

Create Repo in Code Commit & practice Code Build & Pipelines.

Step 1: Create Ec2 instance for the dev.

EC2 > Instances > Launch an instance

Launch an instance

Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Info

Name

AWS_Task

Add additional tags

▼ Application and OS Images (Amazon Machine Image)

Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

macOS

Ubuntu

Windows

Red Hat

SUSE Linux

Browse more AMIs

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), 55D Volume Type

Free tier eligible

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2024-02-07

Architecture

AMI ID

Verified provider

▼ Summary

Number of instances

Info

1

Software Image (AMI)

Canonical, Ubuntu, 22.04 LTS, ...read more

Virtual server type (instance type)

t2.micro

Firewall (security group)

launch-wizard-2

Storage (volumes)

1 volume(s) - 8 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 month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

Review commands

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

EC2 > Instances > Launch an instance

Launch an instance

Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Info

Name

AWS_deploy

Add additional tags

▼ Application and OS Images (Amazon Machine Image)

Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

macOS

Ubuntu

Windows

Red Hat

SUSE Linux

Browse more AMIs

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), 55D Volume Type

Free tier eligible

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2024-02-07

Architecture

AMI ID

Verified provider

▼ Summary

Number of instances

Info

1

Software Image (AMI)

Canonical, Ubuntu, 22.04 LTS, ...read more

Virtual server type (instance type)

t2.micro

Firewall (security group)

launch-wizard-2

Storage (volumes)

1 volume(s) - 8 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 month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

Review commands

▼ Advanced details

Info

Domain join directory

Info

Select

Create new directory

IAM instance profile

Info

Task_deploy

Create new IAM profile

Hostname type

Info

IP name

DNS Hostname

Info

Enable IP name IPv4 (A record) DNS requests

Enable resource-based IPv4 (A record) DNS requests

Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery

Info

Select

Shutdown behavior

Info

Stop

Stop - Hibernate behavior

Info

Select

Termination protection

Info

Select

Stop protection

Info

Select

▼ Summary

Software Image (AMI)

Canonical, Ubuntu, 22.04 LTS, ...read more

Virtual server type (instance type)

t2.micro

Firewall (security group)

launch-wizard-2

Storage (volumes)

1 volume(s) - 8 GiB

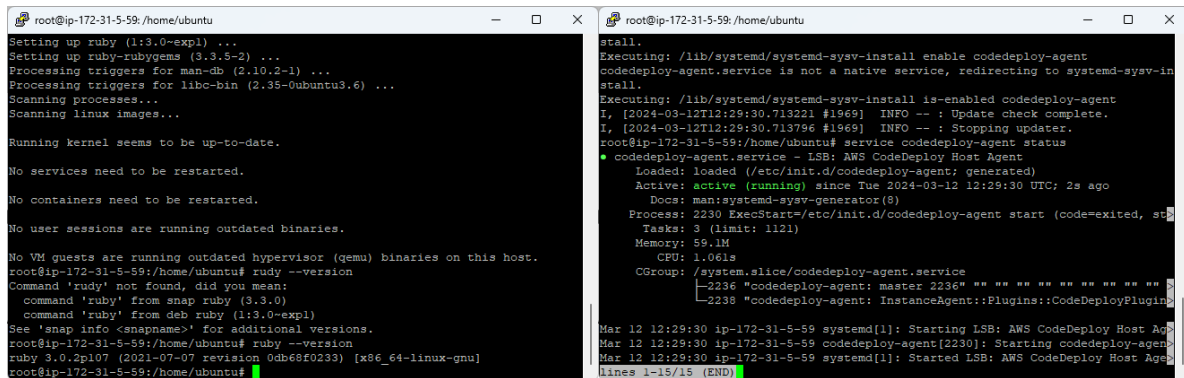
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

