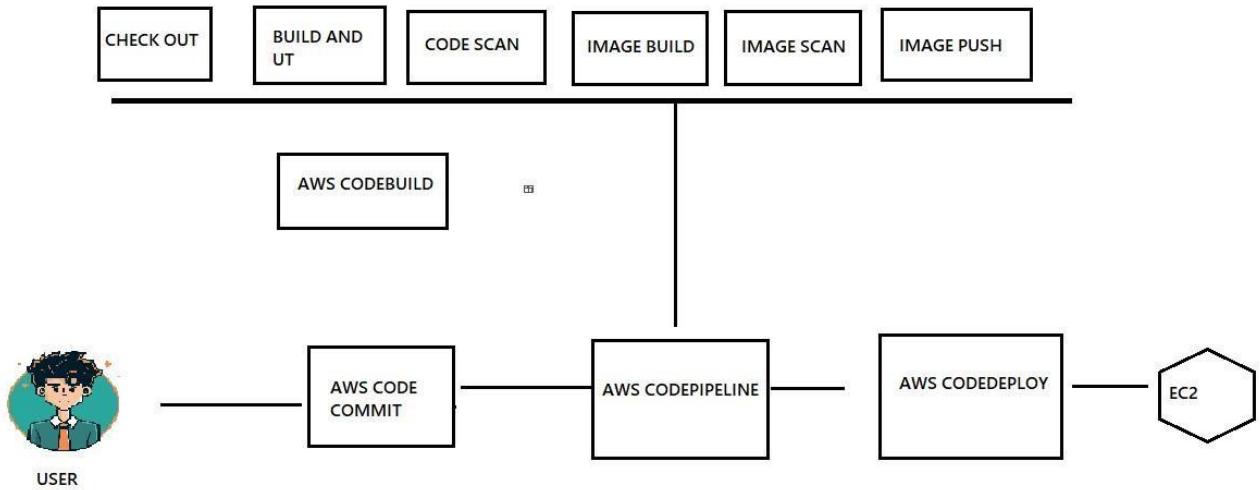


NAME – VICKY SHARMA

AIM- CI CD PIPELINE AUTOMATION USING AWS CODE COMMIT

STEPS -1 SO BASICALLY IN THESE PROJECTWE ARE GOING TO DEPLOY OUR PYTHON SIMPLE FLASK APPLICATION USING CICD PIPELINE ARCHITECTURE:-



STEP 1-

Divide the complete project into two part which is CI and

CD now we will go for part 1 which is continuous

integration

FIRST CREATE A REPOSITORIES IN GITHUB AND PUSH YOUR CODE WITH REQUIREMENT TXT, DOCKER

FILE IN WHICH WE WILL HAVE SCRIPT TO CREATE IMAGE OF CONTAINER AND

SIMPLE FLASK APP INTO GITHUB FROM YOUR LOCAL REPOSITORY

SO BASICALLY I WILL START FROM SCRATCH AND CREATE FIRST DOCKER FILE VISUAL STUDIO WHICH

WAS MY LOCAL REPOSITORY AND PUSHING IT INTO MY GITHUB REPOSITORY

SO BEFORE CREATING

# DOCKER FILE WE NEED TO CREATE A REPOSITORY IN GITHUB

The screenshot shows the GitHub interface for creating a new repository. The repository name is set to "Python flask app". The "Public" option is selected. Under "Initialize this repository with:", the "Add a README file" checkbox is checked. Below the interface, there is a "Quick setup" section with instructions for setting up the repository on desktop or via HTTPS/SSH, and a command-line setup section with a sample git commit message.

Quick setup — if you've done this kind of thing before

or   <https://github.com/Vickyproject/Python-flask-app.git>

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# Python-flask-app" >> README.md  
git init  
git add README.md  
git commit -m "first commit"
```

now create a docker file

To create a docker file go to github desktop and clone your repository and from there open in visual studio select

The screenshot shows the Visual Studio Code (VS Code) interface. At the top, the menu bar includes File, Edit, Selection, View, Go, Run, and ... On the right, there's a search bar with the text "2024-CICD-PROJECT". The main area has a dark theme. On the left, the Explorer sidebar shows a folder named "PYTHON-FLASK-APP" containing a file named "docker file". The "docker file" is selected and its content is displayed in the center-right panel. The content of the "docker file" is:

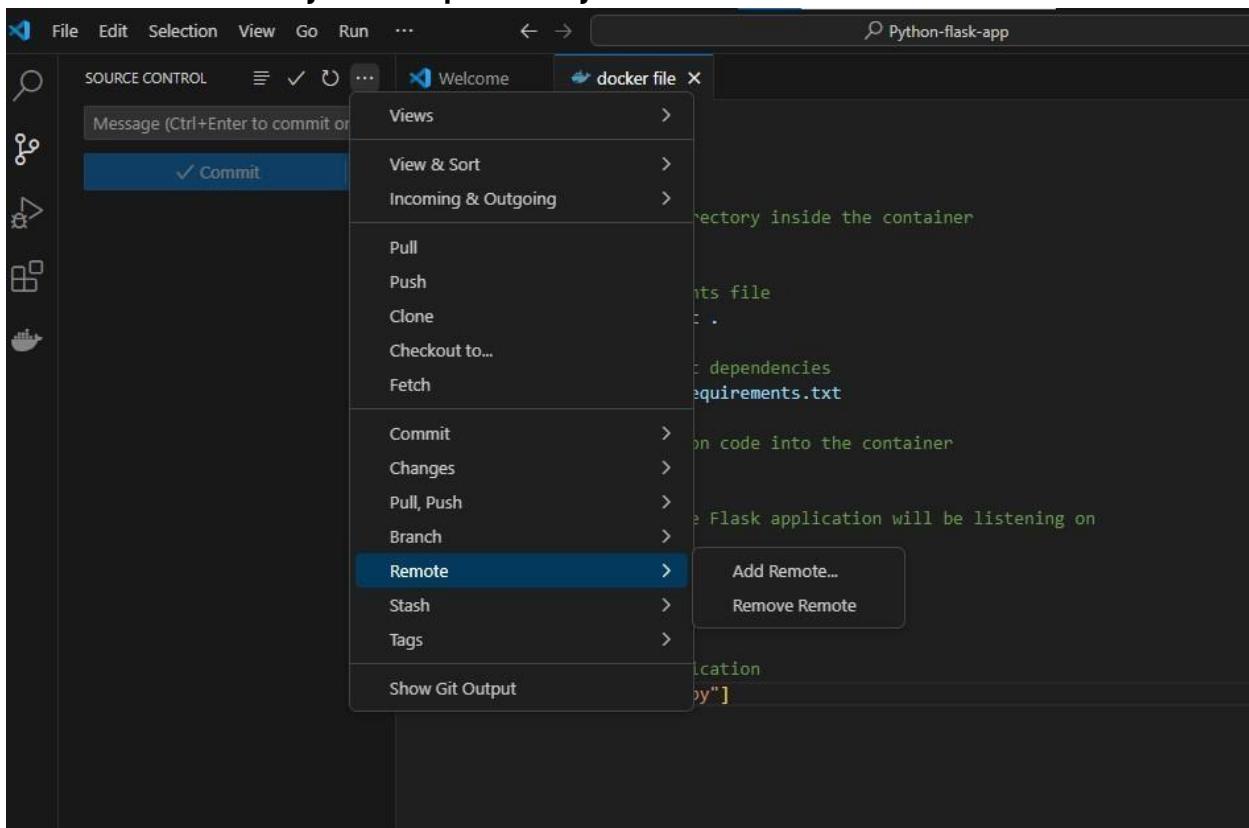
```
1 # Base image
2 FROM python:3.8
3
4 # Set the working directory inside the container
5 WORKDIR /app
6
7 # Copy the requirements file
8 COPY requirements.txt .
9
10 # Install the project dependencies
11 RUN pip install -r requirements.txt
12
13 # Copy the application code into the container
14 COPY . .
15
16 # Expose the port the Flask application will be listening on
17 EXPOSE 5000
18
19 # Set environment variables, if necessary
20 # ENV MY_ENV_VAR=value
21
22 # Run the Flask application
23 CMD ["python", "app.py"]
```

Now commit and push these into our create repository in github

A screenshot of the Visual Studio Code interface. The title bar shows "Python-flask-app". The left sidebar has icons for search, source control, and file operations. The main area shows a "docker file" tab with the following Dockerfile content:

```
1 # Base image
2 FROM python:3.8
3
4 # Set the working directory inside the container
5 WORKDIR /app
6
7 # Copy the requirements file
8 COPY requirements.txt .
9
10 # Install the project dependencies
11 RUN pip install -r requirements.txt
12
13 # Copy the application code into the container
14 COPY . .
15
16 # Expose the port the Flask application will be listening on
17 EXPOSE 5000
18
19 # Set environment variables, if necessary
20 # ENV MY_ENV_VAR=value
21
22 # Run the Flask application
23 CMD ["python", "app.py"]
```

Now add remote your repository url link



Quick setup — if you've done this kind of thing before

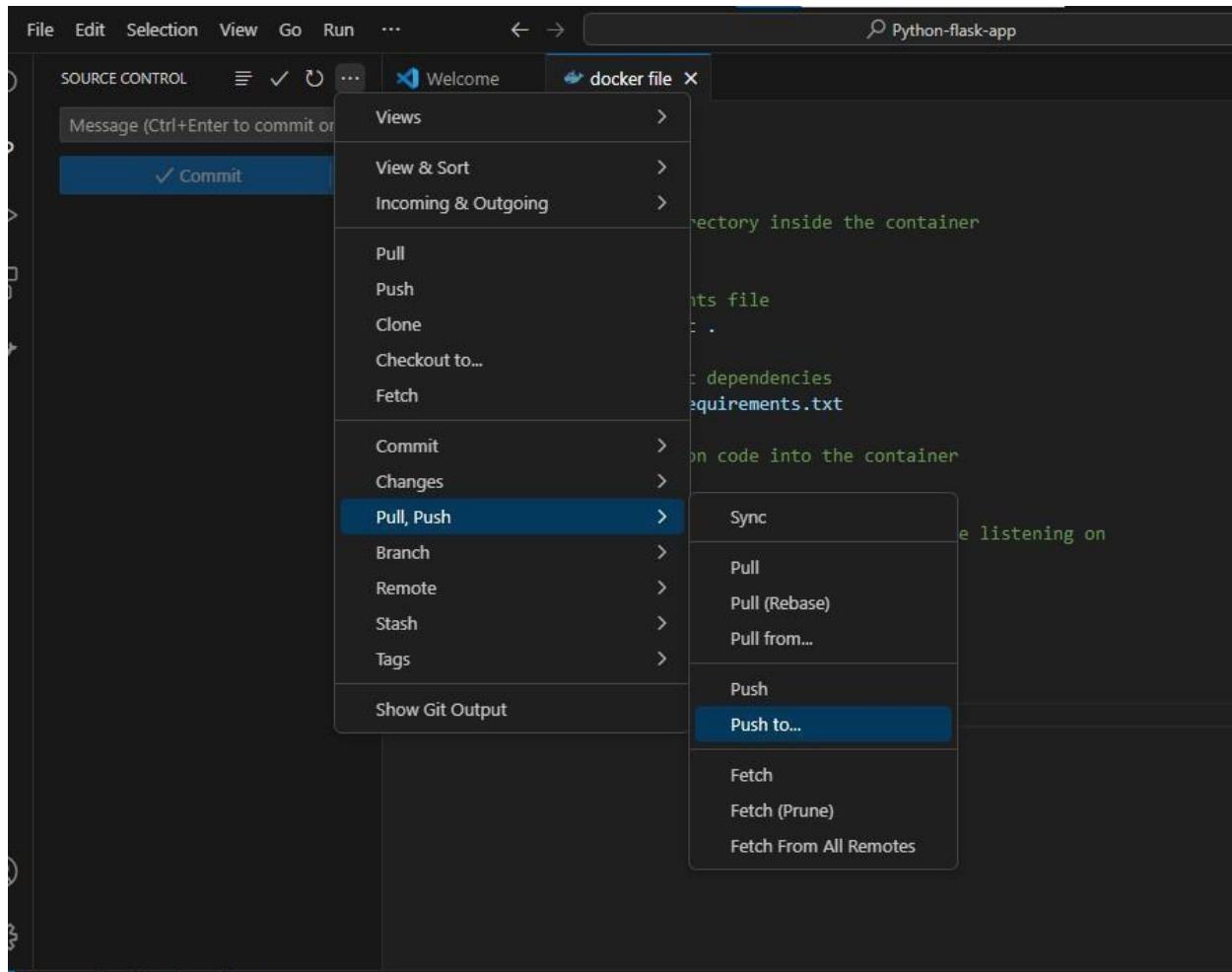
[Set up in Desktop](#) or [HTTPS](#) [SSH](#) <https://github.com/Vickyproject/Python-flask-app.git>

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

The screenshot shows two windows side-by-side. The top window is a web browser displaying the GitHub 'Quick setup' page for a new repository named 'Python-flask-app'. It shows options to 'Set up in Desktop', 'HTTPS', 'SSH', and provides the URL 'https://github.com/Vickyproject/Python-flask-app.git'. Below this, instructions for creating a new file or uploading an existing file are shown, along with a note about including README, LICENSE, and .gitignore files. The bottom window is a terminal session in a dark-themed IDE. The user has run the command 'git init' to initialize a local repository. They then run 'git remote add origin https://github.com/Vickyproject/Python-flask-app.git' to add the GitHub remote. Finally, they have created a 'dockerfile' and committed it with the message 'Message (Ctrl+Enter to commit on...)'.

```
1 # Base image
2 FROM python:3.8
3
4 # Set the working directory inside the container
5 WORKDIR /app
6
7 # Copy the requirements file
8 COPY requirements.txt .
9
10 # Install the project dependencies
11 RUN pip install -r requirements.txt
12
13 # Copy the application code into the container
14 COPY .
15
16 # Expose the port the Flask application will be listening on
17 EXPOSE 5000
18
19 # Set environment variables, if necessary
20 # ENV MY_ENV_VAR=value
21
22 # Run the Flask application
23 CMD ["python", "app.py"]
```



```
# Base image
FROM python:3.8

# Set the working directory inside the container
WORKDIR /app

# Copy the requirements file
COPY requirements.txt .

# Install the project dependencies
RUN pip install -r requirements.txt

# Copy the application code into the container
COPY . .

# Expose the port the Flask application will be listening on
EXPOSE 5000

# Set environment variables, if necessary
# ENV MY_ENV_VAR=value

# Run the Flask application
CMD ["python", "app.py"]
```

Vickyproject / Python-flask-app

Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Python-flask-app Public

Pin Unwatch 1 Fork 0

main 1 Branch 0 Tags Go to file Add file Code

Vickyproject dockerfile 3796dbc · 3 minutes ago 1 Commits

docker file dockerfile 3 minutes ago

README

Add a README

Help people interested in this repository understand your project by adding a README.

About

No description, website

Activity

0 stars

1 watching

0 forks

Releases

No releases published Create a new release

Packages

No packages published

We successfully push our file into github in a similar manner push all file here

Now APP.PY FILE WE NEED TO PUSH

The screenshot shows the VS Code interface. On the left, the Source Control sidebar displays a commit message: "Message (Ctrl+Enter to commit on...)" with a "Commit" button. Below it, the Changes tree shows "Changes 1" with "app.py". The main editor area shows the following Python code:

```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def hello():
    return 'Hello, world!'
if __name__ == '__main__':
    app.run()
```

The screenshot shows a GitHub repository page for "Python-flask-app". The top navigation bar includes "Code", "Issues", "Pull requests", "Actions", "Projects", "Wiki", "Security", "Insights", and "Settings". The repository details show "Vickyproject / Python-flask-app" and a search bar. The code tab is selected, showing the "main" branch with 1 branch and 0 tags. The file list includes "APP.PY" (by Vickyproject, 1 minute ago), "app.py" (1 minute ago), and "dockerfile" (8 minutes ago). The README file is also listed. The right sidebar provides repository statistics: "No description", "Activity", "0 stars", "1 watch", "0 forks", and a "Releases" section.

Now IN SAME MANNER LET PUSH ALL FILE

The screenshot shows a code editor interface with a dark theme. The top bar includes standard menu items like File, Edit, Selection, View, Go, Run, and a search bar labeled "Python-flask-app". Below the menu is a toolbar with icons for Source Control, Changes, and a Commit button. The main area displays a file named "buildspec.yml" with the following content:

```
version: 0.2
env:
  parameter-store:
    DOCKER_REGISTRY_USERNAME: /myapp/docker-credentials/username
    DOCKER_REGISTRY_PASSWORD: /myapp/docker-credentials/password
    DOCKER_REGISTRY_URL: /myapp/docker-registry/url
phases:
  install:
    runtime-versions:
      python: 3.11
  pre_build:
    commands:
      - echo "Installing dependencies..."
      - pip install -r day-13/simple-python-app/requirements.txt
  build:
    commands:
      - echo "Running tests..."
      - cd day-13/simple-python-app/
      - echo "Building Docker image..."
      - echo "$DOCKER_REGISTRY_PASSWORD" | docker login -u "$DOCKER_REGISTRY_USERNAME" --password-stdin "$DOCKER_REGISTRY_PASSWORD"
      - docker build -t "$DOCKER_REGISTRY_URL/$DOCKER_REGISTRY_USERNAME/simple-python-flask-app:latest".
      - docker push "$DOCKER_REGISTRY_URL/$DOCKER_REGISTRY_USERNAME/simple-python-flask-app:latest"
  post_build:
    commands:
      - echo "Build completed successfully!"
artifacts:
  files:
    - '**/*'
base-directory: ./simple-python-app
```

main ▾ 1 Branch 0 Tags Go to file Add file <> Code About

Vickyproject	buildspec.yml	668547b · 1 minute ago	3 Commits
app.py	APP.PY	6 minutes ago	
buildspec.yml	buildspec.yml	1 minute ago	
docker file	dockerfile	13 minutes ago	
<hr/>			
README			
<hr/>			

No descriptive

Activity

0 stars

1 watching

0 forks

---

#### Releases

No releases published

Create a new release

---

#### Packages

The screenshot shows a GitHub repository named 'Python-flask-app' and a corresponding project in VS Code. In the GitHub interface, the repository details are visible, including the README file. The VS Code window shows the code editor with the 'requirements.txt' file open, containing the word 'flask'. The GitHub interface includes navigation links like 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. The VS Code interface shows a sidebar with 'SOURCE CONTROL' and 'Sync Changes 1 ↑'.

SO HERE I SUCCESFULL PUSH ALL MY REQUIRE FILE INTO MY GITHUB AND WE WILL USE AWS CODECOMMIT

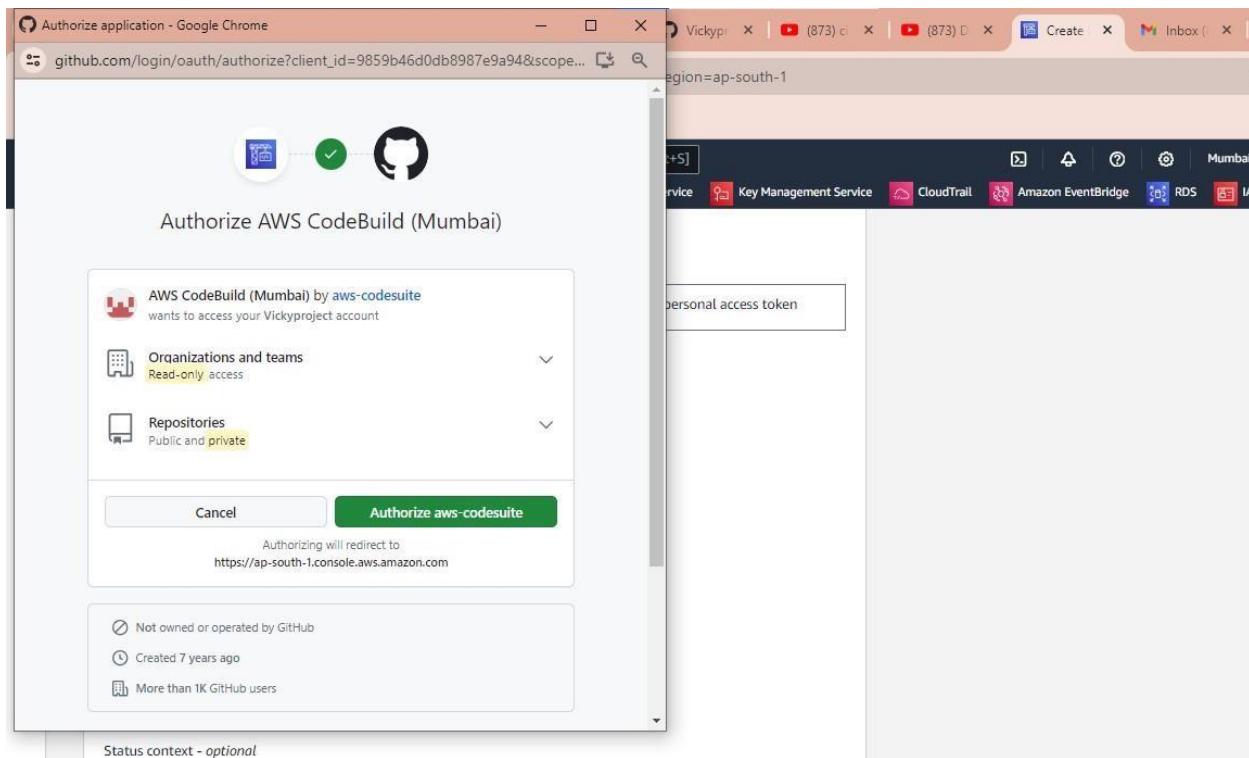
BUT ONE THING DON'T WORRY ABOUR YAML FILE WE WILL RE CREATE IT AGAIN STEP WISE FURTHER  
NOW WE WILL START WITH CODE BUILD AND GO TO AWS CONSOLE AND  
SEARCH AWS CODEBUILD AND CREATE CODE BUILD

The screenshot shows the AWS Lambda console search results for 'CODEBUILD'. The search bar at the top contains the text 'CODEBUILD'. Below it, a search result for 'Services' is shown, with 'CodeBuild' highlighted. The 'CodeBuild' service is described as 'Build and Test Code'. The left sidebar shows other services like EC2 and VPC.

The screenshot shows the AWS CodeBuild console. On the left, a sidebar navigation includes 'Developer Tools' and 'CodeBuild' sections. Under 'CodeBuild', there are links for 'Source • CodeCommit', 'Artifacts • CodeArtifact', 'Build • CodeBuild' (which is expanded), 'Getting started', 'Build projects', and 'Build history'. The main content area displays a 'Build projects' list titled 'Info'. It features a search bar, action buttons ('Actions', 'Create trigger', 'View details', 'Start build', 'Create project'), and a dropdown for 'Your projects'. A message 'No results' is shown, followed by a note: 'There are no results to display.' Below this is a table with columns: Name, Source provider, Repository, Latest build status, Description, and Last Modified. A large central window is titled 'Project configuration' for a project named 'SIMPLE-PYTHON-FLASK APP'. It contains fields for 'Description - optional' (empty), 'Build badge - optional' (unchecked), 'Enable concurrent build limit - optional' (unchecked), and a 'Tags' section (empty). The right side of the configuration window shows a preview of the project's build logs.

## SELECT TOOL SO I WILL ADD GITHUB WHICH I USED

This screenshot shows the 'Source' configuration screen for a build project. At the top, there is a 'Source' button and an 'Add source' button. Below it, the 'Source 1 - Primary' section is displayed. The 'Source provider' dropdown is set to 'GitHub', which is highlighted with a blue checkmark. Other options listed include 'No source', 'Amazon S3', 'AWS CodeCommit', 'Bitbucket', 'GitHub Enterprise', 'GitLab', and 'GitLab Self Managed'. The 'Environment image' section at the bottom shows 'Managed image' selected. A tooltip for the environment image dropdown states: 'The type of Amazon Lambda instances for builds. A fleet's default instance type will be used for the build if this is not specified.'



Connect it to your github account

ap-south-1.console.aws.amazon.com/codesuite/codebuild/oauth/github?code=7ac...   Mum  Vicky Omprakash Sh 

EC2 VPC S3 AWS Auto Scaling Simple Queue Service Simple Notification Service Key Management Service CloudTrail 

**Processing OAuth request**

Choose Confirm to connect CodeBuild to your GitHub account

Repository

Repository in my GitHub account  Public repository

GitHub repository

Use: "https://github.com/Vickyproject/Python-flask-app.gitD"

No results

You are connected to GitHub using OAuth.

**Disconnect from GitHub**

Connection status  
You are connected to GitHub using OAuth.

Source version - *optional* Info  
Enter a pull request, branch, commit ID, tag, or reference and a commit ID.

**Additional configuration**  
Git clone depth, Git submodules, Build status config

Git clone depth - *optional*

Git submodules - *optional*  
 Use Git submodules

Build Status - *optional*  
 Report build statuses to source provider when your builds start and finish

Status context - *optional*  
Configurable context with build status with support for environment variables.

Target URL - *optional*

CloudShell  Feedback

Region=ap-south-1

personal access token

© 2024, Amazon Web Services, Inc. or its affiliates.

Provisioning model info 

**On-demand**  
Automatically provision build infrastructure in response to new builds.

**Reserved capacity**  
Use a dedicated fleet of instances for builds. A fleet's compute and environment type will be used for the project.

Environment image

**Managed image**  
Use an image managed by AWS CodeBuild

**Custom image**  
Specify a Docker image

Compute

**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

Architecture

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-SIMPLE-PYTHON-FLASK APP-service-role

Type your service role name

I need to make a I am role later

**Buildspec**

Build specifications

**Insert build commands**  
Store build commands as build project configuration

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

Build commands

Enter commands you want to run during the build phase. Separate each build command with "&&." For example, "mvn test && mvn package." Use a buildspec file to run commands in other phases or if you have a long list of commands.

e.g. echo "Hello World" Switch to editor

**Batch configuration**

You can run a group of builds as a single execution. Batch configuration is also available in advanced option when starting build.

**Define batch configuration - optional**  
You can also define or override batch configuration when starting a build batch.

**Artifacts** Add artifact

Now write a script yaml file which is also known as build spec file so for that click on switch editor and info

AWS CodeBuild User Guide

What is AWS CodeBuild? Getting started Samples Plan a build Buildspec reference Batch buildspec reference GitHub Action runner buildspec reference Build environment reference Build locally VPC support Working with build projects and builds Working with AWS Lambda compute Working with reserved capacity Working with test reporting Logging and monitoring

User Guide

User Guide

("). The following command is enclosed in quotation marks because a colon (:) followed by a space is not allowed in YAML. The quotation marks are escaped (\").

```
"export PACKAGE_NAME=$(cat package.json | grep name | head -1 | awk -F: '{ print $2 }' | sed 's/[\"\",]/\\\"\\\"')"
```

The buildspec has the following syntax:

```
version: 0.2

run-as: Linux-user-name

env:
  shell: shell-tag
  variables:
    key: "value"
    key: "value"
  parameter-store:
    key: "value"
    key: "value"
  exported-variables:
    - variable
    - variable
  secrets-manager:
    key: secret-id:json-key:version-stage:version-id
  git-credential-helper: no | yes

proxy:
  upload-artifacts: no | yes
```

**version: 0.2**

**env:**

**parameterstore:**  
**key: "value" key:**  
**"value"**

**phases:**

**install:**

**run-as: Linux-user-name**

**on-failure: ABORT | CONTINUE**

**runtime-versions: runtime:**

**version runtime: version**

**commands:**

- comm

a nd

- comm

a

nd

finally:

- comm

a nd

- comma

nd #

steps:

pre\_bui

ld:

n-as: Linux-user-name

ru  
on

#

il

bu<sup>ru</sup>  
on

#

ru  
on

-failure: ABORT | CONTINUE

commands:

- comm

a nd

- comma

nd

finally:

- comm

a nd

- comm

a nd

steps:

d:

n-as: Linux-user-name

-failure: ABORT | CONTINUE

commands:

comm

a nd

- comma

nd

finally:

comm

a nd

comm

a nd

steps:

post\_build:

n-as: Linux-user-name

-failure: ABORT | CONTINUE

commands:

comm

a nd

comma

nd

finally:

```
comm  
and
```

SO BASICALLY WE GONNA CHANGE THESE DEFAULT CODE INTO OUR REQUIREMENT SCRIPT ONLY

FIRST WE WILL GO FOR INSTALL

BEFORE EDIT

phases:

```
install:  
    n -as: Linux- user-name  
    ru failure: ABORT |  
on-    CONTINUE e-versions:  
    | runtime:  
runtimversion    runtime: version  
    commands:  
  
    -  
  
    comm  
    and  
    -  
    comm  
    and  
    finally:  
    -  
    comm  
    and  
    -  
    comm
```

a nd  
steps:

```
# pre_build:  
run-as: Linux-user-name  
on-failure: ABORT |  
CONTINUE commands:
```

- comm

a nd

```
- comma  
nd  
finally:
```

- comm

a nd -

comm

a nd d:

buil n-as: Linux-user-name  
ru -failure: ABORT | CONTINUE  
on commands:

#

ru  
on

- comm  
a nd  
- comma  
nd  
finally:

- comm  
a nd

- comm

```
a nd  
steps:  
post_build:  
  n-as: Linux-user-name  
  -failure: ABORT | CONTINUE  
  commands:  
    -  
      comm  
      a nd  
    - comma  
      nd  
      finally:  
    -  
      comm  
      a nd
```

## AFTER EDIT

```
phases:  
  install :  
  runtime-versions:  
    python: 3.11
```

#so after that our next aim to edit prebuilt code since for our python application we require flask which was in requirements.txt so we need to mention these

```
pre_build:
```

```
  command  
  s:
```

```
pip install -r requirements.txt #now we need to build our code mean  
according to diagram we need to #build and ut , image build and image  
push
```

**build:**      **commands:**      echo

“Building Docker Image”

Docker build -t “<>”

Docker push <>

```
#basically these build command is incompletre because we need to  
enter security cereditial and docker login to build image so for these  
we will use aws system management and here we need to access to  
role of aws sms#
```

```
# NOW IN POST BUILD WE SIMPPLY GIVE A MESSAGE THAT BUILD  
DOCKER IS SUCCESS
```

**post\_build:**

-

**command**

**s:**

```
echo “build is success” #for our part one which is CI we just need  
these  
build spect only so proceed with these create build spect
```

Final code:-

**phases:**

**install:**

**runtimeversions:**

**python:3.11 pre\_build:**

**commands:**

`pip install -r requirements.txt`

**build:**

**commands:**

`echo "Building Docker  
Image"`

`Docker build -t "<>"`

`Docker push`

`<>`

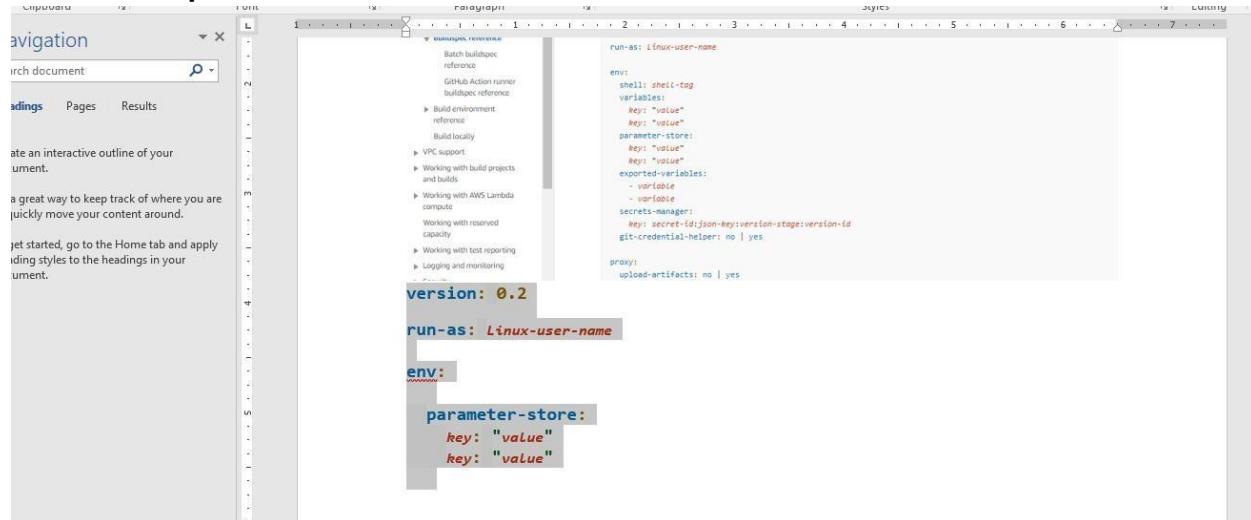
**post\_build:**

**command**

**s:**

`echo "build is success"`

Now add parameter which is our secret creditinal



**version: 0.2**

**env:**

```
parameters
e: key: "value"
key:
"value"

version : 0.2
env: parameter-store: docker_user: /python-
app/dockercredential/username docker_password: /python-
app/dockercredential/password docker_registry_url: /python-
app/dockercredential/username
```

so final code will be

```
version : 0.2
env:
parameter-
store:
docker_user:
/pythonapp/doc
kercredential/u
serna me
docker_passwo
rd:
/pythonapp/doc
kercredential/p
asswo rd
docker_registry_
ur l:
/pythonapp/dock
ercredential/use
```

**rna me phases:**

**install:**

**runtimeversions:**

**python:3.11**

**pre\_build:**

**command**

**s:**

**pip install -r requirements.txt**

**build:**

**commands:**

echo "Building Docker  
Image"

Docker build -t "<>"

Docker push

<>

But still we need to add tag and docker push

So replace these om tag

"\$docker\_registry\_url/\$docker\_user/Pythonflask-

app:latest" And same for push also

**version : 0.2**

**env: parameter-store: docker\_user: /python-  
app/dockercredential/username docker\_password: /python-  
app/dockercredential/password docker\_registry\_url: /python-  
app/dockercredential/docker-urlregistry phases:**

**install :**

**runtim e-versions:**

**python:3.11**

```
pre_build:  
commands:  
- echo "Building Docker Image"  
- docker build -t  
"${docker_registry_url}/${docker_user}/Pythonflaskapp:latest"  
- echo "Pushing Docker Image"  
- docker push "${docker_registry_url}/${docker_user}/Python-  
flaskapp:latest"
```

So basically these is our build spec yaml file

Now re edit and  
run

New  
code

version: 0.2

```

env:
  parameter-store:
    docker user: "/python-app/docker-credential/username"
    docker password: "/python-app/docker-credential/password"
    docker_registry_url: "/python-app/docker-credential/docker-url-
registry"

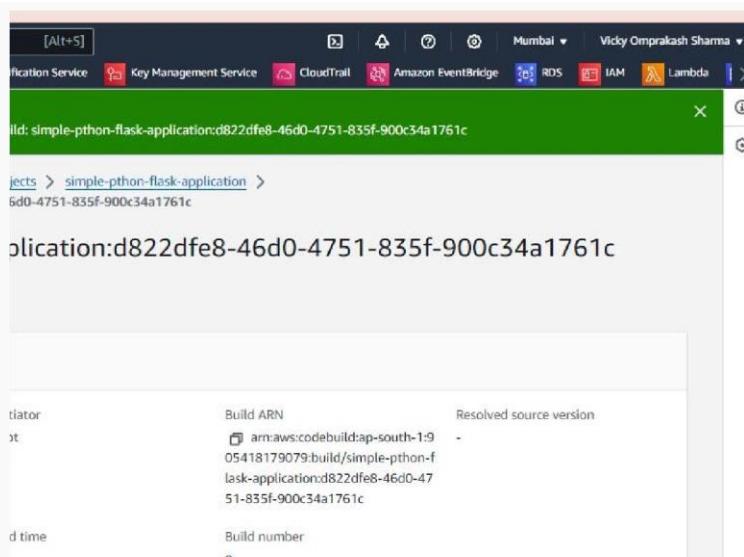
phases:
  install:
    runtime-versions:
      python: 3.11

  pre build:
    commands:
      - echo "Building Docker Image"
      - docker build -t "${docker_registry_url}/${docker_user}/Python-
flask-app:latest" .
      - echo "Pushing Docker Image"
      - docker push "${docker_registry_url}/${docker_user}/Python-
flask-app:latest"

```

**post build:**  
**commands:**

-echo "Build is  
 Sucessfull"



```

11 [Container] 2024/04/02 16:34:51.146670 Processing environment variables
12 [Container] 2024/04/02 16:34:51.175899 Decrypting parameter store environment variables
13 [Container] 2024/04/02 16:34:51.268228 Phase complete: DOWNLOAD_SOURCE State: FAILED
14 [Container] 2024/04/02 16:34:51.268272 Phase context status code: Decrypted Variables Error Message: parameter does not exist: /python-
app/docker-credential/registry-url
15

```

So here we need to check the path of register url and its correct parameter name in build spec

```
26 [Container] 2024/04/02 16:58:05.226009 INSTALL: 0 commands
27 [Container] 2024/04/02 16:58:05.226015 PRE_BUILD: 4 commands
28 [Container] 2024/04/02 16:58:05.226306 Phase complete: DOWNLOAD_SOURCE State: SUCCEEDED
29 [Container] 2024/04/02 16:58:05.226317 Phase context status code: Message:
30 [Container] 2024/04/02 16:58:05.298751 Entering phase INSTALL
31 [Container] 2024/04/02 16:58:05.300487 Phase complete: INSTALL State: SUCCEEDED
32 [Container] 2024/04/02 16:58:05.300510 Phase context status code: Message:
33 [Container] 2024/04/02 16:58:05.332156 Entering phase PRE_BUILD
34 [Container] 2024/04/02 16:58:05.332571 Running command echo "Building Docker Image"
35 Building Docker Image
36
37 [Container] 2024/04/02 16:58:05.337164 Running command docker build -t "${docker_registry_url}/${docker_user}/Python-flask-app:latest" .
38 ERROR: invalid tag "****/**/Python-flask-app:latest": repository name must be lowercase
39
40 [Container] 2024/04/02 16:58:05.478107 Command did not exit successfully docker build -t "${docker_registry_url}/${docker_user}/Python-flask-
app:latest" . exit status 1
41 [Container] 2024/04/02 16:58:05.481470 Phase complete: PRE_BUILD State: FAILED
42 [Container] 2024/04/02 16:58:05.481489 Phase context status code: COMMAND_EXECUTION_ERROR Message: Error while executing command: docker build
"${docker_registry_url}/${docker_user}/Python-flask-app:latest" .. Reason: exit status 1
43
```

Again we are facing some issue so we need to check that issue and troubleshoot it

Here these error saying your tag is invalid since we enter our project name in uppercase at some point so we need to make it lowercase

**version: 0.2**

**env:**

**parameter-**

**store:**

```
  docker_user: "/python-
  app/dockercredential/username"
  docker_password: "/pythonapp/docker-
  credential/password"    docker_registry_url:
  "/python-app/dockercredential/docker-url-
  registry"
```

**phases:**

**install:**

**runtime-versions:**

**python: 3.11**

**pre\_build:**

**commands:**

- echo "Building Docker Image"
- docker build -t "\${docker\_registry\_url}/\${docker\_user}/python-flaskapp:latest" .
- echo "Pushing Docker Image"
- docker push "\${docker\_registry\_url}/\${docker\_user}/python-flaskapp:latest"

**post\_build:**

**commands:**

**-echo "Build is Sucessfull"**

**Now again do build spec**

**Finally after a lot of effort we finally get result**

simple-python-flask-application:7b4a1c3a-5987-4410-a549-a7d67b47080a

Status	Initiator	Build ARN	Resolved source version
Succeeded	root	arn:aws:codebuild:ap-south-1:905418179079:build/simple-python-flask-application:7b4a1c3a-5987-4410-a549-a7d67b47080a	4755613e4964ff0023e00fea701aea41ec27d3

Start time	End time	Build number
Apr 3, 2024 12:47 AM (UTC+5:30)	Apr 3, 2024 12:48 AM (UTC+5:30)	16

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

```

15 [Container] 2024/04/02 19:18:09.425819 Running command echo "Installing Python version 3.11 ..."
16 Installing Python version 3.11 ...
17
18 [Container] 2024/04/02 19:18:09.430349 Running command pyenv global ${PYTHON_311_VERSION}
19
20 [Container] 2024/04/02 19:18:10.460967 Moving to directory /codebuild/output/src869693519/src/github.com/Vickyproject/Python-flask-app
21 [Container] 2024/04/02 19:18:10.462415 Unable to initialize cache download: no paths specified to be cached
22 [Container] 2024/04/02 19:18:10.553802 Configuring ssm agent with target id: codebuild:7b4a1c3a-5987-4410-a549-a7d67b47080a
23 [Container] 2024/04/02 19:18:10.576751 Successfully updated ssm agent configuration
24 [Container] 2024/04/02 19:18:10.576998 Registering with agent
25 [Container] 2024/04/02 19:18:10.615663 Phases Found in YAML: 1
26 [Container] 2024/04/02 19:18:10.615678 INSTALL: 0 commands
27 [Container] 2024/04/02 19:18:10.615949 Phase complete: DOWNLOAD_SOURCE State: SUCCEEDED
28 [Container] 2024/04/02 19:18:10.615960 Phase context status code: Message:
29 [Container] 2024/04/02 19:18:10.681331 Entering phase INSTALL
30 [Container] 2024/04/02 19:18:10.682890 Phase complete: INSTALL State: SUCCEEDED
31 [Container] 2024/04/02 19:18:10.682902 Phase context status code: Message:
32 [Container] 2024/04/02 19:18:10.713196 Entering phase PRE_BUILD
33 [Container] 2024/04/02 19:18:10.714808 Phase complete: PRE_BUILD State: SUCCEEDED
34 [Container] 2024/04/02 19:18:10.714820 Phase context status code: Message:
35 [Container] 2024/04/02 19:18:10.744313 Entering phase BUILD
36 [Container] 2024/04/02 19:18:10.745810 Phase complete: BUILD State: SUCCEEDED
37 [Container] 2024/04/02 19:18:10.745822 Phase context status code: Message:
38 [Container] 2024/04/02 19:18:10.778774 Entering phase POST_BUILD
39 [Container] 2024/04/02 19:18:10.780177 Phase complete: POST_BUILD State: SUCCEEDED
40 [Container] 2024/04/02 19:18:10.780186 Phase context status code: Message:
41

```

Build started You have successfully started the following build: simple-python-flask-application:7b4a1c3a-5987-4410-a549-a7d67b47080a

simple-python-flask-application:7b4a1c3a-5987-4410-a549-a7d67b47080a

Status	Initiator	Build ARN	Resolved source version
Succeeded	root	arn:aws:codebuild:ap-south-1:905418179079:build/simple-python-flask-application:7b4a1c3a-5987-4410-a549-a7d67b47080a	4755613e4964ff0023e00fea701aea41ec27d3

Start time	End time	Build number
Apr 3, 2024 12:47 AM (UTC+5:30)	Apr 3, 2024 12:48 AM (UTC+5:30)	16

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

Showing the last 40 lines of the build log. [View entire log](#)

Tail logs

No previous logs

But here we didn't getting output because our docker file name is incorrect and content of docker file so basically I had firsrt rename my docker file name and run the code again

Build projects Info

C Actions ▾ Create trigger View details Start build ▾ Create project

Q Your projects ▾ < 1 > ⌂

Name	Source provider	Repository	Latest build status	Description	Last Modified
simple-python-flask-app	GitHub	Vickyproject/Python-flask-app ↗	Succeeded	-	9 minutes ago

