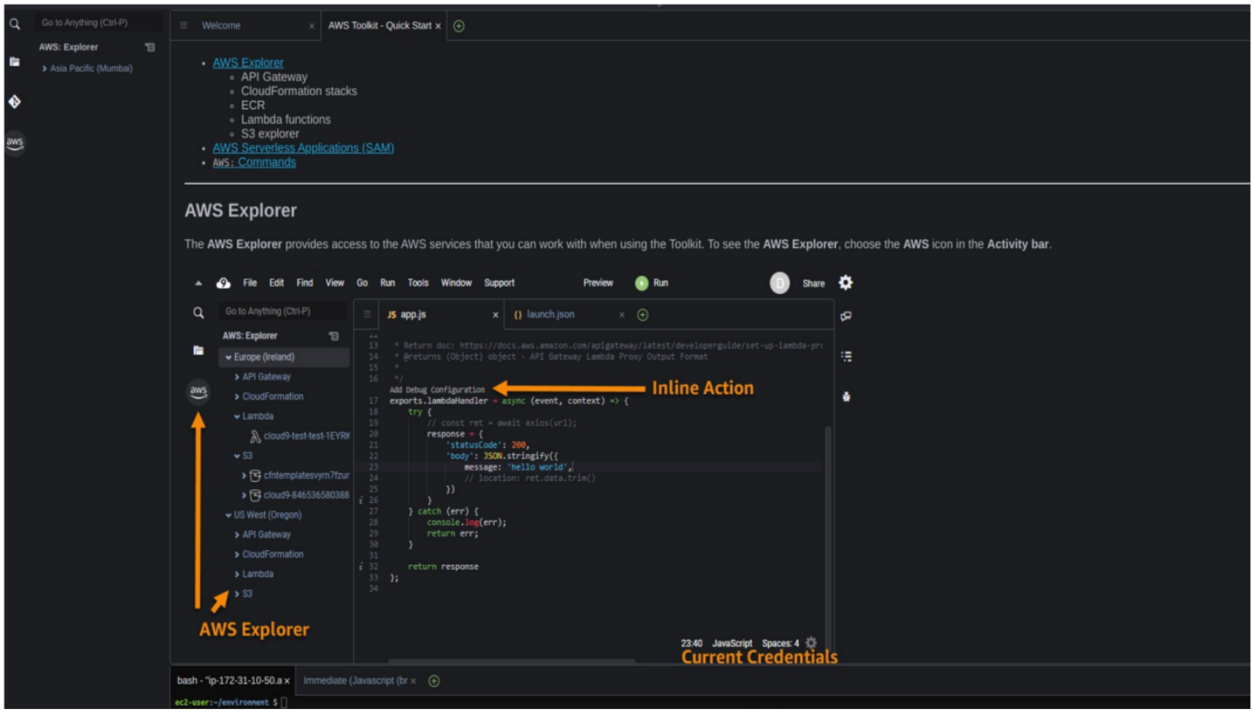
You can attach up to 10 managed policies to this user group. All of the users in this group inherit the attached permissions.

Filter policies by property or policy name and press enter — 4 matches

"awscloud9" ✕ — Clear filters

□	Policy name	Type	Description
□	AWSCloud9EnvironmentMember	AWS managed	Provides the ability to be invite
□	AWSCloud9Administrator	AWS managed	Provides administrator access
□	AWSCloud9User	AWS managed	Provides permission to create
□	AWSCloud9SSMInstanceProfile	AWS managed	This policy will be used to atta

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



- AWS Explorer
  - API Gateway
  - CloudFormation stacks
  - ECR
  - Lambda functions
  - S3 explorer
- AWS Serverless Applications (SAM)
- AWS: Commands

