

Project 1:

AWS CodePipeline is a fully managed continuous delivery service that helps you automate your release pipelines for fast and reliable application and infrastructure updates. CodePipeline automates the build, test, and deploy phases of your release process every time there is a code change, based on the release model you define. This enables you to rapidly and reliably deliver features and updates. You can easily integrate AWS CodePipeline with third-party services such as GitHub or with your own custom plugin. With AWS CodePipeline, you only pay for what you use. There are no upfront fees or long-term commitments

This activity guide cover steps for:

1. Create a CodeCommit repository
2. Add sample code to your CodeCommit repository
3. Create an EC2 Linux instance and install the CodeDeploy agent
4. To launch an instance
5. Create an application in CodeDeploy
6. Create your first pipeline in CodePipeline
7. To verify that your pipeline ran successfully
8. Modify code in your CodeCommit repository
9. To verify your pipeline ran successfully
10. Clean up resources

a. Unzip the files from

https://docs.aws.amazon.com/codepipeline/latest/userguide/samples/SampleApp_Linux.zip into the local directory (for example, /tmp/MyDemoRepo or c:\temp\MyDemoRepo).

Be sure to place the files directly into your local repository. Do not include a SampleApp_Linux folder. On your local machine for example, your directory and file hierarchy should look like this:

b. Use git commands to upload the code in the local directory to public repository in Codecommit

```
| -- appspec.yml
```

```
| -- index.html
```

```
|-- LICENSE.txt
L-- scripts
    |-- install_dependencies
    |-- start_server
    L-- stop_server
```

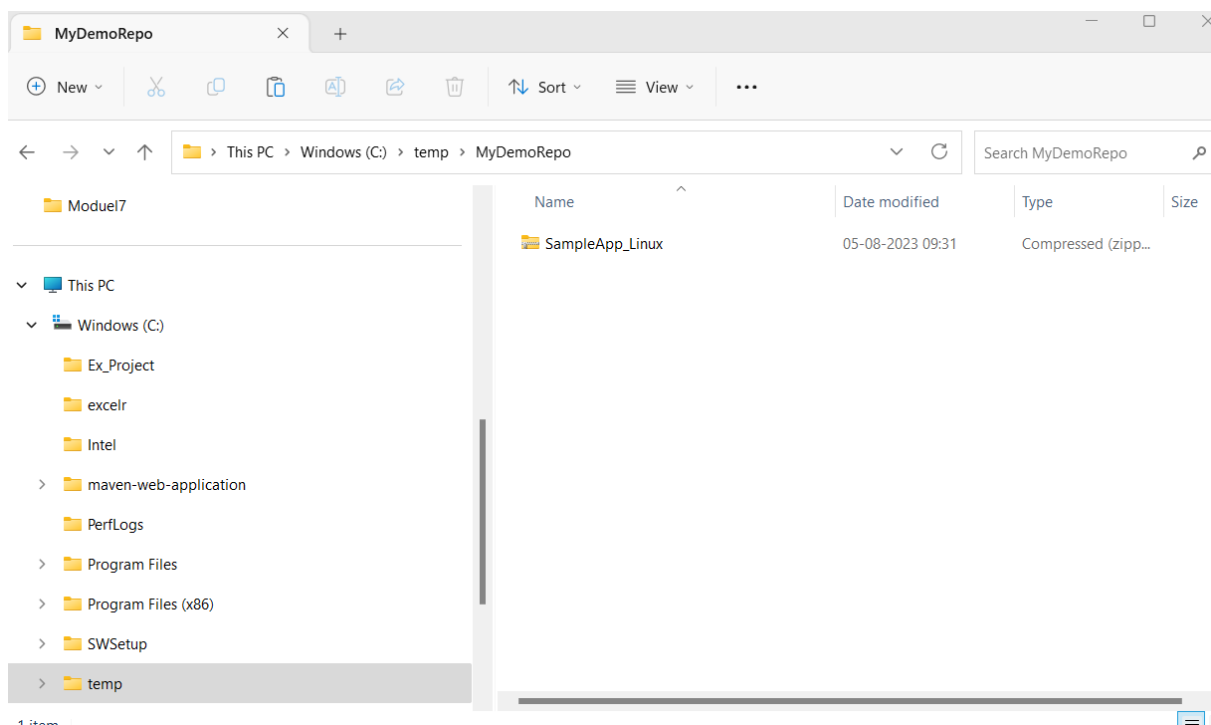
c.Create and configure Code Deploy (Amazon EC2 instance)

d. Create a pipeline using AWS CodePipeline , AWS Code Commit and Code Deploy to deploy index.html to Amazon Linux ec2 instance