```
245 b6f975b275ed: Waiting
246 9a6b1661f2d4: Waiting
247 e077e19b6682: Waiting
248 21e1c4948146: Waiting
249 68866beb2ed2: Waiting
250 e6e2ab10dba6: Waiting
251 0238a1790324: Waiting
252 bcf931b51217: Mounted from library/python
253 9f6e3ee457ae: Pushed
254 0bccfb2f92ac: Pushed
255 a1d2d8442d1a: Pushed
256 b6f975b275ed: Mounted from library/python
257 e050618ab2ba: Pushed
258 21e1c4948146: Mounted from library/python
259 68866beb2ed2: Mounted from library/python
260 9a6b1661f2d4: Mounted from library/python
261 e077e19b6682: Mounted from library/python
262 e6e2ab10dba6: Mounted from library/python
263 0238a1790324: Mounted from library/python
264 latest: digest: sha256:b8c439bb2da0b5fa0c2bcb90052fb7860173ecbc02f4f4bf493b690129f71184 size: 2841
265
266 [Container] 2024/04/04 16:09:43.741168 Phase complete: BUILD State: SUCCEEDED
267 [Container] 2024/04/04 16:09:43.741189 Phase context status code: Message:
268 [Container] 2024/04/04 16:09:43.778521 Entering phase POST_BUILD
269 [Container] 2024/04/04 16:09:43.778961 Running command echo "Build is Successful"
270 Build is Successful
271
272 [Container] 2024/04/04 16:09:43.783229 Phase complete: POST_BUILD State: SUCCEEDED
273 [Container] 2024/04/04 16:09:43.783242 Phase context status code: Message:
274
```

Main source code mean build spec file last and final successful  
update version: 0.2

env: parameter-

store:

```
DOCKERHUB_USER: "/python-app/docker-credential/user"
DOCKER_PASSWORD: "/python-app/docker-credential/password"
```

```
    DOCKER_REGISTRY_URL: "/python-app/docker-credential/docker-  
urlregistry"
```

```
    IMAGE_NAME: "/python-app/image"
```

```
phases: install:
```

```
runtime-
```

```
versions:
```

```
    python: 3.11
```

```
pre_build:
```

```
commands:
```

```
- echo "Installing dependencies..."
```

```
- pip install -r requirements.txt
```

```
build:
```

```
commands:
```

```
- echo "Logging in to Docker Hub..."
```

```
- echo "$DOCKER_PASSWORD" | docker login -u  
"$DOCKERHUB_USER" -password-stdin "$DOCKER_REGISTRY_URL"
```

```
- echo "Building Docker Image"
```

```
- docker build -t "${DOCKERHUB_USER}/${IMAGE_NAME}:latest" -f  
Dockerfile .
```

```
- echo "Pushing Docker Image"
```

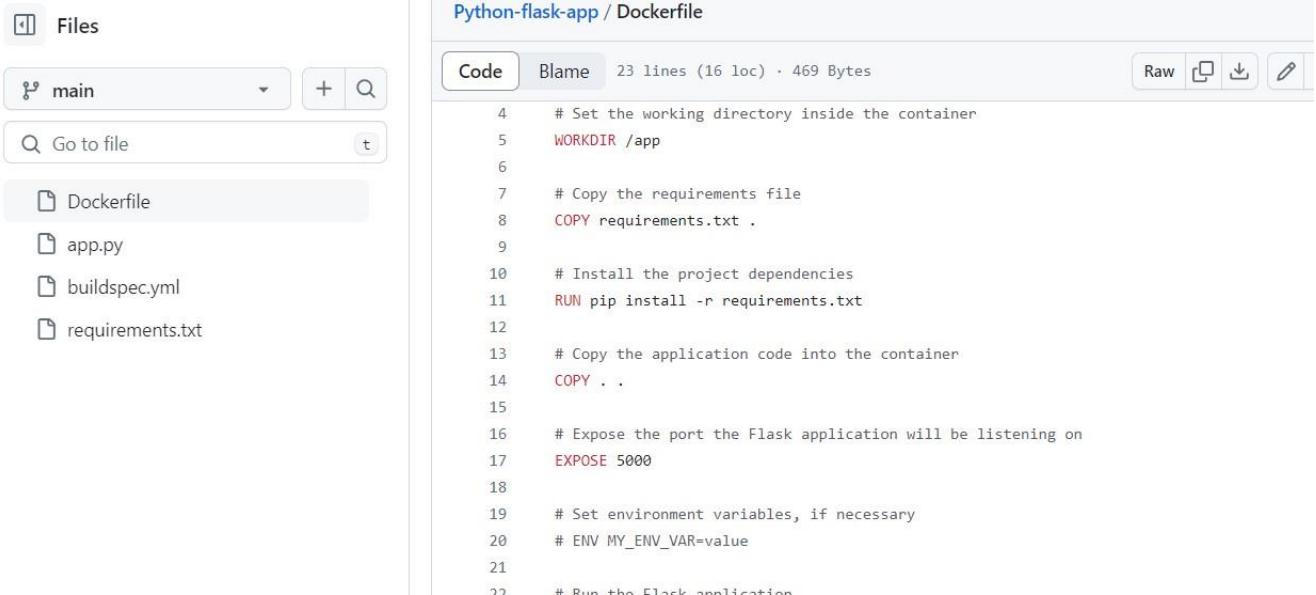
```
- docker push "${DOCKERHUB_USER}/${IMAGE_NAME}:latest"
```

```
post_build:
```

```
commands:
```

```
- echo "Build is Successful"
```

And one important thing also in build -t command give path which is -f and your dockerfile name



The screenshot shows a GitHub repository named 'Python-flask-app'. On the left, there's a sidebar titled 'Files' with a tree view showing 'main' (a folder), 'Dockerfile' (selected), 'app.py', 'buildspec.yml', and 'requirements.txt'. The main area is titled 'Python-flask-app / Dockerfile' and contains the following code:

```
4 # Set the working directory inside the container
5 WORKDIR /app
6
7 # Copy the requirements file
8 COPY requirements.txt .
9
10 # Install the project dependencies
11 RUN pip install -r requirements.txt
12
13 # Copy the application code into the container
14 COPY . .
15
16 # Expose the port the Flask application will be listening on
17 EXPOSE 5000
18
19 # Set environment variables, if necessary
20 # ENV MY_ENV_VAR=value
21
22 # Run the Flask application
```

## Dockerfile content

```
# Base image
```

```
FROM python:3.8
```

```
# Set the working directory inside the container
```

```
WORKDIR /app
```

```
# Copy the requirements file
```

```
COPY requirements.txt .
```

```
# Install the project dependencies
```

```
RUN pip install -r requirements.txt
```

```
# Copy the application code into the  
container COPY ..
```

```
# Expose the port the Flask application will be listening on  
EXPOSE 5000
```

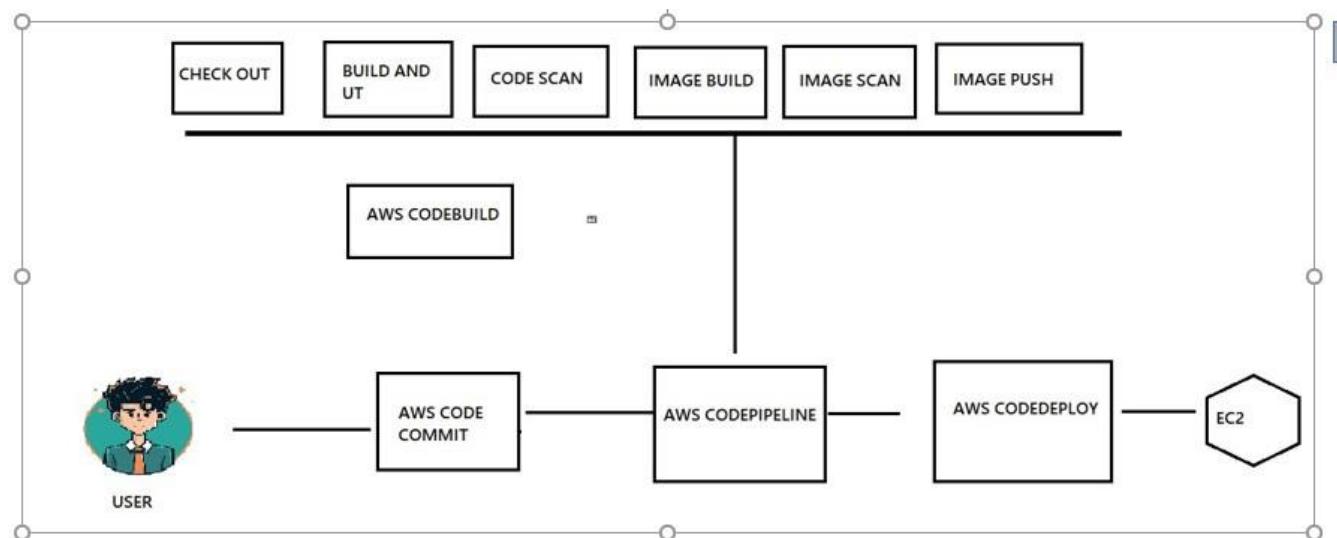
```
# Set environment variables, if necessary
```

```
# ENV MY_ENV_VAR=value
```

```
# Run the Flask application
```

```
CMD ["python", "app.py"]
```

NOW GO FOR FURTHER PART TILL NOW WE HAVE DONE WITH CODEBUILD



NOW AS PER DIAGRAM WE NEED TO INTEGRATE THESE WITH OUR AWS CODEPIPELINES SO WE DON'T NEED TO CHANGE IT AGAIN AND AGAIN IT WILL AUTOMATE WHENEVER WE MADE CHANGE IN GITHUB CODE

## SO GO TO CODEPIPELINES

The screenshot shows the AWS CodePipeline console interface. At the top, there's a navigation bar with various AWS services like EC2, VPC, S3, AWS Auto Scaling, Simple Queue Service, Simple Notification Service, Key Management Service, CloudTrail, Amazon EventBridge, RDS, IAM, Lambda, and DynamoDB. Below the navigation bar, the main area has a sidebar titled "Developer Tools" with sections for "CodePipeline" and "Getting started". The main content area is titled "Pipelines" and shows a table with columns for Name, Latest execution status, Latest source revisions, Latest execution started, and Most recent executions. A message says "No results" and "There are no results to display." On the right side, there's a "Create pipeline" button. The central part of the screen is titled "Pipeline settings" and shows a form to create a new pipeline. The "Pipeline name" field contains "simple-python-flask-app". The "Pipeline type" section shows "V2" selected. The "Execution mode" section shows "Superseded" selected. The "Service role" section is partially visible. To the right of the form, there's a sidebar with sections for "Choose your pipeline type", "For information about pricing for CodePipeline, see Pricing.", and "Create or choose your CodePipeline service role".

**Pipeline settings**

**Pipeline name**  
Enter the pipeline name. You cannot edit the pipeline name after it is created.  
  
No more than 100 characters

**Pipeline type**  
The pipeline type determines the pipeline structure and availability of parameters such as triggers. Pipeline type selection will impact features and pricing. [Which pipeline is right for me?](#)

**V1**

**V2**

**Execution mode**  
Choose the execution mode for your pipeline. This determines how the pipeline is run.

**Superseded**  
A more recent execution can overtake an older one. This is the default.

**Queued (Pipeline type V2 required)**  
Executions are processed one by one in the order that they are queued.

**Parallel (Pipeline type V2 required)**  
Executions don't wait for other runs to complete before starting or finishing.

**Service role**

**Choose your pipeline type**  
CodePipeline provides the following pipeline types, which differ in characteristics and price. Choose one of the following pipeline types:

- **V1** type pipelines have a JSON structure that contains standard pipeline, stage, and action-level parameters.
- **V2** type pipelines have the same structure as a V1 type, along with additional parameter support, such as triggers on Git tags and pipeline-level variables.

**For information about pricing for CodePipeline, see [Pricing](#).**

**Create or choose your CodePipeline service role**  
You only need to create a service role the first time you create a pipeline in CodePipeline. The

#### 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-ap-south-1-simple-python-flask-app

Type your service role name

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

#### Variables

You can add variables at the pipeline level. You can choose to assign the value when you start the pipeline. Choosing this option requires pipeline type V2. Learn more [\[?\]](#)

No variables defined at the pipeline level in this pipeline.

#### Source

##### Source provider

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

GitHub (Version 2)



##### New GitHub version 2 (app-based) action

To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. Learn more

##### Connection

Choose an existing connection that you have already configured, or create a new one and then return to this task.



or

[Connect to GitHub](#)

##### Repository name

Choose a repository in your GitHub account.



following pipeline types, which differ in characteristics and pricing. Choose one of the following pipeline types:

- **V1** type pipelines have a JSON structure that contains standard pipeline, stage, and action-level parameters.
- **V2** type pipelines have the same structure as a V1 type, along with additional parameter support, such as triggers on Git tags and pipeline-level variables.

For information about pricing for CodePipeline, see [Pricing](#).

[Create or choose your CodePipeline service role](#)

You only need to create a service role if you choose the V1 type.

#### Choose a provider

CodePipeline follows the differences between the two providers. Choose the provider that matches your pipeline requirements.

- **V1** provides a simple JSON structure for defining pipeline stages and actions.
- **V2** provides a more complex structure with support for triggers on Git tags and pipeline-level variables.

For more information, see [CodePipeline providers](#).

[Create a new pipeline](#)

You can also edit an existing pipeline.

Now connect to your github

Developer Tools > ... > Create connection

## Create a connection Info

**Create GitHub App connection Info**

Connection name  
vicky-python-app

▶ Tags - optional

**Connect to GitHub**

To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. [Learn more](#)

Choose an existing connection that you have already configured, or create a new one and then return to this task.

arn:aws:codestar-connections:ap-south-1:905418179079:connection/4adb3:  or  Connect to GitHub

**Ready to connect**  
Your GitHub connection is ready for use.

Repository name  
Choose a repository in your GitHub account.

You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

Choose the provider and then provide the connection details.

Service Key Management Service CloudTrail Amazon EventBridge

Feedback Privacy Terms Cookie preferences © 2024 Amazon Web Services, Inc. or its affiliates.

arn:aws:codestar-connections:ap-south-1:905418179079:connection/4adb3: X

or

Connect to GitHub



### Ready to connect

Your GitHub connection is ready for use.

#### Repository name

Choose a repository in your GitHub account.

Vickyproject/github-project



You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

#### Default branch

Default branch will be used only when pipeline execution starts from a different source or manually started.

main



#### 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.

## Trigger

#### Trigger type

Choose the trigger type that starts your pipeline.

##### No filter

Starts your pipeline on any push and clones the HEAD.

##### Specify filter

Starts your pipeline on a specific filter and clones the exact commit. Pipeline type V2 is required.

##### Do not detect changes

Don't automatically trigger the pipeline.

#### Event type

Choose the event type for the trigger that starts your pipeline.

##### Push

##### Pull request

#### Filter type

Choose the filter type for the event that starts your pipeline.

##### Branch

##### Tag

C  
fc  
di  
C  
pi  
•  
•  
•  
F  
C  
C  
C  
Yi  
rc  
pi  
• follow  
differ  
Choos  
pipeli  
• V1  
JS  
st:  
ac  
• V2  
sa  
alr  
pa  
tri  
pij  
For in  
CodeI  
Creat  
Code  
You c  
role t  
pipeli  
•  
•

For in  
CodeI  
Creat  
Code  
You c  
role t  
pipeli  
•  
•

#### Branches

You can specify the target branch or branches you are pushing to. Use a comma to specify multiple entries.

#### Include

*Enter branches or patterns*

#### Exclude

*Enter branches or patterns*

#### File paths - optional

You can specify the file path or file paths you are pushing to. Use a comma to separate multiple entries.

#### Include

*Enter file paths or patterns*

#### Exclude

*Enter file paths or patterns*

 You can add additional sources and triggers by editing the pipeline after it is created.

Home > Pipelines > Create new pipeline

## Add build stage Info

Step 3 of 5

### Build - optional

#### Build provider

This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output

AWS CodeBuild



AWS CodeBuild

Add Jenkins

#### Project name

Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.



or

#### Environment variables - optional

### Build - optional

**Build provider**  
This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.

AWS CodeBuild ▾

**Region**

Asia Pacific (Mumbai) ▾

**Project name**  
Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.

or Create project

---

**Project name**  
Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.

or Create project

**Environment variables - optional**  
Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. [Learn more](#)

Add environment variable

**Build type**

Single build  
Triggers a single build.

Batch build  
Triggers multiple builds as a single execution.

- **V1** type pipeline JSON structure standard pipeline action-level pipeline
- **V2** type pipeline same structure along with additional parameter supports triggers on GitHub pipeline-level

For information about creating a CodePipeline, see [Create or choose a CodePipeline](#)

You only need to differ in the [Choose pipeline type](#)

- **V1** type pipeline JSON structure standard pipeline action-level pipeline
- **V2** type pipeline same structure along with additional parameters supports triggers on GitHub pipeline-level

For information about creating a CodePipeline, see [Create or choose a CodePipeline](#)

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1 Choose pipeline settings Review Info Step 5 of 5

Step 2 Add source stage

Step 3 Add build stage

Step 4 Add deploy stage

Step 5 Review

### Step 1: Choose pipeline settings

**Pipeline settings**

Pipeline name: simple-python-flask-app

Pipeline type: V2

Execution mode: SUPERSEDED

Artifact location: A new Amazon S3 bucket will be created as the default artifact store for your pipeline

Service role name: AWSCodePipelineServiceRole-20180614-1-simple-python-flask-app

V2

Execution mode

SUPERSEDED

Artifact location

A new Amazon S3 bucket will be created as the default artifact store for your pipeline

Service role name

AWSCodePipelineServiceRole-ap-south-1-simple-python-flask-app

pipelines

- V1 type

JSON structure

standard pipeline

artifact

- V2 type

same structure

along with additional

parameters

triggers

pipeline

For info

CodePip

Create

CodePip

You only

role the

pipeline

## Variables

Name	Default value	Description
No variables		

Name	Default value	Description
No variables		

No variables defined at the pipeline level in this pipeline.

Choose your pipeline

CodePipeline provides

following pipeline

differ in character:

Choose one of the

pipeline types:

- V1 type pipeline

JSON structure

standard pipeline

action-level par

- V2 type pipeline

same structure

along with additional

parameters supported

triggers on Git commit

pipeline-level variables

For information about

CodePipeline, see the

Create or choose your

CodePipeline service

You only need to do

role the first time you

create a pipeline in CodePip

## Step 2: Add source stage

### Source action provider

Source action provider

GitHub (Version 2)

OutputArtifactFormat

CODE\_ZIP

DetectChanges

false

ConnectionArn

arn:aws:codestar-connections:ap-south-1:905418179079:connection/4adb332a-11c2-4d50-8ce7-5f3fe5189ad2

FullRepositoryUrl

```
-->-->
DetectChanges
false

ConnectionArn
arn:aws:codestar-connections:ap-south-1:905418179079:connection/4adb332a-11c2-4d50-8ce7-5f3fe5189ad2