AWS Explorer

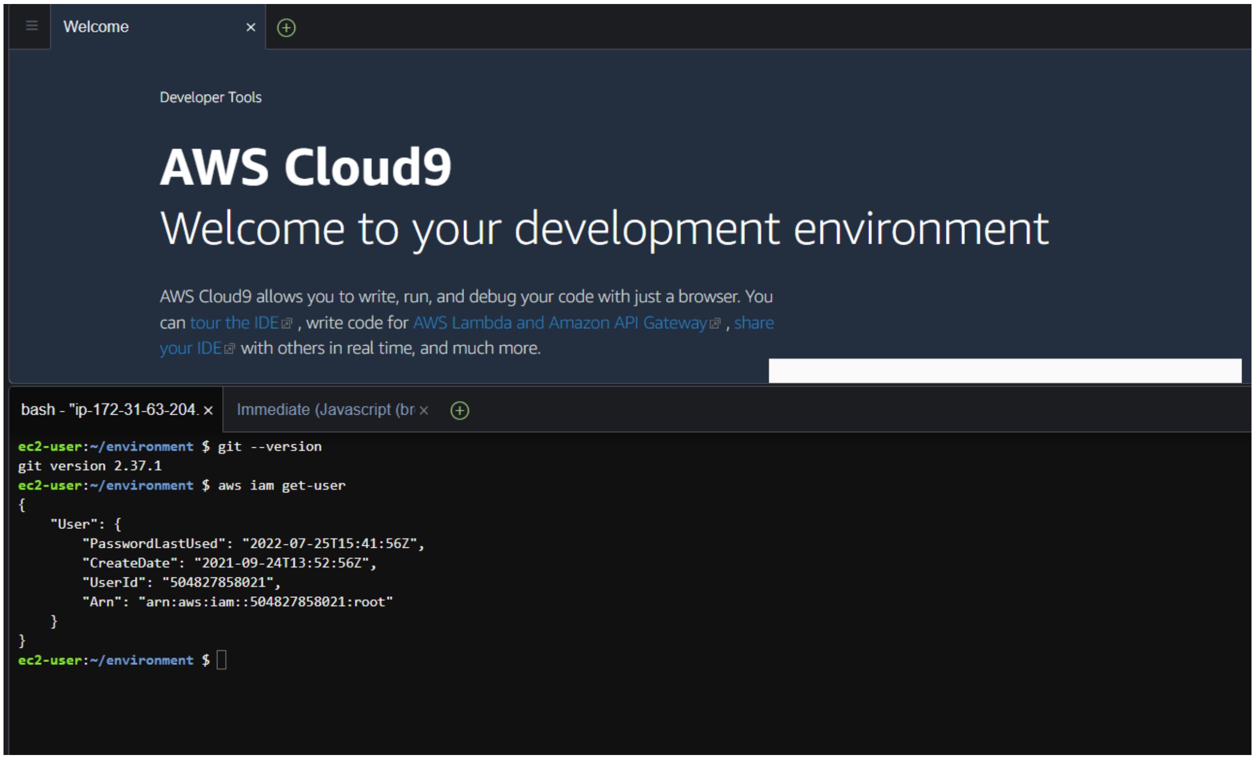
The **AWS Explorer** provides access to the AWS services that you can work with when using the Toolkit. To see the **AWS Explorer**, choose the **AWS** icon in the **Activity bar**.

```
 * Return doc: https://docs.aws.amazon.com/apigateway/latest/developerguide/set-up-lambda-pr
 * @returns {Object} object - API Gateway Lambda Proxy Output Format
 *
 */
Add Debug Configuration
exports.lambdaHandler = async (event, context) => {
    try {
        // const ret = await axios(url);
        response = {
            'statusCode': 200,
            'body': JSON.stringify({
                message: 'hello world',
                // location: ret.data.trim()
            })
        }
    } catch (err) {
        console.log(err);
        return err;
    }

    return response
};
```

```
ec2-user:~/environment $
```

16. If you check at the bottom side Cloud9 IDE also gives you and aws CLI for command operations: as we here checked the git version, iam user details and so on…



