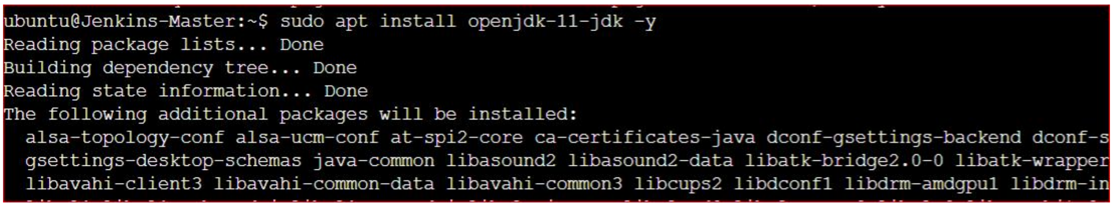
Module 6: Jenkins Assignment - 1

1. Trigger a pipeline using Git when push on develop branch.
2. Pipeline should pull Git content to a folder.

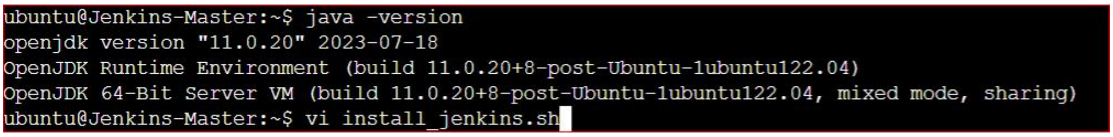
Solution:-

```
$ sudo apt install openjdk-11-jdk -y
```



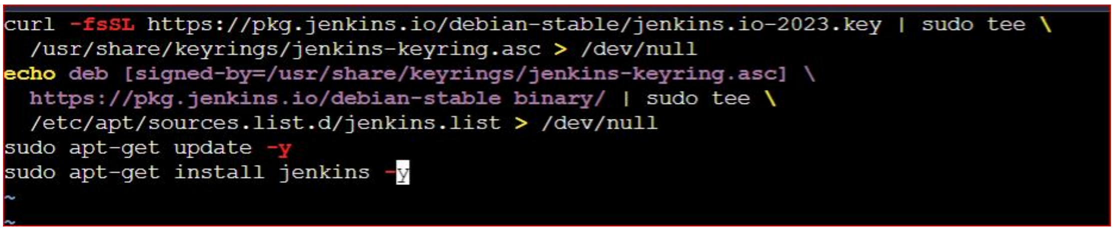
```
ubuntu@Jenkins-Master:~$ sudo apt install openjdk-11-jdk -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  alsa-topology-conf alsa-ucm-conf at-spi2-core ca-certificates-java dconf-gsettings-backend dconf-s
  gsettings-desktop-schemas java-common libasound2 libasound2-data libatk-bridge2.0-0 libatk-wrapper
  libavahi-client3 libavahi-common-data libavahi-common3 libcups2 libdconf1 libdrm-amdgpu1 libdrm-in
```

```
$ java -version
$ vi install_jenkins.sh
```



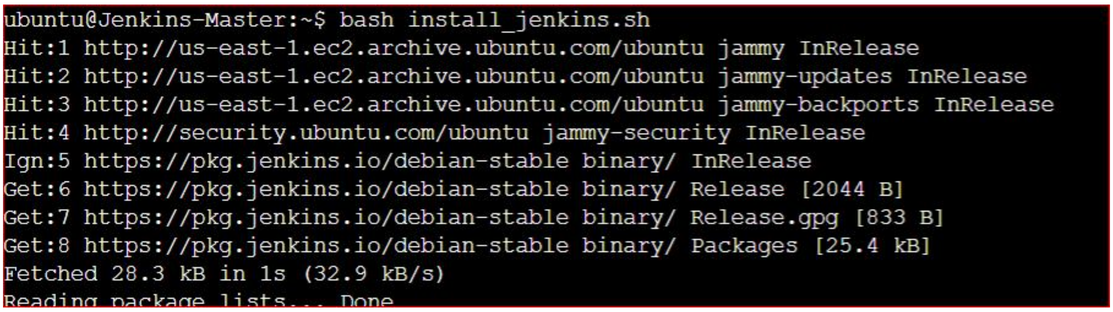
```
ubuntu@Jenkins-Master:~$ java -version
openjdk version "11.0.20" 2023-07-18
OpenJDK Runtime Environment (build 11.0.20+8-post-Ubuntu-1ubuntu122.04)
OpenJDK 64-Bit Server VM (build 11.0.20+8-post-Ubuntu-1ubuntu122.04, mixed mode, sharing)
ubuntu@Jenkins-Master:~$ vi install_jenkins.sh
```

```
curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee \
  /usr/share/keyrings/jenkins-keyring.asc > /dev/null
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 -y
sudo apt-get install jenkins -y
```



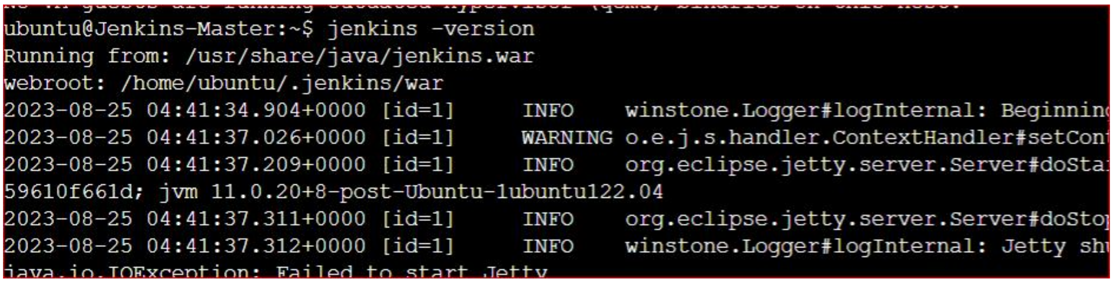
```
curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee \
  /usr/share/keyrings/jenkins-keyring.asc > /dev/null
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update -y
sudo apt-get install jenkins -y
~
~
```

$ bash install_jenkins.sh



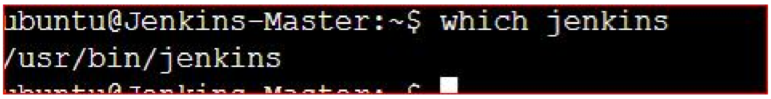
```
ubuntu@Jenkins-Master:~$ bash install_jenkins.sh
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign:5 https://pkg.jenkins.io/debian-stable binary/ InRelease
Get:6 https://pkg.jenkins.io/debian-stable binary/ Release [2044 B]
Get:7 https://pkg.jenkins.io/debian-stable binary/ Release.gpg [833 B]
Get:8 https://pkg.jenkins.io/debian-stable binary/ Packages [25.4 kB]
Fetched 28.3 kB in 1s (32.9 kB/s)
Reading package lists... Done
```

