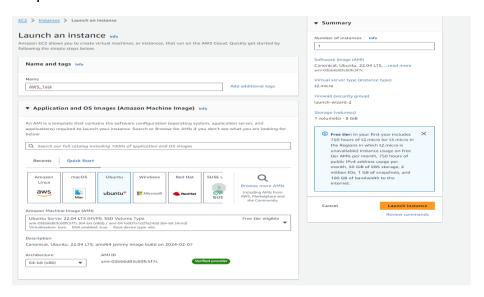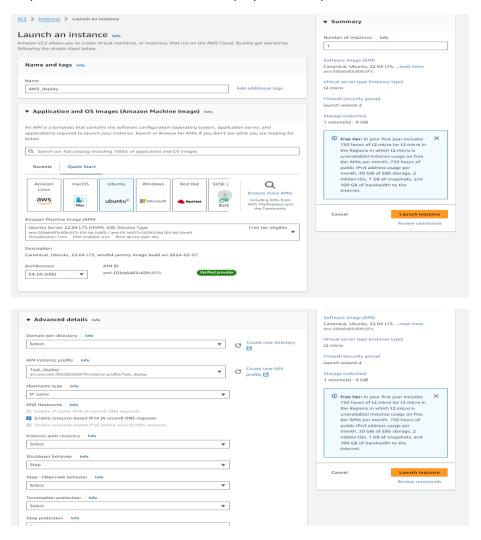# Create Repo in Code Commit & practice Code Build & Pipelines.
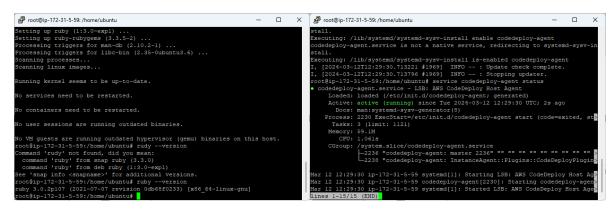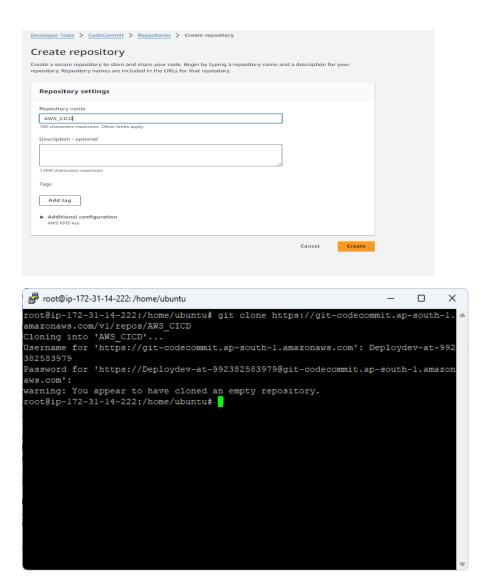
Step 1: Create Ec2 instance for the dev.

Step 2: Create Ec2 instance for the deploy with IAM profile.

Step 3: Install Rudy and code deploy agent in the deploy Instance.
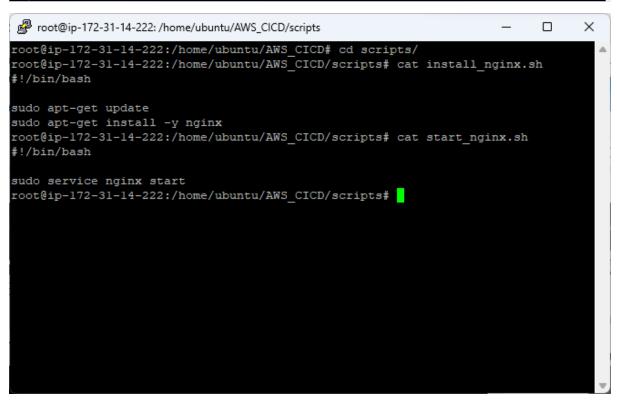

Step 4: Create repository in AWS code commit and clone the repository in dev instance using git.
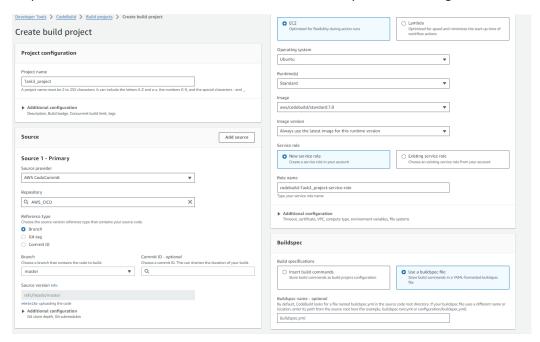
Step 5 : In Dev instance, create the appspec.yml, buildspec.yml,index.html, install_nginx.sh & start_nginx.sh


```
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD# cat appspec.yml
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html
hooks:
  AfterInstall:
    - location: scripts/install_nginx.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: scripts/start_nginx.sh
      timeout: 300
      runas: root
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD# cat buildspec.yml
version: 0.2

phases:
  install:
    commands:
      - echo Installing NGINX
      - sudo apt-get update
      - sudo apt-get install nginx -y
  build:
    commands:
      - echo Build started on `date`
      - cp index.html /var/www/html/
  post_build:
    commands:
      - echo Configuring NGINX

artifacts:
  files:
    - '**/*'
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD# cat index.html
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        font-family: Arial, sans-serif;
        background-color: #f2f2f2;
        color: #333;
        text-align: center;
      }

      h1 {
        font-size: 36px;
        margin-top: 50px;
        color: #6130e8;
      }

      p {
        font-size: 18px;
        margin: 20px 0;
      }
    </style>
  </head>
  <body>
    <h1>Hello Devops</h1>
    <p>This is Jayaprabakara from DO1WE Tamil Batch</p>
  </body>
```



```
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD# cd scripts/
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD/scripts# cat install_nginx.sh
#!/bin/bash

sudo apt-get update
sudo apt-get install -y nginx
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD/scripts# cat start_nginx.sh
#!/bin/bash

sudo service nginx start
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD/scripts#
```