Developer Tools

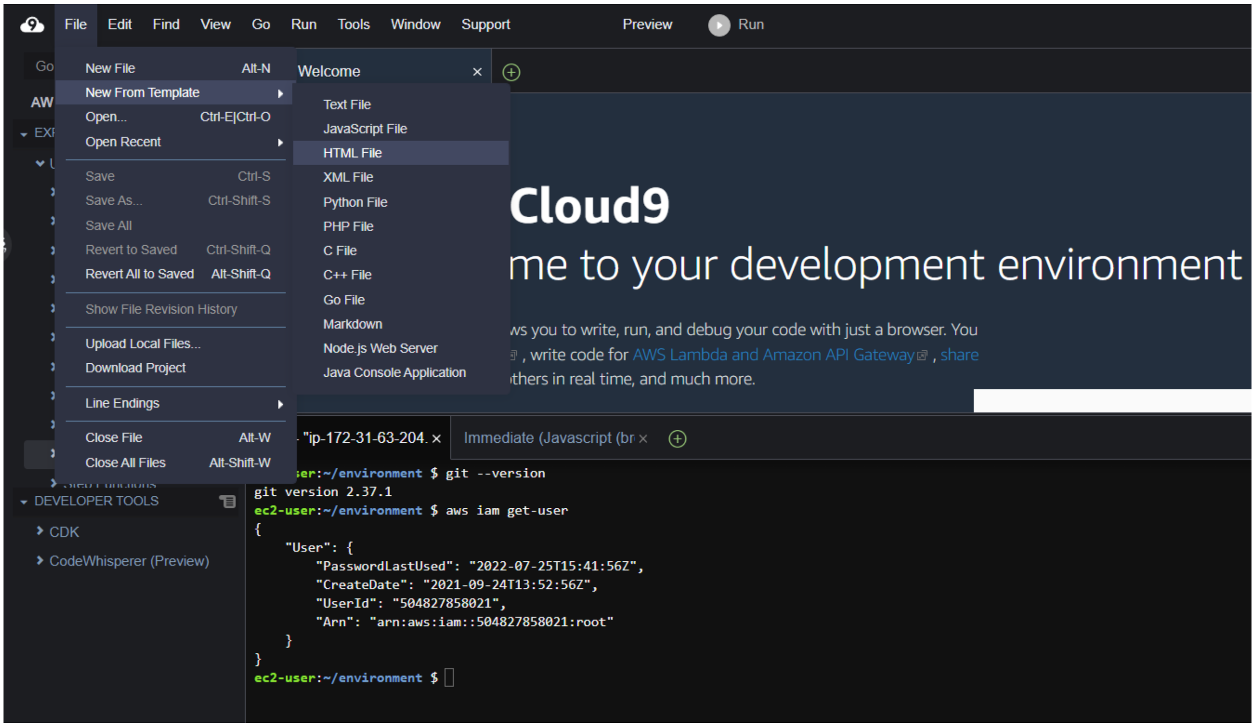
AWS Cloud9

Welcome to your development environment

AWS Cloud9 allows you to write, run, and debug your code with just a browser. You can tour the IDE, write code for AWS Lambda and Amazon API Gateway, share your IDE with others in real time, and much more.

```
ec2-user:~/environment $ git --version
git version 2.37.1
ec2-user:~/environment $ aws iam get-user
{
    "User": {
        "PasswordLastUsed": "2022-07-25T15:41:56Z",
        "CreateDate": "2021-09-24T13:52:56Z",
        "UserId": "504827858021",
        "Arn": "arn:aws:iam::504827858021:root"
    }
}
ec2-user:~/environment $ 
```

17. Now we will set up a collaborative environment. Click on File you can create a new file or choose from template, here opting for an html file to collaborate.



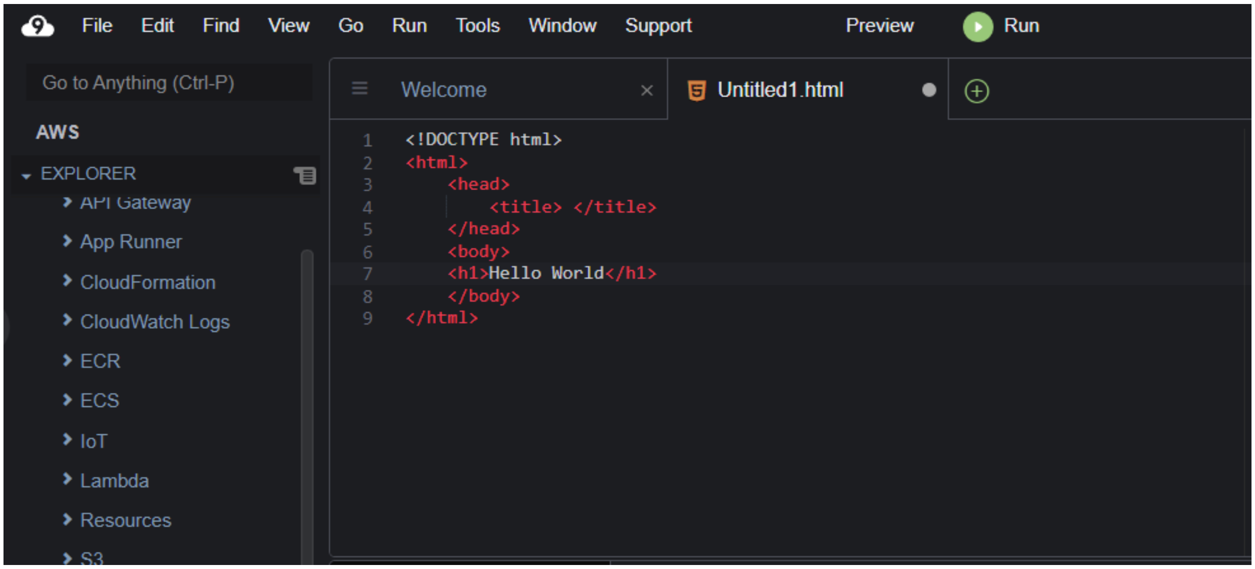
Cloud9

me to your development environment

ws you to write, run, and debug your code with just a browser. You , write code for AWS Lambda and Amazon API Gateway , share thers in real time, and much more.