$ jenkins -version



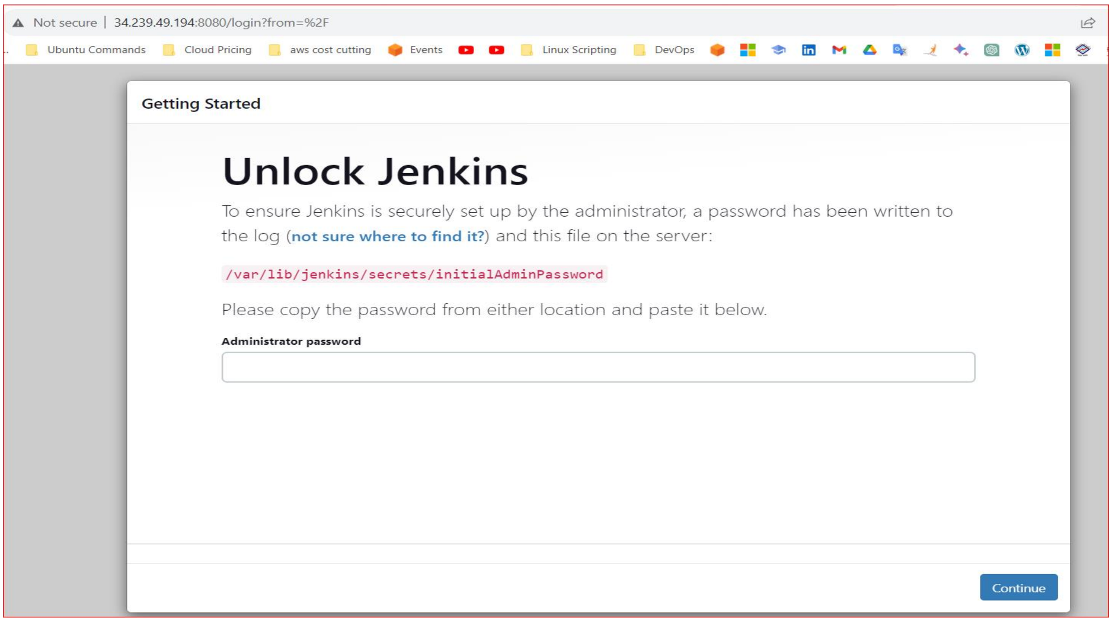
```
ubuntu@Jenkins-Master:~$ jenkins -version
Running from: /usr/share/java/jenkins.war
webroot: /home/ubuntu/.jenkins/war
2023-08-25 04:41:34.904+0000 [id=1]     INFO    winstone.Logger#logInternal: Beginnin
2023-08-25 04:41:37.026+0000 [id=1]     WARNING o.e.j.s.handler.ContextHandler#setCon
2023-08-25 04:41:37.209+0000 [id=1]     INFO    org.eclipse.jetty.server.Server#doSta
59610f661d; jvm 11.0.20+8-post-Ubuntu-1ubuntu122.04
2023-08-25 04:41:37.311+0000 [id=1]     INFO    org.eclipse.jetty.server.Server#doSto
2023-08-25 04:41:37.312+0000 [id=1]     INFO    winstone.Logger#logInternal: Jetty sh
java.io.IOException: Failed to start Jetty
```

$ which jenkins



```
ubuntu@Jenkins-Master:~$ which jenkins
/usr/bin/jenkins
ubuntu@Jenkins-Master:~$
```

* Browse jenkins using url :- http://34.239.49.194:8080



Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

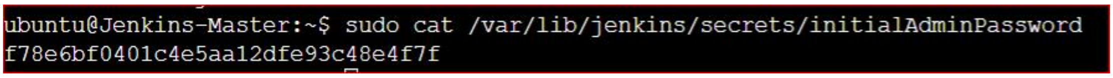
`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

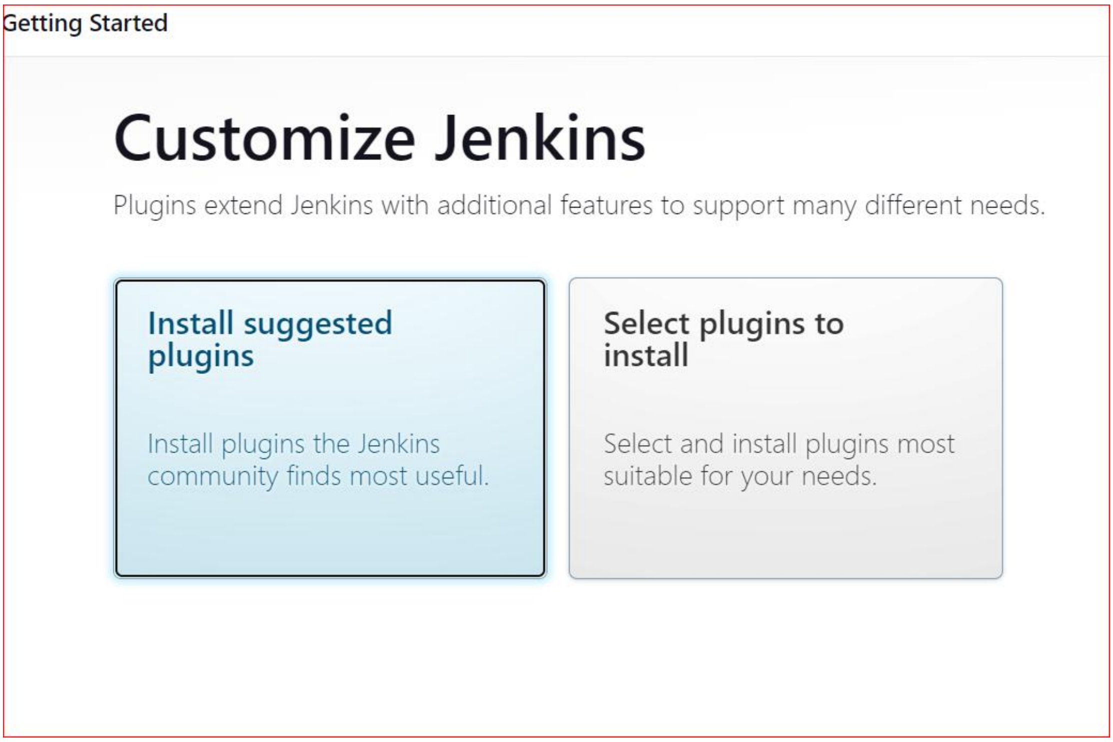
Continue

$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword



```
ubuntu@Jenkins-Master:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
f78e6bf0401c4e5aa12dfe93c48e4f7f
```

* Click on install suggested plugins.



Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

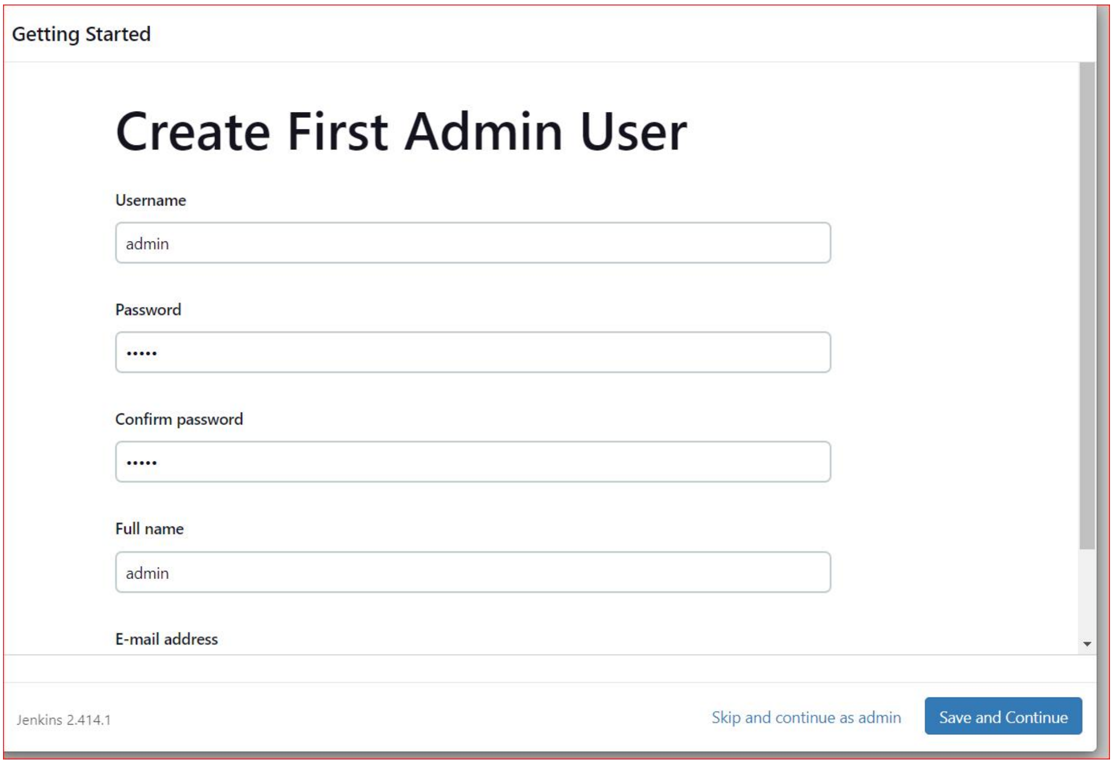
Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

