Project 1 : Capstone 1

You have been hired as a Sr. DevOps Engineer in Abode Software. They want to implement DevOps Lifecycle in their company. You have been asked to implement this lifecycle as fast as possible. Abode Software is a product-based company and their product is available on this GitHub link.

https://github.com/hshar/website.git

Following are the specifications of the lifecycle:

1. Install the necessary software on the machines using a configuration management tool

2. Git workflow has to be implemented

3. CodeBuild should automatically be triggered once a commit is made to master branch or develop branch.

a. If a commit is made to master branch, test and push to prod

b. If a commit is made to develop branch, just test the product, do not push to prod

4. The code should be containerized with the help of a Dockerfile. The Dockerfile should be built every time there is a push to GitHub. Use the following pre-built container for your application: hshar/webapp

The code should reside in '/var/www/html'

5. The above tasks should be define i jenkins pipelines with the following jobs

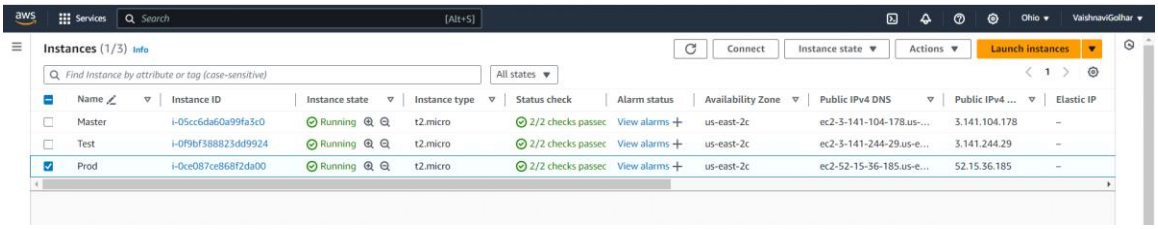
a. Job1 : build

b. Job2 : test

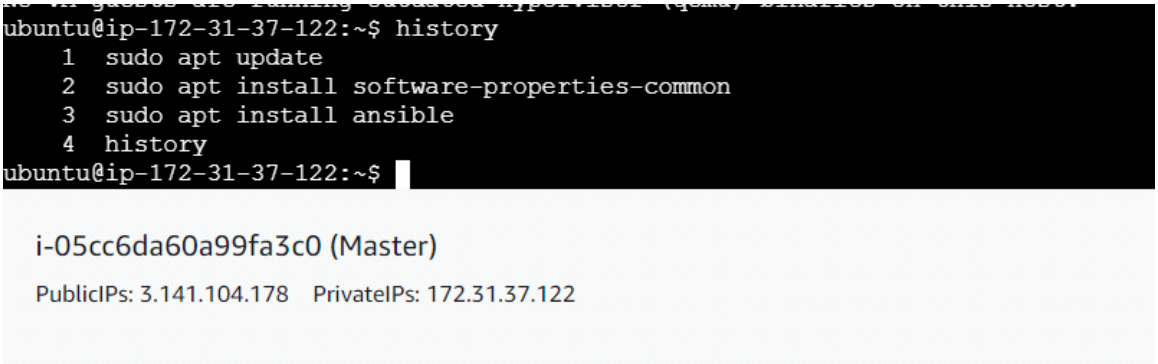
c. Job3 : prod

SOLUTION:

1. Set Up AWS EC2 Instances



2. Install Necessary Software



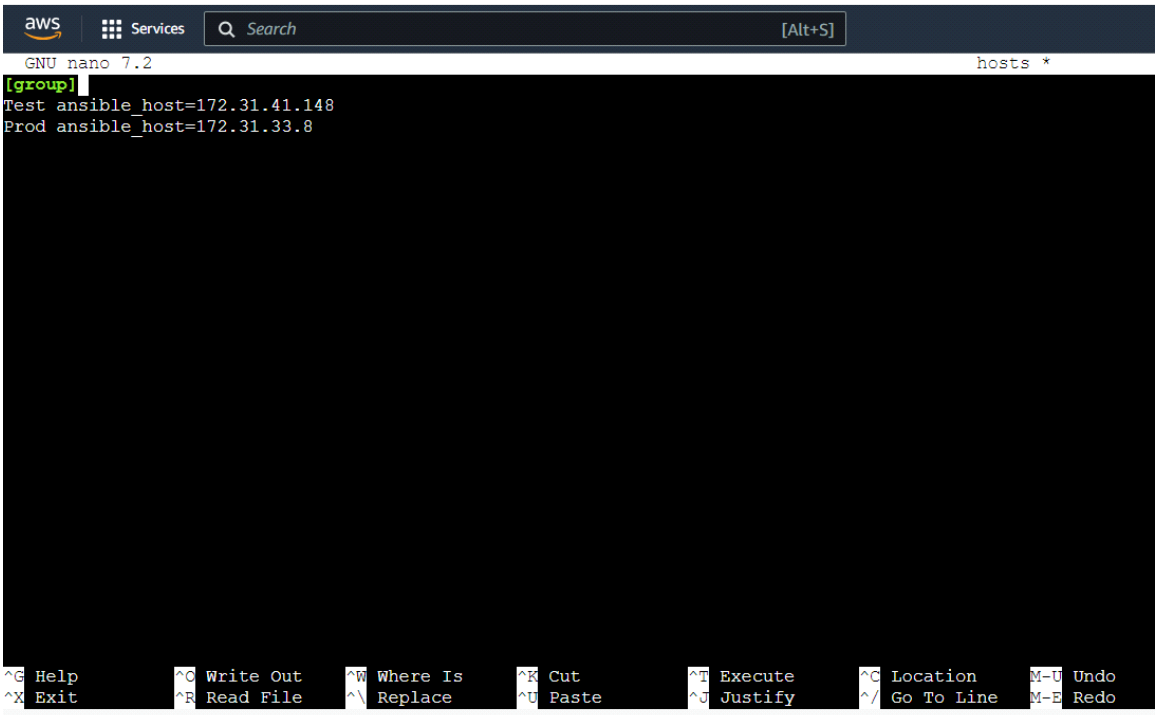
3. Create ssh conection between Master and Slaves

```
    5  ssh-keygen
    6  cd .ssh
    7  ls
    8  sudo cat id_ed25519.pub
    9  cd
   10  ssh 172.31.41.148
   11  ssh 172.31.33.8
   12  history
ubuntu@ip-172-31-37-122:~$
```

i-05cc6da60a99fa3c0 (Master)

PublicIPs: 3.141.104.178 PrivateIPs: 172.31.37.122

4. Create an Ansible Inventory File:



i-05cc6da60a99fa3c0 (Master)

PublicIPs: 3.141.104.178 PrivateIPs: 172.31.37.122

```
ubuntu@ip-172-31-37-122:~$ cd /etc/ansible
ubuntu@ip-172-31-37-122:/etc/ansible$ ls
ansible.cfg  hosts  roles
ubuntu@ip-172-31-37-122:/etc/ansible$ sudo nano hosts
ubuntu@ip-172-31-37-122:/etc/ansible$ sudo cat hosts
[group]
Test ansible_host=172.31.41.148
Prod ansible_host=172.31.33.8
ubuntu@ip-172-31-37-122:/etc/ansible$
```

i-05cc6da60a99fa3c0 (Master)

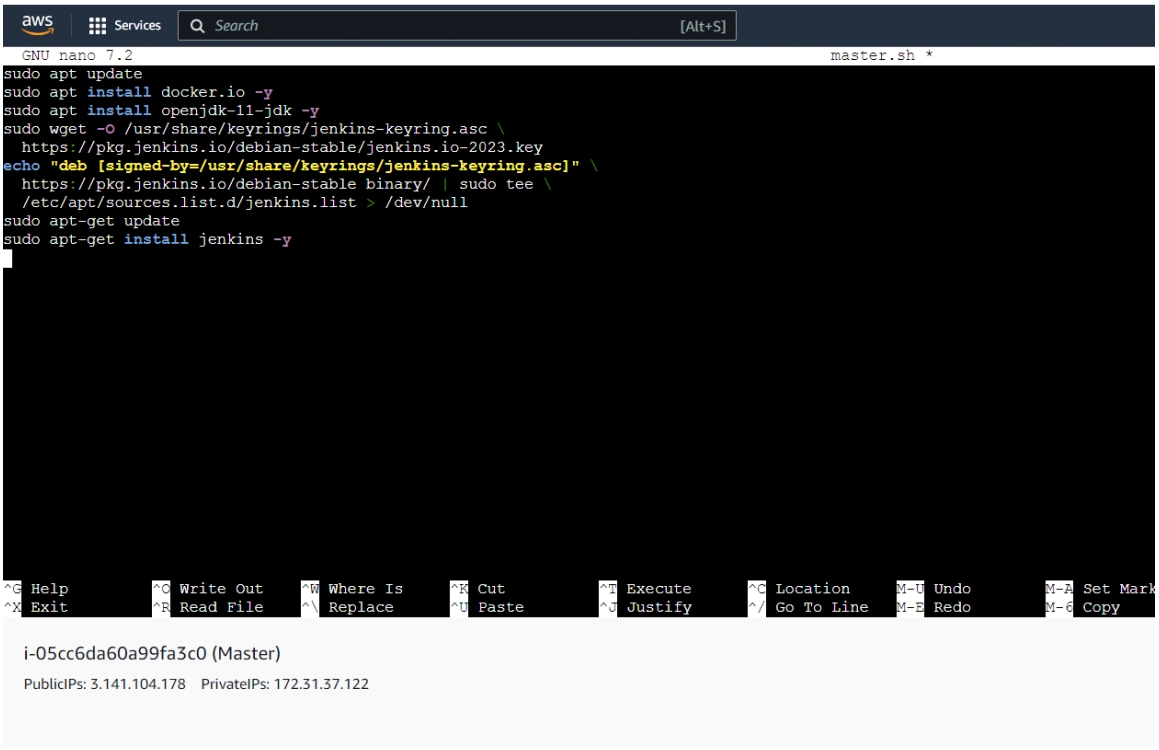
PublicIPs: 3.141.104.178 PrivateIPs: 172.31.37.122

```
ubuntu@ip-172-31-37-122:/etc/ansible$ ansible -m ping all
Prod | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
Test | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ubuntu@ip-172-31-37-122:/etc/ansible$
```

i-05cc6da60a99fa3c0 (Master)