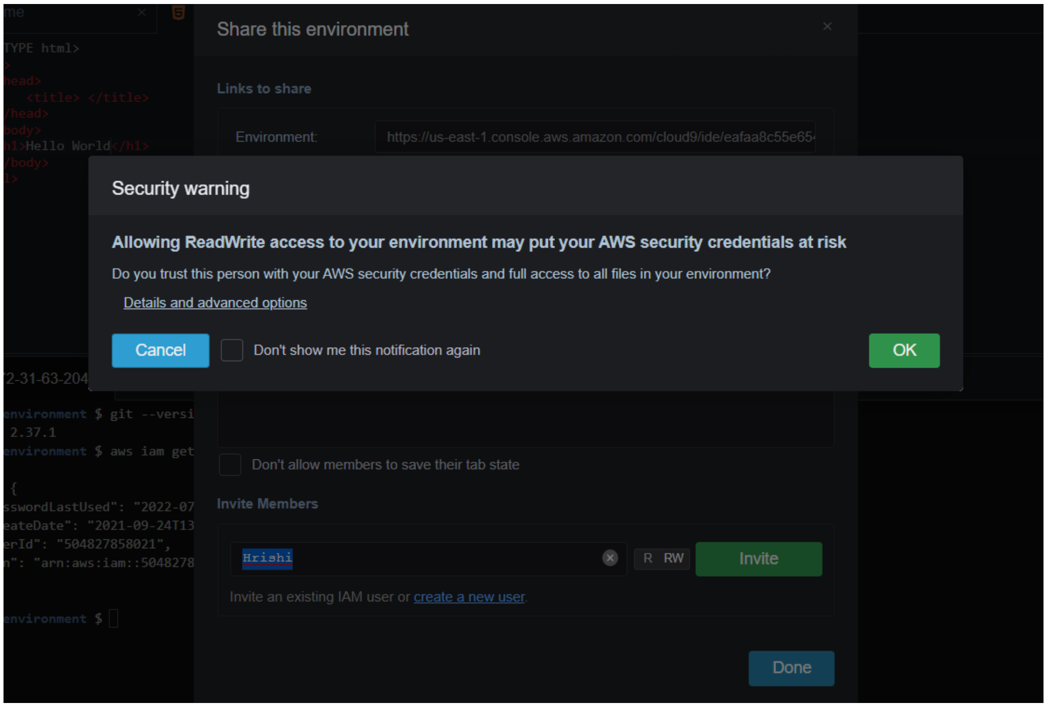
```
ser:~/environment $ git --version
git version 2.37.1
ec2-user:~/environment $ aws iam get-user
{
    "User": {
        "PasswordLastUsed": "2022-07-25T15:41:56Z",
        "CreateDate": "2021-09-24T13:52:56Z",
        "UserId": "504827858021",
        "Arn": "arn:aws:iam::504827858021:root"
    }
}
ec2-user:~/environment $
```

18. Edit html file and save it.



```
<!DOCTYPE html>
<html>
    <head>
        <title> </title>
    </head>
    <body>
    <h1>Hello World</h1>
    </body>
</html>
```

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.



```
TYPE html>
>
head>
    <title> </title>
/head>
body>
h1>Hello World</h1>
/body>
l>
```

```
2-31-63-204
environment $ git --versi
 2.37.1
environment $ aws iam get

 {
sswordLastUsed": "2022-07
eateDate": "2021-09-24T13
erId": "504827858021",
n": "arn:aws:iam::5048278

environment $
```

Share this environment

Links to share

Environment: https://us-east-1.console.aws.amazon.com/cloud9/ide/eafaa8c55e65

Security warning

Allowing ReadWrite access to your environment may put your AWS security credentials at risk

Do you trust this person with your AWS security credentials and full access to all files in your environment?