Review commands

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



The image shows two terminal windows side-by-side. The left window is a root shell on an Ubuntu instance (ip-172-31-5-59) showing the output of 'snap install rudy --version' and 'snap info rudy'. The right window shows the installation and status of the 'awscli' snap and the 'codedeploy-agent' service. It includes commands like 'systemctl enable codedeploy-agent', 'systemctl start codedeploy-agent', and 'service codedeploy-agent status', along with detailed logs of the service starting.

```
root@ip-172-31-5-59:/home/ubuntu# snap install rudy --version
Setting up ruby (1:3.0-expl) ...
Setting up ruby-rubygems (3.3.5-2) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.6) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

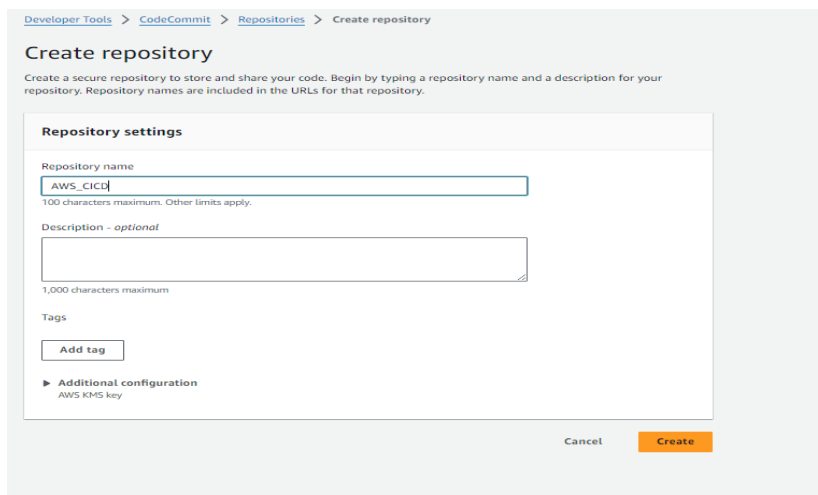
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-5-59:/home/ubuntu# rudy --version
Command 'rudy' not found, did you mean:
  Command 'ruby' from snap ruby (3.3.0)
  Command 'ruby' from deb ruby (1:3.0-expl)
See 'snap info <snapname>' for additional versions.
root@ip-172-31-5-59:/home/ubuntu# rudy --version
ruby 3.0.2p107 (2021-07-07 revision 0db68f0233) [x86_64-linux-gnu]
root@ip-172-31-5-59:/home/ubuntu#

root@ip-172-31-5-59:/home/ubuntu# snap install awscli
Installing from the stable channel...
Done.

root@ip-172-31-5-59:/home/ubuntu# systemctl enable codedeploy-agent
codedeploy-agent.service is not a native service, redirecting to systemd-sysv-in
stall.
Executing: /lib/systemd/systemd-sysv-install enable codedeploy-agent
I, [2024-03-12T12:29:30.713221 #1969] INFO -- : Update check complete.
I, [2024-03-12T12:29:30.713796 #1969] INFO -- : Stopping updater.
root@ip-172-31-5-59:/home/ubuntu# service codedeploy-agent status
codedeploy-agent.service - LSB: AWS CodeDeploy Host Agent
   Loaded: loaded (/etc/init.d/codedeploy-agent; generated)
   Active: active (running) since Tue 2024-03-12 12:29:30 UTC; 2s ago
     Docs: man:systemd-sysv-generator(8)
   Process: 2230 ExecStart=/etc/init.d/codedeploy-agent start (code=exited, st
   Tasks: 3 (limit: 1121)
   Memory: 59.1M
   CPU: 1.061s
   CGroup: /system.slice/codedeploy-agent.service
           └─2236 "codedeploy-agent: master 2236"
             └─2238 "codedeploy-agent: InstanceAgent::Plugins::CodeDeployPlugi
Mar 12 12:29:30 ip-172-31-5-59 systemd[1]: Starting LSB: AWS CodeDeploy Host Ag
Mar 12 12:29:30 ip-172-31-5-59 codedeploy-agent[2230]: Starting codedeploy-age
Mar 12 12:29:30 ip-172-31-5-59 systemd[1]: Started LSB: AWS CodeDeploy Host Age
lines 1-15/15 (END)
```

Step 4: Create repository in AWS code commit and clone the repository in dev instance using git.