* Create and insert user details:-



Getting Started

Create First Admin User

Username: admin

Password: •••••

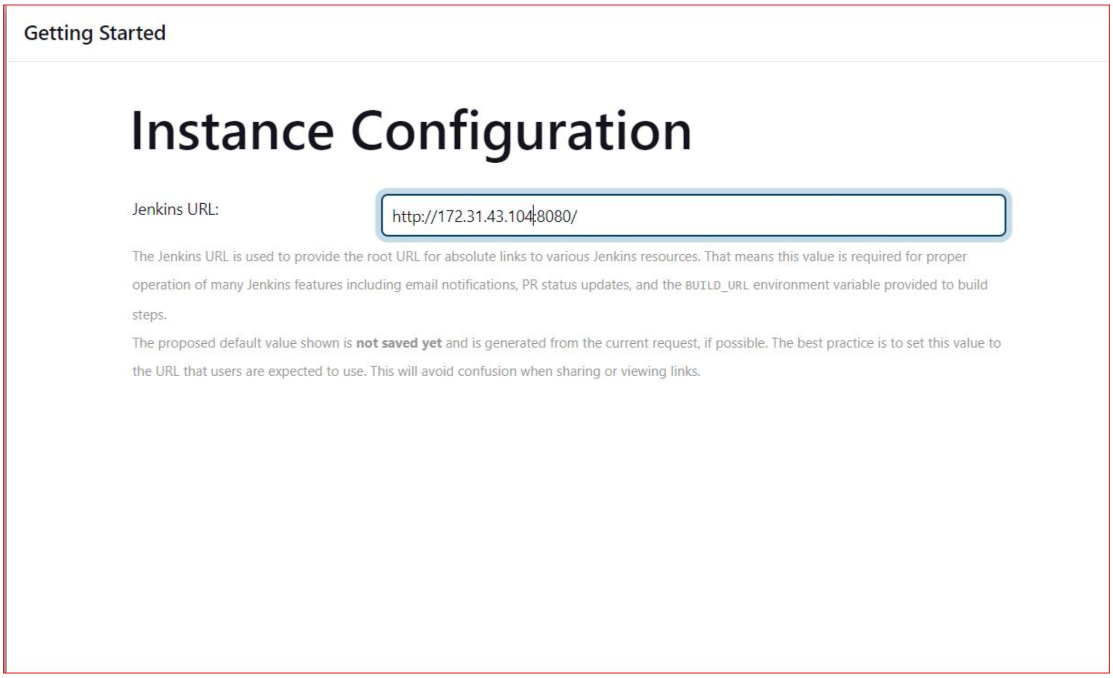
Confirm password: •••••

Full name: admin

E-mail address

Jenkins 2.414.1

* Replace public ip to private ip:-



Getting Started

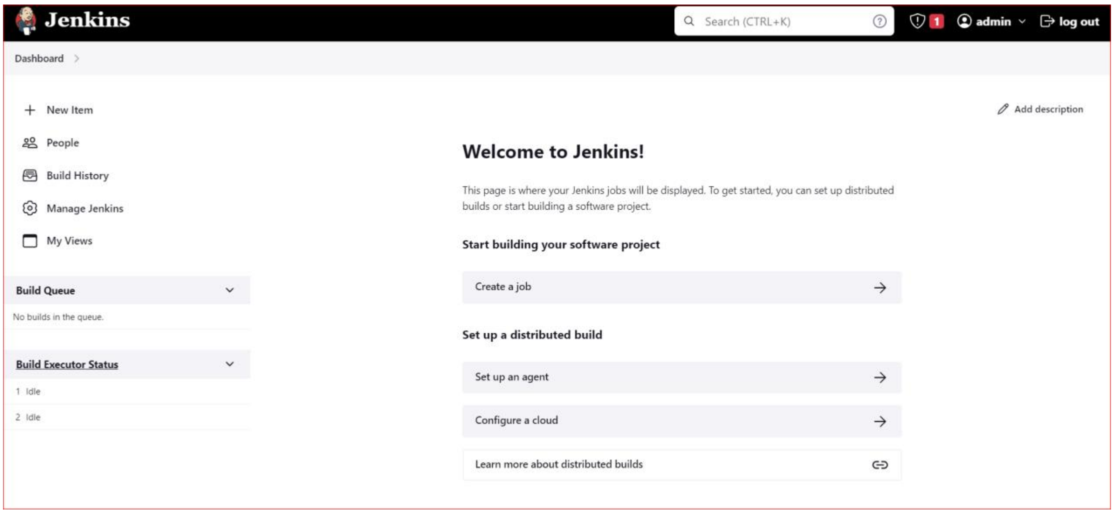
Instance Configuration

Jenkins URL: http://172.31.43.104:8080/

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD_URL environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

* Jenkins page now we can start using jenkins :-



Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job →

Set up a distributed build

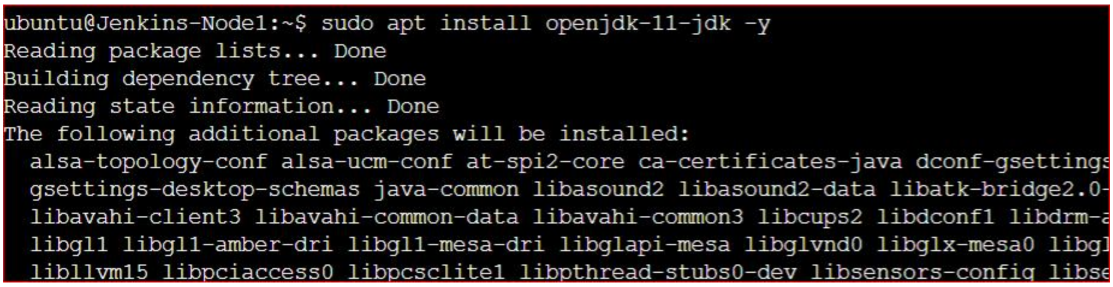
Set up an agent →

Configure a cloud →

Learn more about distributed builds

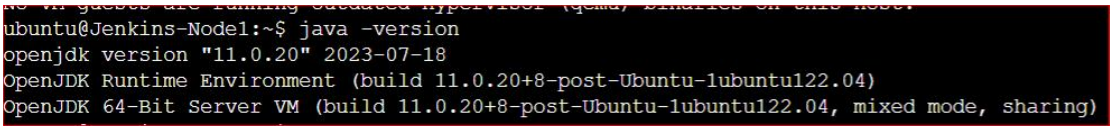
* In node-1:-

$ sudo apt install openjdk-11-jdk -y



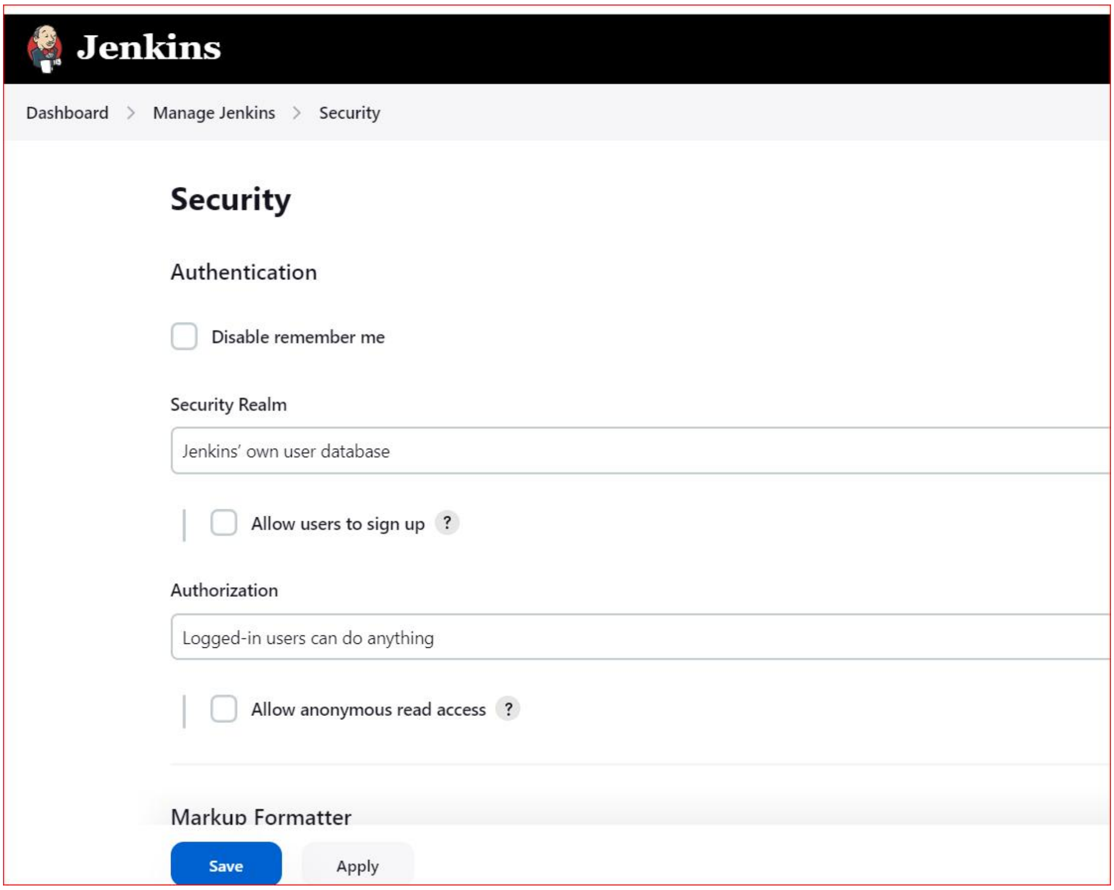
```
ubuntu@Jenkins-Node1:~$ sudo apt install openjdk-11-jdk -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  alsa-topology-conf alsa-ucm-conf at-spi2-core ca-certificates-java dconf-gsettings
  gsettings-desktop-schemas java-common libasound2 libasound2-data libatk-bridge2.0-
  libavahi-client3 libavahi-common-data libavahi-common3 libcups2 libdconf1 libdrm-a
  libgl1 libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libgl
  libllvm15 libpciaccess0 libpcsclite1 libpthread-stubs0-dev libsensors-config libse
```