FullRepositoryId
Vickyproject/github-project

Default branch
main
```

## Trigger configuration

You can add additional pipeline triggers after the pipeline is created.

Trigger type

Specify filter

Event type

### Trigger configuration

You can add additional pipeline triggers after the pipeline is created.

Trigger type

Specify filter

Event type

Push

Filter type

Branch

Include branches

-

Exclude branches

-

Include file paths

-

Exclude file paths

-

following pipeline types differ in characteristics:  
Choose one of the following pipeline types:

- **V1** type pipelines: JSON structure follows standard pipeline action-level parameters

- **V2** type pipelines: same structure as V1 along with additional parameter support triggers on Git tags and pipeline-level variables

For information about CodePipeline, see [Create or choose your CodePipeline service](#)

You only need to create the first time you choose your pipeline type:  
CodePipeline provides the following pipeline types that differ in characteristics:  
Choose one of the following pipeline types:

- **V1** type pipelines: JSON structure follows standard pipeline action-level parameters

- **V2** type pipelines: same structure as V1 along with additional parameter support triggers on Git tags and pipeline-level variables

For information about CodePipeline, see [Create or choose your CodePipeline service](#)

You only need to create the first time you choose your pipeline in CodePipeline.

**Step 3: Add build stage**

Build action provider	AWS CodeBuild
ProjectName	simple-python-flask-app

**Step 4: Add deploy stage**

Deploy action provider	No deploy
------------------------	-----------

Build action provider	AWS CodeBuild
ProjectName	simple-python-flask-app

**Step 4: Add deploy stage**

Deploy action provider	No deploy
------------------------	-----------

[Cancel](#)
[Previous](#)
[Create pipeline](#)

following pipeline type differ in characteristics  
Choose one of the following pipeline types:

- **V1** type pipelines have the same structure as standard pipeline, so they support action-level parameters.
- **V2** type pipelines have the same structure as a standard pipeline along with additional parameter support, triggers on Git tags, and pipeline-level variables.

For information about creating a V1 pipeline in CodePipeline, see [Pricing](#).

Create or choose your CodePipeline service role

You only need to create a service role the first time you create a pipeline in CodePipeline. For more information, see [Creating a Pipeline](#).

© 2024 Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy](#) [Terms of Use](#)

Choose one of the following pipeline types:

- **V1** type pipelines have the same structure as standard pipeline, so they support action-level parameters.
- **V2** type pipelines have the same structure as a standard pipeline along with additional parameter support, triggers on Git tags, and pipeline-level variables.

For information about creating a V2 pipeline in CodePipeline, see [Pricing](#).

Create or choose your CodePipeline service role

You only need to create a service role the first time you create a pipeline in CodePipeline. For more information, see [Creating a Pipeline](#).

FOR NOW I DIDN'T SELECT DEPLOY WE WILL LOOK FORWARD JUST CREATE PIPELINES NOW AND INTEGRATE IT WITH OUR CODEBUILD

IN TRIGGER PART SELECT “MAIN” IN INCLUDING AND IN EXCLUDING SELECT FEATURE BRANCH AND AT END CREATE ON CODEPIPELINE

The screenshot shows the AWS CodePipeline console with two separate pipeline executions side-by-side.

**Pipeline 1 (Top):** Status: Success. Pipeline execution ID: [a05f429e-834a-48ce-8120-2eb7e88342b7](#). Stage: Source (In progress). Sub-stage: Didn't Run. Message: No executions yet.

**Pipeline 2 (Bottom):** Status: Success. Pipeline execution ID: [a05f429e-834a-48ce-8120-2eb7e88342b7](#). Stage: Source (Succeeded). Sub-stage: GitHub (Version\_2) - Succeeded 8 minutes ago. Details: Initial commit (commit hash: [0f03dcb](#)). Stage: Build (Failed). Sub-stage: Failed. Details: Pipeline execution ID: [a05f429e-834a-48ce-8120-2eb7e88342b7](#).

SO HERE AGAIN WE ARE FACING ERROIR AGAIN

## IN THESE ERROR IT INDICATE THAT OUR REQUIREMENT.TXT IS MISSING \

The screenshot shows the AWS CodePipeline console. On the left, a sidebar lists navigation options: Developer Tools, CodePipeline, Source, Artifacts, Build, Deploy, Pipeline, History, Settings, and Settings. The main area displays a pipeline named 'CodePipeline'. The first stage, 'Source', is labeled 'Succeeded' with a green circle icon. It shows a GitHub source provider with a commit ID '0f03dcbe' and a message 'Source: Initial commit'. Below this is a 'Disable transition' button. The second stage, 'Build', is labeled 'Failed' with a red circle icon. It shows a pipeline execution ID 'a05f429e-834a-48ce-8120-2eb7e88342b7'. A log panel on the right contains the following text:

```
35
36 [Container] 2024/04/04 17:24:57.662570 Running command pip install -r requirements.txt
37 ERROR: Could not open requirements file: [Errno 2] No such file or directory: 'requirements.txt'
38
39 [notice] A new release of pip is available: 23.3.1 -> 24.0
40 [notice] To update, run: pip install --upgrade pip
41
42 [Container] 2024/04/04 17:25:14.928108 Command did not exit successfully pip install -r requirements.txt exit status 1
43 [Container] 2024/04/04 17:25:14.931483 Phase complete: PRE_BUILD State: FAILED
44 [Container] 2024/04/04 17:25:14.931502 Phase context status code: COMMAND_EXECUTION_ERROR Message: Error while executing command: pip install -r requirements.txt.
Reason: exit status 1
45
```

SO BASICALLY WE HAD FIX THESE ISSUE ALSO BY JUST UPDATING OUR REQUIREMENTS.TXT FILE AND ENETER CONTENT LIKE:

Here's how your `requirements.txt` file should look:

```
makefile
Flask==2.0.2
requests==2.26.0
```

After updating the `requirements.txt` file, commit it to your source code repository and trigger the pipeline again.

NOW AGAIN GO TO CODEPIPELINE AND CHECK THE ERROR IS RESOLVE OR NOT

Pipelines	Info	C	Notify	View history	Release change	Delete pipeline	Create pipeline
<input type="text"/> <span>Q</span>		<	1	>	@		
Name	Latest execution status	Latest source revisions		Latest execution started	Most recent executions		
 simple-python-flask-app (Type: V2   Execution mode: SUPERSEDED)	 Failed	Source – <a href="#">Of03dcbe</a> 	Initial commit	29 minutes ago	 View details		

LET CHECK AGAIN

The screenshot shows a CI pipeline interface with a failed build stage. A red vertical bar on the left indicates the failure. A green arrow points down to a 'Disable transition' button. The main area shows a 'Build' stage failed by AWS CodeBuild, with a 'Failed - 29 minutes ago' status and a 'View details' button. Below the build log, a commit hash '0f03dcbe' is shown with a 'Source: Initial commit' link. The build log itself is as follows:

```
33 [Container] 2024/04/04 17:54:40.772596 Running command echo "Installing dependencies..."  
34 Installing dependencies...  
35  
36 [Container] 2024/04/04 17:54:40.777207 Running command pip install -r requirements.txt  
37 ERROR: Could not open requirements file: [Errno 2] No such file or directory: 'requirements.txt'  
38  
39 [notice] A new release of pip is available: 23.3.1 -> 24.0  
40 [notice] To update, run: pip install --upgrade pip  
41  
42 [Container] 2024/04/04 17:54:57.243415 Command did not exit successfully pip install -r requirements.txt  
43 [Container] 2024/04/04 17:54:57.247021 Phase complete: PRE_BUILD State: FAILED  
44 [Container] 2024/04/04 17:54:57.247041 Phase context status code: COMMAND_EXECUTION_ERROR Message: Error  
Reason: exit status 1  
45
```

At the bottom, there are 'Go to' and 'Feed' buttons.

So here we basically update our requirement.txt but still facing some error so may be I had done wrong code pipelines configuration

**Pipeline settings**

**Pipeline name**  
Enter the pipeline name. You cannot edit the pipeline name after it is created.  
  
No more than 100 characters

**Pipeline type**  
The pipeline type determines the pipeline structure and availability of parameters such as triggers. Pipeline type selection will impact features and pricing. [Which pipeline is right for me?](#)

V1       V2

**Execution mode**  
Choose the execution mode for your pipeline. This determines how the pipeline is run.

Superseded  
A more recent execution can overtake an older one. This is the default.

Queued (Pipeline type V2 required)  
Executions are processed one by one in the order that they are queued.

Parallel (Pipeline type V2 required)  
Executions don't wait for other runs to complete before starting or finishing.

**Service role**

Now we remember that we didn't attach role in last time

**Service role**

New service role  
[Create a service role in your account](#)

Existing service role  
[Choose an existing service role from your account](#)

AWSCodePipelineServiceRole-ap-south-1-simple-python-flask-app  
arn:aws:iam::905418179079:role/service-role/AWSCodePipelineServiceRole-ap-south-1-simple-python-flask-app

**Variables**  
You can add variables at the pipeline level. You can choose to assign the value when you start the pipeline. Choosing

## Source

### Source provider

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

GitHub (Version 2) ▾



#### New GitHub version 2 (app-based) action

To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. [Learn more](#)

### Connection

Choose an existing connection that you have already configured, or create a new one and then return to this task.

arn:aws:codestar-connections:ap-south-1:905418179079:connection/4adb3: X or [Connect to GitHub](#)



#### Ready to connect

Your GitHub connection is ready for use.

arn:aws:codestar-connections:ap-south-1:905418179079:connection/4adb3: X or [Connect to GitHub](#)



#### Ready to connect

Your GitHub connection is ready for use.

### Repository name

Choose a repository in your GitHub account.

Vickypoint/Python-flask-app X

You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

### Default branch

Default branch will be used only when pipeline execution starts from a different source or manually started.

main X

### 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.

Choose the trigger type that starts your pipeline.

**No filter**

Starts your pipeline on any push and clones the HEAD.

**Specify filter**

Starts your pipeline on a specific filter and clones the exact commit. Pipeline type V2 is required.

**Do not detect changes**

Don't automatically trigger the pipeline.

**Event type**

Choose the event type for the trigger that starts your pipeline.

**Push**

**Pull request**

**Filter type**

Choose the filter type for the event that starts your pipeline.

**Branch**

**Tags**

**Branches**

You can specify the target branch or branches you are pushing to. Use a comma to specify multiple entries.

**Include**

main

**Exclude**

Enter branches or patterns

**File paths - optional**

**Include**

main

**Exclude**

Enter file paths or patterns

**File paths - optional**

You can specify the file path or file paths you are pushing to. Use a comma to separate multiple entries.

**Include**

Enter file paths or patterns

**Exclude**

Enter file paths or patterns

**ⓘ You can add additional sources and triggers by editing the pipeline after it is created.**

Cancel

Previous

Next

## Add build stage Info

Step 3 of 5

### Build - optional

#### Build provider

This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.



#### Region



#### Project name

Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.



or

#### Environment variables - optional

Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. [Learn more](#)

#### Project name

Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.



or

#### Environment variables - optional

Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. [Learn more](#)

#### Build type

 Single build

Triggers a single build.

 Batch build

Triggers multiple builds as a single execution.

The screenshot shows a modal dialog box titled "Skip deployment stage". Inside the dialog, a message reads: "Your pipeline will not include a deployment stage. Are you sure you want to skip this stage?". Below the message are two buttons: "Cancel" and "Skip".

Below the dialog, there is a navigation bar with buttons: "Cancel", "Previous", "Skip deploy stage" (which is highlighted in orange), and "Next".

At the bottom of the screen, there is a table showing the latest execution details for a pipeline named "BICKY". The table has columns: Name, Latest execution status, Latest source revisions, Latest execution started, and Most recent executions.

Name	Latest execution status	Latest source revisions	Latest execution started	Most recent executions
BICKY (Type: V2   Execution mode: SUPERSEDED)	<span>Success</span> Succeeded	Source – <a href="#">cc5da17e</a> Update requirements.txt	19 minutes ago	<span>Success</span> View details

So with adding role and update requirements.txt we successfully create a aws codepipeline

SO TILL NOW ACCORDING TO DIAGRAM WHAT WE HAD DONE

WE HAD PUSH OUR CODE FROM VISUAL STUDIO TO GITHUB AND CONNECT GITHUB TO CODE BUILD AND AFTER BUILDING CODE WE HAVE INTEGRATE IT WITH CODEPIPELINES

SO LETS VERIFY OUR CODEPIPELINE BY COMMITTED CHANGE IN GITHUB WHERE UT GET TRIGGER OR NOT

Edit Preview S

```
1  from flask import Flask
2
3  app = Flask(__name__)
4
5  @app.route('/')
6  def hello():
7      return 'Hello, world!'
8
9  if __name__ == '__main__':
10     app.run()
11 |
```

Adding EXTRA LINE

```
1  from flask import Flask
2
3  app = Flask(__name__)
4
5  @app.route('/')
6  def hello():
7      return 'Hello, world!'
8
9  if __name__ == '__main__':
10     app.run()
11 |
12 |
```

## MPW COMMIT CHANGE

Python-flask-app / app.py in main

Cancel changes Commit changes...

Edit Preview Spaces 4 No wrap

```
1 from flask import Flask
2
3 app = Flask(__name__)
4
5 @app.route('/')
6 def hello():
7     return 'Hello, world!'
8
9 if __name__ == '__main__':
10    app.run()
11
12 |
```

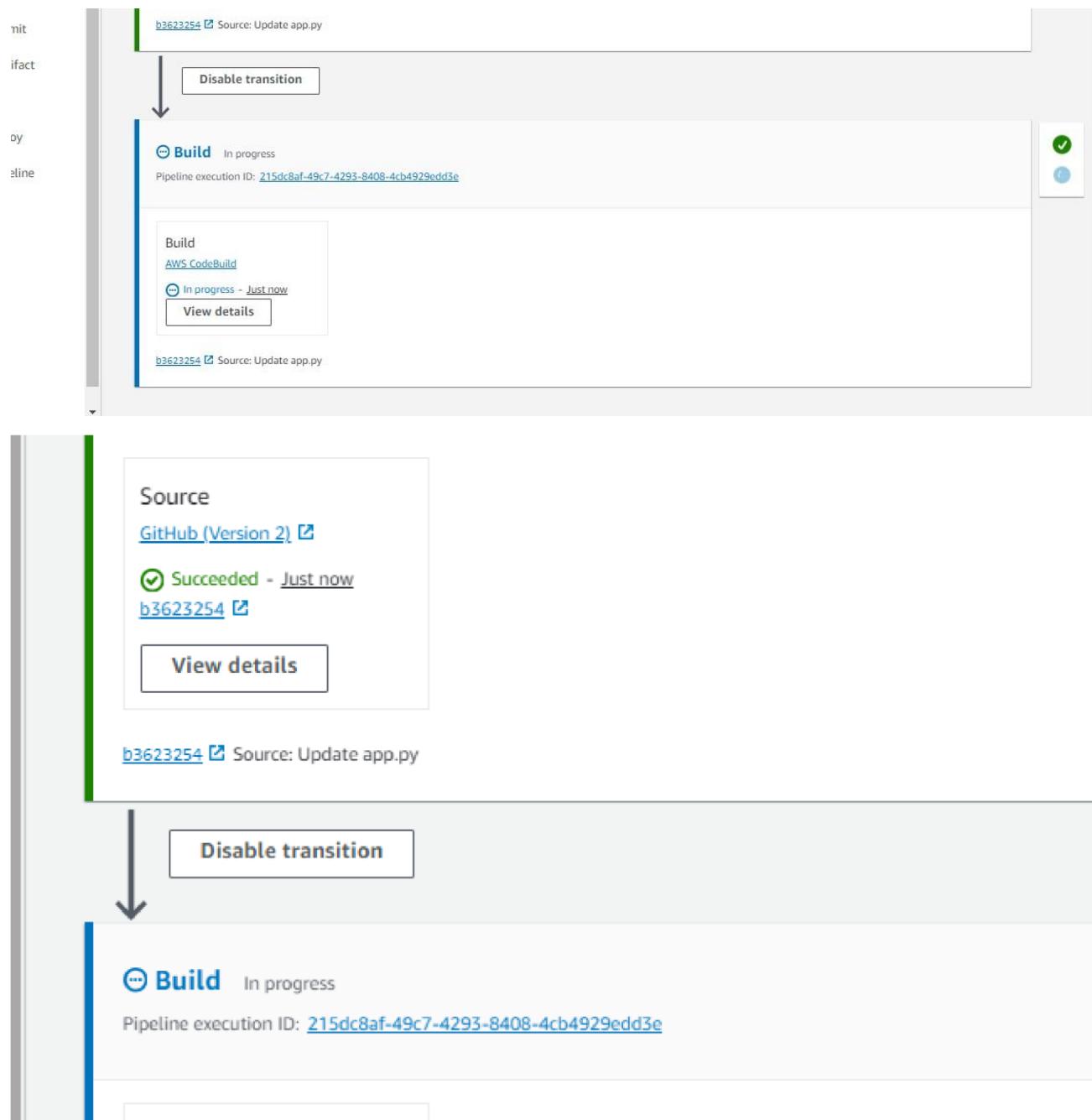
Commit message: Update app.py

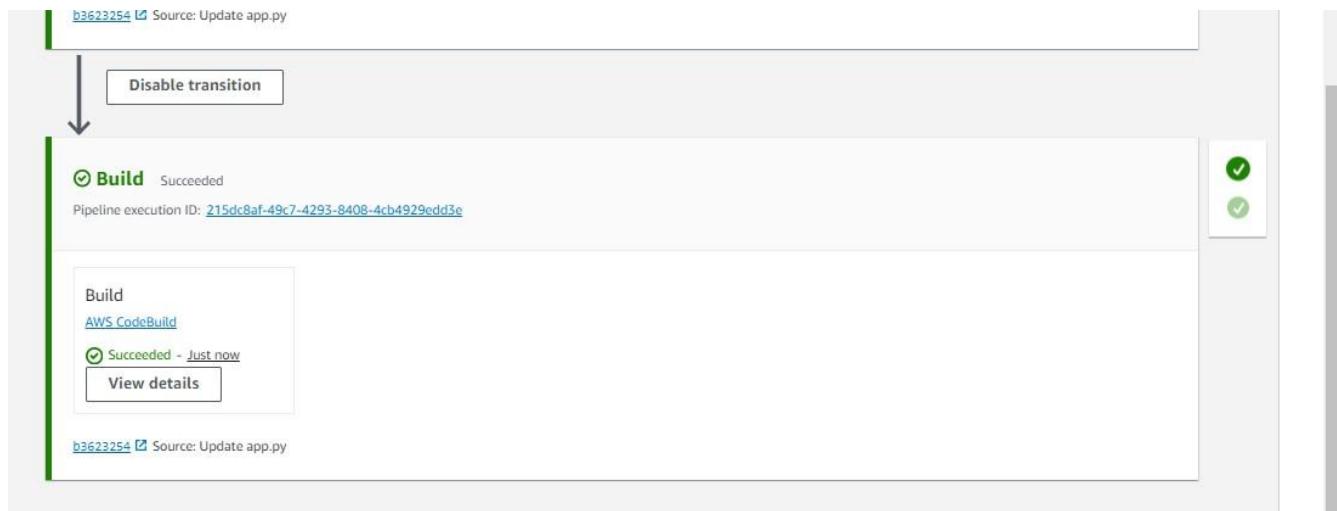
Extended description: Add an optional extended description..

Commit directly to the main branch

Cancel Commit changes

ONCE IT GET UPDATE WE WILL SEE NEW IMAGE IN OUR DOCKERHUB REPOSITORY





LET CHECK DOCKER HUB

The screenshot shows a Docker image page for the repository lionpull123/shwtank-ka-bacha-aagya-hai. The image was last pushed a minute ago by the user lionpull123. The "Tags" tab is selected, showing the "latest" tag. A command box at the top right contains "docker pull lionpull123/shwtank-ka-bacha-aagya-hai:latest" with a "Copy" button. The table below lists the digest (9b749373a343), OS/ARCH (linux/amd64), Last pull (---), and Compressed Size (361.6 MB).

TAG	Digest	OS/ARCH	Last pull	Compressed Size
latest	<a href="#">9b749373a343</a>	linux/amd64	---	361.6 MB

So we get successful update 1 min ago

SO HERE AUTOMATICALLY OUR UPDATE IS TRIGGER

NOW LETS MOVE TO OUR PART 2 WHICH WAS CONTINOUS DEWPLOYMENT (CD)

IN PART 2 OUR GOAL IS TO DEPLOY OUR CODE TO EC2 INSTANT THROUGH AWS CODE DEPLOY SO FIRST CREATE AN APPLICATION CODE DEPLOY

The screenshot shows two separate AWS Management Console sessions.

**Top Session (Search Results):**

- Search bar: CODE DEploy
- Results: Services (136), Features (285), Resources (New), Documentation (544,709), Knowledge Articles (1,181), Marketplace (98), Blogs (24,673), Events (705), Tutorials (143).
- Services List:
  - CodeDeploy: Automate Code Deployments
  - CodeArtifact: Secure, scalable, and cost-effective artifact management for software development
  - CodeBuild: Build and Test Code
- See all 136 results ▶

**Bottom Session (CodeDeploy Application Creation):**

- Developer Tools > CodeDeploy > Applications
- Applications Table:
  - No results: There are no results to display.
- Create application button
- Application configuration form:
  - Application name: PYTHON\_FLASK\_APP (selected)
  - Compute platform: EC2/On-premises
  - Tags: Add tag

Now create an ec2 instant where we want to deploy our code

## 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  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

applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Recents | Quick Start

Amazon Linux | macOS | Ubuntu | Windows | Red Hat | SUSE L  [Browse more AMIs](#)  
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type Free tier eligible  
ami-007020fd9c84e18c7 (64-bit (x86)) / ami-09c443d9277298026 (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: ebs

Number of instances Info

Software Image (AMI)  
Amazon Linux 2023 AMI 2023.4.2...read more  
ami-09298640a92b2d12c

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

 [Free tier eligible](#)

Cancel [Launch instance](#) Review commands

▼ Summary

Number of instances Info

Software Image (AMI)  
Amazon Linux 2023 AMI 2023.4.2...read more  
ami-09298640a92b2d12c

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

 [Free tier eligible](#)

Cancel [Launch instance](#) Review commands

Instance type

t2.micro Free tier eligible  
Family: t2 1 vCPU 1 GiB Memory Current generation: true  
On-Demand Linux base pricing: 0.0124 USD per Hour  
On-Demand Windows base pricing: 0.017 USD per Hour  
On-Demand RHEL base pricing: 0.0724 USD per Hour  
On-Demand SUSE base pricing: 0.0124 USD per Hour

 All generations [Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*   Create new key pair

▼ Network settings 

Number of instances Info

Software Image (AMI)  
Canonical, Ubuntu, 22.04 LTS, ...read more  
ami-007020fd9c84e18c7

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

 [Free tier eligible](#)

Cancel [Launch instance](#)

▼ Summary

Number of instances Info

Software Image (AMI)  
Canonical, Ubuntu, 22.04 LTS, ...read more  
ami-007020fd9c84e18c7

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

 [Free tier eligible](#)

Cancel [Launch instance](#)

**VPC - required** | [Info](#)

vpc-047a6142d07f8704a	(default)	<a href="#">Edit</a>
172.31.0.0/16		<a href="#">Create new subnet</a>

**Subnet** | [Info](#)

subnet-0137d0bbddffcb2e1	sub2	<a href="#">Edit</a>
VPC: vpc-047a6142d07f8704a	Owner: 905418179079	<a href="#">View details</a>
Availability Zone: ap-south-1a	IP addresses available: 250	CIDR: 172.31.1.0/24

**Auto-assign public IP** | [Info](#)

Enable	<a href="#">Edit</a>
--------	----------------------

**Additional charges apply** when outside of **free tier allowance**

**Firewall (security groups)** | [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

[Create security group](#)

[Select existing security group](#)

**Security group name - required**

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and \_-:/()#@[]+=;&{}!\$\*

**Description - required** | [Info](#)

**Inbound Security Group Rules**

**Security group rule 1 (TCP, 22, 0.0.0.0/0)** [Remove](#)

Type   <a href="#">Info</a>	Protocol   <a href="#">Info</a>	Port range   <a href="#">Info</a>
ssh	TCP	22
Source type   <a href="#">Info</a>	Source   <a href="#">Info</a>	Description - optional   <a href="#">Info</a>
Anywhere	<a href="#">Add CIDR, prefix list or security group</a>	e.g. SSH for admin desktop
<input type="text" value="0.0.0.0"/> <a href="#">X</a>		

**⚠️** Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. [X](#)

[Add security group rule](#)

[Advanced network configuration](#)

**Summary**

Number of instances | [Info](#)

1
---

**Software Image (AMI)**  
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)  
ami-007020fd9c84e18c7

**Virtual server type (instance type)**  
t2.micro

**Firewall (security group)**  
New security group

**Storage (volumes)**  
1 volume(s) - 8 GiB

[Free Tier for your instance](#)

[Cancel](#)

**Summary**

Number of instances | [Info](#)

1
---

**Software Image (AMI)**  
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)  
ami-007020fd9c84e18c7

**Virtual server type (instance type)**  
t2.micro

**Firewall (security group)**  
New security group

**Storage (volumes)**  
1 volume(s) - 8 GiB

[Free Tier for your instance](#)

[Cancel](#) [Launch instance](#) [Review commands](#)

**aws** | [Services](#) | [Search](#) [Alt+S]

[EC2](#) [VPC](#) [S3](#) [AWS Auto Scaling](#) [Simple Queue Service](#) [Simple Notification Service](#) [Key Management Service](#) [CloudTrail](#) [Amazon EventBridge](#) [RDS](#) [IAM](#) [Lambda](#) [DynamoDB](#) >

EC2 > Instances > Launch an instance

**Success**  
Successfully initiated launch of instance [i-0ea328cdaf24ded7](#)

[Launch log](#)

**Next Steps**

Now connect with our command prompt