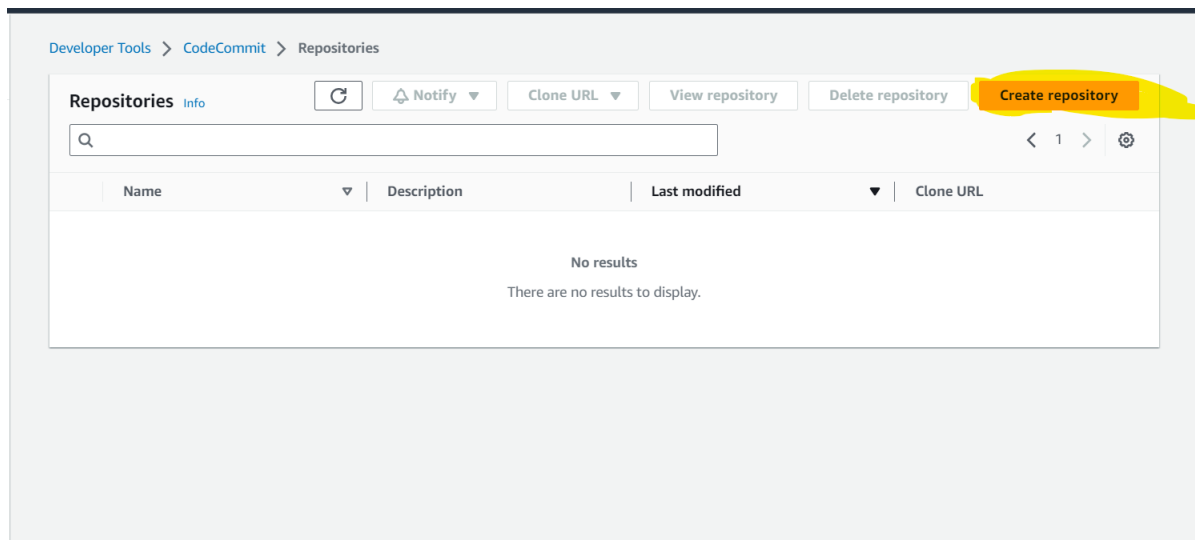
e.Check the o/p , by accessing the public ip of the ec2 instance . Contents of the index.html must be displayed

Download SampleApp_Linux file and unzip it.

https://docs.aws.amazon.com/codepipeline/latest/userguide/samples/SampleApp_Linux.zip



1. Navigate to CodeCommit and create a repository.



2. Enter the name and description and later, create the repository.

[Developer Tools](#) > [CodeCommit](#) > [Repositories](#) > [Create repository](#)

Create repository

Create a secure repository to store and share your code. Begin by typing a repository name and a description for your repository. Repository names are included in the URLs for that repository.

Repository settings

Repository name

MyDemoRepo

100 characters maximum. Other limits apply.

Description - *optional*

A pipeline using AWS [CodePipeline](#), AWS Code Commit, and Code Deploy to deploy index.html to Amazon Linux ec2 instance