PublicIPs: 3.141.104.178 PrivateIPs: 172.31.37.122

5. Create sript to run master and slaves

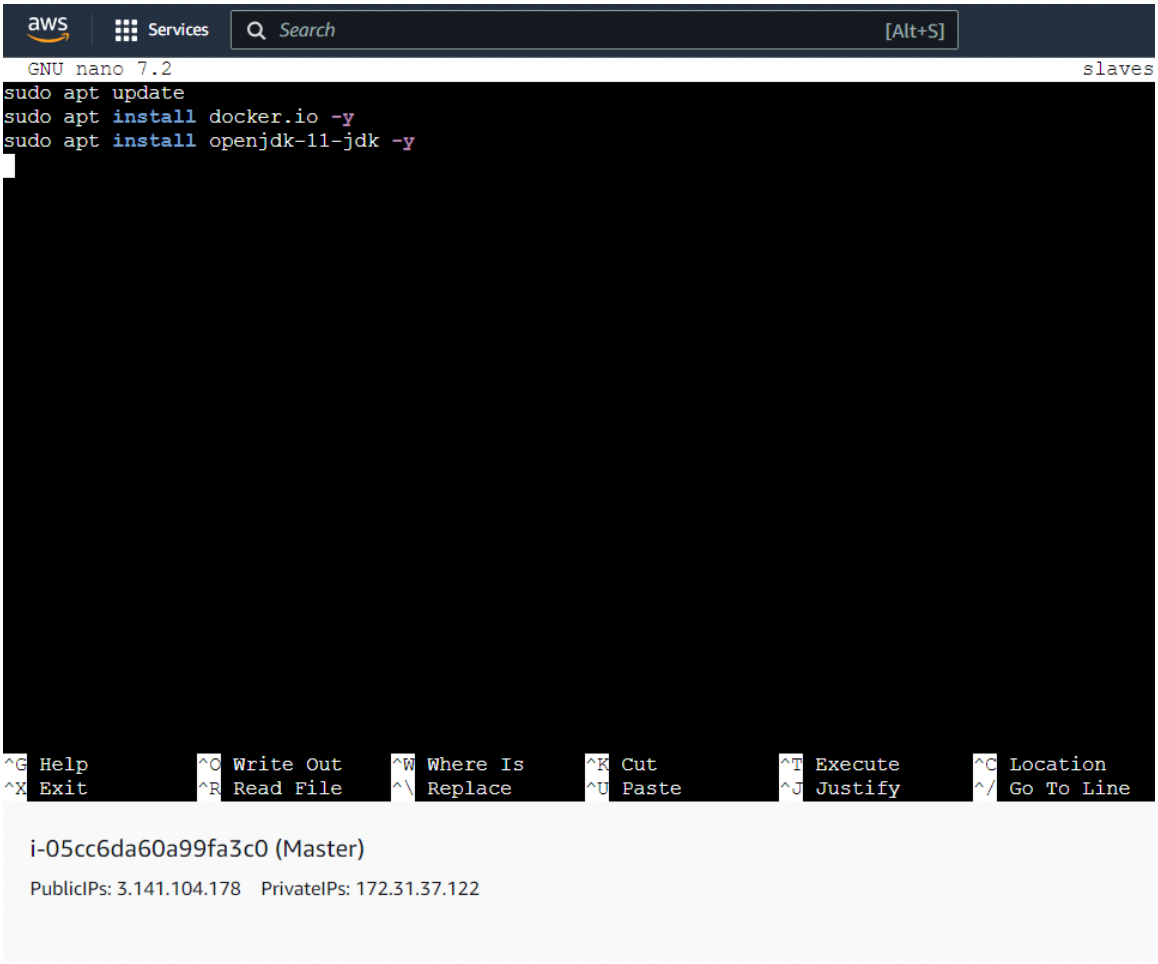


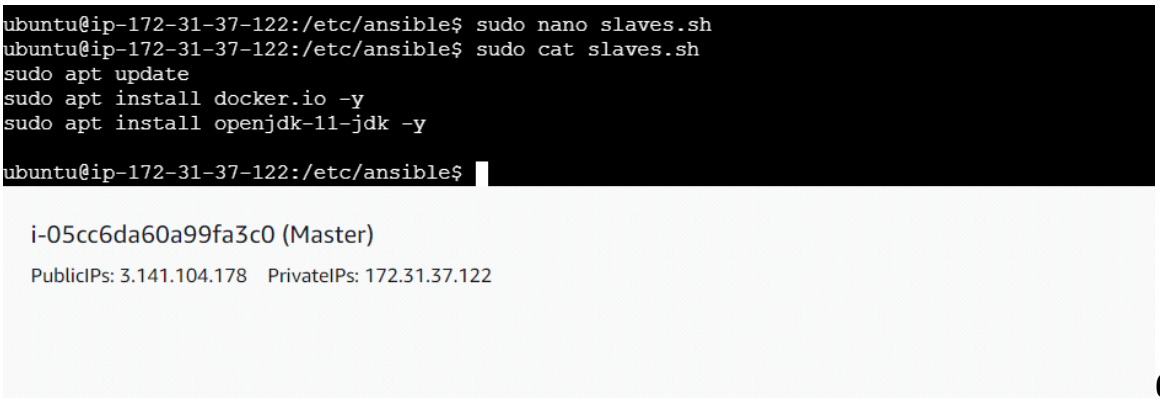

```
ubuntu@ip-172-31-37-122:/etc/ansible$ sudo nano master.sh
ubuntu@ip-172-31-37-122:/etc/ansible$ sudo cat master.sh
sudo apt update
sudo apt install docker.io -y
sudo apt install openjdk-11-jdk -y
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins -y

ubuntu@ip-172-31-37-122:/etc/ansible$
```

i-05cc6da60a99fa3c0 (Master)

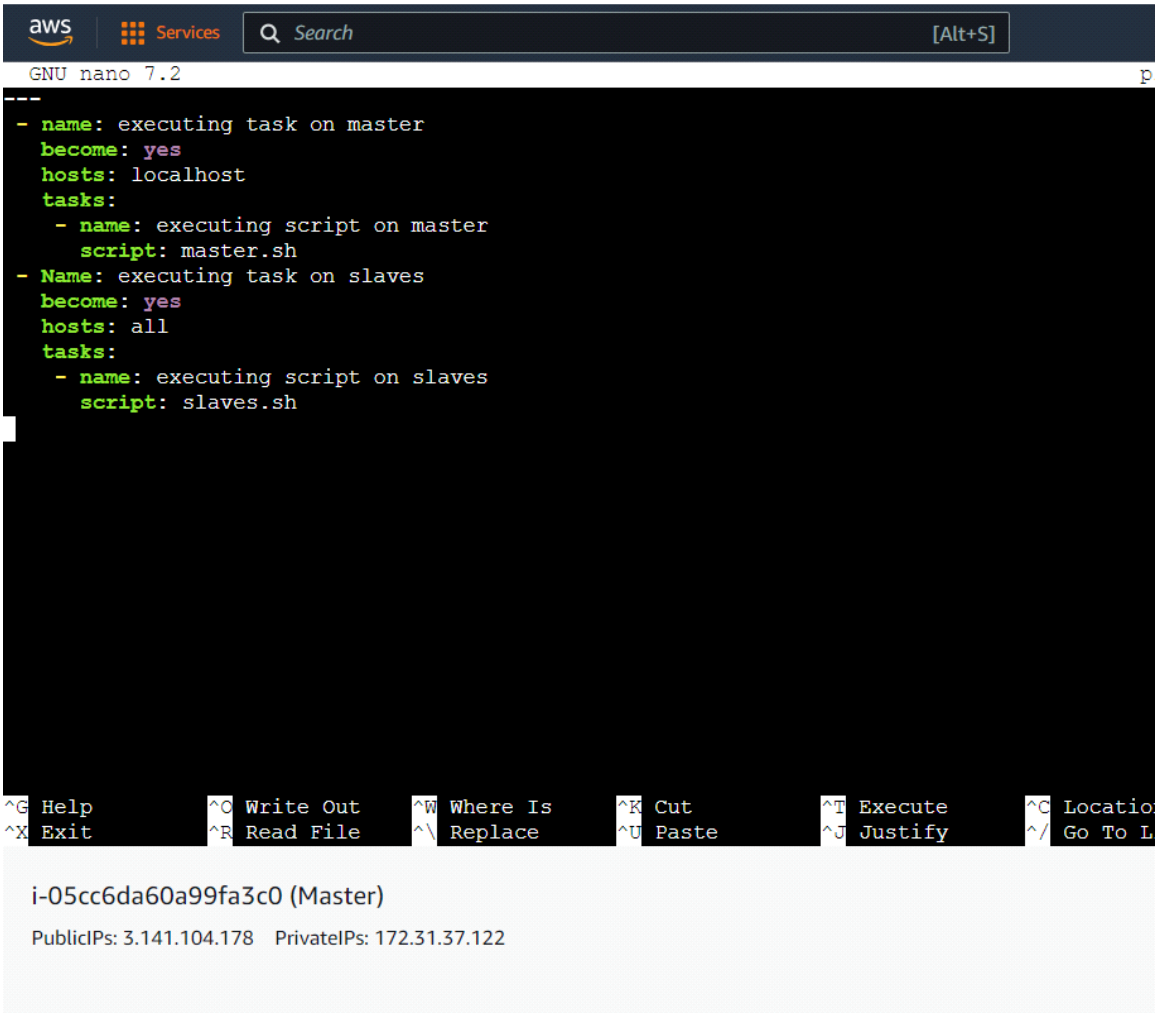
PublicIPs: 3.141.104.178 PrivateIPs: 172.31.37.122