```

C:\Users\sam\Downloads>ssh -i "newkey.pem" ubuntu@3.228.14.181
The authenticity of host '3.228.14.181 (3.228.14.181)' can't be established.
ECDSA key fingerprint is SHA256:Kh8oXZg3sbvf+bWrPDAs/7Vkbm0A/2e+jWxXw/S/Smg.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.228.14.181' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1014-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Apr  5 17:01:01 UTC 2024

 System load:  0.9658203125   Processes:          104
 Usage of /:   20.4% of 7.57GB  Users logged in:    0
 Memory usage: 21%           IPv4 address for eth0: 172.31.11.59
 Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

```

Now WE NEED TO INSTALL DEPLOY AGENT IN EC2 TO USED AWS CODE DEPLOY]

SO HERE IS AWS DOCUMENTATION

Q HOW TO INSTALL CODE DEPLOY AGENT FOR UBUNTU EC2 INSTANT

WS Contact Us English ▾

S > Documentation > Search results

Manager.

AWS Docs > kinesisvideostreams > latest > dg > gs-edge-gg

**Deploy the Amazon Kinesis Video Streams Edge Agent to AWS IoT Greengrass**

Use this topic to deploy the Amazon Kinesis Video Streams Edge Agent to AWS IoT Greengrass to record and upload media from IP cameras.

AWS Docs > systems-manager > latest > userguide > sysman-ins...

**Manually installing and uninstalling SSM Agent on EC2 instances for Windows Server**

Manually install or uninstall SSM Agent on EC2 instances for Windows Server.

AWS Docs > codedeploy > latest > userguide > codedeploy-agent...

**Install the CodeDeploy agent for Ubuntu Server**

We recommend installing the CodeDeploy agent with AWS Systems Manager to be able to configure scheduled updates of the agent. For more information, see Sign in to the instance....

AWS Docs > amazoncloudwatch > latest > monitoring > install-c...

LINK - <https://docs.aws.amazon.com/codedeploy/latest/userguide/codedeploy-agent-operations-install-ubuntu.html>

FOLLOW ALL RTHE BASIC COMMAND

# Install the CodeDeploy agent for Ubuntu Server

[PDF](#) | [RSS](#)

## Note

We recommend installing the CodeDeploy agent with AWS Systems Manager to be able to configure scheduled updates of the agent. For more information, see [Install the CodeDeploy agent using AWS Systems Manager](#).

### To install the CodeDeploy agent on Ubuntu Server

1. Sign in to the instance.
2. Enter the following commands, one after the other:

```
sudo apt update
```



```
sudo apt install ruby-full
```



```
sudo apt install wget
```



3. Enter the following command:

```
ubuntu@ip-172-31-11-59:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1519 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [293 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1648 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [275 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1060 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [241 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [22.1 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [49.6 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [12.0 kB]
```

```

Reading state information... done
41 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-11-59:~$ sudo apt install ruby-full
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
fonts-lato javascript-common libgmp-dev libgmpxx4ldbl libjs-jquery libruby3.0 rake ri ruby ruby-dev ruby-net-
ruby-rubygems ruby-webrick ruby-xmlrpc ruby3.0 ruby3.0-dev ruby3.0-doc rubygems-integration unzip zip
Suggested packages:
apache2 | lighttpd | httpd gmp-doc libgmp10-doc libmpfr-dev bundler
The following NEW packages will be installed:
fonts-lato javascript-common libgmp-dev libgmpxx4ldbl libjs-jquery libruby3.0 rake ri ruby ruby-dev ruby-full
ruby-net-telnet ruby-rubygems ruby-webrick ruby-xmlrpc ruby3.0 ruby3.0-dev ruby3.0-doc rubygems-integration u
zip
0 upgraded, 21 newly installed, 0 to remove and 41 not upgraded.
Need to get 11.4 MB of archives.
After this operation, 59.7 MB of additional disk space will be used.

ubuntu@ip-172-31-11-59:~$ sudo apt install wget
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
wget is already the newest version (1.21.2-2ubuntu1).
wget set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 41 not upgraded.
ubuntu@ip-172-31-11-59:~$
```

SO BOW WE NEED TO FOCUS BECAUSE WE NEED TO ADD BUCKET NAME AND ITS SPECIFIC REGION FROM DOCUMENT ITSELF

4. Enter the following command:

```
wget https://bucket-name.s3.region-identifier.amazonaws.com/latest/install
```

*bucket-name* is the name of the Amazon S3 bucket that contains the CodeDeploy Resource Kit files for your region, and *region-identifier* is the region where you want to deploy the application. For example:

```
https://aws-codedeploy-us-east-2.s3.us-east-2.amazonaws.com/latest/install
```

For a list of bucket names and region identifiers, see [Resource kit bucket names by Region](#).

5. Enter the following command:

FOR REGION NAME amd bucket NAME CLICK ON RESOURCES KIT

## Resource kit bucket names by Region

This table lists the names of *bucket-name* replacements required for some procedures in the guide. These are the names of the Amazon S3 buckets that contain the CodeDeploy Resource Kit files.

 Note

To access the Amazon S3 bucket in the Asia Pacific (Hong Kong) Region, you must enable the region in your AWS account. For more information, see [Managing AWS Regions](#).

Region name	<i>Bucket-name</i> replacement	Region identifier
US East (N. Virginia)	aws-codedeploy-us-east-1	us-east-1
US East (Ohio)	aws-codedeploy-us-east-2	us-east-2
US West (N. California)	aws-codedeploy-us-west-1	us-west-1
US West (Oregon)	aws-codedeploy-us-west-2	us-west-2
Africa (Cape Town)	aws-codedeploy-af-south-1	af-south-1
Asia Pacific (Hong Kong)	aws-codedeploy-ap-east-1	ap-east-1
Asia Pacific (Hyderabad)	aws-codedeploy-ap-south-2	ap-south-2
Asia Pacific (Jakarta)	aws-codedeploy-ap-southeast-3	ap-southeast-3
Asia Pacific (Melbourne)	aws-codedeploy-ap-southeast-4	ap-southeast-4

\*Untitled - Notepad

File Edit Format View Help

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

```
root@ip-172-31-11-59:~$ wget set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 41 not upgraded.
ubuntu@ip-172-31-11-59:~$ cd /home/ubuntu
ubuntu@ip-172-31-11-59:~$ wget https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install
```

```
ubuntu@ip-172-31-11-59:~$ cd /home/ubuntu
ubuntu@ip-172-31-11-59:~$ wget https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install
--2024-04-05 17:10:27-- https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install
Resolving aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com (aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com)... 4.231.170.58, 54.231.199.26, 54.231.226.226, ...
Connecting to aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com (aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com) 54.231.170.58|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 19045 (19K) []
Saving to: 'install'

install                                         100%[=====] 18.60K  --.KB/s   in 0.001s

2024-04-05 17:10:27 (14.8 MB/s) - 'install' saved [19045/19045]
ubuntu@ip-172-31-11-59:~$
```

<https://aws-codedeploy-us-east-2.s3.us-east-2.amazonaws.com/latest/install>  
For a list of bucket names and region identifiers, see [Resource kit bucket names by Region](#).

5. Enter the following command:

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

6. Do one of the following:

- To install the latest version of the CodeDeploy agent on any supported version of Ubuntu Server except 20.04:

```
sudo ./install auto
```

- To install the latest version of the CodeDeploy agent on Ubuntu Server 20.04:

```
ubuntu@ip-172-31-11-59:~$ chmod +x ./install
ubuntu@ip-172-31-11-59:~$ sudo ./install auto
I, [2024-04-05T17:11:49.144720 #2101]  INFO -- : Starting Ruby version check.
I, [2024-04-05T17:11:49.145061 #2101]  INFO -- : Starting update check.
I, [2024-04-05T17:11:49.145278 #2101]  INFO -- : Attempting to automatically detect supported package manager...
W, [2024-04-05T17:11:49.155640 #2101]  WARN -- : apt-get found but no gdebi. Installing gdebi with `apt-get -y ...
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  gdebi-core
0 upgraded, 1 newly installed, 0 to remove and 41 not upgraded.
Need to get 133 kB of archives.
After this operation, 876 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 gdebi-core all 0.9.5.7+nmu6 [133 kB]
Fetched 133 kB in 0s (6260 kB/s)
Selecting previously unselected package gdebi-core.
(Reading database ... 81692 files and directories currently installed.)
Preparing to unpack .../gdebi-core_0.9.5.7+nmu6_all.deb ...
Unpacking gdebi-core (0.9.5.7+nmu6) ...
Setting up gdebi-core (0.9.5.7+nmu6) ...
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
```

#### To check that the service is running

1. Enter the following command:

```
systemctl status codedeploy-agent
```

If the CodeDeploy agent is installed and running, you should see a message like The AWS CodeDeploy agent is running.

```
ubuntu@ip-172-31-11-59:~$ systemctl status codedeploy-agent
● codedeploy-agent.service - LSB: AWS CodeDeploy Host Agent
  Loaded: loaded (/etc/init.d/codedeploy-agent; generated)
  Active: active (running) since Fri 2024-04-05 17:11:58 UTC; 37s ago
    Docs: man:systemd-sysv-generator(8)
  Process: 2385 ExecStart=/etc/init.d/codedeploy-agent start (code=exited, status=0/SUCCESS)
    Tasks: 2 (limit: 1121)
   Memory: 57.5M
      CPU: 1.102s
     CGroup: /system.slice/codedeploy-agent.service
             └─2405 "codedeploy-agent: master" 2405" ...
                  ├─2407 "codedeploy-agent: InstanceAgent::Plugins::CodeDeployPlugin::CommandPoller of master" 2407" ...
Apr 05 17:11:58 ip-172-31-11-59 systemd[1]: Starting LSB: AWS CodeDeploy Host Agent...
Apr 05 17:11:58 ip-172-31-11-59 codedeploy-agent[2385]: Starting codedeploy-agent:
Apr 05 17:11:58 ip-172-31-11-59 systemd[1]: Started LSB: AWS CodeDeploy Host Agent.
lines 1-15/15 (END)
```

## NOW ASSIGN ROLE TO EC2 TO ACCESS CODE DEPLOY

Screenshot of the AWS IAM Roles page showing the creation of a new role named "DEPLOY ROLE". The search bar shows "DEPLOY ROLE" and the results table is empty.

**Step 2: Add permissions**

**Trusted entity type**

- AWS service: Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- AWS account: Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- Web identity: Allows users federated by external web identity provider to assume this role to perform actions in this account.
- SAML 2.0 federation: Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- Custom trust policy: Create a custom trust policy to enable others to perform actions in this account.

**Use case**

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

**Service or use case**

EC2

Choose a use case for the specified service.

Use case

- EC2: Allows EC2 instances to call AWS services on your behalf.
- EC2 Role for AWS Systems Manager

**Permissions policies (1/948) Info**

Choose one or more policies to attach to your new role.

Filter by Type: All types, 11 matches

Policy name	Type	Description
<input type="checkbox"/> <a href="#">AmazonEC2RoleforAWSCodeDeploy</a>	AWS managed	Provides EC2 access to S3 bucket to do...
<input type="checkbox"/> <a href="#">AmazonEC2RoleforAWSCodeDeployLimited</a>	AWS managed	Provides EC2 limited access to S3 buck...
<input type="checkbox"/> <a href="#">AWSCodeDeployDeployerAccess</a>	AWS managed	Provides access to register and deploy ...
<input checked="" type="checkbox"/> <a href="#">AWSCodeDeployFullAccess</a>	AWS managed	Provides full access to CodeDeploy res...
<input type="checkbox"/> <a href="#">AWSCodeDeployReadOnlyAccess</a>	AWS managed	Provides read only access to CodeDepl...
<input type="checkbox"/> <a href="#">AWSCodeDeployRole</a>	AWS managed	Provides CodeDeploy service access to ...
<input type="checkbox"/> <a href="#">AWSCodeDeployRoleForCloudFormation</a>	AWS managed	Provides CodeDeploy service access to ...

**Role details**

**Role name**  
Enter a meaningful name to identify this role.  
**code-deploy-role**  
Maximum 64 characters. Use alphanumeric and '+=\_@-' characters.

**Description**  
Add a short explanation for this role.  
**Allows EC2 instances to call AWS services on your behalf.**  
Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: \_+=,. @-^[]#\$%^&\*();<>

**Step 1: Select trusted entities**

**Trust policy**

```

1+ [
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "sts:AssumeRole"
8       ],
9       "Principal": {
10         "Service": [
11           "ec2.amazonaws.com"
12         ]
13       }
14     }
15   ]
16 ]

```

**Identity and Access Management (IAM)**

**Role code-deploy-role created.**

**Roles (3) Info**  
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name	Trusted entities	Last activity
AWSCodePipelineServiceRole-ap-south-1-simple-python-flask-app	AWS Service: codepipeline	Yesterday
AWSServiceRoleForAmazonEKS	AWS Service: eks (Service-Linked Role)	4 days ago
AWSLambdaRoleForAmazonEventBridge	AWS Lambda for Amazon EventBridge	4 days ago

**EC2 Instances (1/1) Info**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
project	i-08a20e1e3c67ef7b5	Running	t2.micro	Initializing	View alarms +	us-east-1b

**Actions** ▾ **Launch instances** ▾

- Connect
- View details
- Manage instance state
- Instance settings
- Networking
- Security**
- Image and templates
- Monitor and troubleshoot

The screenshot shows the AWS CloudWatch Instances console. At the top, there's a navigation bar with various AWS services: AWS Auto Scaling, Simple Queue Service, Simple Notification Service, Key Management Service, CloudTrail, Amazon EventBridge, RDS, IAM, Lambda, DynamoDB, CloudFront, and Route 53. Below the navigation bar is a search bar with placeholder text "Find Instance by attribute or tag (case-sensitive)" and a dropdown menu set to "All states". A button labeled "Clear filters" is also present. The main area displays a table of instances. The first row, which is selected, has a checked checkbox and the name "project". The table columns include Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. The instance "project" is listed as "Running" with the instance ID "i-08a20e1e3c67ef7b5". The "Status check" column shows a green circle with "2/2 checks passed". The "Actions" menu for this instance includes options like Connect, View details, Manage instance state, Instance settings, Networking, Security (which is highlighted with a blue border), Image and templates, and Monitor and troubleshoot.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
project	i-08a20e1e3c67ef7b5	Running	t2.micro	2/2 checks passed	View alarms	us-east-1b

Actions ▾

- Connect
- View details
- Manage instance state
- Instance settings
- Networking
- Security
- Image and templates
- Monitor and troubleshoot

Launch instances ▾

NOW RESTART THE AGENT\

Cmd-

```
sudo service codedeploy-agent restart
```

```
ubuntu@ip-172-31-11-59:~$ sudo service codedeploy-agent restart
ubuntu@ip-172-31-11-59:~$ systemctl status codedeploy-agent
● codedeploy-agent.service - LSB: AWS CodeDeploy Host Agent
```

CONFIGURE OF EC2 IS DONE NOW GO FOR CODEDEPLOY SIDE

## CREATE AN DEPLOYMENT GROUP TO ATTACH EC2

Application details

Name PYTHON-FLASK-APP	Compute platform EC2/On-premises
--------------------------	-------------------------------------

Deployments Deployment groups Revisions

Deployment groups

Name	Status	Last attempted deployment	Last successful deployment	Trigger count
No deployment groups				

View details Edit Create deployment group < 1 > ⌂

No deployment groups

Before you can deploy your application using CodeDeploy, you must create a deployment group.

Deployment group name

Enter a deployment group name

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.

Now either create a role with new policy and qattach ec2 full accces or use same role just add new policy of ec2 full access

1 hour

Permissions Trust relationships Tags Access Advisor Revoke sessions

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

Filter by Type  
Q Search All types

Add permissions ▲ Attach policies Create inline policy

< 1 > ⌂

Policy name	Type	Attached entities
AWSCodeDeployFullAccess	AWS managed	2

Filter by Type  
Q ec2 All types 31 matches

Policy name

Policy name	Type
AmazonEC2ContainerRegistryFullAccess	AWS managed
AmazonEC2ContainerRegistryPowerUser	AWS managed
AmazonEC2ContainerRegistryReadOnly	AWS managed
AmazonEC2ContainerServiceAutoscaleRole	AWS managed
AmazonEC2ContainerServiceEventsRole	AWS managed
AmazonEC2ContainerServiceforEC2Role	AWS managed
AmazonEC2ContainerServiceRole	AWS managed
AmazonEC2FullAccess	AWS managed
AmazonEC2ReadOnlyAccess	AWS managed

Policy was successfully attached to role.

You can attach up to 10 managed policies.

Filter by Type  
Q Search All types

Policy name ▲ Type Attached entities

Policy name	Type	Attached entities
AmazonEC2FullAccess	AWS managed	3
AWSCodeDeployFullAccess	AWS managed	2

▶ Permissions boundary (not set)

▼ Generate policy based on CloudTrail events

Attach these role now

## Deployment group name

Enter a deployment group name

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.



## 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

0 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.

## 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.

[Enable load balancing](#)

### ► Advanced - optional

[AWS CodeDeploy does not have the permissions required to assume the role](#)  
`arn:aws:iam::905418179079:role/code-deploy-role.`

[Cancel](#)

[Create deployment group](#)

We need to create new role and add code deplo as service

	others to perform actions in this account.	Belonging to you or a 3rd party to perform actions in this account.	External web identity provider to assume this role to perform actions in this account.
--	--	---	--

**SAML 2.0 federation**  
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

**Custom trust policy**  
Create a custom trust policy to enable others to perform actions in this account.

### Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

#### Service or use case

CodeDeploy

Choose a use case for the specified service.

#### Use case

**CodeDeploy**

Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.

**CodeDeploy for Lambda**

Allows CodeDeploy to route traffic to a new version of an AWS Lambda function version on your behalf.

**CodeDeploy - ECS**

Allows CodeDeploy to read S3 objects, invoke Lambda functions, publish to SNS topics, and update ECS services on your behalf.

### Role details

I create

#### Role name

Enter a meaningful name to identify this role.

new

Maximum 64 characters. Use alphanumeric and '+,-,\_.' characters.

#### Description

Add a short explanation for this role.

Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.

Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: \_+=,. @-/\{()\}#\$%^&\*";;"<>

### Step 1: Select trusted entities

Edi

#### Trust policy