1,000 characters maximum

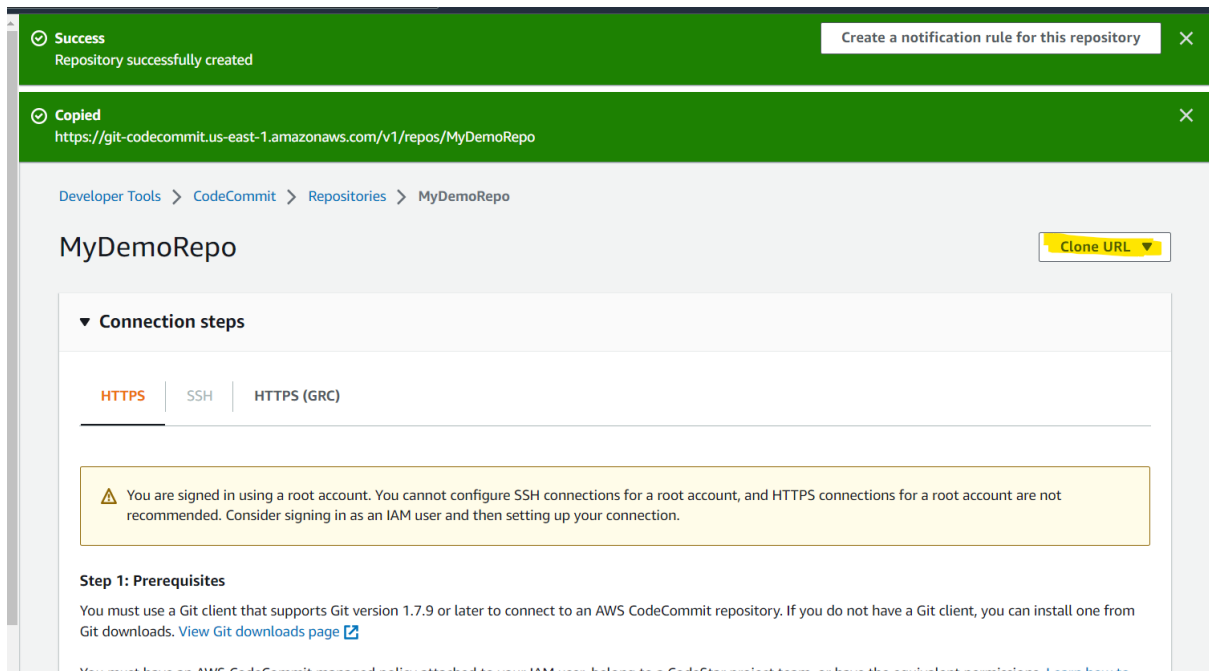
Tags

Add tag

☐ Enable Amazon CodeGuru Reviewer for Java and Python - *optional*

Get recommendations to improve the quality of the Java and Python code for all pull requests in this repository.

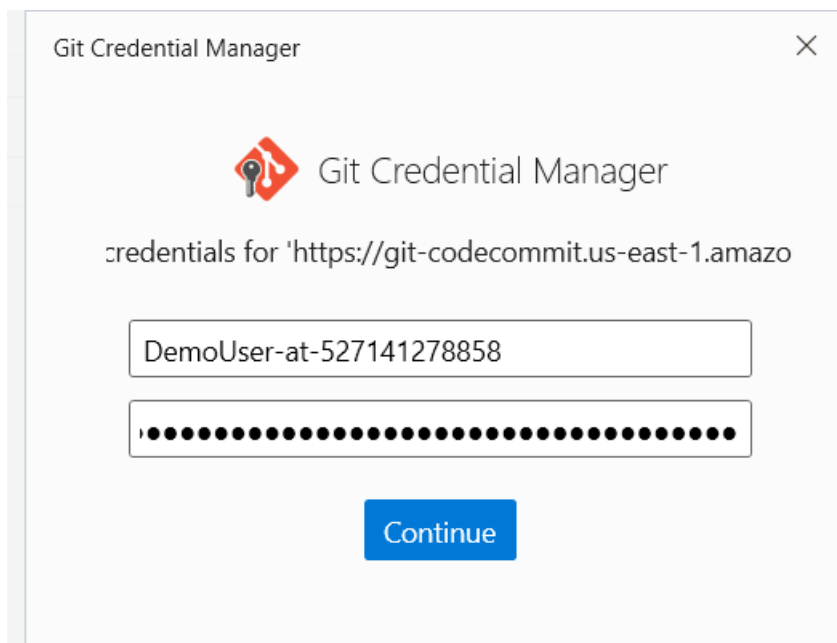
A service-linked role will be created in IAM on your behalf if it does not exist.



3. Now, you need to push the SampleApp_Linux file to the remote repository you have created. Copy the http link which you will use to clone the repository locally.
 - **Git clone “url_link”**

```
C:\Users\Manjunath S>git clone https://git-codecommit.us-east-1.amazonaws.com/v1/repos/MyDemoRepo
Cloning into 'MyDemoRepo'...
warning: You appear to have cloned an empty repository.
```

- It will ask for the credentials, you need to download the credentials from IAM->Git credentials.



4. Initialize and check the git status. You can the files but it is not yet committed.

```
C:\Users\Manjunath S>cd C:\temp\MyDemoRepo\SampleApp_Linux

C:\temp\MyDemoRepo\SampleApp_Linux>git init
Initialized empty Git repository in C:/temp/MyDemoRepo/SampleApp_Linux/.git/

C:\temp\MyDemoRepo\SampleApp_Linux>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    LICENSE.txt
    appspec.yml
    index.html
    scripts/

nothing added to commit but untracked files present (use "git add" to track)

C:\temp\MyDemoRepo\SampleApp_Linux>|
```