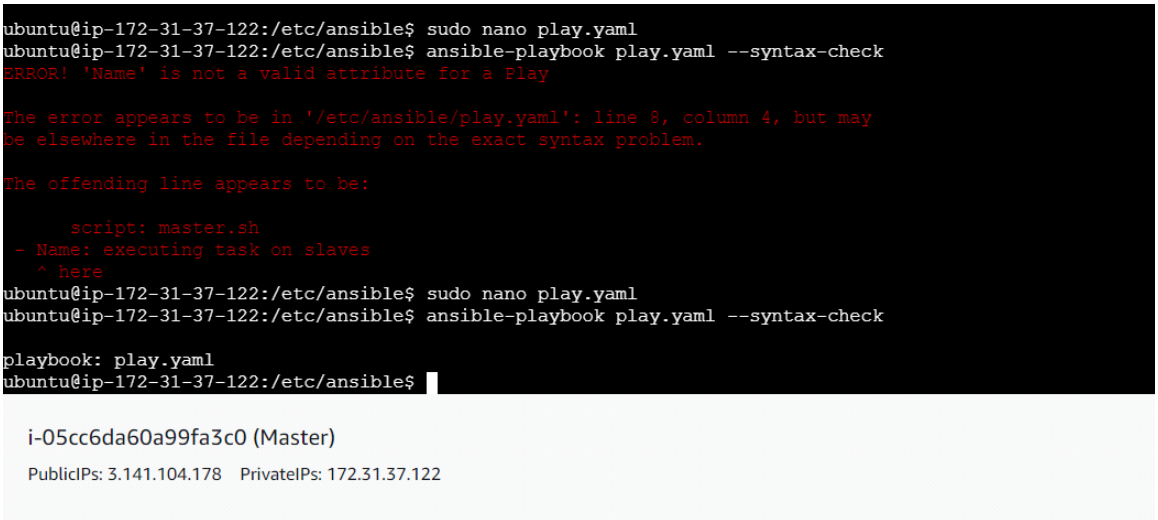


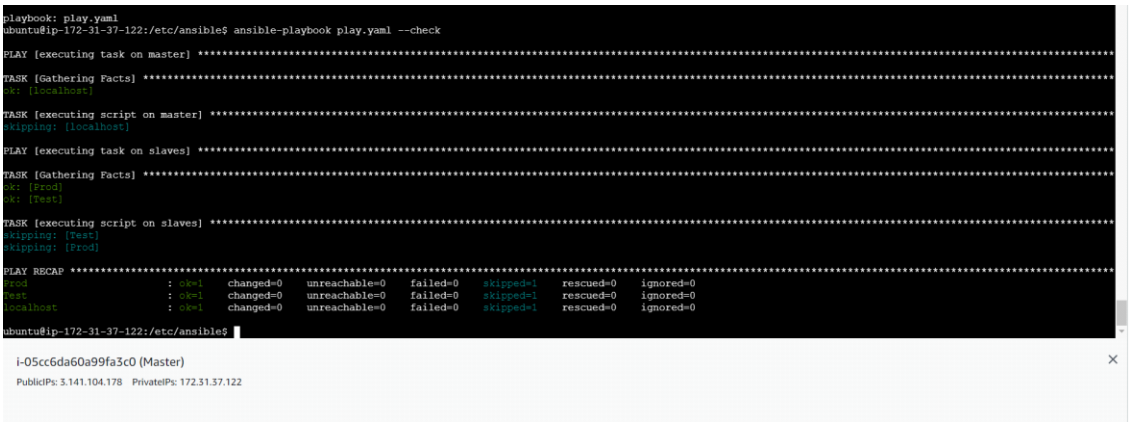
6.

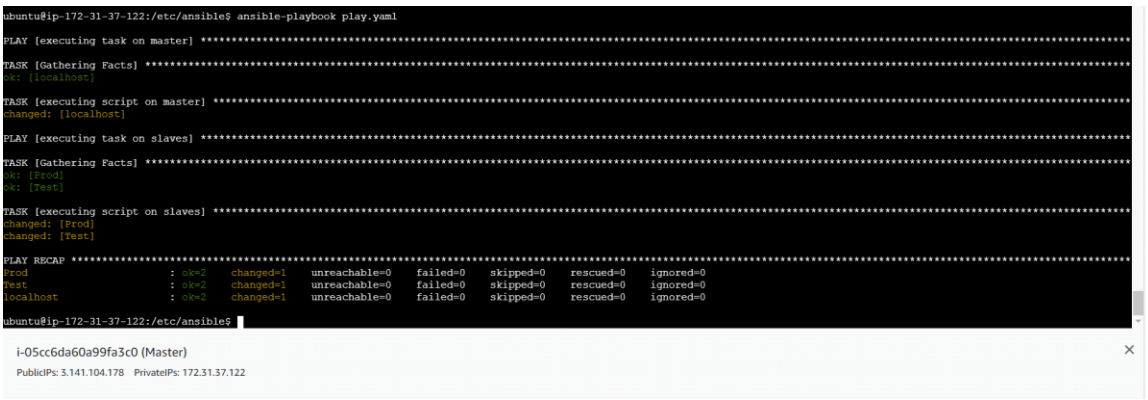
Create an Ansible Playbook:



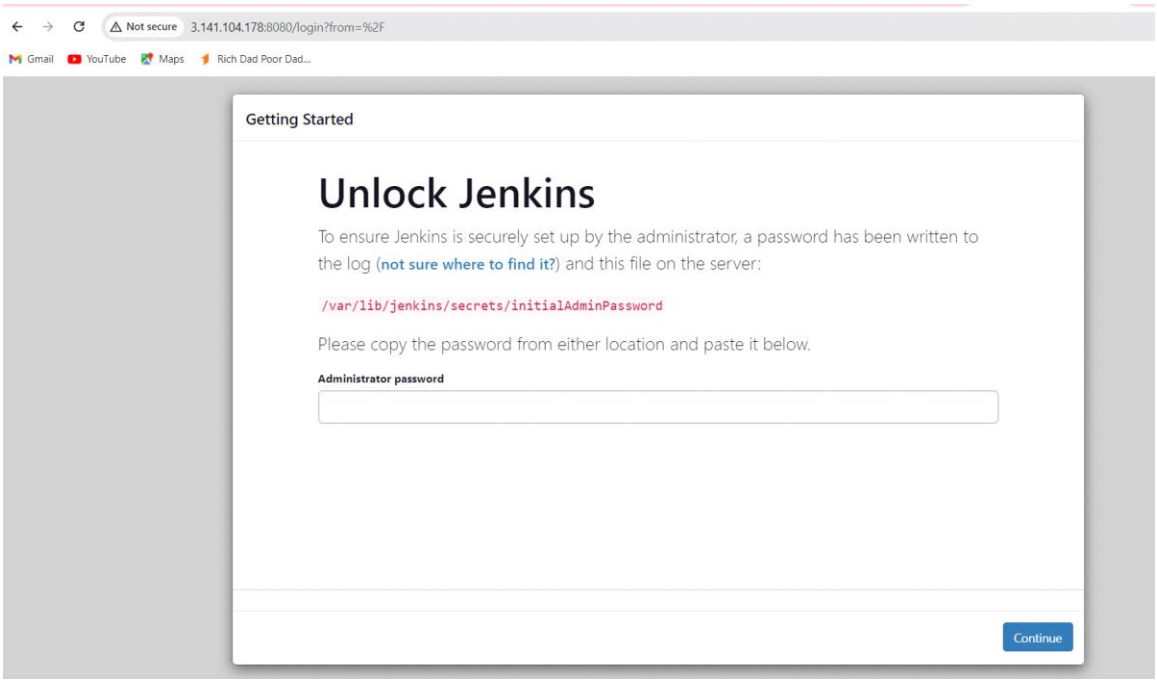
7. Run the Ansible Playbook:

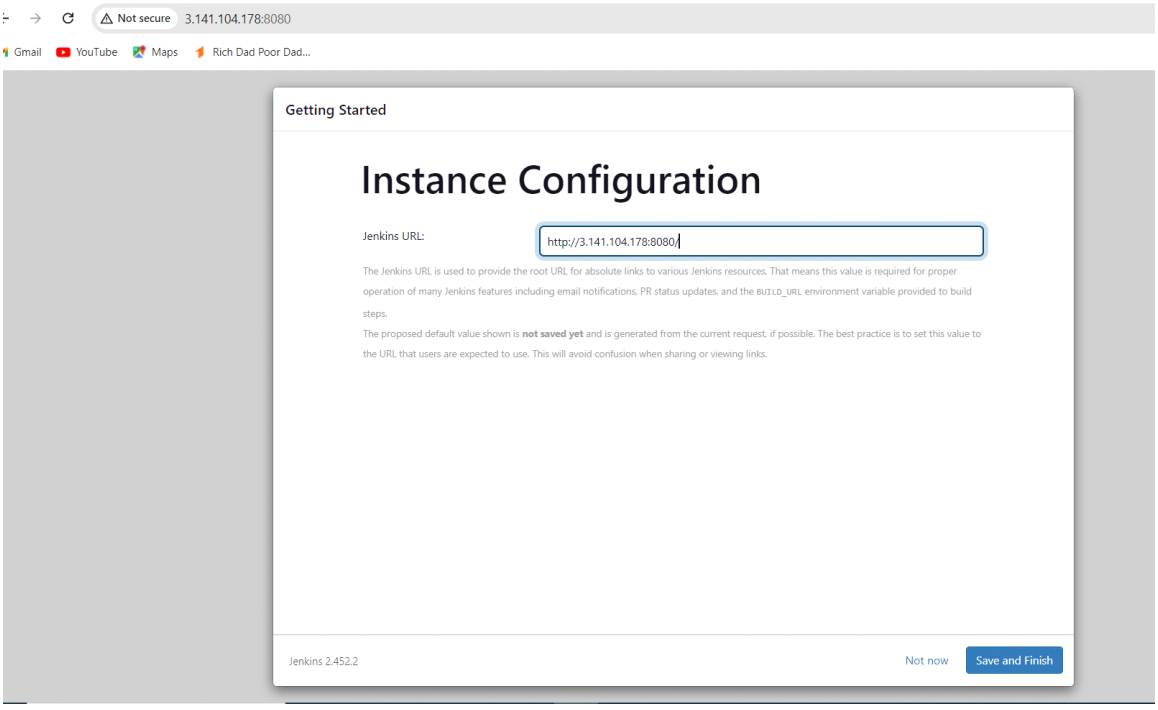




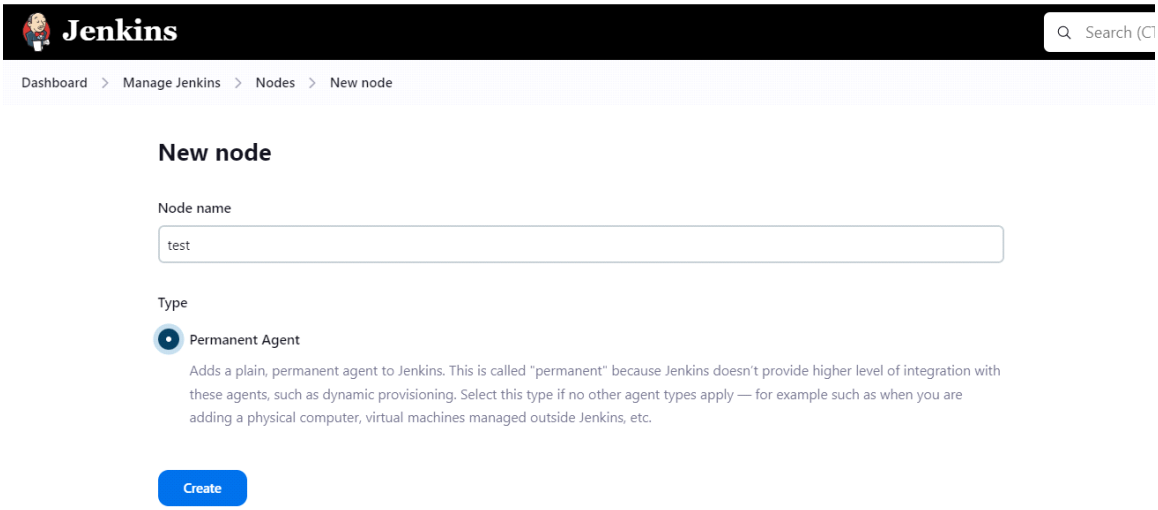


8. Configure Jenkins



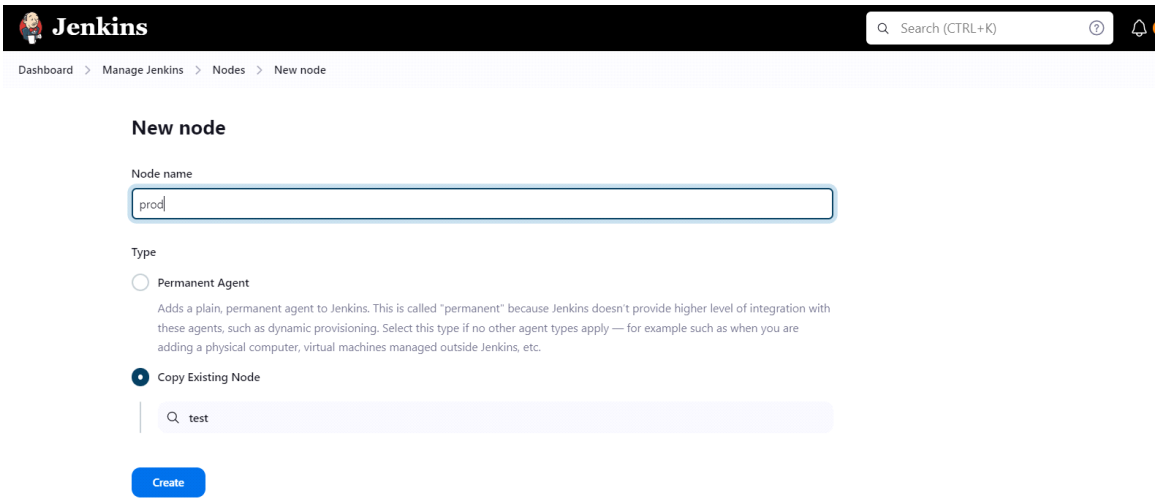


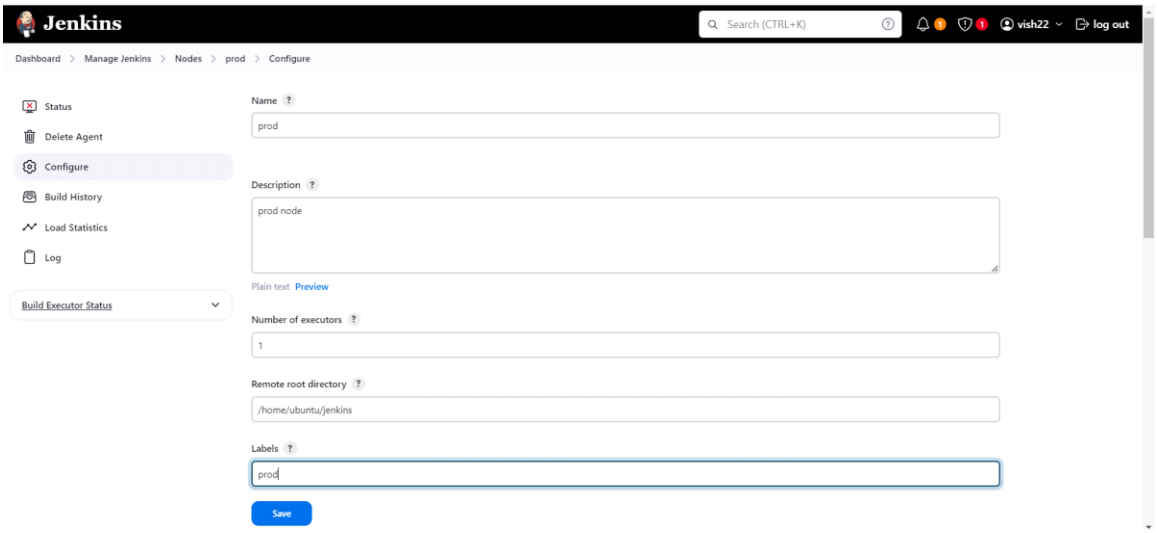
9. Created 2 node

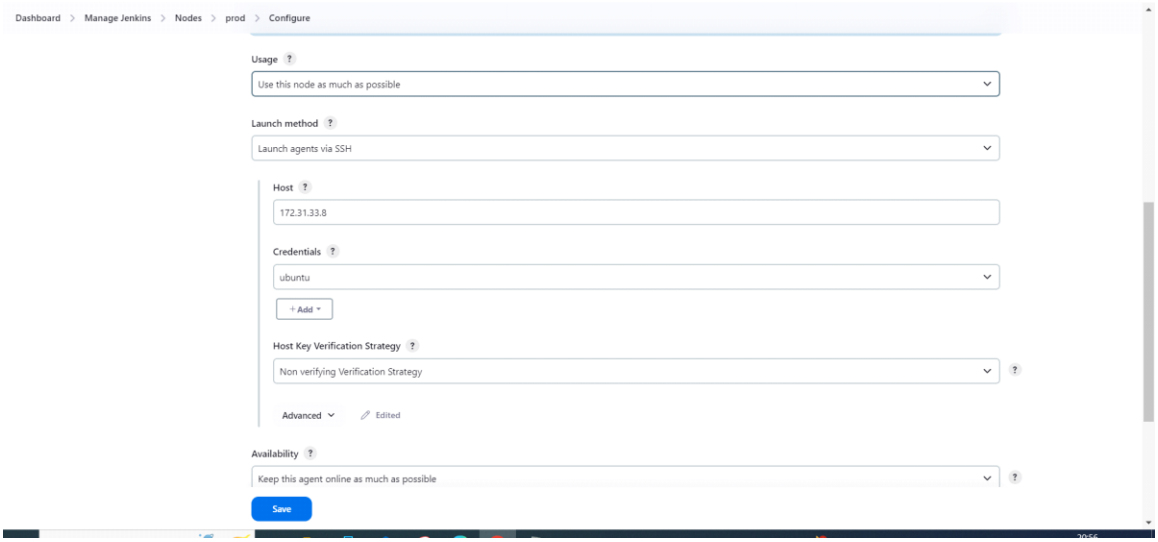

Jenkins
Search (CT
Dashboard > Manage Jenkins > Nodes > New node
New node
Node name
test
Type
Permanent Agent
Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.
Create