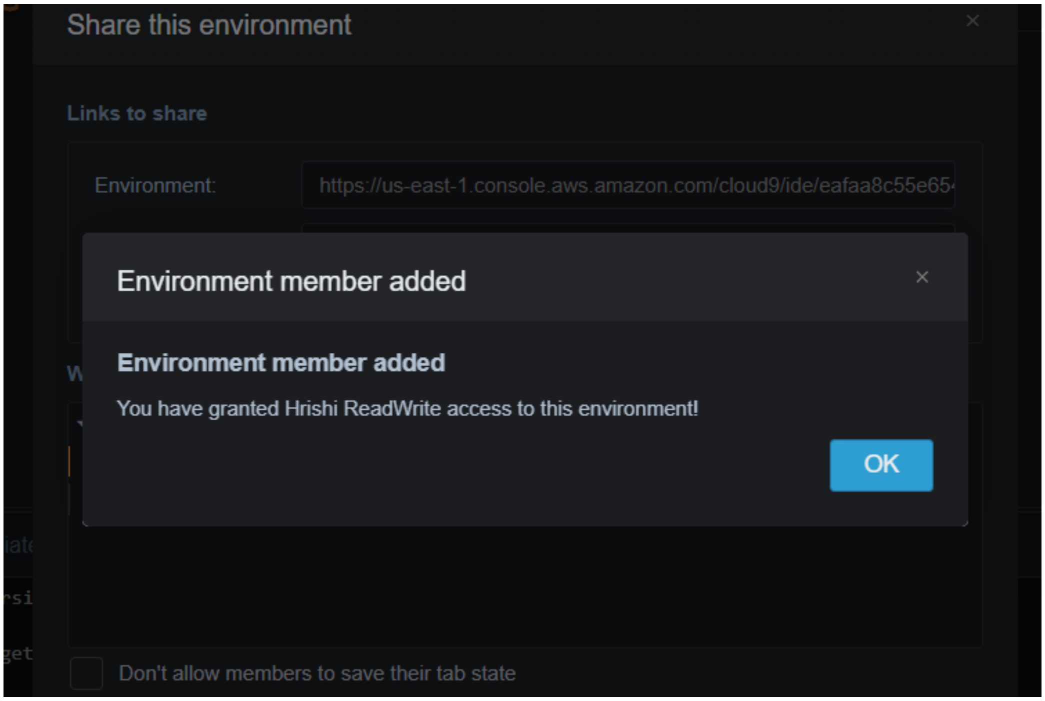
Details and advanced options

□ Don't allow members to save their tab state

Invite Members

Hrishi

Invite an existing IAM user or create a new user



Share this environment

Links to share

Environment: https://us-east-1.console.aws.amazon.com/cloud9/ide/eafaa8c55e65

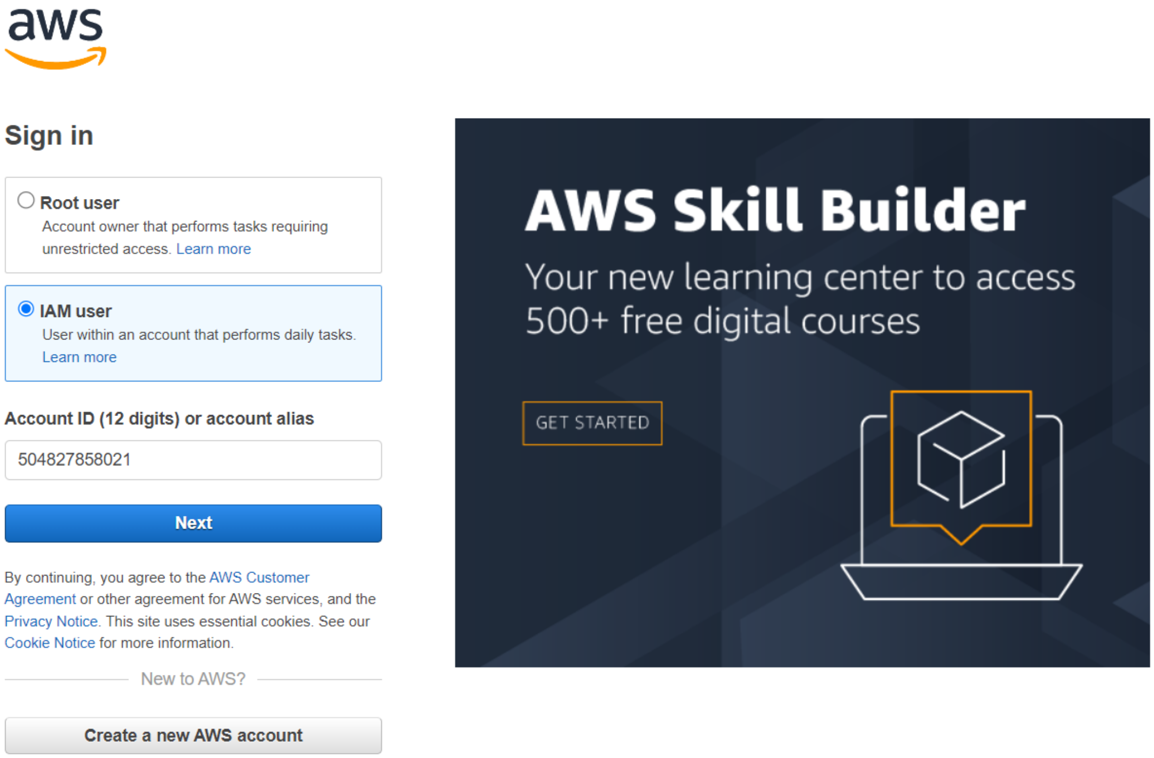
Environment member added

Environment member added

You have granted Hrishi ReadWrite access to this environment!