5. Add and commit the changes later, check the status.

```
C:\temp\MyDemoRepo\SampleApp_Linux>git add -A
warning: in the working copy of 'LICENSE.txt', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'appspec.yml', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'index.html', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'scripts/install_dependencies', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'scripts/start_server', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'scripts/stop_server', LF will be replaced by CRLF the next time Git touches it

C:\temp\MyDemoRepo\SampleApp_Linux>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   LICENSE.txt
    new file:   appspec.yml
    new file:   index.html
    new file:   scripts/install_dependencies
    new file:   scripts/start_server
    new file:   scripts/stop_server
```

```
C:\temp\MyDemoRepo\SampleApp_Linux>git commit -m "uploading the contents"
[master (root-commit) 9699651] uploading the contents
 6 files changed, 266 insertions(+)
 create mode 100644 LICENSE.txt
 create mode 100644 appspec.yml
 create mode 100644 index.html
 create mode 100644 scripts/install_dependencies
 create mode 100644 scripts/start_server
 create mode 100644 scripts/stop_server

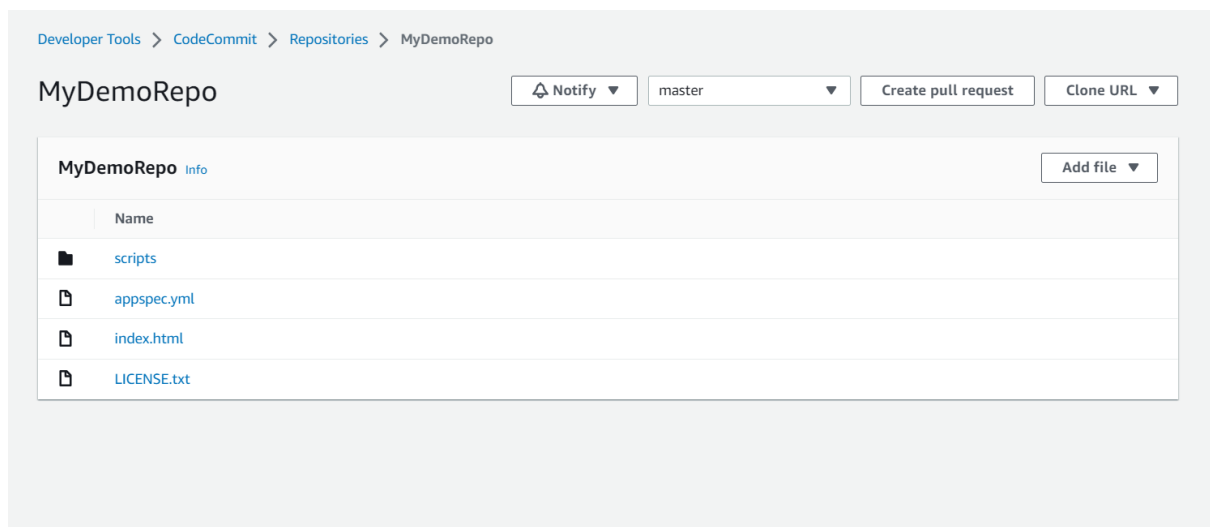
C:\temp\MyDemoRepo\SampleApp_Linux>git push
```

6. And next, push the changes into the remote repository.

```
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (9/9), 5.01 KiB | 1.00 MiB/s, done.
Total 9 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To https://git-codecommit.us-east-1.amazonaws.com/v1/repos/MyDemoRepo
 * [new branch]      master -> master
branch 'master' set up to track 'MyDemoRepo/master'.

C:\temp\MyDemoRepo\SampleApp_Linux>
```

7. Verify whether you can see the files in the remote repository.



8. The next step is to launch EC2 Instance. Select the AMI image and required credentials.
 - And use the below commands to run the agent in your instance.

```
sudo apt-get update
```

```
sudo apt-get install -y awscli
```

```
sudo apt-get install -y ruby
```

```
wget https://aws-coddeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install
```

```
chmod +x ./install
```

```
sudo ./install auto
```

```
sudo service coddeploy-agent start
```

```
sudo service coddeploy-agent status
```

Allow tags in metadata [Info](#)

Select

User data - optional [Info](#)
Upload a file with your user data or enter it in the field.

Choose file

```
#!/bin/bash
yum -y update
yum install -y ruby
yum install -y aws-cli
cd /home/ec2-user
aws s3 cp s3://aws-coddeploy-us-east-1/latest/install --region us-east-1
chmod +x ./install
./install auto
```

☐ User data has already been base64 encoded

Number of instances [Info](#)

1

Software Image (AMI)

Amazon Linux 2 Kernel 5.10 AMI...[read more](#)
ami-09538990a0c4fe9be

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 20 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per

Cancel

Launch instance

[Review commands](#)

9. And in your instance, modify the IAM role and attach the role.

EC2 > Instances > i-04a4f1cccc854b180 > Modify IAM role

Modify IAM role [Info](#)

Attach an IAM role to your instance.

Instance ID

i-04a4f1cccc854b180 (SampleDemoRepo)

IAM role

Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.