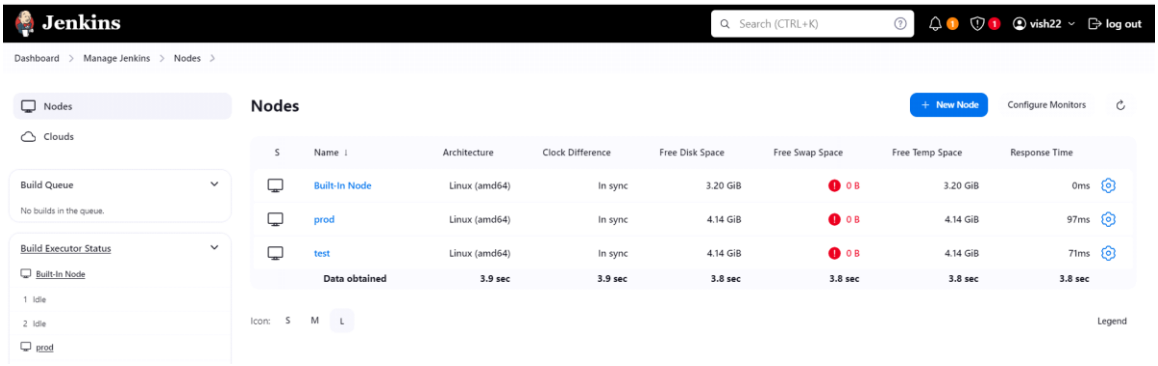

Jenkins
Search (CTRL+K)
vish22
log out
Dashboard > Manage Jenkins > Nodes >
Name
test
Description
test node
Plain text Preview
Number of executors
1
Remote root directory
/home/ubuntu/jenkins
Labels
test
Save


Dashboard > Manage Jenkins > Nodes >
Usage
Use this node as much as possible
Launch method
Launch agents via SSH
Host
172.31.41.148
Credentials
ubuntu
+ Add
Host Key Verification Strategy
Non verifying Verification Strategy
Advanced

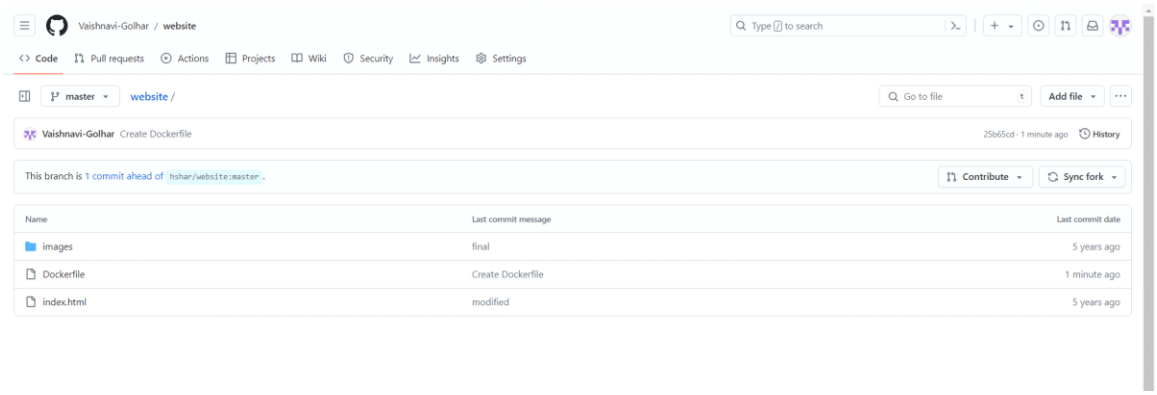

Jenkins
Search (CTRL+K)
Dashboard > Manage Jenkins > Nodes > New node
New node
Node name
prod
Type
Permanent Agent
Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.
Copy Existing Node
test
Create


Jenkins
Search (CTRL+K)
vish22
log out
Dashboard > Manage Jenkins > Nodes > prod > Configure
Status
Delete Agent
Configure
Build History
Load Statistics
Log
Build Executor Status
Name
prod
Description
prod node
Plain text Preview
Number of executors
1
Remote root directory
/home/ubuntu/jenkins
Labels
prod
Save





10. Set Up Git Workflow



Vaishnavi-Golhar / website

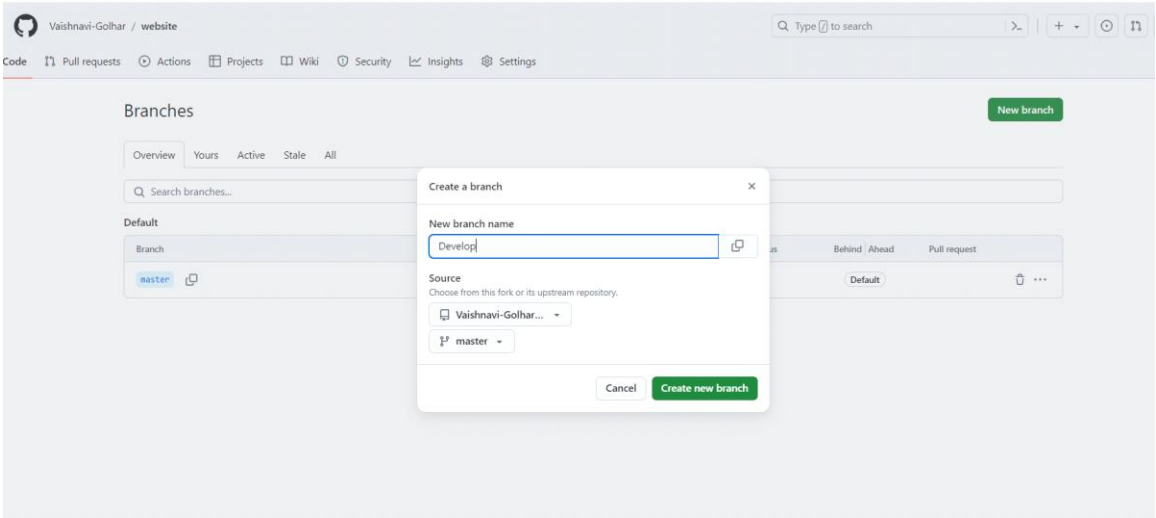
Code | Pull requests | Actions | Projects | Wiki | Security | Insights | Settings