The screenshot shows the 'Create repository' page in the AWS CodeCommit console. The 'Repository name' field is filled with 'AWS_CICD'. The 'Description - optional' field is empty. There are 'Add tag' and 'Additional configuration' (AWS KMS key) options. At the bottom, there are 'Cancel' and 'Create' buttons.

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

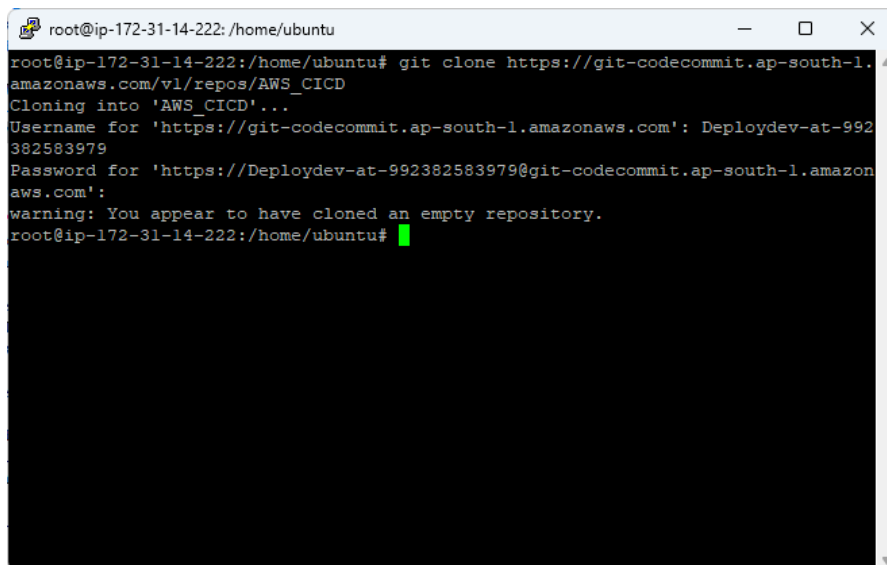
100 characters maximum. Other limits apply.

Description - optional

1,000 characters maximum

Tags

► Additional configuration
AWS KMS key



The terminal window shows the command 'git clone https://git-codecommit.ap-south-1.amazonaws.com/v1/repos/AWS_CICD' being executed. It prompts for a username and password, and then shows the warning 'warning: You appear to have cloned an empty repository.' followed by a prompt for the next command.

```
root@ip-172-31-14-222:/home/ubuntu# git clone https://git-codecommit.ap-south-1.amazonaws.com/v1/repos/AWS_CICD
Cloning into 'AWS_CICD'...
Username for 'https://git-codecommit.ap-south-1.amazonaws.com': Deploydev-at-992382583979
Password for 'https://Deploydev-at-992382583979@git-codecommit.ap-south-1.amazonaws.com':
warning: You appear to have cloned an empty repository.
root@ip-172-31-14-222:/home/ubuntu#
```

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: #666666;
  }
  p {
    font-size: 18px;
    margin: 20px 0;
  }
</style>
</head>
<body>
<h1>Hello Devops</h1>
<p>This is Jayaprakash from DOINE Tamil Batch</p>
</body>
```

```
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD/scripts
root@ip-172-31-14-222:/home/ubuntu/AWS_CICD/scripts# 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
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.

Developer Tools > CodeBuild > Build projects > Create build project

Create build project

Project configuration

Project name

A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.

► **Additional configuration**
Description, build badge, Concurrent build limit, tags

Source

Add source

Source 1 - Primary

Source provider

AWS CodeCommit

Repository

Q AWS_CICD

Reference type
Choose the source version reference type that contains your source code.

☒ Branch
☐ Git tag
☐ Commit ID

Branch
Choose a branch that contains the code to build.

master

Commit ID - optional
Choose a commit ID. This can shorten the duration of your build.