```

1- {
2-   "Version": "2012-10-17",
3-   "Statement": [
4-     {
5-       "Sid": "",
6-       "Effect": "Allow",
7-       "Principal": {
8-         "Service": [
9-           "codedeploy.amazonaws.com"
10      ],
11    },
12    "Action": [

```

## Service role

Enter a service role

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



## 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.

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

[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](#)

The screenshot shows a success message: "Deployment group created". The navigation path is: Developer Tools > CodeDeploy > Applications > PYTHON-FLASK-APP > deploy-group. There are buttons for Edit, Delete, and Create deployment.

Deployment group name	Application name	Compute platform
deploy-group	PYTHON-FLASK-APP	EC2/On-premises

Deployment type	Service role ARN	Deployment configuration
In-place	arn:aws:iam::905418179079:role/new	CodeDeployDefault.AllAtOnce

Rollback enabled	Agent update scheduler

Since TILL NOW WE HAD CREATE EC2 INSTANCE WITH DEPLOY AGENT INSIDE IT AND ASSIGN ROLE TO ACCESS CODE DEPLOY AND ON OTHER HAND CREATED CODE DEPLOY AND CODE DEPLOYMENT GROUP ASSIGN ROLE TO ACCESS EC2 INSTANCE NOW WE WILL DEPLOY OUR APPLICATION BUT BEFORE THAT WE NEED TO BUILD SPEC FILE MEAN APP SPEC FILE SIMILAR LIKE BUILD SPEC FILE SO LET DO SOME CPONFIGURATION FOR THESE \

The screenshot shows the application details for "PYTHON-FLASK-APP". It has tabs for Deployments, Deployment groups (which is selected), and Revisions. Under Deployment groups, there is a table with one entry: "deploy-group".

Name	Status	Last attempted deployment	Last successful deployment	Trigger count
deploy-group	-	-	-	0

## GO TO DEPLOYMENT

The screenshot shows the AWS CodeDeploy 'Create deployment' interface. At the top, there's a navigation bar with links to EC2, VPC, S3, AWS Auto Scaling, Simple Queue Service, Simple Notification Service, Key Management Service, CloudTrail, and Amazon EventBridge. Below the navigation, a breadcrumb trail indicates the path: Developer Tools > CodeDeploy > Applications > PYTHON-FLASK-APP > Create deployment. The main title is 'Create deployment'. A 'Deployment settings' section contains fields for Application (PYTHON-FLASK-APP), Deployment group (search bar containing 'deploy-group'), Compute platform (EC2/On-premises), and Deployment type (In-place). A note about Managed hook execution role is present. Below this, the 'In-place' section is expanded, showing options for Revision type (radio buttons for 'My application is stored in Amazon S3' and 'My application is stored in GitHub', with the latter being selected), and a GitHub token name input field containing 'alias'. A note says 'Type alias for first time when you connect to your github account'.

Developer Tools > CodeDeploy > Applications > PYTHON-FLASK-APP > Create deployment

### Create deployment

#### Deployment settings

Application  
PYTHON-FLASK-APP

Deployment group  
 X

Compute platform  
EC2/On-premises

Deployment type  
In-place

Managed hook execution role  
The IAM role used by the CodeDeploy Managed Hook function to perform actions. [Edit Managed Hook execution role](#).

In-place

Managed hook execution role  
The IAM role used by the CodeDeploy Managed Hook function to perform actions. [Edit Managed Hook execution role](#).

Revision type

My application is stored in Amazon S3

My application is stored in GitHub

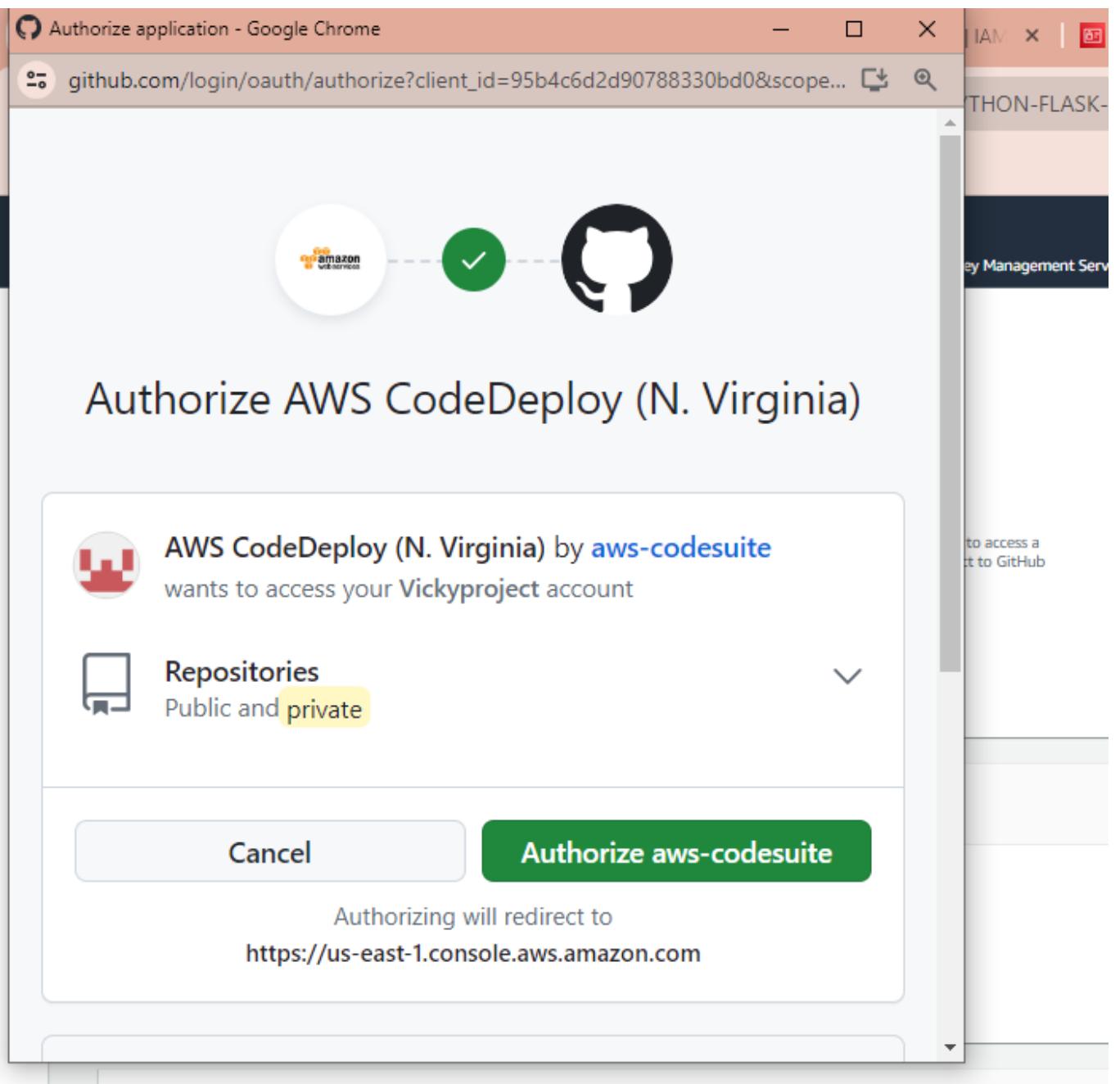
GitHub token name  
Select the name of the token associated to an account you have already connected, or grant AWS CodeDeploy permission to access a different account. To connect to a GitHub account for the first time, type an alias for the account, and then choose Connect to GitHub

X

Use entered value: alias

[Connect to GitHub](#)

Type alias for first time when you connect to your github account



Processing OAuth request

Choose Confirm to connect CodeDeploy to your GitHub account

[Cancel](#) [Confirm](#)

**GitHub token name**  
Select the name of the token associated to an account you have already connected, or grant AWS CodeDeploy permission to access a different account. To connect to a GitHub account for the first time, type an alias for the account, and then choose Connect to GitHub.

alias [X](#)

[Connected](#)

**✓ Application PYTHON-FLASK-APP successfully bound to alias GitHub token** [X](#)

**Repository name**

**Commit ID**

## For repository name

Repository name  
Vickyproject/Python-flask-app

Commit ID

### Deployment description

Deployment description - optional  
Add a brief description about the deployment

### Additional deployment behavior settings

ApplicationStop lifecycle event failure - optional  
Type a deployment group name  
 Don't fail the deployment to an instance if this lifecycle event on the instance fails

Content options - optional

CloudShell   Feedback

github.com/vickyproject/Python-flask-app

Gmail YouTube All Bookmarks

Vickyproject / Python-flask-app Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Python-flask-app Public Unpin Unwatch 1 Fork 0 Star 0

main 1 Branch 0 Tags Go to file + <> Code About

Vickyproject Update app.py b362325 · yesterday 9 Commits

Dockerfile Update and rename docker file to Dock... 2 days ago

app.py Update app.py yesterday

buildspec.yml buildspec.yml last week

requirements.txt Update requirements.txt yesterday

No description, website, or topics provided.

Activity 0 stars 1 watching 0 forks

Releases

For commit id go to github and select recent commit and copy the id

The screenshot shows a GitHub repository named "Python-flask-app" which is public. The main branch is "main". There is 1 branch and 0 tags. A search bar allows going to a file. A green "Code" button is visible. Below the header, a list of recent commits is shown:

Author	Commit Message	Date
Vickyproject	Update app.py	b362325 · yesterday
	Dockerfile	Update and rename docker file to Docke... 2 days ago
	app.py	Update app.py yesterday
	buildspec.yml	buildspec.yml last week

Below the commits, there are navigation links for Code, Issues, Pull Requests, Actions, Projects, Wiki, Security, Insights, and Settings.

## Commits

The screenshot shows the commit history for the "main" branch. It includes filters for All users and All time. The commits are listed as follows:

- o- Commits on Apr 5, 2024
  - Update app.py**  
Vickyproject committed yesterday Verified b362325 ✓ ↗
- o- Commits on Apr 4, 2024
  - Update requirements.txt**  
Vickyproject committed yesterday Verified cc5da17 ↗ ↗

Repository name

Vickyproject/Python-flask-app

Commit ID

b3623254210d1fd3ab8bd82611764381237e8c8

## Deployment description

Deployment description - optional

Add a brief description about the deployment

## Additional deployment behavior settings

ApplicationStop lifecycle event failure - *optional*

Repository name

Vickyproject/Python-flask-app

Commit ID

b3623254210d1fd3ab8bd82611764381237e8c8

## Deployment description

Deployment description - optional

Add a brief description about the deployment

## Additional deployment behavior settings

ApplicationStop lifecycle event failure - *optional*

Type a deployment group name

Don't fail the deployment to an instance if this lifecycle event on the instance fails

Content options - *optional*

Repository name  
Vickyproject/Python-flask-app

Commit ID  
b3623254210d1fd3ab8bdf82611764381237e8c8

**Deployment description**

Deployment description - optional  
Add a brief description about the deployment

**Additional deployment behavior settings**

ApplicationStop lifecycle event failure - *optional*  
Type a deployment group name  
 Don't fail the deployment to an instance if this lifecycle event on the instance fails

Content options - *optional*  
Choose what to do during a deployment when a file on a target instance has the same name as a file in the application revision

Fail the deployment  
An error is reported and the deployment status is changed to Failed.

Overwrite the content  
The file in the application revision is copied to the target location on the instance, replacing the previous file.

Retain the content  
The file in the application revision is not copied to the instance. The existing file is kept at the target location and treated as part of the new deployment.

▶ Deployment group overrides

▶ Rollback configuration overrides

[Cancel](#) [Create deployment](#)

Go with default and create on create deployment

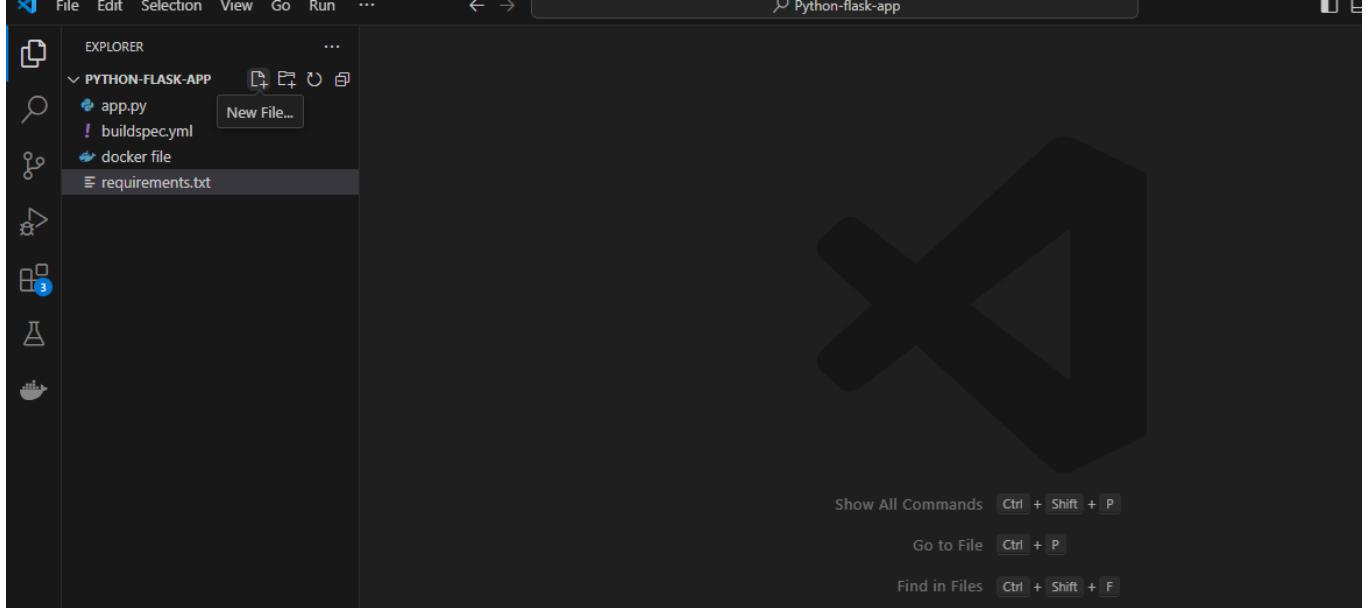
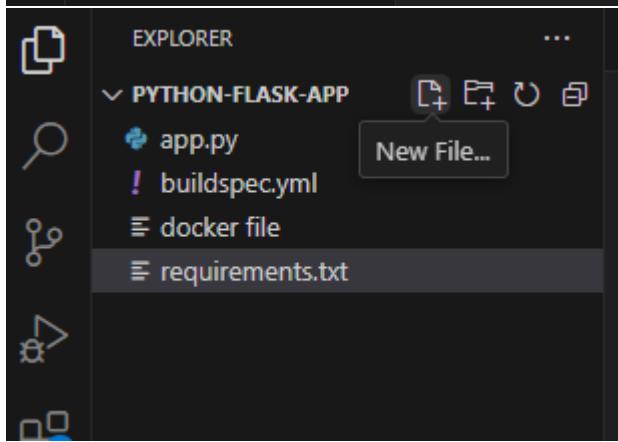
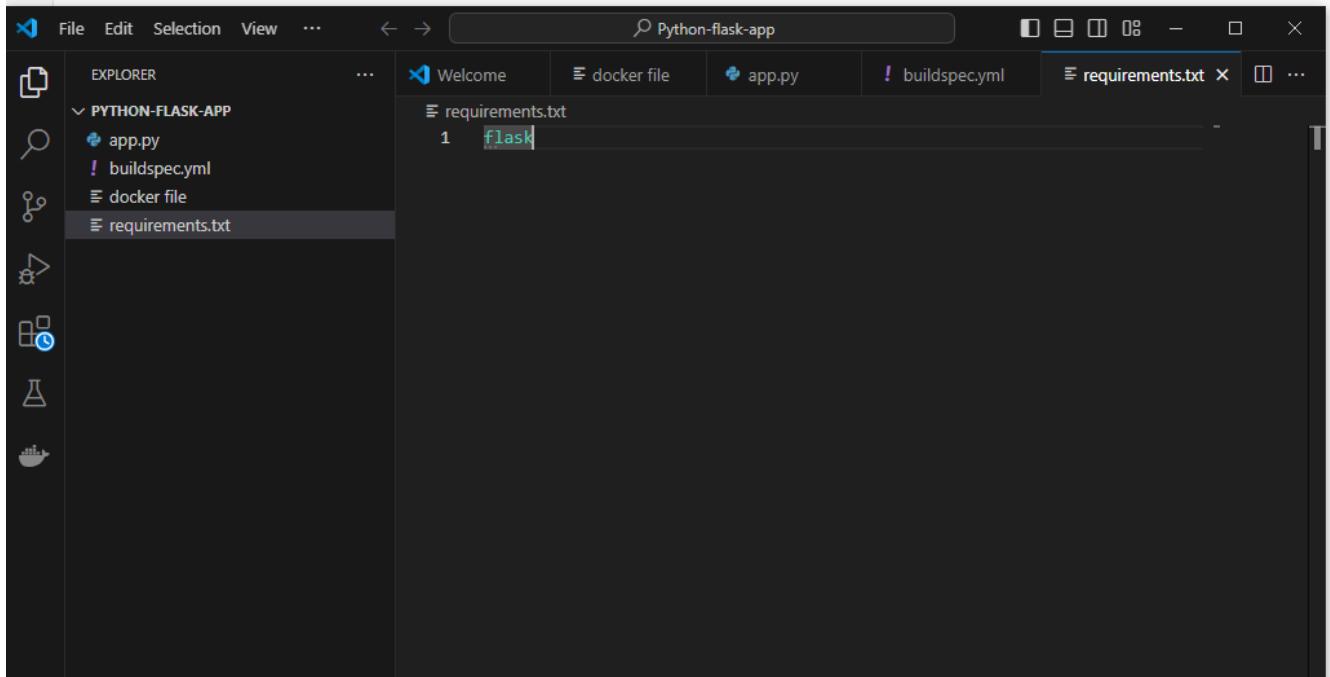
The screenshot shows the AWS CodeDeploy console. At the top, there's a navigation bar with various services like EC2, VPC, S3, AWS Auto Scaling, Simple Queue Service, Simple Notification Service, Key Management Service, CloudTrail, Amazon EventBridge, RDS, IAM, Lambda, DynamoDB, and CloudFront. A search bar and a [Alt+S] keybinding are also present. On the right, it shows "N. Virginia" and a user "Vicky Omprakash S". Below the navigation bar, a sidebar on the left lists "Developer Tools", "Source • CodeCommit", "Artifacts • CodeArtifact", "Build • CodeBuild", "Deploy • CodeDeploy", "Getting started", "Deployments", "Deployment", "Applications", "Deployment configurations", "On-premises instances", "Pipeline • CodePipeline", and "Settings". A "Go to resource" link is at the bottom of the sidebar. The main content area has a green header bar with a success icon and the message "Deployment created". It shows the deployment path: Developer Tools > CodeDeploy > Deployments > d-WWZ7RY1F5. Below this, there's a "d-WWZ7RY1F5" section with "Deployment details" and "Revision details" tabs. Under "Deployment details", it shows Application: PYTHON-FLASK-APP, Deployment ID: d-WWZ7RY1F5, Status: Created, Deployment configuration: CodeDeployDefault.AllAtOnce, Deployment group: deploy-group, and Deployment description: -. Under "Revision details", there are columns for Revision location, Revision created, and Revision description. At the bottom, a large "Deployment status" box shows the progress of "Installing application on your instances": 0 of 1 instances updated, 0%, and In progress. A circular progress bar is also shown.

It will take some time

This screenshot shows the same AWS CodeDeploy interface as the previous one, but the deployment has failed. The "Deployment status" box now shows 0 of 1 instances updated, 1%, Failed, and Failed. The "Deployment details" section shows the same information as before, but the status is now Failed. The deployment ID is d-WWZ7RY1F5. The deployment configuration is CodeDeployDefault.AllAtOnce, and the deployment group is deploy-group. The deployment description is -. The "Revision details" section is partially visible at the bottom.

Basically it is fail so I intentionally show failure to show all of you the importance of appspec.yml file we didn't create app.spec file yet we need to create in visual studio and push it to the main repository let start

Open your github desktop version and clone your repository and open in visual studio



The screenshot shows the VS Code interface with the Python-Flask-App project open. The Explorer sidebar on the left lists files: app.py, ! appspec.yml (which is selected), ! buildspec.yml, docker file, and requirements.txt. The bottom half of the screen shows the contents of the appspec.yml file in the Editor tab.