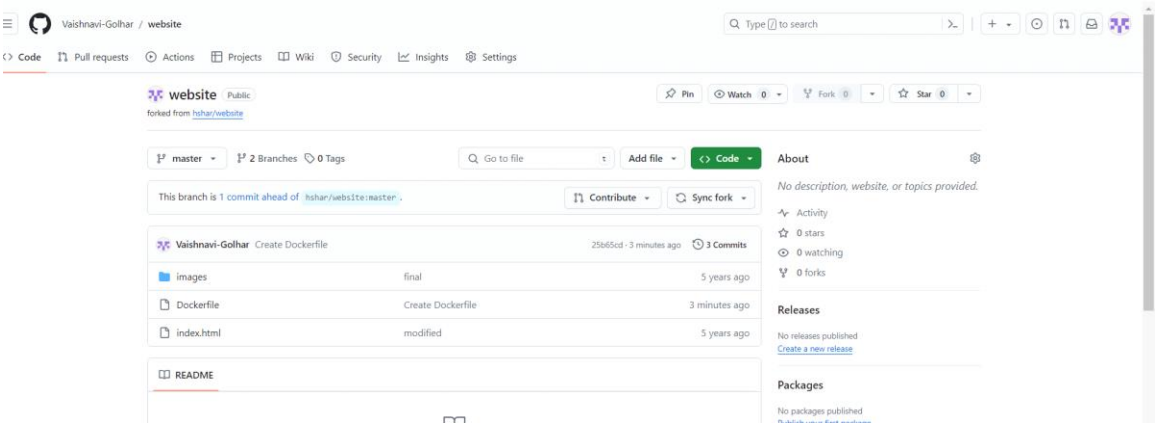
master | website / Dockerfile

Vaishnavi-Golhar Create Dockerfile

Code | Blame | 5 lines (5 loc) · 109 Bytes | Code 55% faster with GitHub Copilot

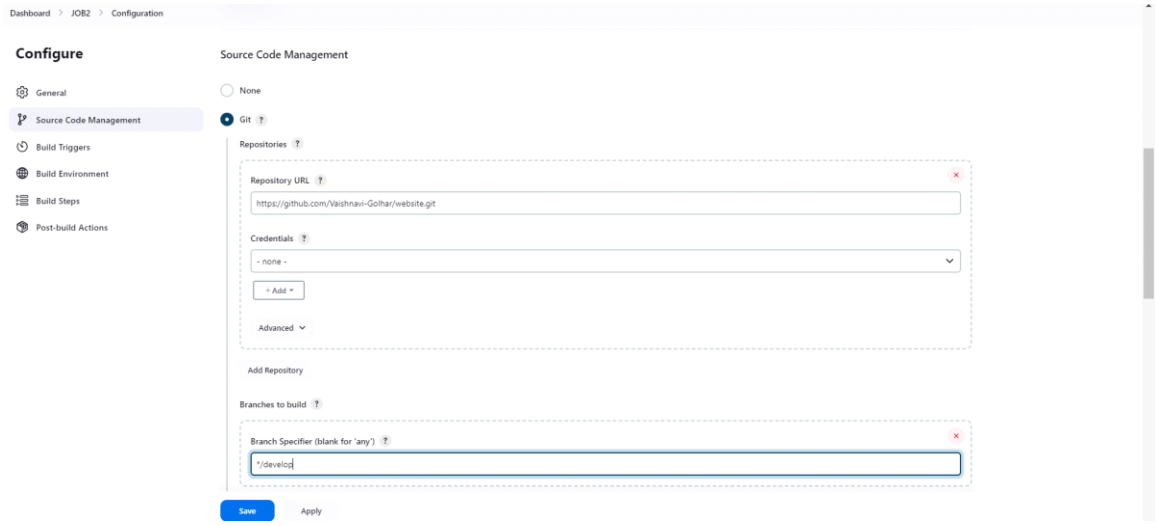
```
1  FROM ubuntu
2  RUN apt update
3  RUN apt install apache2 -y
4  ADD . /var/www/html
5  ENTRYPOINT apachectl -D FOREGROUND
```

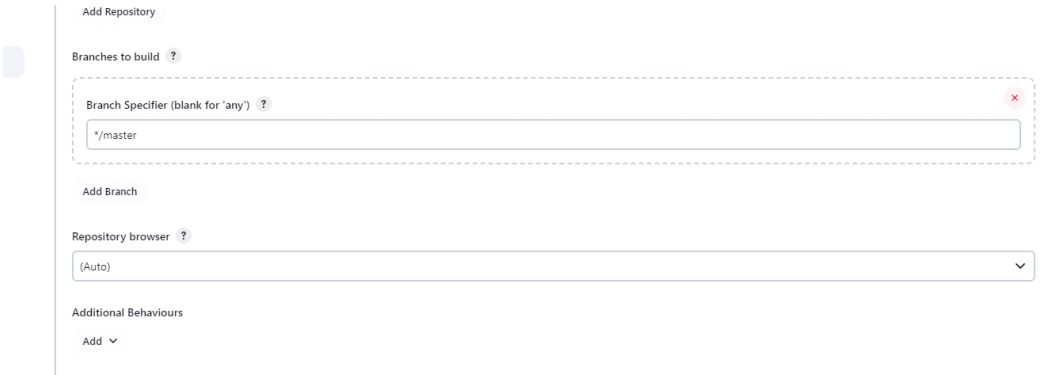


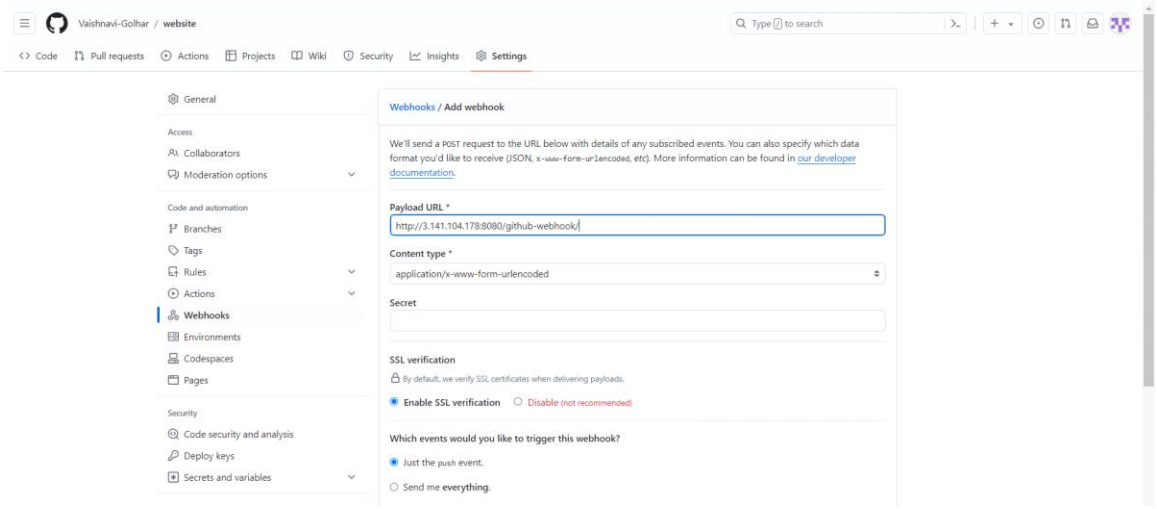


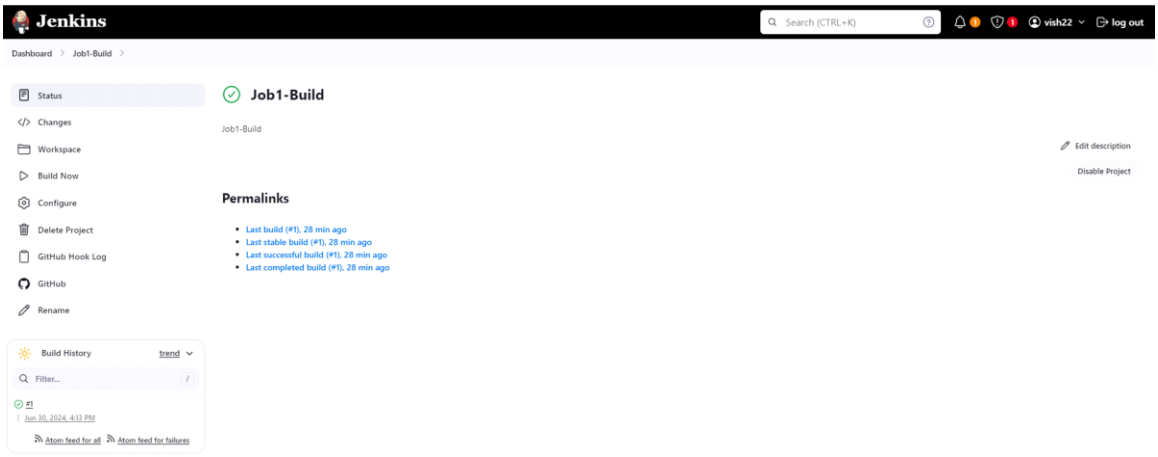
11. Jenkins Pipeline Configuration

Job1:

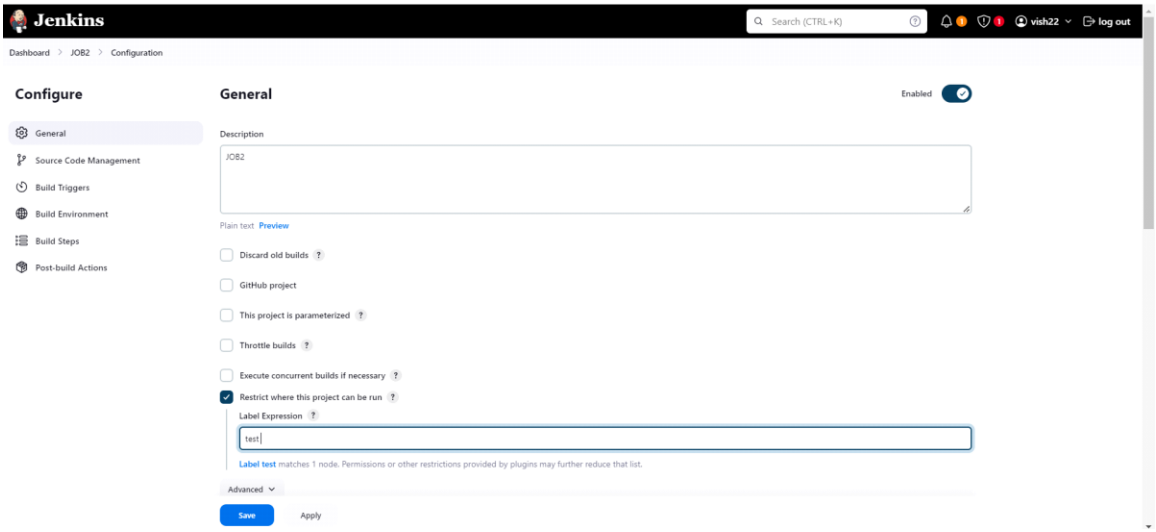




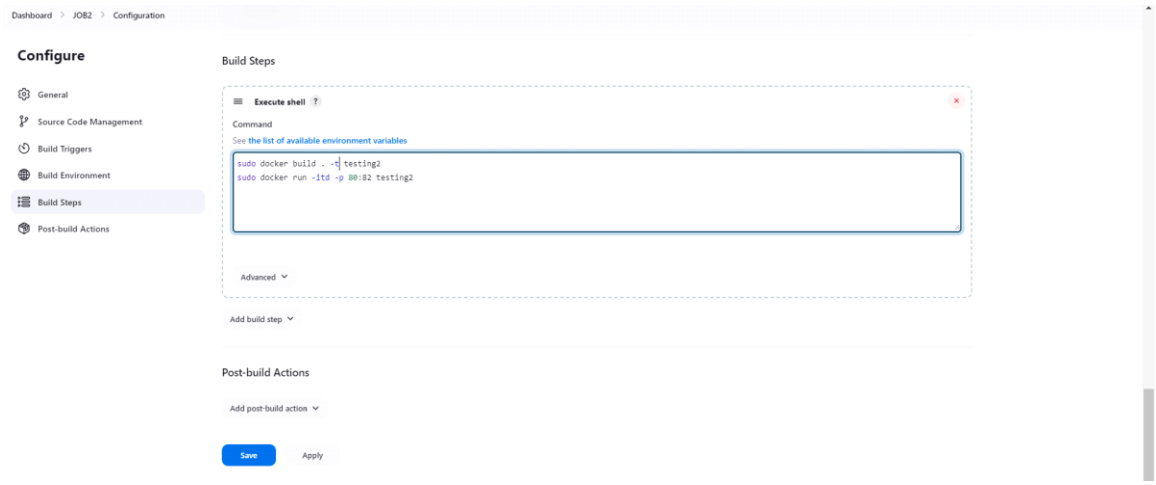

Vaishnavi-Golhar / website
Type / to search
Code
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
General
Access
Collaborators
Moderation options
Code and automation
Branches
Tags
Rules
Actions
Webhooks
Environments
Codespaces
Pages
Security
Code security and analysis
Deploy keys
Secrets and variables
Webhooks / Add webhook
We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our developer documentation.
Payload URL *
http://3.141.104.178:8080/github-webhook/
Content type *
application/x-www-form-urlencoded
Secret
SSL verification
By default, we verify SSL certificates when delivering payloads.
Enable SSL verification
Disable (not recommended)
Which events would you like to trigger this webhook?
Just the push event.
Send me everything.


Jenkins
Search (CTRL+K)
vish22
log out
Dashboard
Job1-Build
Status
Changes
Workspace
Build Now
Configure
Delete Project
GitHub Hook Log
GitHub
Rename
Job1-Build
Job1-Build
Edit description
Disable Project
Permalinks
Last build (#1), 28 min ago
Last stable build (#1), 28 min ago
Last successful build (#1), 28 min ago
Last completed build (#1), 28 min ago
Build History
trend
Filter...
#1
Jun 30, 2024, 4:13 PM
Atom feed for all
Atom feed for failures

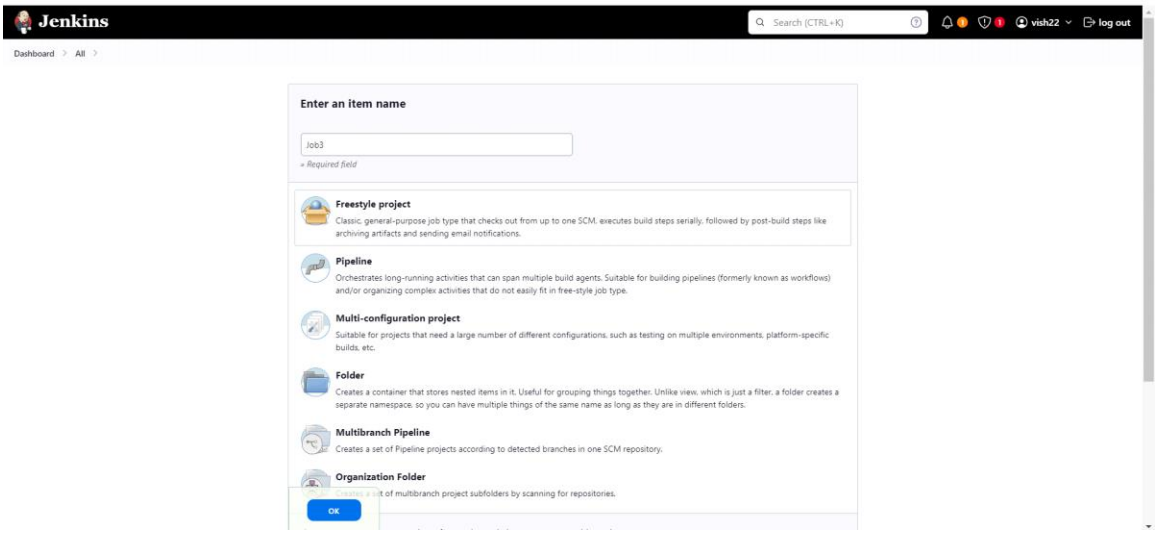
Job2:

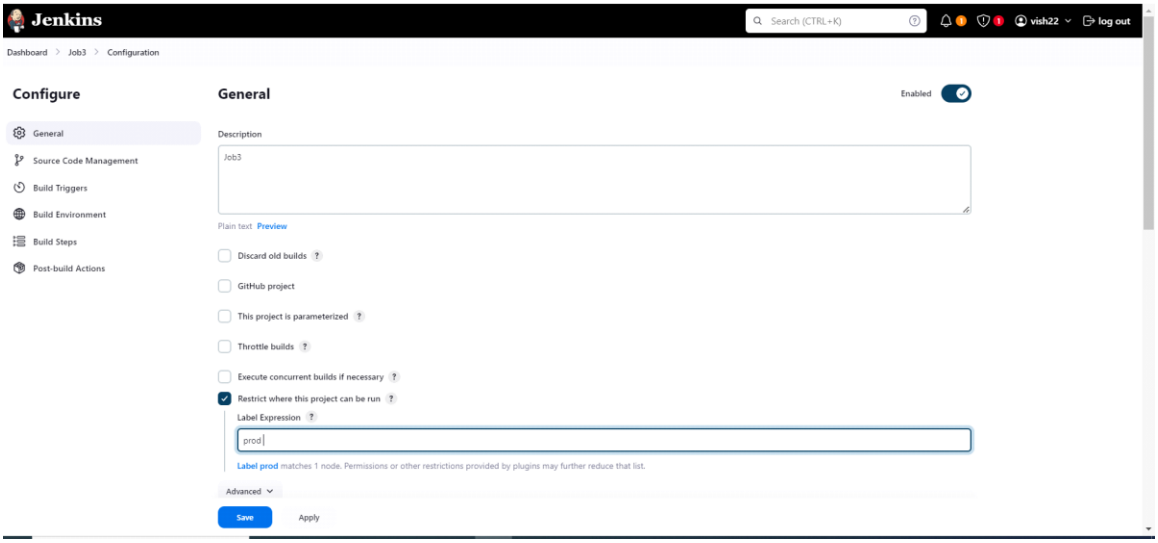

Jenkins
Search (CTRL+K)
vish22
log out
Dashboard > JOB2 > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
General
Enabled
Description
JOB2
Plain text Preview
Discard old builds
GitHub project
This project is parameterized
Throttle builds
Execute concurrent builds if necessary
Restrict where this project can be run
Label Expression
test
Label test matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.
Advanced
Save
Apply

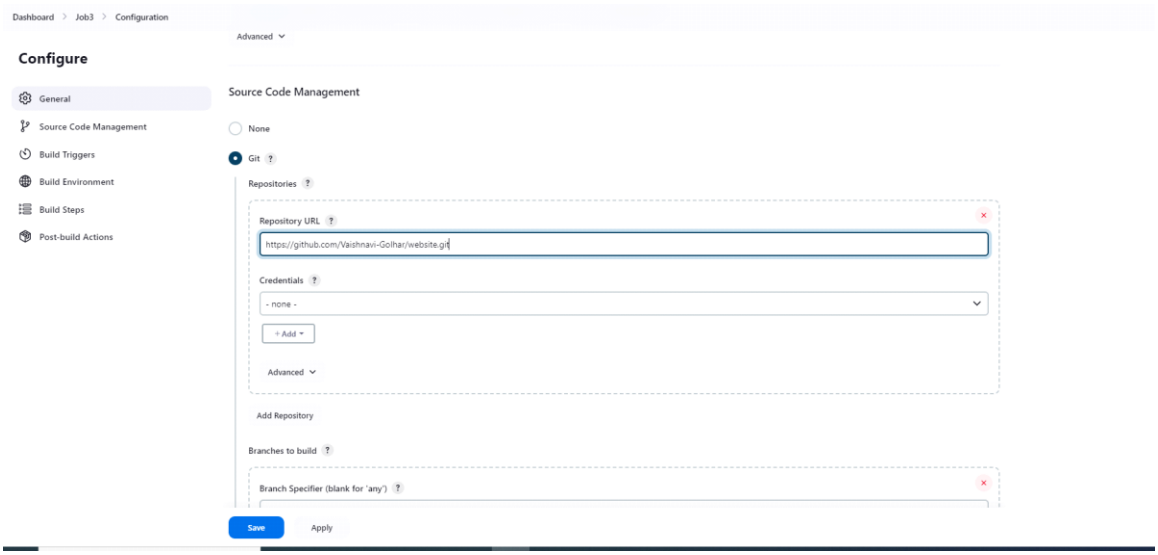

Dashboard > JOB2 > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Git
Repositories
Repository URL
https://github.com/Vaishnavi-Golhar/website.git
Credentials
- none -
+ Add
Advanced
Add Repository
Branches to build
Branch Specifier (blank for 'any')
*/develop
Add Branch