Step 6: git add * →git commit -m → git push origin master.


```
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD/scripts# cd ..
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD# git add *
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD# git commit -m "uploading the code"
[master (root-commit) e0e4e28] uploading the code
 Committer: root <root@ip-172-31-14-222.ap-south-1.compute.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 6 files changed, 68 insertions(+)
 create mode 100644 appspec.yml
 create mode 100644 buildspec.yml
 create mode 100644 index.html
 create mode 100644 scripts/.start_nginx.sh.swp
 create mode 100644 scripts/install_nginx.sh
 create mode 100644 scripts/start_nginx.sh
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD# git push orgini master
fatal: 'orgini' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD# git push origin master
Username for 'https://git-codecommit.ap-south-1.amazonaws.com': Deploydev-at-992
382583979
Password for 'https://Deploydev-at-992382583979@git-codecommit.ap-south-1.amazon
aws.com':
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 1.33 KiB | 680.00 KiB/s, done.
Total 9 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To https://git-codecommit.ap-south-1.amazonaws.com/v1/repos/AWS_CICD
 * [new branch]      master -> master
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD#
```

Step 7: create code Build → AWS code commit as source provider→change the OS to ubuntu.

Step 8: start the build

Step 9: create application in AWS codedeploy

Step 10: create deployment group in codedeploy → select deploy instance in Environment config
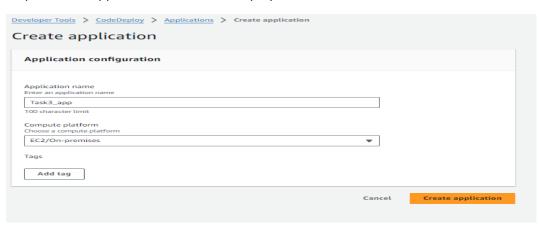
Create deployment group

Application

Application
Task3_app
Compute type
EC2/On-premises

Deployment group name

Enter a deployment group name
Task3_deploy
100 character limit

Service role

Enter a service role
Enter a service role with CodeDeploy permissions that grants AWS CodeDeploy access to your target instances.

arn:aws:iam::992382583979:role/CodeDeployRole

Deployment type

Choose how to deploy your application

○ In-place
Updates the instances in the deployment group with the latest application revisions. During a deployment, each instance will be briefly taken offline for its update.

○ Blue/green
Replaces the instances in the deployment group with new instances and deploys the latest application revision to them. After instances in the replacement environment are registered with a load balancer, instances from the original environment are deregistered and can be terminated.

Environment configuration

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
1 unique matched instance. 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
Q Name

Value - optional
Q AWS_deploy

Remove tag

Add tag

+ Add tag group

☐ On-premises instances

Matching instances
1 unique matched instance. Click here for details

Agent configuration with AWS Systems Manager Info

ⓘ **Complete the required prerequisites before AWS Systems Manager can install the CodeDeploy Agent.** Make sure the AWS Systems Manager Agent is installed on all instances and attach the required IAM policies to them. Learn more

Install AWS CodeDeploy Agent
○ Never
○ Only once
● Now and schedule updates

Basic scheduler | Cron expression

14 | Days ▼

Agent configuration with AWS Systems Manager Info

ⓘ **Complete the required prerequisites before AWS Systems Manager can install the CodeDeploy Agent. Make sure the AWS Systems Manager Agent is installed on all instances and attach the required IAM policies to them.** Learn more

Install AWS CodeDeploy Agent
○ Never
○ Only once
● Now and schedule updates

Basic scheduler | Cron expression

14 | Days ▼

Deployment settings

Deployment configuration
Choose from a list of default and custom deployment configurations. A deployment configuration is a set of rules that determines how fast an application is deployed and the success or failure conditions for a deployment.

CodeDeployDefault.AllAtOnce ▼   or   **Create deployment configuration**

Load balancer

Select a load balancer to manage incoming traffic during the deployment process. The load balancer blocks traffic from each instance while it's being deployed to and allows traffic to it again after the deployment succeeds.

☐ Enable load balancing

▶ Advanced – optional

Cancel   **Create deployment group**

Step 11: Create Pipeline → select Code commit as source stage→select newly created build →

Select newly created application and deployment group.

Step 12 : after creating pipeline source →bulid→deploy will run successfully.

Task3_Pipeline

Pipeline type: **V2**   Execution mode: **SUPERSEDED**

⊘ **Source** Succeeded
Pipeline execution ID: d2bd32d2-7a3e-41b3-80e0-c5b4f0931e7d

Source
AWS CodeCommit

⊘ Succeeded - 4 minutes ago
e0e4e28a

View details

e0e4e28a Source: uploading the code

Disable transition

⊘ **Build** Succeeded
Pipeline execution ID: d2bd32d2-7a3e-41b3-80e0-c5b4f0931e7d

Build
AWS CodeBuild

⊘ Succeeded - 3 minutes ago
View details

e0e4e28a Source: uploading the code

Disable transition

⊘ **Deploy** Succeeded
Pipeline execution ID: d2bd32d2-7a3e-41b3-80e0-c5b4f0931e7d

Deploy
AWS CodeDeploy

⊘ Succeeded - 3 minutes ago
View details

e0e4e28a Source: uploading the code

OUTPUT

← C ⌂ ▲ Not secure | 43.204.103.83   A⁴ ☆

Hello Devops

This is Jayaprabakara from DO1WE Tamil Batch