$ java -version



```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@Jenkins-Node1:~$ java -version
openjdk version "11.0.20" 2023-07-18
OpenJDK Runtime Environment (build 11.0.20+8-post-Ubuntu-1ubuntu122.04)
OpenJDK 64-Bit Server VM (build 11.0.20+8-post-Ubuntu-1ubuntu122.04, mixed mode, sharing)
```

* Go to:- dashboard -> manage jenkins -> configure global security



Security

Authentication

□ Disable remember me

Security Realm

Jenkins' own user database

□ Allow users to sign up ?

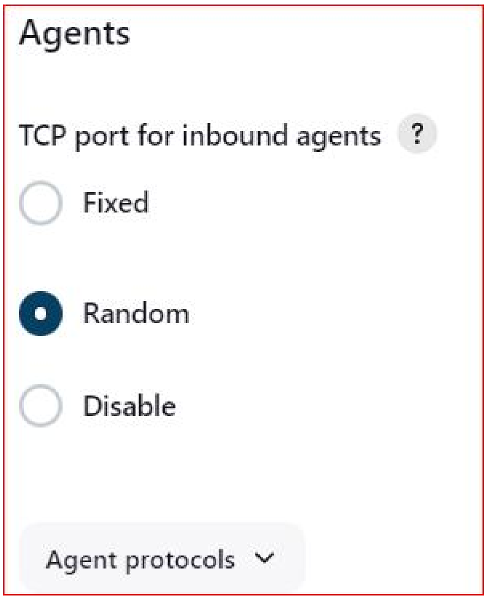
Authorization

Logged-in users can do anything

□ Allow anonymous read access ?

Markup Formatter

* Select:- tcp port for inbound agents -> rendom -> save



Agents

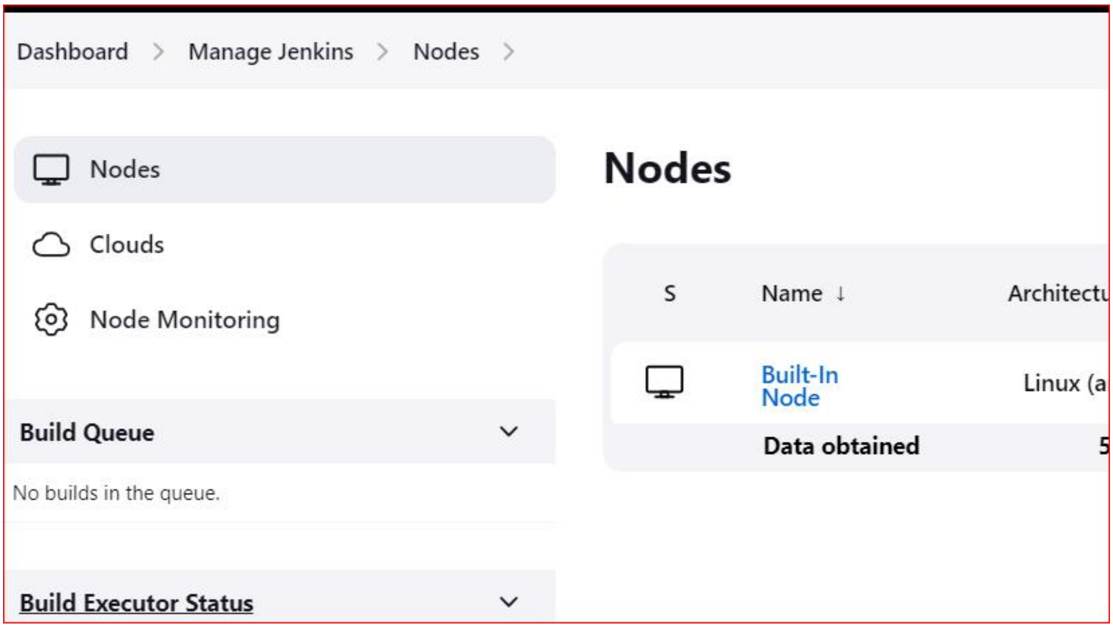
TCP port for inbound agents ?

○ Fixed

● Random

○ Disable

* Go to:- manage nodes and clouds -> new node

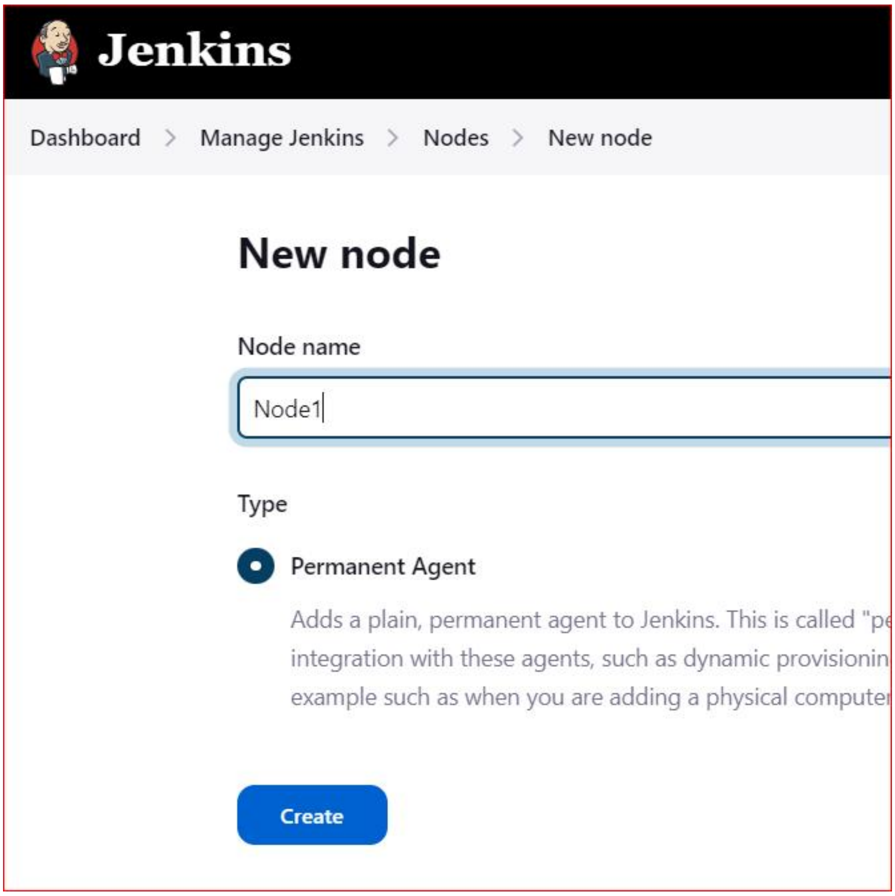


No builds in the queue.