Deploy-weAPP

[Create new IAM role](#)

Cancel

Update IAM role

10. The next step is to create an application, Navigate CodeDeploy > Applications > Create applications.

Here, you need to enter the name of the application and select EC2/On-premises.

Create application

Application configuration

Application name

Enter an application name

MyDemoRepo

100 character limit

Compute platform

Choose a compute platform

EC2/On-premises

Tags

Add tag

Cancel

Create application

11. After creating the application, create code-deploy group, you need to enter the service role and select Amazon EC2 instance. Select the key, the key will be your instance name. Next, unselect the load balance.

Services

Search

[Alt+S]

Select any combination of Amazon EC2 Auto Scaling groups, Amazon EC2 instances, and on-premises instances to add to this deployment

☐ Amazon EC2 Auto Scaling groups

☒ Amazon EC2 instances
2 unique matched instances. [Click here for details](#)

You can add up to three groups of tags for EC2 instances to this deployment group.
One tag group: Any instance identified by the tag group will be deployed to.
Multiple tag groups: Only instances identified by all the tag groups will be deployed to.

Tag group 1

Key

Value - optional

Q Name X

Q MyDemoRepo X

Remove tag

Add tag

+ Add tag group

☐ On-premises instances

Matching instances
2 unique matched instances. [Click here for details](#)

Below are the sample permission for the role.

Permissions
Trust relationships
Tags
Access Advisor
Revoke sessions

Permissions policies (6)
Info

You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter.

☐
☐
☐
☐
☐
☐

Policy name	Type	Description
AdministratorAccess	AWS managed - job function	Provides full access to AWS services and resources.
AmazonEC2FullAccess	AWS managed	Provides full access to Amazon EC2 via the AWS Management Console.
AWSCodeDeployRole	AWS managed	Provides CodeDeploy service access to expand tags and interact with Auto...
AmazonEC2RoleforAWSCodeDeploy	AWS managed	Provides EC2 access to S3 bucket to download revision. This role is neede...
AWSCodeDeployFullAccess	AWS managed	Provides full access to CodeDeploy resources.
AmazonEC2RoleforSSM	AWS managed	This policy will soon be deprecated. Please use AmazonSSMManagedInst...

12. Now, navigate to the pipeline in codedeploy and create a new pipeline. Enter the name of the pipeline and select next.

Choose pipeline settings
Info

Pipeline settings

Pipeline name

Enter the pipeline name. You cannot edit the pipeline name after it is created.

SamplePipeline

No more than 100 characters

Service role

☒
New service role
Create a service role in your account

☐
Existing service role
Choose an existing service role from your account

Role name

AWSCodePipelineServiceRole-us-east-1-SamplePipeline

Type your service role name

☒
Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

Advanced settings

Cancel
Next

13. The source provider will be AWS Codecommit, select the repository and the branch that you have created.

Source provider
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

AWS CodeCommit

Repository name
Choose a repository that you have already created where you have pushed your source code.

MyDemoRepo

Branch name
Choose a branch of the repository

master

Change detection options
Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

☒ Amazon CloudWatch Events (recommended)
Use Amazon CloudWatch Events to automatically start my pipeline when a change occurs

☐ AWS CodePipeline
Use AWS CodePipeline to check periodically for changes

Output artifact format
Choose the output artifact format.

☒ CodePipeline default
AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.

☐ Full clone
AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions.

Cancel

Previous

Next

14. You can skip the build-stage. In the Deploy stage, the deploy provider will be AWS codedeploy. Select the region. Select the application and deployment group that you have created.

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

You cannot skip this stage
Pipelines must have at least two stages. Your second stage must be either a build or deployment stage. Choose a provider for either the build stage or deployment stage.

Deploy

Deploy provider
Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.

AWS CodeDeploy

Region
US East (N. Virginia)

Application name
Choose an application that you have already created in the AWS CodeDeploy console. Or create an application in the AWS CodeDeploy console and then return to this task.

MyDemoRepo

Deployment group
Choose a deployment group that you have already created in the AWS CodeDeploy console. Or create a deployment group in the AWS CodeDeploy console and then return to this task.

SampleGroup

Cancel

Previous

Next

Click on next, review the pipeline, and create the pipeline. Monitor the results of the pipeline.