```
! appspec.yml U ●
! appspec.yml
1 version: 0.0
2 os: linux
3
4 hooks:
5   ApplicationStop:
6     - location: scripts/stop_container.sh
7       timeout: 300
8       runas: root
9   AfterInstall:
10    - location: scripts/start_container.sh
11      timeout: 300
12      runas: root
```

In these appspsec.yml we have basically use script which contain two thing i.e container start and container stop  
So we need to create new folderr name as scripts  
so inside scripts create two new file which was start conteiner and stop container

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows a project named "PYTHON-FLASK-APP" containing files: app.py, appspec.yml, buildspec.yml, docker file, and requirements.txt.
- Code Editor:** Displays the contents of the appspec.yml file. The file specifies a version of 0.0, an OS of linux, and hooks for ApplicationStop and AfterInstall. Each hook contains a script located at scripts/stop\_container.sh or scripts/start\_container.sh with a timeout of 300 and runas root.

```
version: 0.0
os: linux

hooks:
  ApplicationStop:
    - location: scripts/stop_container.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/start_container.sh
      timeout: 300
      runas: root
```

so inside scripts create two new file which was start container and stop container

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the "scripts" directory containing two new files: startcontainer.sh and stop\_container.sh.
- Code Editor:** Shows the contents of the stop\_container.sh file, which is currently empty (containing only a line 1).

now copy and paste stop and start container.sh command

The screenshot shows the VS Code interface for a Python Flask application. The Explorer sidebar lists the project structure:

- PYTHON-FLASK-APP
  - scripts
    - \$ startconatiner.sh
    - \$ stop\_container.sh
  - app.py
  - appspec.yml
  - buildspec.yml
  - docker file
  - requirements.txt

The Editor tab displays three files:

- ! appspec.yml
- \$ stop\_container.sh
- \$ startconatiner.sh

The content of the startconatiner.sh file is:

```
#!/bin/bash
set -e
# Stop the running container (if any)
echo "Hi"
```

The content of the stop\_container.sh file is:

```
#!/bin/bash
set -e
```

The content of the appspec.yml file is:

```
version: 0.0.1
phases:
  build:
    commands:
      - echo "Building Docker image"
  deploy:
    commands:
      - echo "Deploying Docker image"
```

So in STARTCONATINER.SH

WE NEED TO CONFIGURE SOME THING LIKE OUR LASTEST DOCKER IMAGE PULL HERE

The screenshot shows the Docker Hub interface. At the top, there's a blue header bar with the Docker Hub logo, 'Explore', 'Repositories' (which is underlined), 'Organizations', and a search bar that says 'Search Docker Hub'. Below the header, there's a dropdown menu with 'lionpull123' selected, a search bar with 'Search by repository name', and another dropdown for 'All Content'. A blue button on the right says 'Create repository'. The main content area shows two repository cards: 'lionpull123 / shwetank-ka-bacha-aagya-hai' (last pushed 1 day ago) and 'lionpull123 / bicky' (Created 3 days ago). Both cards show 'Security unknown', 0 stars, 5 forks, and are public.

This screenshot shows the detailed view of the 'lionpull123/shwetank-ka-bacha-aagya-hai' repository. The header is identical to the previous screenshot. The main content includes the repository name, a small image thumbnail, and a summary: 'By lionpull123 · Updated 1 day ago'. Below this is a 'Image' section with a 'No overview available' message. To the right, there's a 'Docker Pull Command' box containing the command 'docker pull lionpull123/shwetank-ka-bacha-aagya-hai' with a 'Copy' button. Navigation tabs at the bottom left are 'Overview' (underlined) and 'Tags'.

COP THE COMMAND AND PULL HERE IN STArtcontainer.sh

A terminal window titled 'Python basic app' is shown. It has three tabs: 'appspec.yml', '\$ stop\_container.sh', and '\$ startconatiner.sh'. The '\$ startconatiner.sh' tab is active and displays the following script content:

```
#!/bin/bash
set -e
# Pull the Docker image from Docker Hub
docker pull lionpull123/shwetank-ka-bacha-aagya-hai
# Run the Docker image as a container
```

Now in run docker image as container go to dockerfile and see the port number and enter here

Cmd – docker run –d –p 5000:5000

The screenshot shows the VS Code interface with the title bar "Python-flask-app". The left sidebar is titled "EXPLORER" and shows a folder named "PYTHON-FLASK-APP" containing files: "scripts", "app.py", "appspec.yml", "buildspec.yml", "docker file", and "requirements.txt". The right side has three tabs: "appspec.yml", "\$ stop\_container.sh", and "\$ startconatiner.sh". The "\$ startconatiner.sh" tab contains the following script:

```
1 2 3 4 5 6 7 8 9 10
scripts > $ startconatiner.sh
1
2 #!/bin/bash
3 set -e
4
5 # Pull the Docker image from Docker Hub
6 docker pull lionpull123/shwtank-ka-bacha-aagya-hai
7
8 # Run the Docker image as a container
9 docker run -d -p 5000:5000 lionpull123/shwtank-ka-bacha-aagya-hai
10
```

Now simpliy commit change to github

The screenshot shows the VS Code interface with the title bar "Python-flask-app". The left sidebar is titled "EXPLORER" and shows a folder named "PYTHON-FLASK-APP" containing files: "scripts", "app.py", "appspec.yml", "buildspec.yml", "docker file", and "requirements.txt". The right side has three tabs: "appspec.yml", "\$ stop\_container.sh", and "\$ startconatiner.sh". The "appspec.yml" tab contains the following configuration:

```
1 2 3 4 5 6 7 8 9 10 11 12
appspec.yml
1 version: 0.0
2 os: linux
3
4 hooks:
5   ApplicationStop:
6     - location: scripts/stop_container.sh
7       timeout: 300
8       runas: root
9   AfterInstall:
10    - location: scripts/start_container.sh
11      timeout: 300
12      runas: root
```

Click on thrid option and cluick on commit changes

SOURCE CONTROL

adding file for code deploy

✓ Commit

Changes

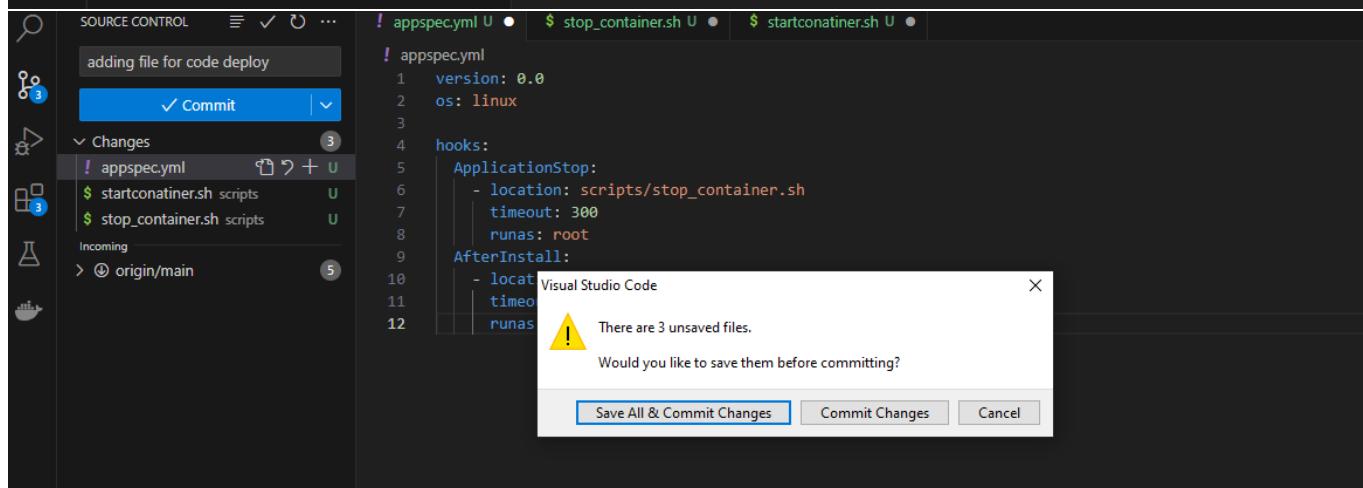
- ! appspec.yml
- \$ startcontainer.sh scripts
- \$ stop\_container.sh scripts

Incoming > origin/main

! appspec.yml

```
version: 0.0
os: linux

hooks:
  ApplicationStop:
    - location: scripts/stop_container.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/start_container.sh
      timeout: 300
      runas: root
```



SOURCE CONTROL

Message (Ctrl+Enter to commit on...)

Sync Changes 5↓ 1↑

Incoming/Outgoing

> origin/main

> main

! appspec.yml

```
version: 0.0
os: linux

hooks:
  ApplicationStop:
    - location: scripts/stop_container.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/start_container.sh
      timeout: 300
      runas: root
```

AFTER COMMITTING WE simply NEED TO click on SYNC CHANGES ON OUR GITHUB REPOSITPRY

Python-flask-app Public

Unpin Unwatch 1 Fork 0 Star 0

main 1 Branch 0 Tags Go to file + Code About

Vickyproject Merge branch 'main' of https://github.com/Vi... a6780d5 · 3 minutes ago 11 Commits

File	Description	Time
scripts	adding file for code deploy	7 minutes ago
Dockerfile	Update and rename docker file to Docke...	2 days ago
app.py	Update app.py	yesterday
appspec.yml	adding file for code deploy	7 minutes ago
buildspec.yml	buildspec.yml	last week
requirements.txt	Update requirements.txt	2 days ago

No description, website, or topics pr

Activity 0 stars 1 watching 0 forks

Releases No releases published Create a new release

Packages

WE SUCCESSFUL ADD OUR APPSPEC FILE

Now DELEET LAST ONE DEPLOYMENT WHICH WAS FAILED AND CREATE NEW DEPLOYMENET AGAIN AGAIN

Developer Tools > CodeDeploy > Applications > PYTHON-FLASK-APP

## PYTHON-FLASK-APP

Notify ▾  Delete application 

### Application details

Name  
PYTHON-FLASK-APP

Compute platform  
EC2/On-premises

Deployments

Deployment groups

Revisions

### Application deployment history



View details

Actions ▾

Copy deployment

Retry deployment

Create deployment



< 1 > 

Deployment Id	Status	Deployment type	Deployment gr...	Revision location	Initiating event	Start time	End time
<input type="radio"/> d-WWZ7RY1F5	 Failed	In-place	deploy-group	github://Vickypr...	User action	1 hour ago	1 hour ago

### Deployment settings

Application  
PYTHON-FLASK-APP

Deployment group

SIMPLE-PYHTON-FINAL-APP 

Use entered value: SIMPLE-PYHTON-FINAL-APP

EC2/On-premises

Revision type

My application is stored in Amazon S3

My application is stored in GitHub

Revision location

Copy and paste the Amazon S3 bucket where your revision is stored



s3://bucket-name/folder/object.[zip|tar|tgz]

## ADD NEW DEPLOYMENT GROUP SINCE LAST ONE IS FAILED

EC2 VPC SNS AWS Auto Scaling Simple Queue Service Simple Notification Service Key Management Service CloudTrail

Deployment group  
SIMPLE-PYTHON-FINAL-APP

Compute platform  
EC2/On-premises

Revision type  
 My application is stored in Amazon S3       My application is stored in GitHub

GitHub token name  
Select the name of the token associated to an account you have already connected, or grant AWS CodeDeploy permission to access a different account. To connect to a GitHub account for the first time, type an alias for the account, and then choose Connect to GitHub

alias

Connect to GitHub

Repository name  
Vickyproject/Python-flask-app

Commit ID  
|

Deployment description

adding file for code deploy



Vickyproject committed 40 minutes ago

4bd0209

Copy full SHA for 4bd0209

Commits on Apr 5, 2024

### Update app.py

Verified

b362325

Connect to GitHub

Repository name

Vickyproject/Python-flask-app

Commit ID

4bd02094eb49e2c26d7b863e8f72ea9858b1f96f

### Deployment description

Deployment description - optional

Add a brief description about the deployment

### Additional deployment behavior settings

#### ▼ Rollback configuration overrides

You can override your deployment group rollback settings for this deployment only.

- Roll back when a deployment fails
- Roll back when alarm thresholds are met
- Disable rollbacks

No Deployment Group found for name: SIMPLE-PYHTON-FINAL-APP

Cancel

Create deployment

SO WE NEED TO SELECT ALREADY DEPLOY GROUP NAME IN TOP

## PYTHON-FLASK-APP

### Deployment group



deploy-group

### Compute platform

EC2/On-premises

### Revision type

Success Deployment created

Developer Tools > CodeDeploy > Deployments > d-6178343F5

d-6178343F5

C Stop deployment Stop and roll back deployment

Deployment details		
Application PYTHON-FLASK-APP	Deployment ID d-6178343F5	Status Created
Deployment configuration CodeDeployDefault.AllAtOnce	Deployment group deploy-group	Initiated by User action
Deployment description ~		

Revision details		
Revision location	Revision created	Revision description

## THESE ALSO FAILED SO LET SOLVE ISSUE

Revision details						
Revision location	Revision created	Revision description				
github://Vickyproject/Python-flask-app/4bd02094eb49e2c26d7b863e8f72ea9858b1f96f	3 minutes ago	Application revision registered by Deployment ID: d-6178343F5				
Event						
Event	Duration	Status	Error code	Start time	End time	
ApplicationStop	less than one second	<span style="color: green;">Succeeded</span>	-	3 minutes ago	3 minutes ago	
DownloadBundle	less than one second	<span style="color: green;">Succeeded</span>	-	3 minutes ago	3 minutes ago	
BeforeInstall	less than one second	<span style="color: green;">Succeeded</span>	-	3 minutes ago	3 minutes ago	
Install	less than one second	<span style="color: green;">Succeeded</span>	-	3 minutes ago	3 minutes ago	
AfterInstall	less than one second	<span style="color: red;">Failed</span>	ScriptMissing	3 minutes ago	3 minutes ago	
ApplicationStart	-	<span style="color: blue;">Skipped</span>	-	-	-	
ValidateService	-	<span style="color: blue;">Skipped</span>	-	-	-	

SO HERE SINCE WE ARE LOOKING THAT DOCKER COMMAND IS FAIL BECAUSE WE DIDN'T INSTALL DOCKER YET

FIRST INSTALL DOCKER.IO on ec2 THROUGH CMD

CMND- sudo apt install docker.io

```
Selecting previously unselected package dns-root-data.
Preparing to unpack .../4-dns-root-data_2023112702~ubuntu0.22.04.1_all.deb ...
Unpacking dns-root-data (2023112702~ubuntu0.22.04.1) ...
Selecting previously unselected package dnsmasq-base.
Preparing to unpack .../5-dnsmasq-base_2.90-0ubuntu0.22.04.1_amd64.deb ...
Unpacking dnsmasq-base (2.90-0ubuntu0.22.04.1) ...
Selecting previously unselected package docker.io.
Preparing to unpack .../6-docker.io_24.0.5-0ubuntu1~22.04.1_amd64.deb ...
Unpacking docker.io (24.0.5-0ubuntu1~22.04.1) ...
Selecting previously unselected package ubuntu-fan.
Preparing to unpack .../7-ubuntu-fan_0.12.16_all.deb ...
Unpacking ubuntu-fan (0.12.16) ...
Setting up dnsmasq-base (2.90-0ubuntu0.22.04.1) ...
Setting up runc (1.1.7-0ubuntu1~22.04.2) ...
Setting up dns-root-data (2023112702~ubuntu0.22.04.1) ...
Setting up bridge-utils (1.7-1ubuntu3) ...
Setting up pigz (2.6-1) ...
Setting up containerd (1.7.2-0ubuntu1~22.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service
```

```
Progress: [ 82%] [#####
.....
```

```
ubuntu@ip-172-31-11-59:~$ 
ubuntu@ip-172-31-11-59:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-doc rinse zfs-fuse | zf
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan
0 upgraded, 8 newly installed, 0 to remove and 15 not upgraded.
Need to get 69.8 MB of archives.
After this operation, 267 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
scanning linux images...

Restarting services...
/etc/needrestart/restart.d/systemd-manager /etc/needrestart/restart.d/systemd-manager
systemctl restart packagekit.service polkit.service rsyslog.service serial-getty@ttyS0.service sys
systemd-networkd.service systemd-resolved.service systemd-udevd.service
Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service
systemctl restart user@1000.service

0 containers need to be restarted.

0 user sessions are running outdated binaries.

0 VM guests are running outdated hypervisor (qemu) binaries on this host.
```

Now retry deployment again

arn:aws:ec2:us-east-1:905418179079:instance/i-08a20e1e3c67ef7b5



scripts/start\_container.sh

Script does not exist at specified location: /opt/codedeploy-agent/deployment-root/79d89d7b-a6bb-4a51-989b-23f99d8ea58c/d-6178343F5/deployment-archive/scripts/start\_container.sh [View more](#).

## Deployment details

Application  
PYTHON-FLASK-APP

Deployment ID  
d-6178343F5

Status  
✖ Failed

Deployment configuration  
CodeDeployDefault.AllAtOnce

Deployment group  
deploy-group

Initiated by  
User action

Deployment description  
-

## Revision details

Revision location

Revision created

Revision description

The screenshot shows the AWS CodeDeploy console. On the left, there's a sidebar with navigation links like Developer Tools, CodeDeploy, Source, Artifacts, Build, Deploy, Getting started, Deployments, Applications, Application (Settings, Deployment configurations), On-premises instances, Pipeline, and Settings. The main area shows the application details for 'PYTHON-FLASK-APP'. It includes the application name, compute platform (EC2/On-premises), and deployment history. The deployment history table lists two entries: one successful deployment (d-6178343F5) and one failed deployment (d-WWZ7RY1F5). Both deployments show a status of 'Failed'.

Deployment Id	Status	Deployment type	Deployment gr...	Revision location	Initiating event	Start time	End time
d-6178343F5	<span style="color: red;">✖ Failed</span>	In-place	deploy-group	github://Vickypr...	User action	9 minutes ago	9 minutes ago
d-WWZ7RY1F5	<span style="color: red;">✖ Failed</span>	In-place	deploy-group	github://Vickypr...	User action	1 hour ago	1 hour ago

## Revision details

Revision location

github://Vickyproject/Python-flask-app/4bd02094eb49e2c26d7b863e8f72ea9858b1f96f

Revision created

9 minutes ago

Revision description

Application revision registered by Deployment ID: d-6178343F5

Event	Duration	Status	Error code	Start time	End time
ApplicationStop	less than one second	<span style="color: green;">✓ Succeeded</span>	-	Just now	Just now
DownloadBundle	less than one second	<span style="color: green;">✓ Succeeded</span>	-	Just now	Just now
BeforeInstall	less than one second	<span style="color: green;">✓ Succeeded</span>	-	Just now	Just now
Install	less than one second	<span style="color: green;">✓ Succeeded</span>	-	Just now	Just now
AfterInstall	less than one second	<span style="color: red;">✖ Failed</span>	ScriptMissing	Just now	Just now
ApplicationStart	-	<span style="color: blue;">ⓘ Skipped</span>	-	-	-
ValidateService	-	<span style="color: blue;">ⓘ Skipped</span>	-	-	-

Scripts is missing so lets check

Noe create new deployment again and these time add lastest recent commit id

Lets do it again

Deployment group

X

Compute platform

EC2/On-premises

Deployment type

In-place

Managed hook execution role

The IAM role used by the CodeDeploy Managed Hook function to perform actions. [Edit Managed Hook execution role.](#)

-

Revision type

My application is stored in Amazon S3  My application is stored in GitHub

Revision location

Copy and paste the Amazon S3 bucket where your revision is stored

[Connect to GitHub](#)

Repository name

Commit ID

---

**Deployment description**

Deployment description - optional

Add a brief description about the deployment

PYTHON-FLASK-APP

Notify ▾ Delete application

**Application details**

Name PYTHON-FLASK-APP	Compute platform EC2/On-premises
--------------------------	-------------------------------------

Deployments Deployment groups Revisions

**Deployment groups**

Name	Status	Last attempted deployment	Last successful deployment	Trigger count
deploy-group	Failed	Apr 6, 2024 10:51 AM (UTC+5:...)	-	0

Developer Tools > CodeDeploy > Applications > PYTHON-FLASK-APP > Create deployment group

## Create deployment group

**Application**

Application PYTHON-FLASK-APP	Compute type EC2/On-premises
---------------------------------	---------------------------------

**Deployment group name**

Enter a deployment group name

100 character limit

**Service role**

Enter a service role

**Event details**

Error code  
✖ ScriptMissing

Script name  
scripts/start\_container.sh

Message  
Script does not exist at specified location: /opt/codedeploy-agent/deployment-root/79d89d7b-a6bb-4a51-989b-23f99d8ea58c/d-1RKAWI3F5/deployment-archive/scripts/start\_container.sh

Logs

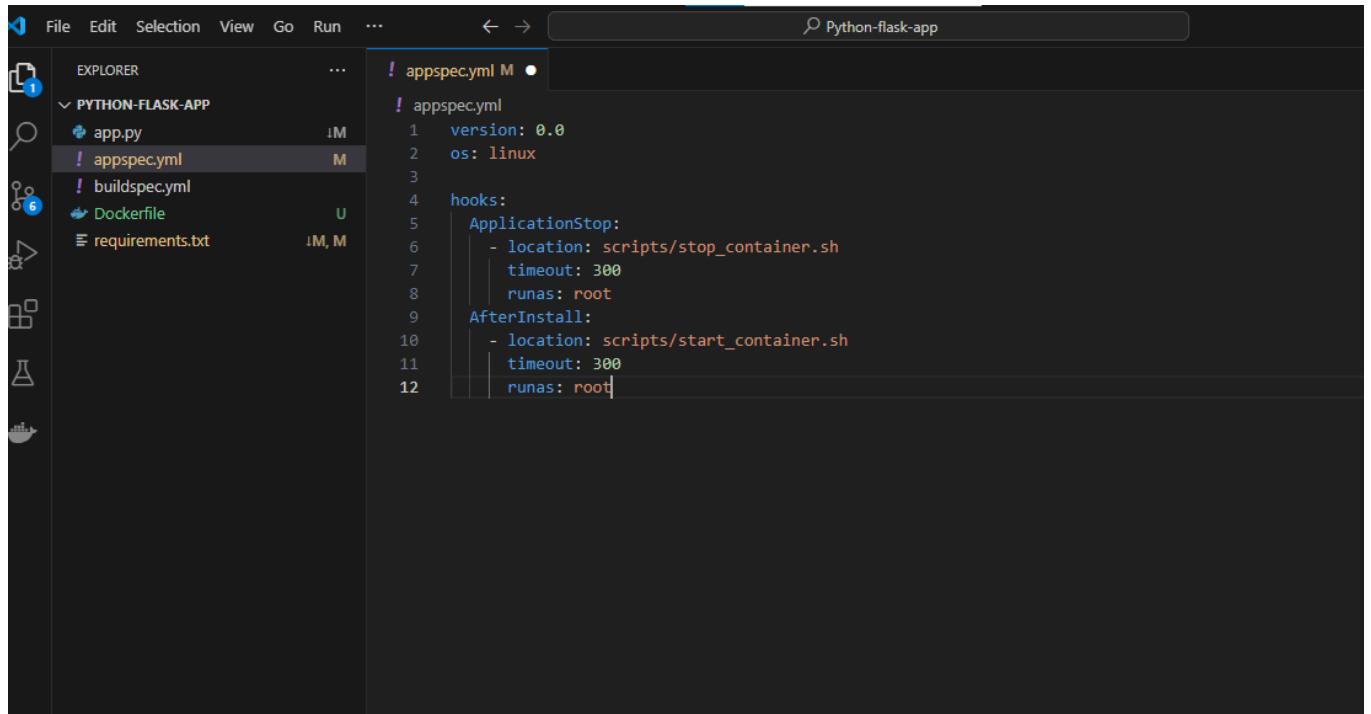
LifecycleEvent - AfterInstall

So we need to trouble shoot these issue

Go to github desktop and open it in visual studio



Create a new file which is appspec.yml



Now create new file as script and inside it add two folder which is startcontainer.sh and stopcontainer.sh

The screenshot shows the Visual Studio Code interface with a dark theme. On the left, the Source Control sidebar displays a list of files: appspec.yml (M), dockerfile (IR, D), Dockerfile (U), requirements.txt (IM, M), scripts (U), start\_container.sh (D), stop\_container.sh (U), startcontainer.sh (scripts) (D), and stop\_container.sh (scripts) (D). A commit dialog is open at the top, showing the command `start_container.sh` with its content:

```
$ start_container.sh
1
2 #!/bin/bash
3 set -e
4
5 # Pull the Docker image from Docker Hub
6 docker pull lionpull123/shwetank-ka-bacha-aagya-hai
7
8 # Run the Docker image as a container
9 docker run -d -p 5000:5000 lionpull123/shwetank-ka-bacha-aagya-hai
```

A modal window titled "Visual Studio Code" is displayed, containing a warning message: "There are 3 unsaved files." followed by "Would you like to save them before committing?". It has three buttons: "Save All & Commit Changes" (highlighted in blue), "Commit Changes", and "Cancel".

Here we successful push our changes to our python repository

The screenshot shows a GitHub repository page for "Python-flask-app". The repository is public and owned by "Vickyproject". The main branch is "main". The repository has 1 branch and 0 tags. The last commit was made 6 minutes ago by "Vickyproject" with the message "new app deploy". The commit history includes:

- scripts: new app deploy (6 minutes ago)
- Dockerfile: add new file (22 minutes ago)
- app.py: Update app.py (2 days ago)
- appspec.yml: new (17 minutes ago)
- buildspec.yml: buildspec.yml (last week)
- requirements.txt: add new file (22 minutes ago)

The "About" section indicates no description, website, or topics. The "Activity" section shows 0 stars, 1 watching, and 0 forks. The "Releases" section shows no releases published. The "Packages" section shows no packages published.

So now again creating new deploy

The screenshot shows the AWS CodeDeploy 'Create deployment' interface. At the top, there's a navigation bar with links to EC2, VPC, SS, AWS Auto Scaling, Simple Queue Service, Simple Notification Service, Key Management Service, CloudTrail, and Amazon Event. Below the navigation, a breadcrumb trail indicates the path: Developer Tools > CodeDeploy > Applications > PYTHON-FLASK-APP > Create deployment. The main title is 'Create deployment'. Under the 'Deployment settings' section, the 'Application' is set to 'PYTHON-FLASK-APP'. The 'Deployment group' field contains 'deploy-group' and has a clear button 'X'. The 'Compute platform' is set to 'EC2/On-premises'. The 'Deployment type' is 'In-place'. Under 'Managed hook execution role', it says 'The IAM role used by the CodeDeploy Managed Hook function to perform actions. [Edit Managed Hook execution role.](#)' followed by a minus sign. The 'Revision type' section shows two options: 'My application is stored in Amazon S3' (unchecked) and 'My application is stored in GitHub' (checked). The entire form is contained within a light gray box.

-

Revision type

My application is stored in Amazon S3

My application is stored in GitHub

GitHub token name

Select the name of the token associated to an account you have already connected, or grant AWS CodeDeploy permission to access a different account. To connect to a GitHub account for the first time, type an alias for the account, and then choose Connect to GitHub

alias X

[Connect to GitHub](#)

Repository name

Vickyproject/Python-flask-app

Commit ID

**Deployment description**

Deployment description - optional

[Connect to GitHub](#)

Repository name

Vickyproject/Python-flask-app

Commit ID

626259b75a6ae8e5911f219e993dd6caf800a19d

**Deployment description**

Deployment description - optional

Add a brief description about the deployment

**Additional deployment behavior settings**

NOW JUST CLICK ON CREATE DEPLOY AND WAIT FOR RESULTK

The screenshot shows the AWS CodeDeploy console. On the left, there's a sidebar with navigation links for Developer Tools, CodeCommit, CodeArtifact, CodeBuild, CodeDeploy, CodePipeline, and Settings. The 'CodeDeploy' link is selected. The main area displays a deployment named 'CodeDeployDefault.AllAtOnce'. The deployment description is '-'. Below this, a 'Revision details' section shows the revision location as 'github://Vickyproject/Python-flask-app/626259b75a6ae8e5911f219e993dd6caf800a19d' and the revision created as 'Just now'. The revision description is 'Application revision registered by Deployment ID: d-NHP58N3F5'. A table below lists deployment events:

Event	Duration	Status	Error code	Start time	End time
ApplicationStop	less than one second	<span style="color: green;">Succeeded</span>	-	Apr 6, 2024 1:17 PM (UTC+5:30)	Apr 6, 2024 1:17 PM (UTC+5:30)
DownloadBundle	less than one second	<span style="color: green;">Succeeded</span>	-	Apr 6, 2024 1:17 PM (UTC+5:30)	Apr 6, 2024 1:17 PM (UTC+5:30)
BeforeInstall	less than one second	<span style="color: green;">Succeeded</span>	-	Apr 6, 2024 1:17 PM (UTC+5:30)	Apr 6, 2024 1:17 PM (UTC+5:30)
Install	less than one second	<span style="color: green;">Succeeded</span>	-	Apr 6, 2024 1:17 PM (UTC+5:30)	Apr 6, 2024 1:17 PM (UTC+5:30)
AfterInstall	27 seconds	<span style="color: green;">Succeeded</span>	-	Apr 6, 2024 1:17 PM (UTC+5:30)	Apr 6, 2024 1:17 PM (UTC+5:30)
ApplicationStart	less than one second	<span style="color: green;">Succeeded</span>	-	Apr 6, 2024 1:17 PM (UTC+5:30)	Apr 6, 2024 1:17 PM (UTC+5:30)
ValidateService	less than one second	<span style="color: green;">Succeeded</span>	-	Apr 6, 2024 1:17 PM (UTC+5:30)	Apr 6, 2024 1:17 PM (UTC+5:30)

Finally after a lot of hardwork I succesfull complete all stage

Now our final STEP IS TO INTEGRATE THESE CODE DEPLOY WHICH WE CREATE JUST NOW WITH CODE PIPELINES AND MERGE PART 1 AND PART 2 I.E CI AND CD WITH AWS CODE PIPWELINES

SO GO TO CODEPIPELINE

The screenshot shows the AWS CodePipeline console. A search bar at the top right contains the text 'CODE Pipeline'. Below it, a sidebar on the left lists various AWS services under 'Developer Tools' and 'CodeDeploy'. The main area displays a search result for 'CODE Pipeline' with three items: 'CodePipeline', 'Amazon Location Service', and 'Elastic Container Service'. On the right, a detailed view of the 'BICKY' pipeline is shown. The pipeline type is V2 and the execution mode is SUPERSEDED. It has one stage named 'Source' which is set to GitHub (Version 2) and has succeeded 17 minutes ago. There are also sections for 'History' and 'Settings'. At the bottom right of the pipeline view, there is a 'Release change' button.

GO TO EDIT

The screenshot shows the 'Edit' interface for the 'BICKY' pipeline. It displays two stages: 'Source' and 'Build'. The 'Source' stage is currently selected, showing its configuration for GitHub (Version 2). Below the stages, there is a button labeled '+ Add stage'.

CLICK ON ADD STAGE

Developer Tools X

**CodePipeline**

- ▶ Source • CodeCommit
- ▶ Artifacts • CodeArtifact
- ▶ Build • CodeBuild
- ▶ Deploy • CodeDeploy
- ▼ Pipeline • CodePipeline
- Getting started
- Pipelines:
  - Pipeline
  - History
  - Settings
- ▶ Settings

Developer Tools X

**CodePipeline**

- ▶ Source • CodeCommit
- ▶ Artifacts • CodeArtifact
- ▶ Build • CodeBuild
- ▶ Deploy • CodeDeploy
- ▼ Pipeline • CodePipeline
- Getting started
- Pipelines
  - Pipeline**
  - History
  - Settings
- ▶ Settings

+ Add stage Edit stage

**Edit: Source**

Source

Cancel Add stage

**Add stage**

Stage name

code-deploy

No more than 100 characters

Cancel Add stage

**Edit: Build**

Build

AWS CodeBuild

+ Add stage

**Edit: code-deploy**

+ Add action group

+ Add stage Edit stage Delete Done

**Edit action**

Action name

Choose a name for your action

code-deploy

No more than 100 characters

Action provider

AWS CodeDeploy

Region

Asia Pacific (Mumbai)

Input artifacts

Choose an input artifact for this action. [Learn more](#)

No more than 100 characters

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.

Q Go to AWS CodeDeploy C

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.

Q Go to AWS CodeDeploy C

**Edit action**

Action name  
Choose a name for your action  
 No more than 100 characters

Action provider

Region

Input artifacts  
Choose an input artifact for this action. [Learn more](#)

Action provider

Region

Input artifacts  
Choose an input artifact for this action. [Learn more](#)

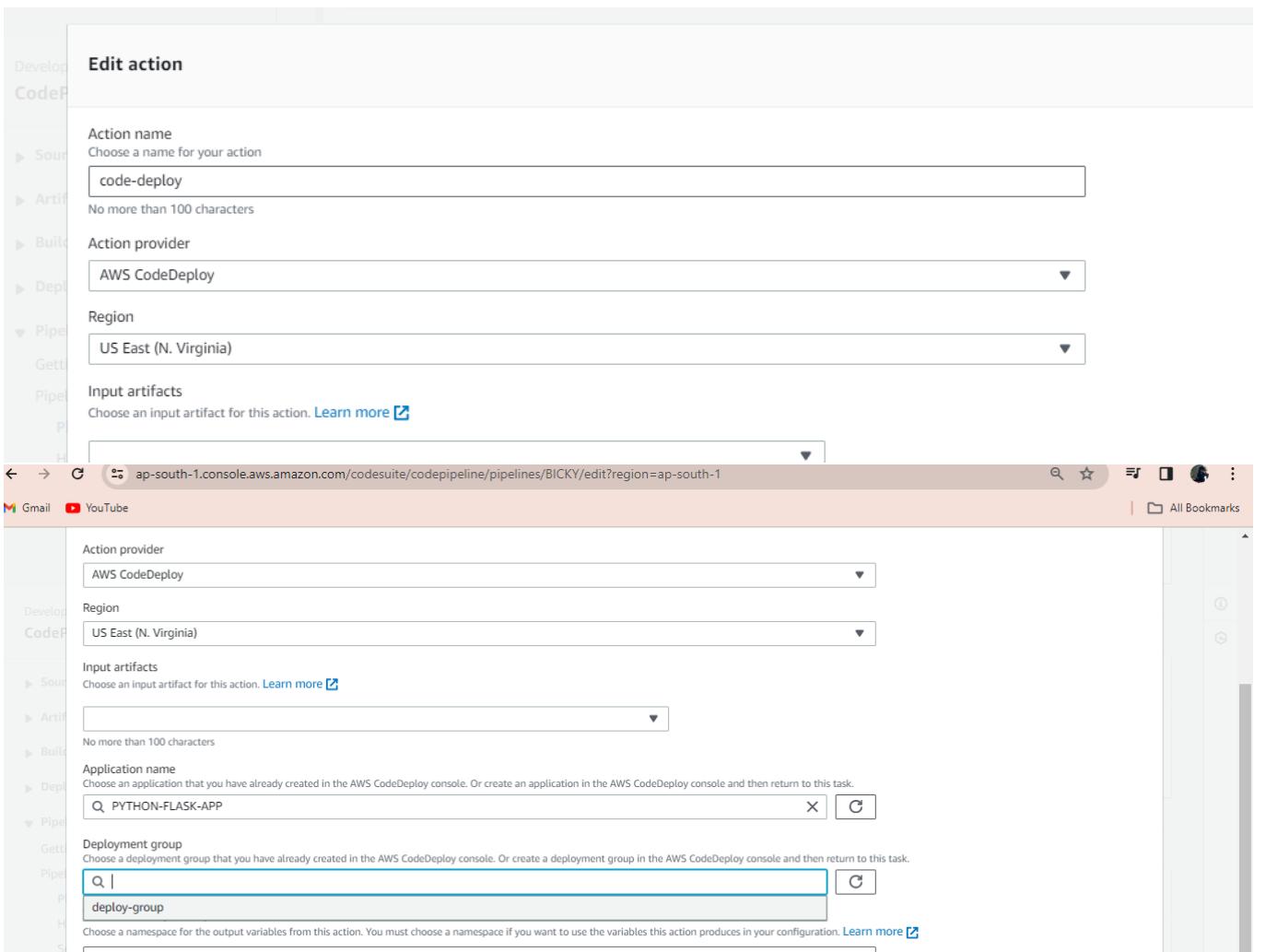
No more than 100 characters

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.  
 X C

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.  
 X C

Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. [Learn more](#)

Gmail YouTube All Bookmarks



Action provider

Region

Input artifacts  
Choose an input artifact for this action. [Learn more](#)

BuildArtifact  
 No more than 100 characters

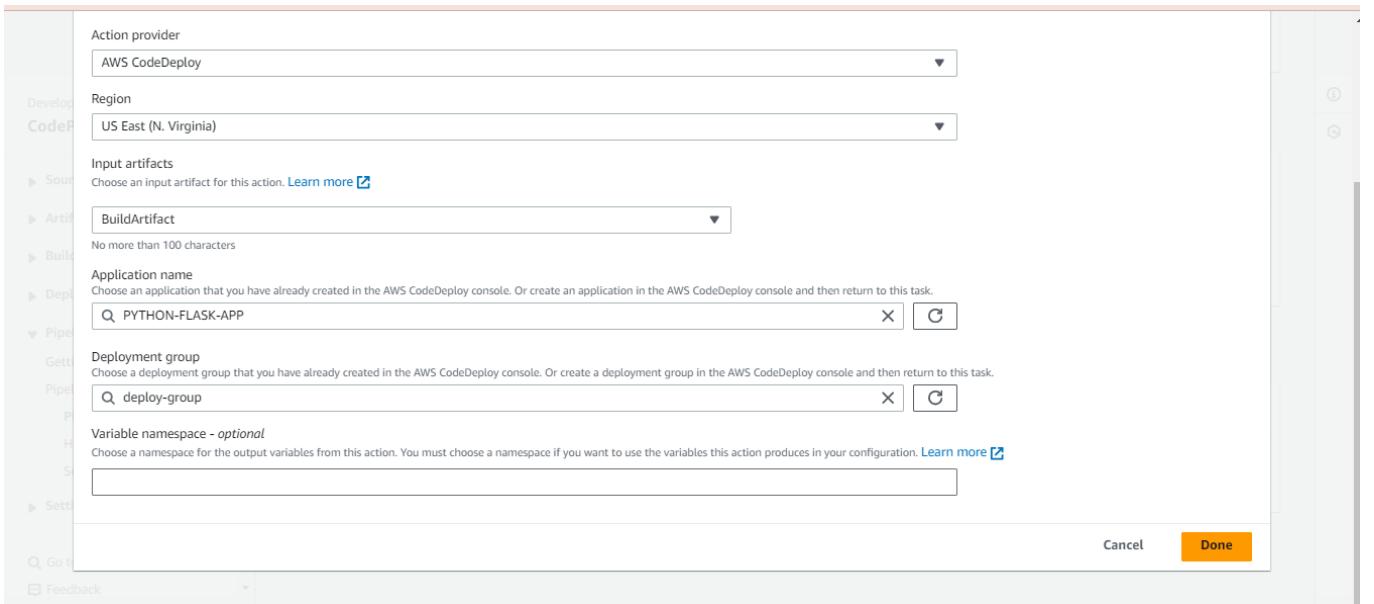
Application name  
Choose an application that you have already created in the AWS CodeDeploy console. Or create an application in the AWS CodeDeploy console and then return to this task.  
 X C

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.  
 X C

Variable namespace - optional  
Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. [Learn more](#)

Cancel Done

Q Go N Feedback



## Click on done

The screenshot shows the AWS CodeDeploy configuration dialog. The 'Action provider' is set to 'AWS CodeDeploy'. The 'Region' is 'US East (N. Virginia)'. Under 'Input artifacts', 'BuildArtifact' is selected. The 'Application name' is 'PYTHON-FLASK-APP'. The 'Deployment group' is 'deploy-group'. There is a section for 'Variable namespace - optional' which is currently empty. At the bottom right are 'Cancel' and 'Done' buttons.

Action provider  
AWS CodeDeploy

Region  
US East (N. Virginia)

Input artifacts  
Choose an input artifact for this action. [Learn more](#)

BuildArtifact

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.  
Q PYTHON-FLASK-APP

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.  
Q deploy-group

Variable namespace - optional  
Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. [Learn more](#)

Cancel Done

AWS Auto Scaling Simple Queue Service Simple Notification Service Key Management Service CloudTrail Amazon EventBridge RDS IAM Lambda DynamoDB

Developer Tools > CodePipeline > Pipelines > BICKY > Edit BICKY

Editing: BICKY

Delete Cancel Save

Edit: Pipeline properties

Pipeline type  
V2

Execution mode  
SUPERSEDED

Click on save button

**Success**  
Pipeline was saved successfully.

[Developer Tools](#) > [CodePipeline](#) > [Pipelines](#) > BICKY

**BICKY**

Pipeline type: V2 Execution mode: SUPERSEDED

**Source** Succeeded  
Pipeline execution ID: [f117a628-8868-4bae-a510-7da93820d60f](#)

Source  
[GitHub \(Version 2\)](#)  
Succeeded - 22 minutes ago  
[626259b7](#)

[View details](#)

[626259b7](#) Source: new app deploy

**Build** Succeeded  
Pipeline execution ID: [f117a628-8868-4bae-a510-7da93820d60f](#)

Build  
[AWS CodeBuild](#)  
Succeeded - 23 minutes ago

[View details](#)

[626259b7](#) Source: new app deploy

[Disable transition](#)

**code-deploy** Didn't Run

code-deploy  
[AWS CodeDeploy](#)  
Didn't Run

SO OUR ENTIRE CI CD PIPELINE IS MADE LET CHECK BY COMMITING SMALL CHANGE IN GITHUB

1  
2  
3  
4  
5  
6