Nodes

S	Name ↓	Architectu
	Built-In Node	Linux (a
	Data obtained	5

* Select:- new node -> name -> permanent agent



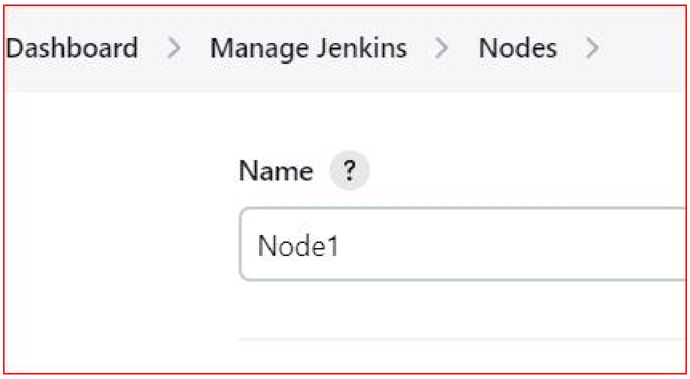
New node

Node name: Node1

Type

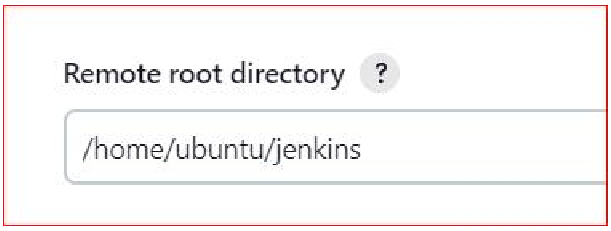
● Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "pe
integration with these agents, such as dynamic provisionin
example such as when you are adding a physical compute



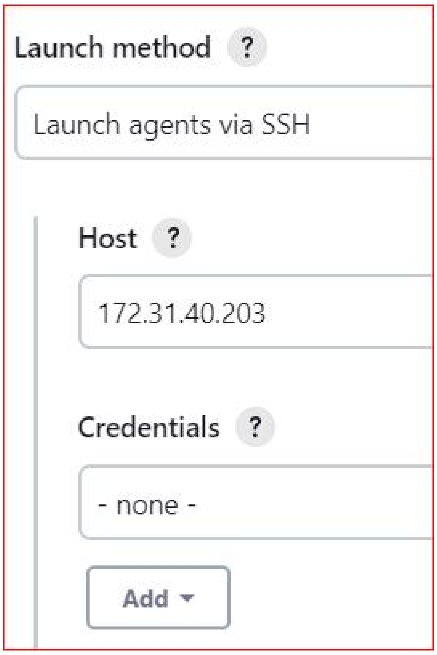
Name: Node1

* Go to:- create -> remote root directory -> /home/ubuntu/jenkins



Remote root directory: /home/ubuntu/jenkins

* Select:- launch method -> launch agents by a ssh -> host -> paste private ip of node not server



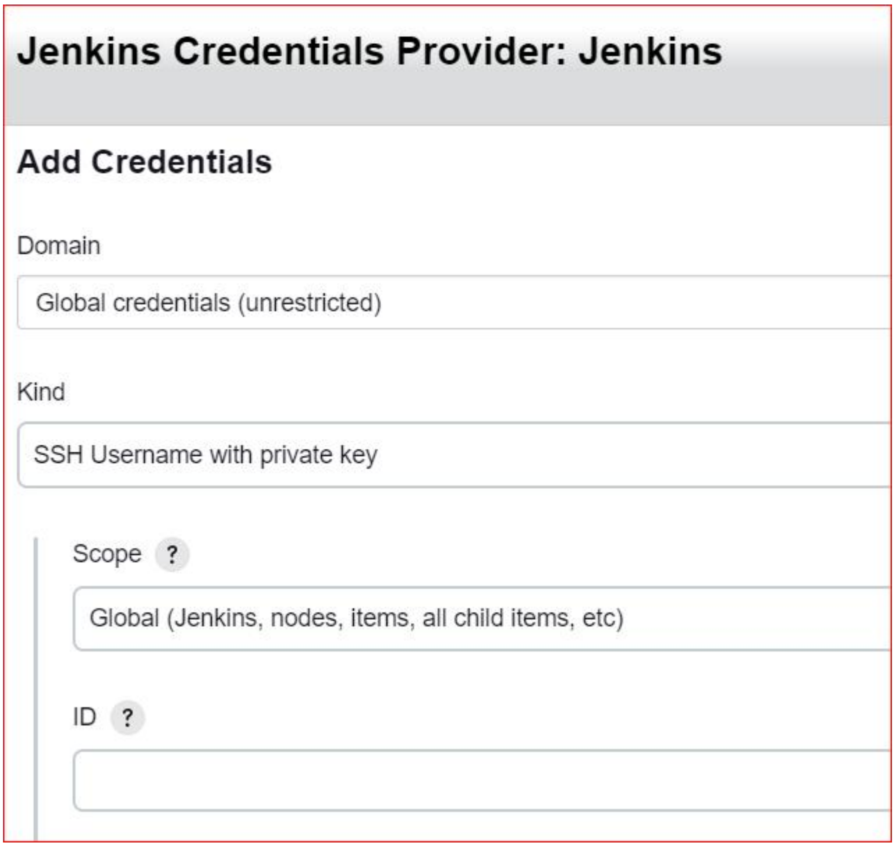
Launch method: Launch agents via SSH

Host: 172.31.40.203

Credentials: - none -

* Select:- credential -> add -> domain-> global credential (unrestricted) -> kind -> ssh username with private key ->

scope -> global (jenkins, nodes, items, all child items, etc.)



Jenkins Credentials Provider: Jenkins

Add Credentials

Domain

Global credentials (unrestricted)

Kind

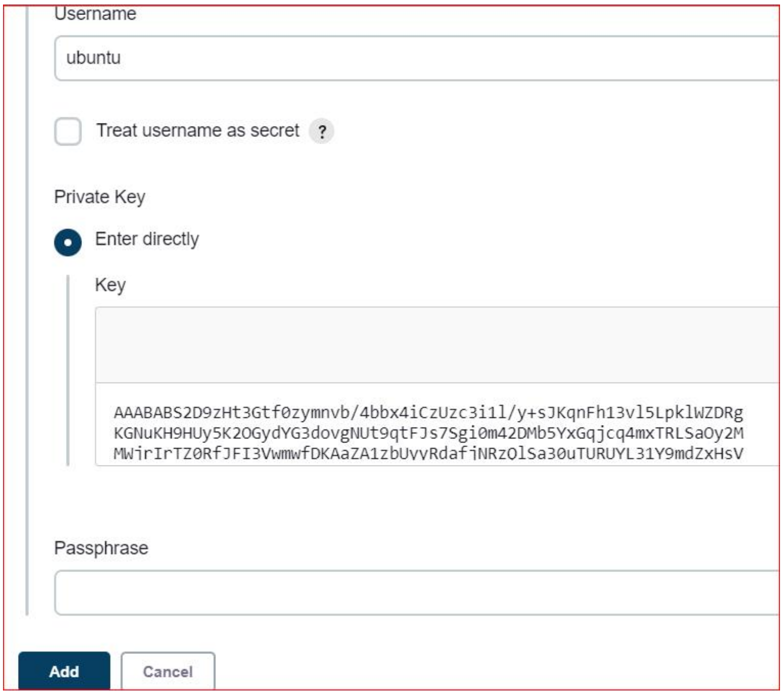
SSH Username with private key

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

ID ?

* Insert:- username -> Jenkins-Node -> private key -> enter directly -> copy and paste key of node



Username

ubuntu

□ Treat username as secret ?