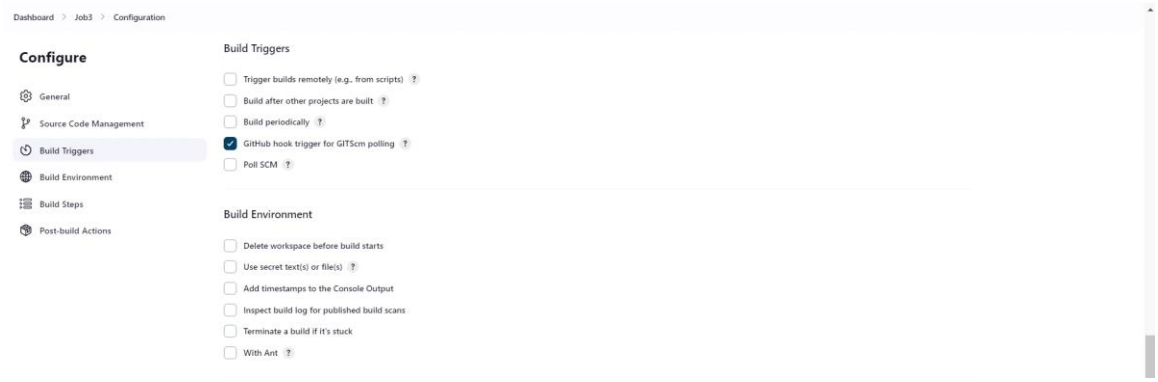

Dashboard > JOB2 > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Build Steps
Execute shell
Command
See the list of available environment variables
sudo docker build . -t testing2
sudo docker run -itd -p 80:82 testing2
Advanced
Add build step
Post-build Actions
Add post-build action
Save
Apply

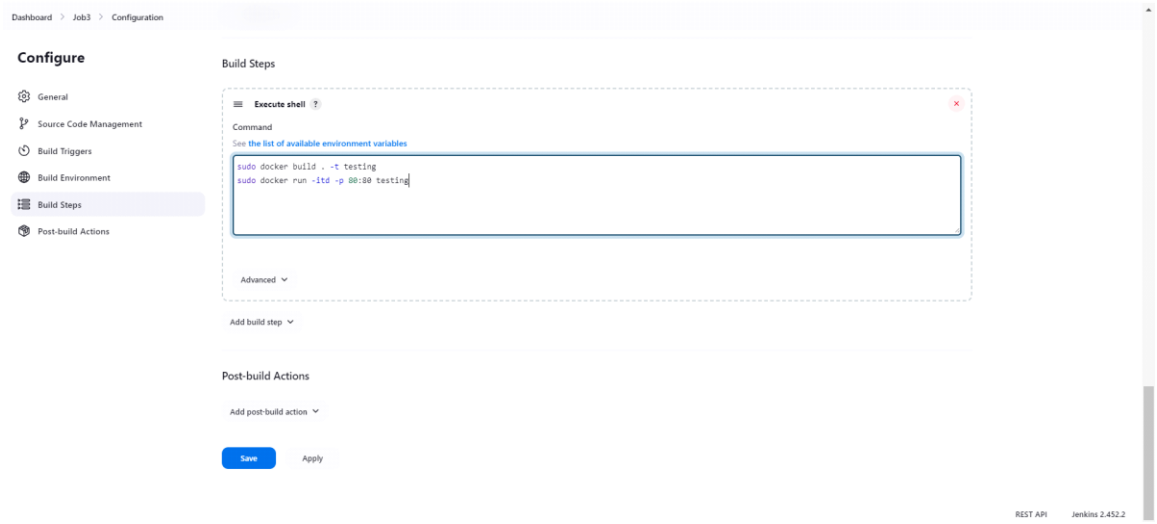
job3:

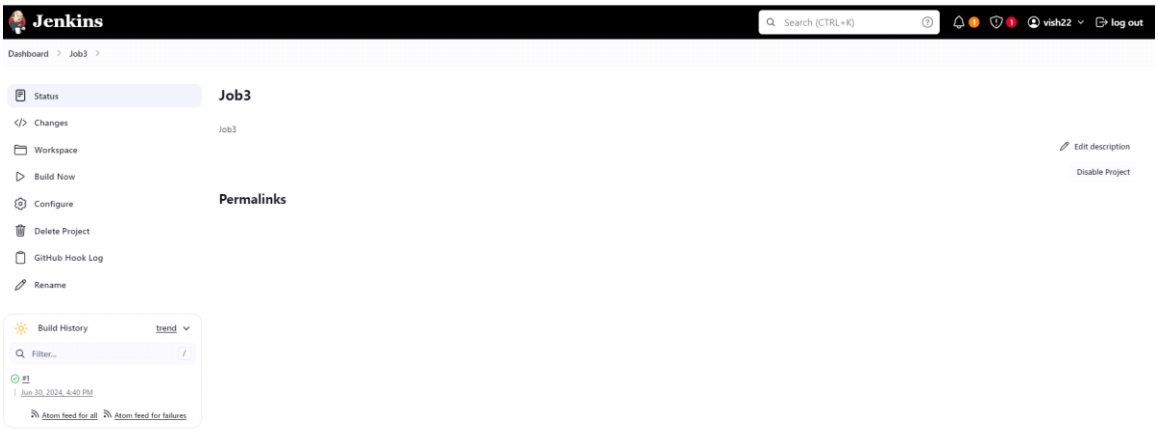

Jenkins
Search (CTRL+K)
vish22
log out
Dashboard
All
Enter an item name
Job3
» Required field
Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
Multibranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.
Organization Folder
OK

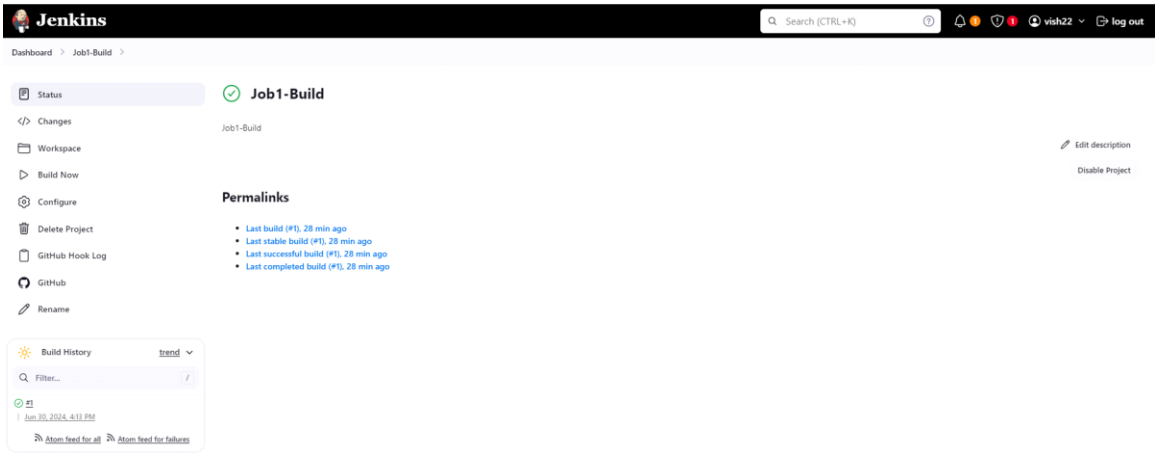

Jenkins
Search (CTRL+K)
vish22
log out
Dashboard
Job3
Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
General
Enabled
Description
Job3
Plain text
Preview
Discard old builds
GitHub project
This project is parameterized
Throttle builds
Execute concurrent builds if necessary
Restrict where this project can be run
Label Expression
prod
Label prod matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.
Advanced
Save
Apply


Dashboard > Job3 > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Advanced
Source Code Management
None
Git
Repositories
Repository URL
https://github.com/Vaishnavi-Golhar/website.git
Credentials
- none -
+ Add
Advanced
Add Repository
Branches to build
Branch Specifier (blank for 'any')
Save
Apply


Dashboard > Job3 > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Build Triggers
Trigger builds remotely (e.g., from scripts)
Build after other projects are built
Build periodically
GitHub hook trigger for GITScm polling
Poll SCM
Build Environment
Delete workspace before build starts
Use secret text(s) or file(s)
Add timestamps to the Console Output
Inspect build log for published build scans
Terminate a build if it's stuck
With Ant


Dashboard > Job3 > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Build Steps
Execute shell
Command
See the list of available environment variables
sudo docker build . -t testing
sudo docker run -itd -p 80:80 testing
Advanced
Add build step
Post-build Actions
Add post-build action
Save
Apply
REST API
Jenkins 2.452.2


Jenkins
Search (CTRL+K)
vish22
log out
Dashboard > Job3 >
Status
Changes
Workspace
Build Now
Configure
Delete Project
GitHub Hook Log
Rename
Build History
trend
Filter...
#1
Jun 30, 2024, 4:40 PM
Atom feed for all
Atom feed for failures
Job3
Job3
Edit description
Disable Project
Permalinks


Jenkins
Search (CTRL+K)
vish22
log out
Dashboard > Job1-Build >
Status
Changes
Workspace
Build Now
Configure
Delete Project
GitHub Hook Log
GitHub
Rename
Build History
trend
Filter...
#1
Jun 30, 2024, 4:13 PM
Atom feed for all
Atom feed for failures
Job1-Build
Job1-Build
Edit description
Disable Project
Permalinks
Last build (#1), 28 min ago
Last stable build (#1), 28 min ago
Last successful build (#1), 28 min ago
Last completed build (#1), 28 min ago