□ Don't allow members to save their tab state

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



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

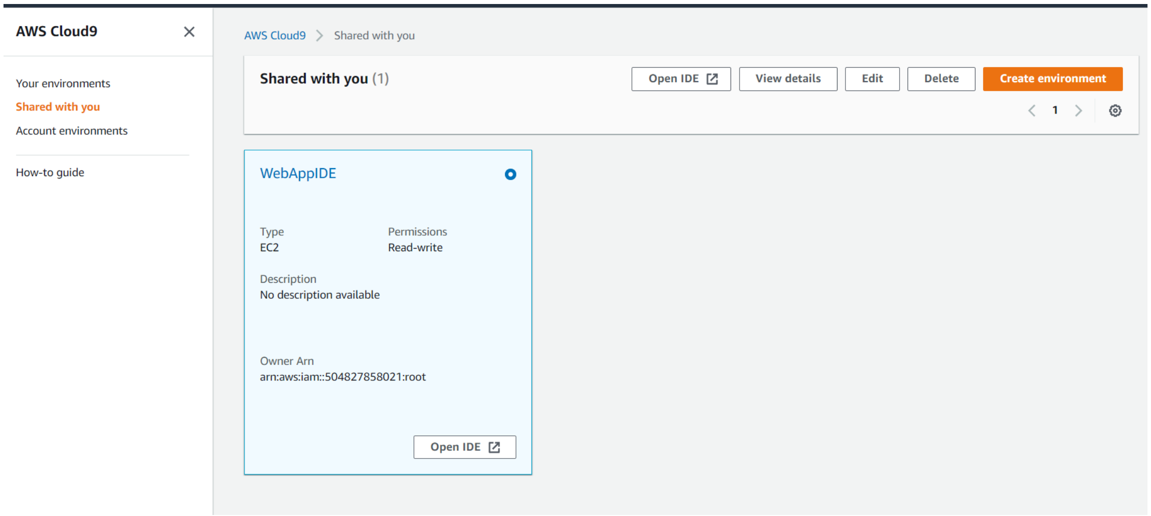
Account ID (12 digits) or account alias

504827858021

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 you environment to collaborate.



Shared with you (1)