Private Key

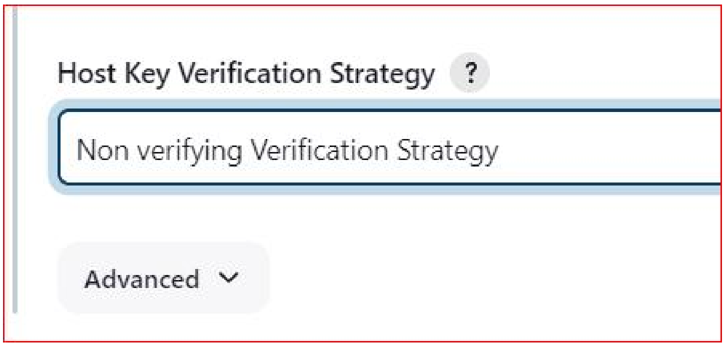
● Enter directly

Key

```
AAABABS2D9zHt3Gtf0zymnvb/4bbx4iCzUzc3i1l/y+sJKqnFh13vl5LpklWZDRg
KGNuKH9HUy5K2OGydYG3dovgNUt9qtFJs7Sgi0m42DMb5YxGqjcq4mxTRLSaOy2M
MWjrIrTZ0RfJFI3VwmwfDKAaZA1zbUyyRdafjNRzOlSa30uTURUYL31Y9mdZxHsV
```

Passphrase

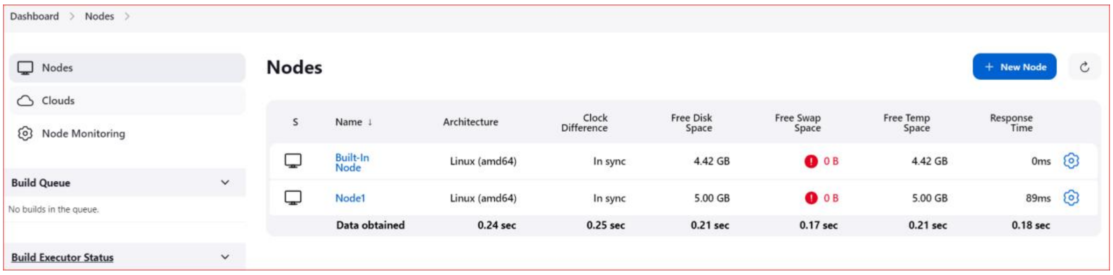
* Select:- host key verification stretagy -> non verifying verification staretegy -> Save



Host Key Verification Strategy

Non verifying Verification Strategy

* refresh status

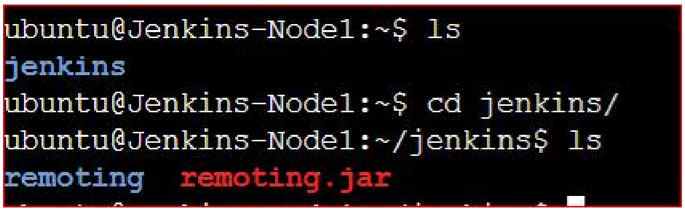


No builds in the queue.

Nodes

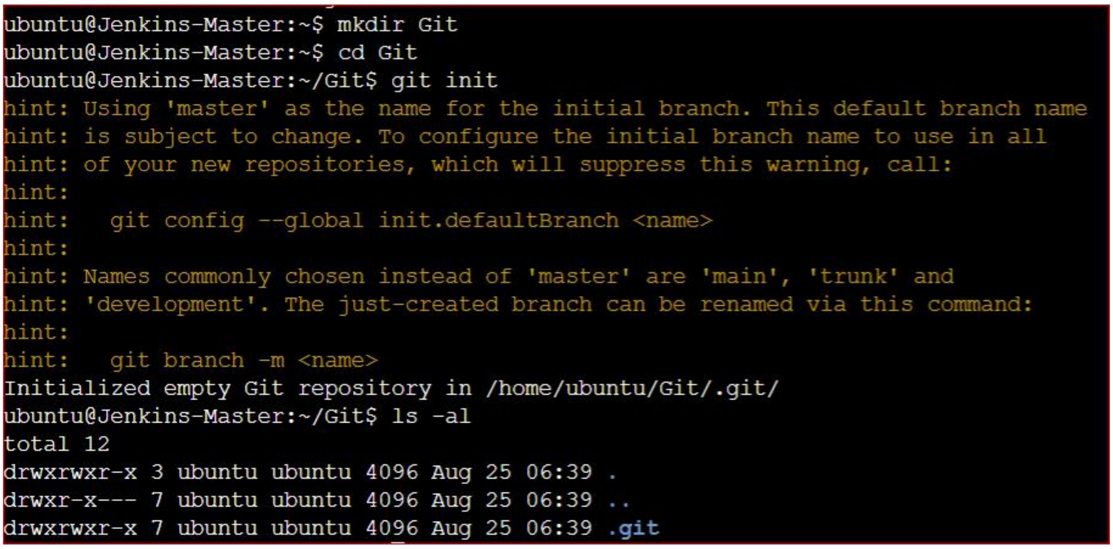
S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
🖥	Built-In Node	Linux (amd64)	In sync	4.42 GB	! 0 B	4.42 GB	0ms
🖥	Node1	Linux (amd64)	In sync	5.00 GB	! 0 B	5.00 GB	89ms
	Data obtained	0.24 sec	0.25 sec	0.21 sec	0.17 sec	0.21 sec	0.18 sec

```
$ ls
$ cd jenkins
$ ls
```



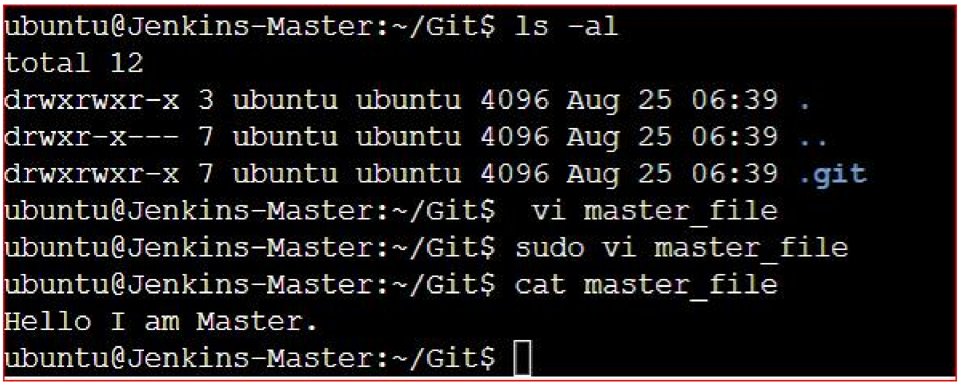
```
ubuntu@Jenkins-Node1:~$ ls
jenkins
ubuntu@Jenkins-Node1:~$ cd jenkins/
ubuntu@Jenkins-Node1:~/jenkins$ ls
remoting  remoting.jar
```

```
$ mkdir Git
$ cd Git
$ git init
$ ls -al
```



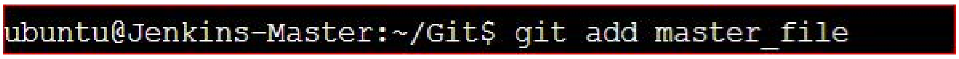
```
ubuntu@Jenkins-Master:~$ mkdir Git
ubuntu@Jenkins-Master:~$ cd Git
ubuntu@Jenkins-Master:~/Git$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ubuntu/Git/.git/
ubuntu@Jenkins-Master:~/Git$ ls -al
total 12
drwxrwxr-x 3 ubuntu ubuntu 4096 Aug 25 06:39 .
drwxr-x--- 7 ubuntu ubuntu 4096 Aug 25 06:39 ..
drwxrwxr-x 7 ubuntu ubuntu 4096 Aug 25 06:39 .git
```

$ ls -al
$ sudo vi master_file
$ cat master_file



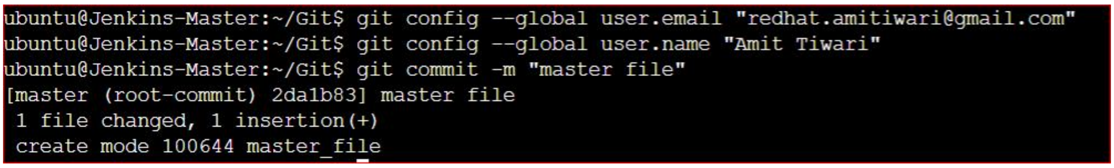
```
ubuntu@Jenkins-Master:~/Git$ ls -al
total 12
drwxrwxr-x 3 ubuntu ubuntu 4096 Aug 25 06:39 .
drwxr-x--- 7 ubuntu ubuntu 4096 Aug 25 06:39 ..
drwxrwxr-x 7 ubuntu ubuntu 4096 Aug 25 06:39 .git
ubuntu@Jenkins-Master:~/Git$  vi master_file
ubuntu@Jenkins-Master:~/Git$ sudo vi master_file
ubuntu@Jenkins-Master:~/Git$ cat master_file
Hello I am Master.
ubuntu@Jenkins-Master:~/Git$ 
```