Source version info

refs/heads/master

eb4e28a uploading the code

► **Additional configuration**
Git clone depth, Git submodules

☒ EC2
Optimized for flexibility during action runs

☐ Lambda
Optimized for speed and minimizes the start up time of workflow actions

Operating system

Ubuntu

Runtime(s)

Standard

Image

aws/codebuild/standard:7.0

Image version

Always use the latest image for this runtime version

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

codebuild-Task3_project-service-role

Type your service role name

► **Additional configuration**
Timeout, certificate, VPC, compute type, environment variables, File systems

Buildspec

☐ Insert build commands
Store build commands as build project configuration

☒ Use a buildspec file
Store build commands in a YAML-formatted buildspec file

Buildspec name - optional
By default, CodeBuild looks for a file named buildspec.yml in the source code root directory. If your buildspec file uses a different name or location, enter its path from the source root here (for example, buildspec-two.yml or configuration/buildspec.yml).

buildspec.yml

Step 8: start the build

Developer Tools > CodeBuild > Build projects > Task3_project > Task3_project:1038afcb-8a33-43ce-b683-62b77fe273c7

Task3_project:1038afcb-8a33-43ce-b683-62b77fe273c7

Stop build Retry build

Build status

Status Succeeded	Initiator root	Build ARN arn:aws:codebuild:ap-south-1:992382583979:build/Task3_project:1038afcb-8a33-43ce-b683-62b77fe273c7	Resolved source version e0e4c28aa4746c2933db931c8f5e47d1900e8148
Start time Mar 12, 2024 6:17 PM (UTC+5:30)	End time Mar 12, 2024 6:18 PM (UTC+5:30)	Build number 1	

Build logs | Phase details | Reports | Environment variables | Build details | Resource utilization

Logs

CloudWatch logs DISABLED	CloudWatch group name /aws/codebuild/Task3_project	CloudWatch stream name -
S3 logs DISABLED	S3 location -	Encryption disabled False

Step 9: create application in AWS codedeploy

Developer Tools > CodeDeploy > Applications > Create application

Create application

Application configuration

Application name
Enter an application name

100 character limit

Compute platform
Choose a compute platform

EC2/On-premises

Tags

Add tag

Cancel Create application

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

Developer Tools > CodeDeploy > Applications > Task3_app > Create deployment group

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
Name

Value - optional
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

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 schedulerCron expression

14Days

Agent configuration with AWS Systems Manager

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 schedulerCron expression

14Days

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.

Source action provider

Source action provider
AWS CodeCommit
RepositoryName
AWS_CICD
Default branch
master
PollForSourceChanges
true
OutputArtifactFormat
CODE_ZIP

Step 3: Add build stage

Build action provider

Build action provider
AWS CodeBuild
ProjectName
Task3_project

Step 4: Add deploy stage

Deploy action provider

Deploy action provider
AWS CodeDeploy
ApplicationName
Task3_app
DeploymentGroupName
Task3_deploy

Cancel Previous Create pipeline

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

Developer Tools > CodePipeline > Pipelines > Task3_Pipeline

Task3_Pipeline

Pipeline type: V2 Execution mode: SUPERSEDED

Source Succeeded

Pipeline execution ID: [d7bd32d2-7a3e-41b3-80e0-c5b4f0931e7d](#)

Source

[AWS CodeCommit](#)

Succeeded - 4 minutes ago

[v0e4e28a](#)

[View details](#)

[v0e4e28a](#) Source: uploading the code

↓ [Disable transition](#)

Build Succeeded

Pipeline execution ID: [d7bd32d2-7a3e-41b3-80e0-c5b4f0931e7d](#)

Build

[AWS CodeBuild](#)

Succeeded - 3 minutes ago

[View details](#)

[v0e4e28a](#) Source: uploading the code

↓ [Disable transition](#)

Deploy Succeeded

Pipeline execution ID: [d7bd32d2-7a3e-41b3-80e0-c5b4f0931e7d](#)

Deploy

[AWS CodeDeploy](#)

Succeeded - 3 minutes ago

[View details](#)

[v0e4e28a](#) Source: uploading the code

OUTPUT

← ↻ 🏠 ⚠ Not secure | 43.204.103.83 ⌕ ☆

Hello Devops

This is Jayaprabakara from DO1WE Tamil Batch