WebAppIDE

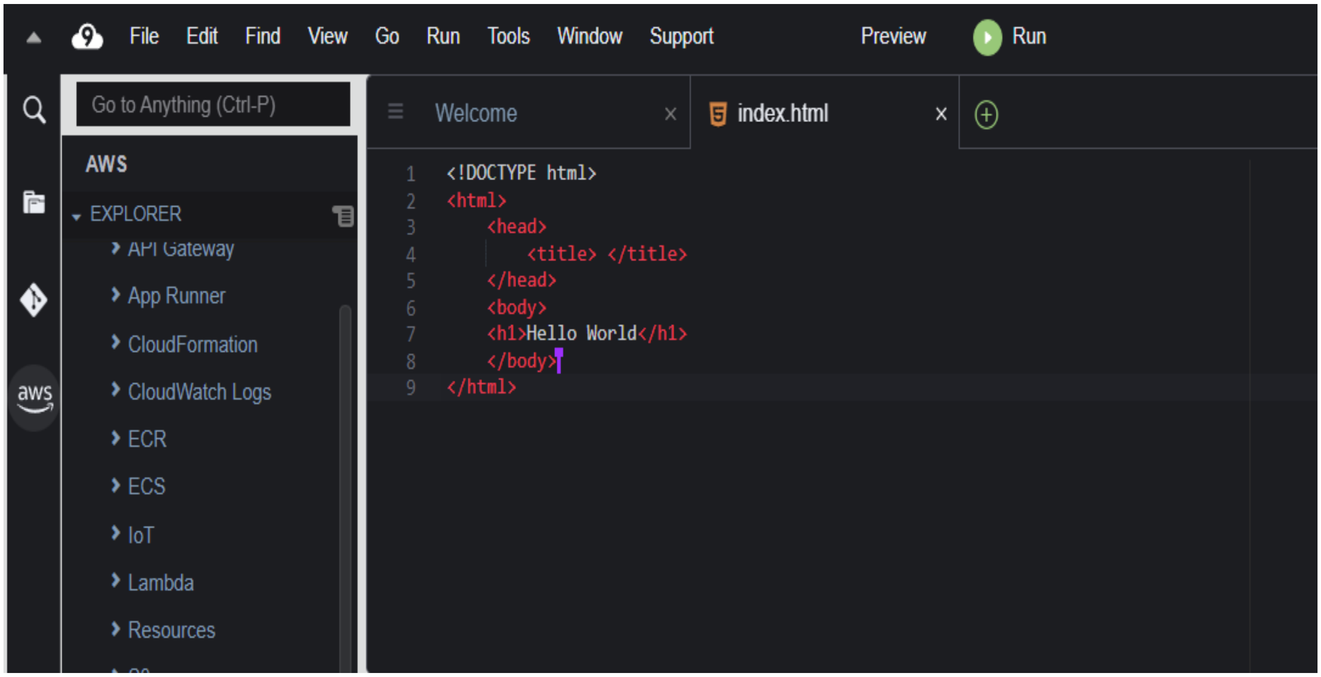
Type
EC2

Permissions
Read-write

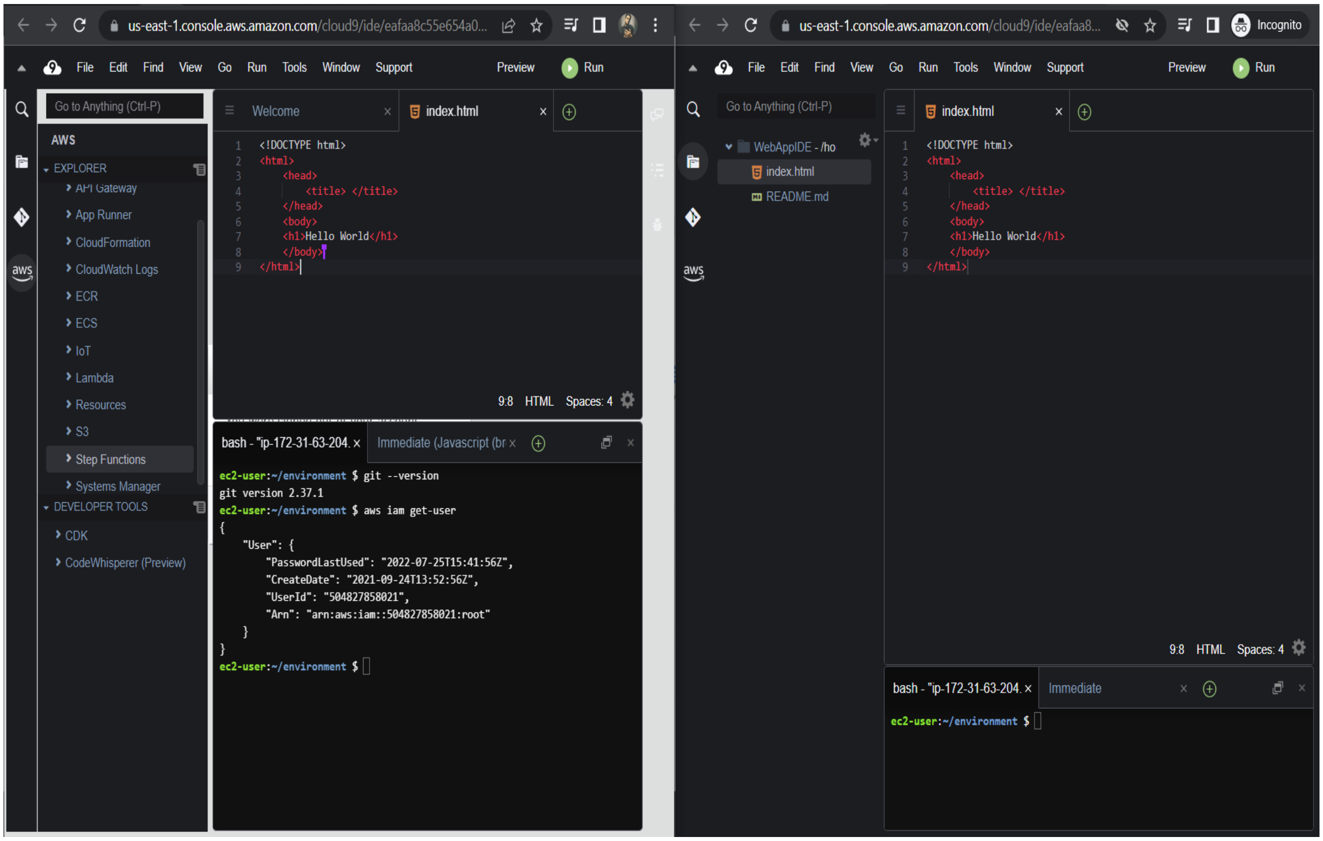
Description
No description available

Owner Arn
arn:aws:iam::504827858021:root

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



```html
<!DOCTYPE html>
<html>
    <head>
        <title> </title>
    </head>
    <body>
    <h1>Hello World</h1>
    </body>
</html>
```