$ git add master_file



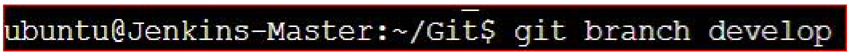
```
ubuntu@Jenkins-Master:~/Git$ git add master_file
```

$ git config --global user.email
“redhat.amitiwari@gmail.com”
$ git config --global user.name “Amit Tiwari”
$ git commit -m “master file”



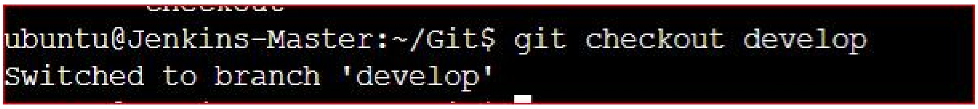
```
ubuntu@Jenkins-Master:~/Git$ git config --global user.email "redhat.amitiwari@gmail.com"
ubuntu@Jenkins-Master:~/Git$ git config --global user.name "Amit Tiwari"
ubuntu@Jenkins-Master:~/Git$ git commit -m "master file"
[master (root-commit) 2da1b83] master file
 1 file changed, 1 insertion(+)
 create mode 100644 master_file
```

```
$ git branch develop
```



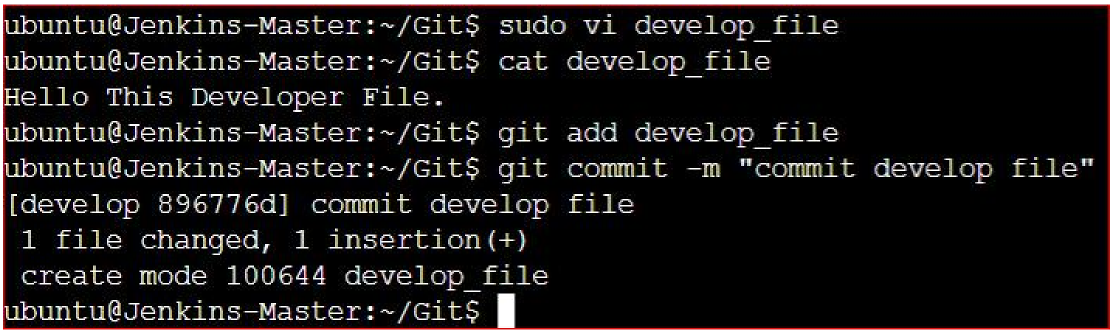
```
ubuntu@Jenkins-Master:~/Git$ git branch develop
```

```
$ git checkout develop
```



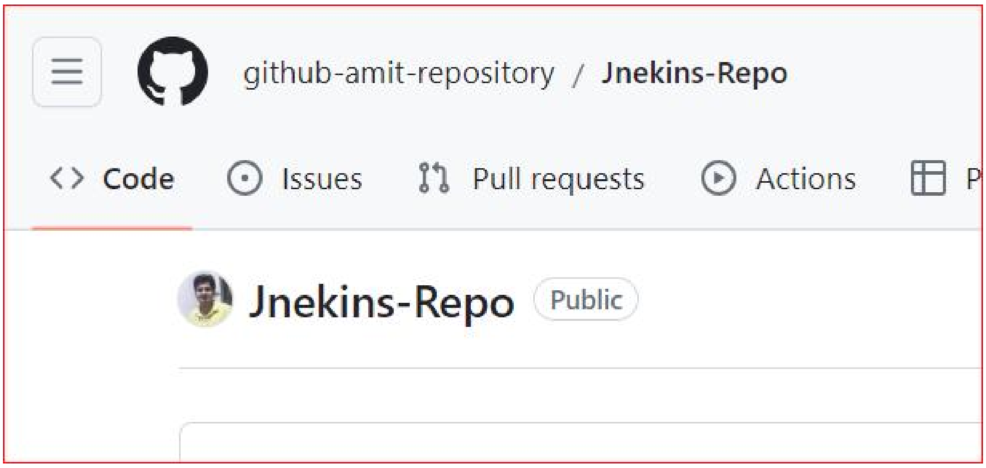
```
ubuntu@Jenkins-Master:~/Git$ git checkout develop
Switched to branch 'develop'
```

```
$ sudo vi develop_file
$ cat develop_file
$ git add develop_file
$ git commit -m “commit develop file”
```

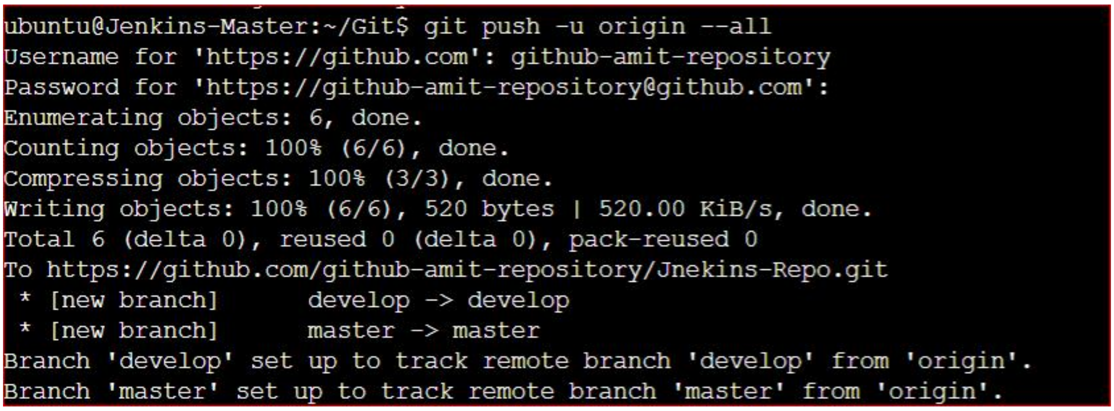


```
ubuntu@Jenkins-Master:~/Git$ sudo vi develop_file
ubuntu@Jenkins-Master:~/Git$ cat develop_file
Hello This Developer File.
ubuntu@Jenkins-Master:~/Git$ git add develop_file
ubuntu@Jenkins-Master:~/Git$ git commit -m "commit develop file"
[develop 896776d] commit develop file
 1 file changed, 1 insertion(+)
 create mode 100644 develop_file
ubuntu@Jenkins-Master:~/Git$
```

* We can see remote repo is created in github:-

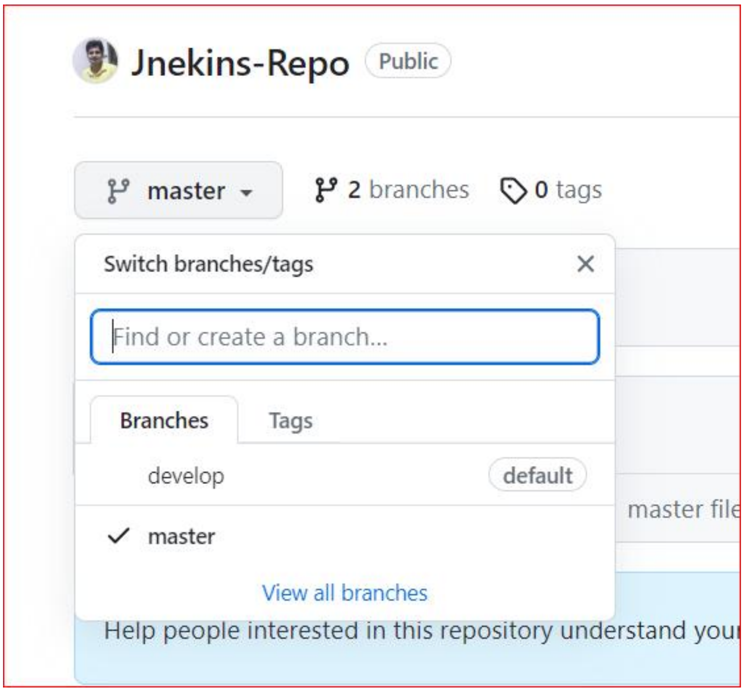


$ git push -u origin --all
* Enter github username and password.