Buttons: Notify, Edit, Stop execution, Clone pipeline, Release change

Source Succeeded
Pipeline execution ID: f4a9c62e-ef43-43e6-ad25-bbbcd188624b

Source
AWS CodeCommit
Succeeded - 13 minutes ago
9a8cc3ef
9a8cc3ef Source: file added ready to deploy

Disable transition

Deploy Succeeded
Pipeline execution ID: f4a9c62e-ef43-43e6-ad25-bbbcd188624b

Deploy
AWS CodeDeploy
Succeeded - 6 minutes ago
Details

Developer Tools CodeDeploy

- Source • CodeCommit
- Artifacts • CodeArtifact
- Build • CodeBuild
- ▼ Deploy • CodeDeploy
 - Getting started
 - Deployments
 - Deployment
 - Applications
 - Deployment configurations
 - On-premises instances
- Pipeline • CodePipeline
- Settings

Go to resource Feedback

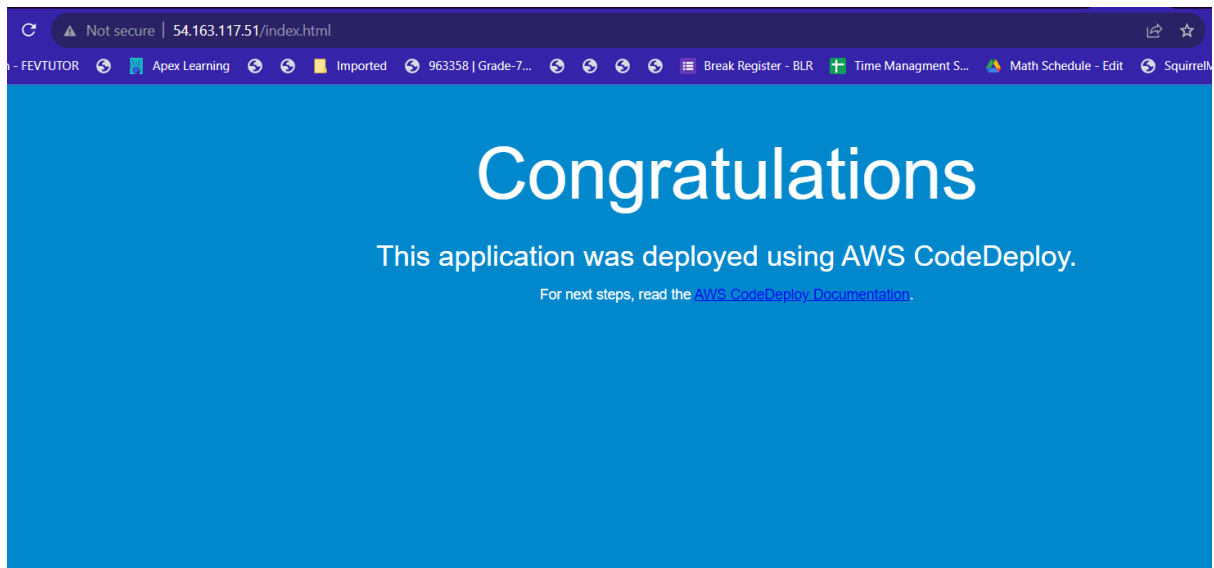
Deployment description
-

Revision details

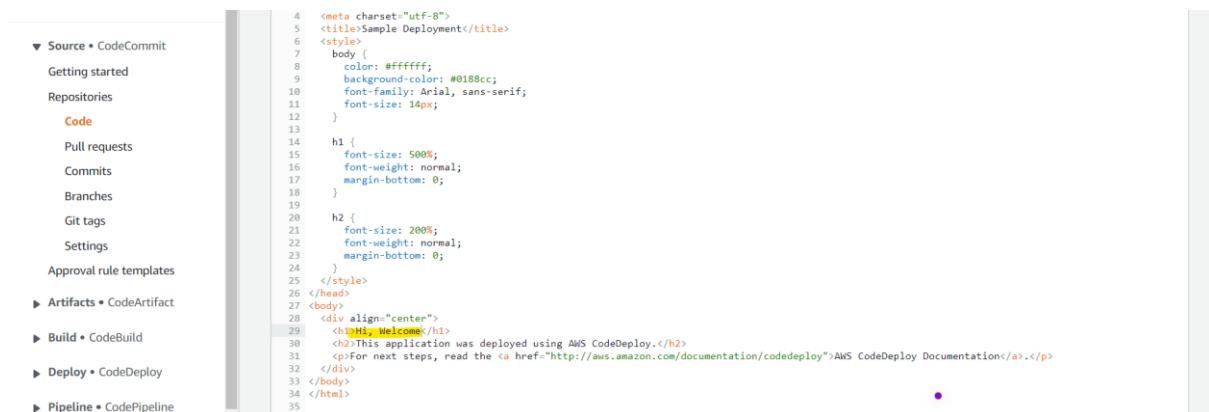
Revision location s3://codepipeline-us-east-1-256922894148/SampleProject2/SourceArti/1wn84EJ?eTag=454f035aaaf1821760bd6821b08de3b5-1	Revision created 25 minutes ago	Revision description Application revision registered by Deployment ID: d-PFR7PEC31
-----------------------------------------------------------------------------------------------------------------------------------------	------------------------------------	---------------------------------------------------------------------------------------