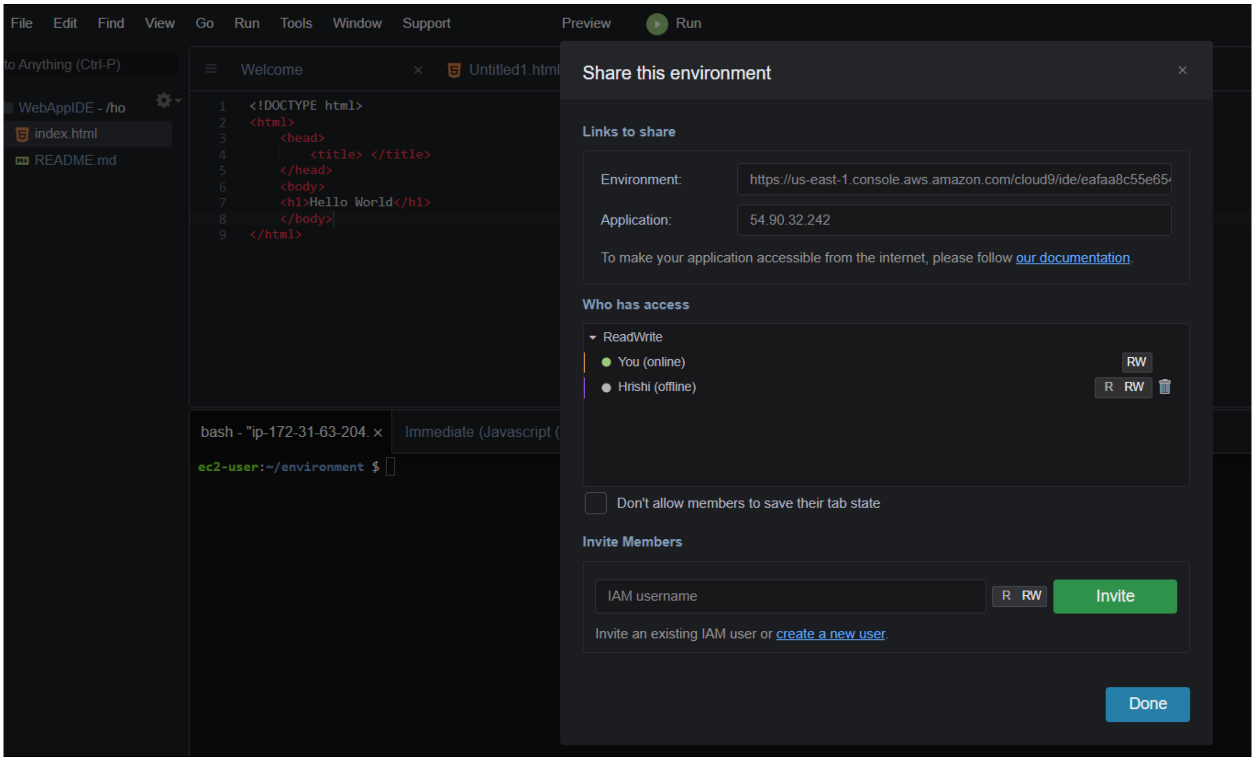


```html
<!DOCTYPE html>
<html>
    <head>
        <title> </title>
    </head>
    <body>
    <h1>Hello World</h1>
    </body>
</html>
```

```
ec2-user:~/environment $ git --version
git version 2.37.1
ec2-user:~/environment $ aws iam get-user
{
    "User": {
        "PasswordLastUsed": "2022-07-25T15:41:56Z",
        "CreateDate": "2021-09-24T13:52:56Z",
        "UserId": "504827858021",
        "Arn": "arn:aws:iam::504827858021:root"
    }
}
ec2-user:~/environment $
```

```
ec2-user:~/environment $
```

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



```
<!DOCTYPE html>
<html>
    <head>
        <title> </title>
    </head>
    <body>
    <h1>Hello World</h1>
    </body>
</html>
```

```
ec2-user:~/environment $
```

Share this environment

Links to share

Environment: https://us-east-1.console.aws.amazon.com/cloud9/ide/eafaa8c55e65

Application: 54.90.32.242

To make your application accessible from the internet, please follow our documentation.

Who has access

ReadWrite

● You (online) RW

● Hrishi (offline) R RW

☐ Don't allow members to save their tab state

Invite Members

IAM username R RW Invite

Invite an existing IAM user or create a new user.

Done

Conclusion:-

We learnt to use the basic cloud9 services provided by AWS. We learnt to add IAM users into the group in order to allow them to collaborate on the project. We also learnt about various permissions given to the IAM users.