Event	Duration	Status	Error code	Start time	End time
ApplicationStop	less than one second	Succeeded	-	Aug 13, 2023 4:39 PM (UTC+5:30)	Aug 13, 2023 4:39 PM (UTC+5:30)
DownloadBundle	less than one second	Succeeded	-	Aug 13, 2023 4:39 PM (UTC+5:30)	Aug 13, 2023 4:39 PM (UTC+5:30)
BeforeInstall	1 second	Succeeded	-	Aug 13, 2023 4:39 PM (UTC+5:30)	Aug 13, 2023 4:39 PM (UTC+5:30)
Install	less than one second	Succeeded	-	Aug 13, 2023 4:39 PM (UTC+5:30)	Aug 13, 2023 4:39 PM (UTC+5:30)
AfterInstall	less than one second	Succeeded	-	Aug 13, 2023 4:39 PM (UTC+5:30)	Aug 13, 2023 4:39 PM (UTC+5:30)
ApplicationStart	less than one second	Succeeded	-	Aug 13, 2023 4:39 PM (UTC+5:30)	Aug 13, 2023 4:39 PM (UTC+5:30)
ValidateService	less than one second	Succeeded	-	Aug 13, 2023 4:39 PM (UTC+5:30)	Aug 13, 2023 4:39 PM (UTC+5:30)

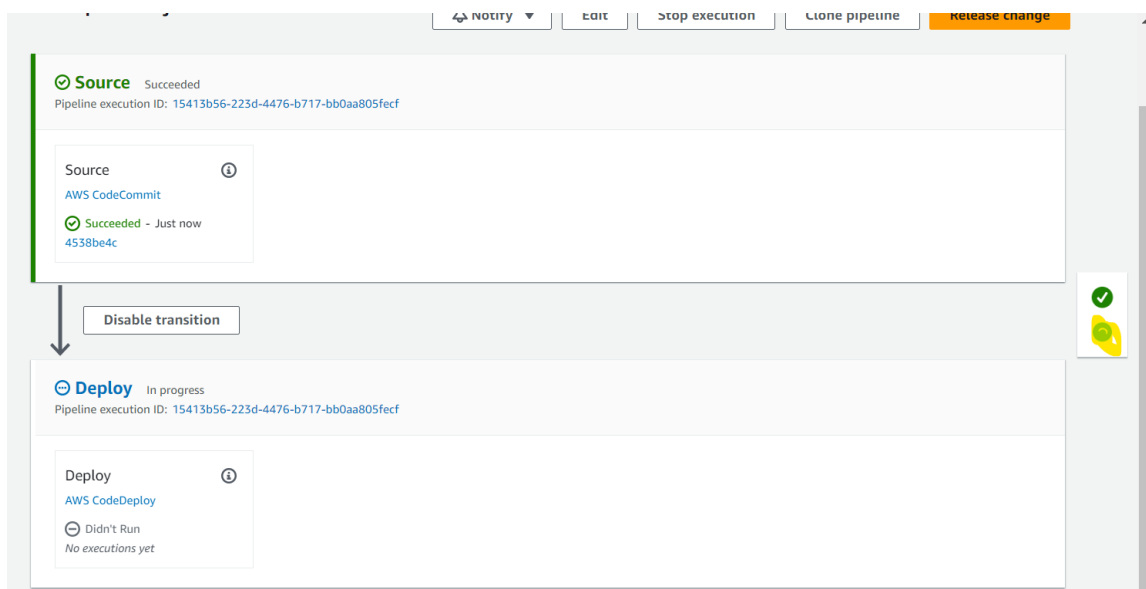
To see the output, copy the public IP address of your instance and browse it.

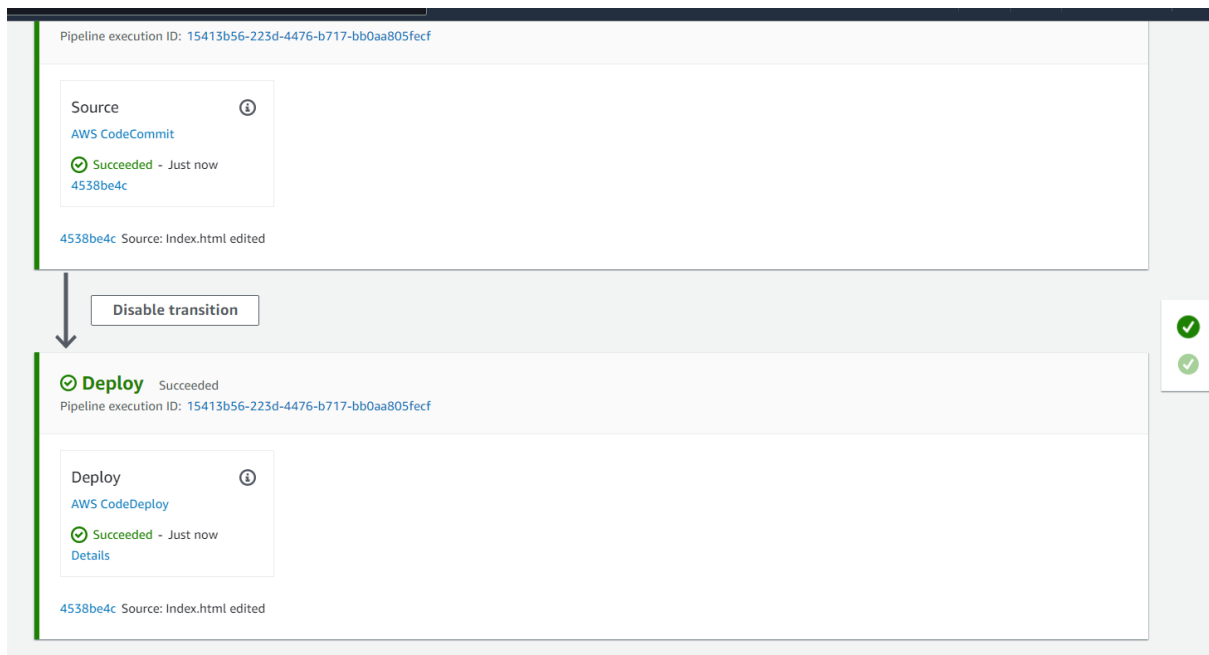


15. Modify code in your CodeCommit repository.

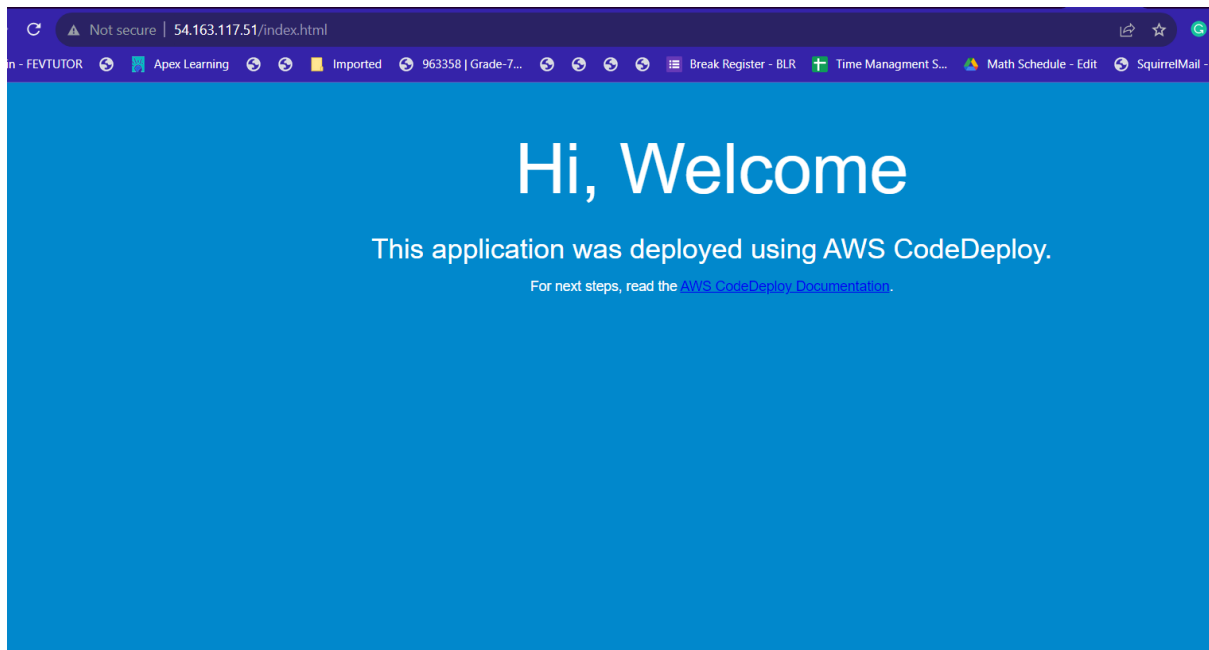


To verify your pipeline ran successfully.





And check the result on your browser once again.



Later, clean-up the resources if it is not needed anymore.