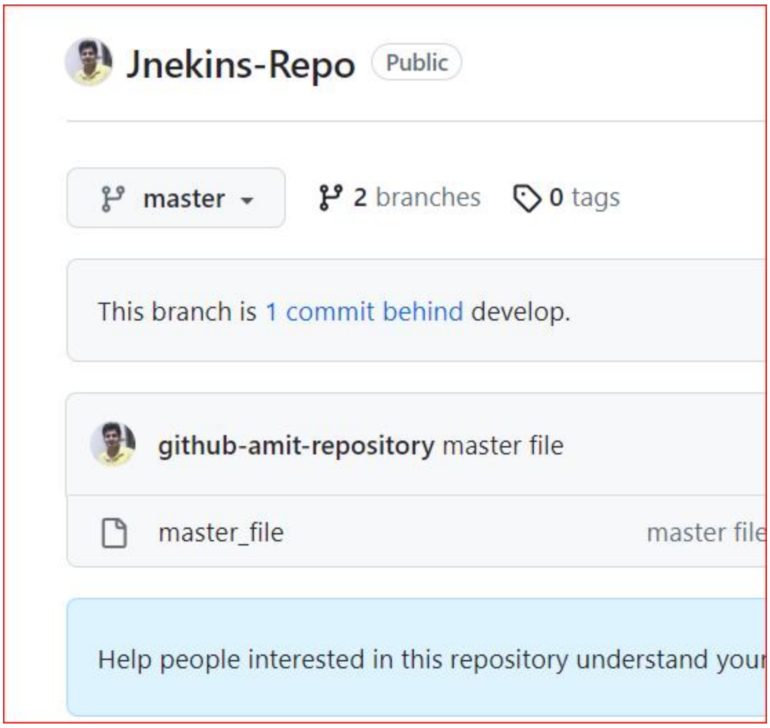


```
ubuntu@Jenkins-Master:~/Git$ git push -u origin --all
Username for 'https://github.com': github-amit-repository
Password for 'https://github-amit-repository@github.com':
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (6/6), 520 bytes | 520.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/github-amit-repository/Jnekins-Repo.git
 * [new branch]      develop -> develop
 * [new branch]      master -> master
Branch 'develop' set up to track remote branch 'develop' from 'origin'.
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

* Github repo of Jenkins-Repo name:-



Help people interested in this repository understand your



Jnekins-Repo Public

master 2 branches 0 tags

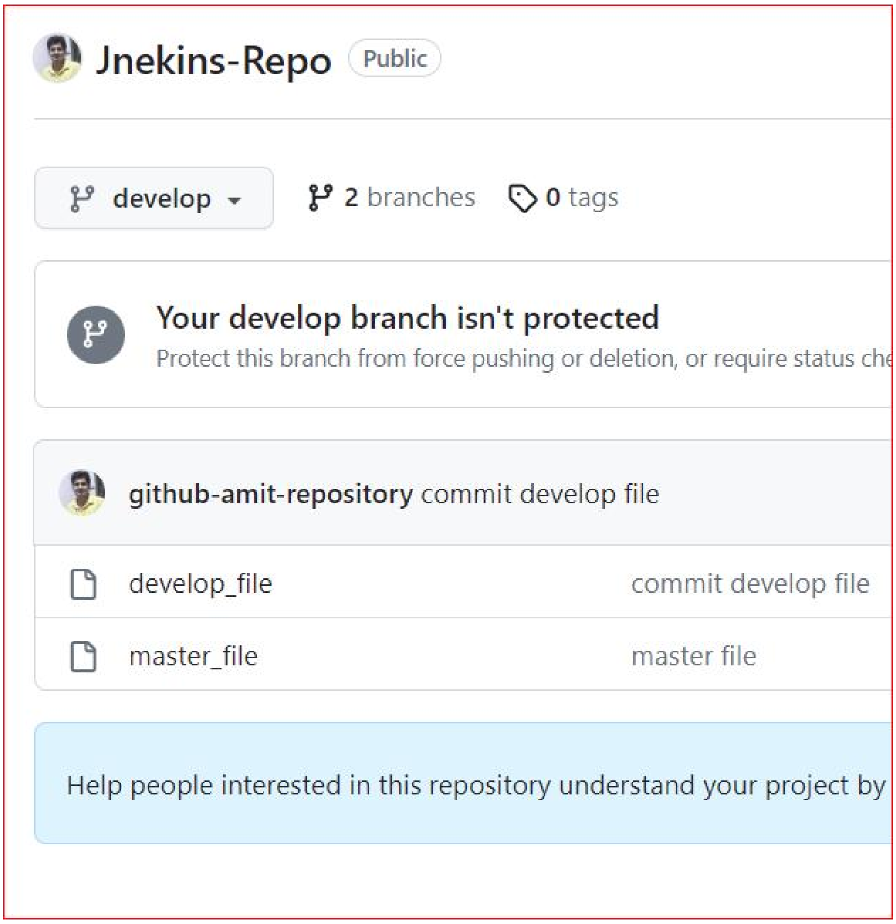
This branch is 1 commit behind develop.

github-amit-repository master file

master_file	master file

Help people interested in this repository understand your

* Now we can see files in the repo:-



Jnekins-Repo Public

develop 2 branches 0 tags

Your develop branch isn't protected

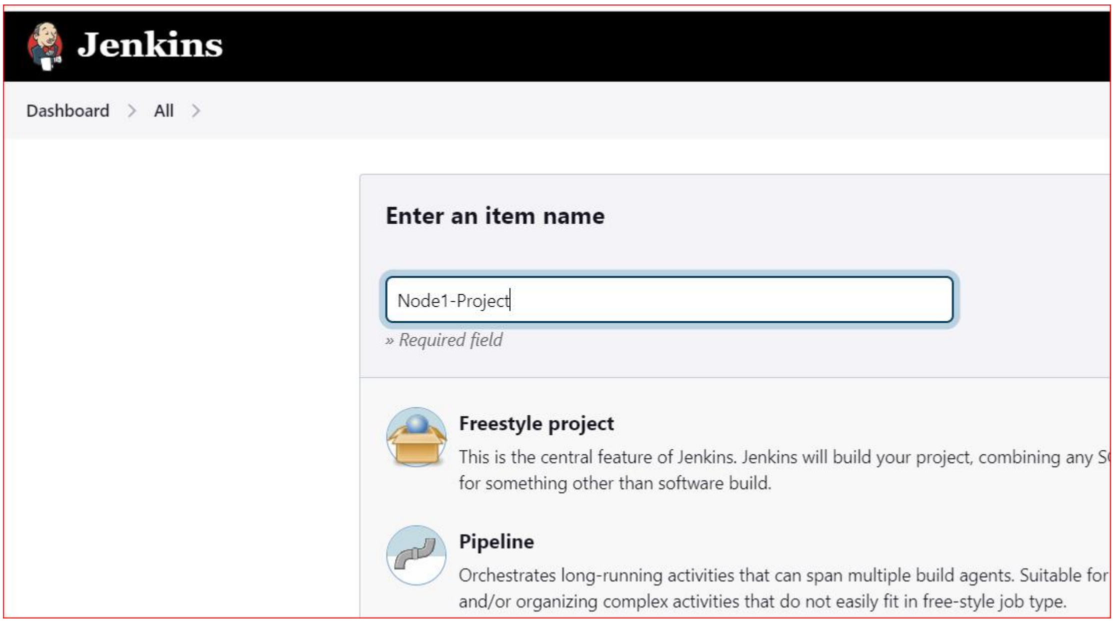
Protect this branch from force pushing or deletion, or require status che

github-amit-repository commit develop file

develop_file	commit develop file
master_file	master file

Help people interested in this repository understand your project by

* Now in jenkins:- new items -> name -> free style project -> ok



Jenkins

Dashboard > All >

Enter an item name

Node1-Project

» Required field

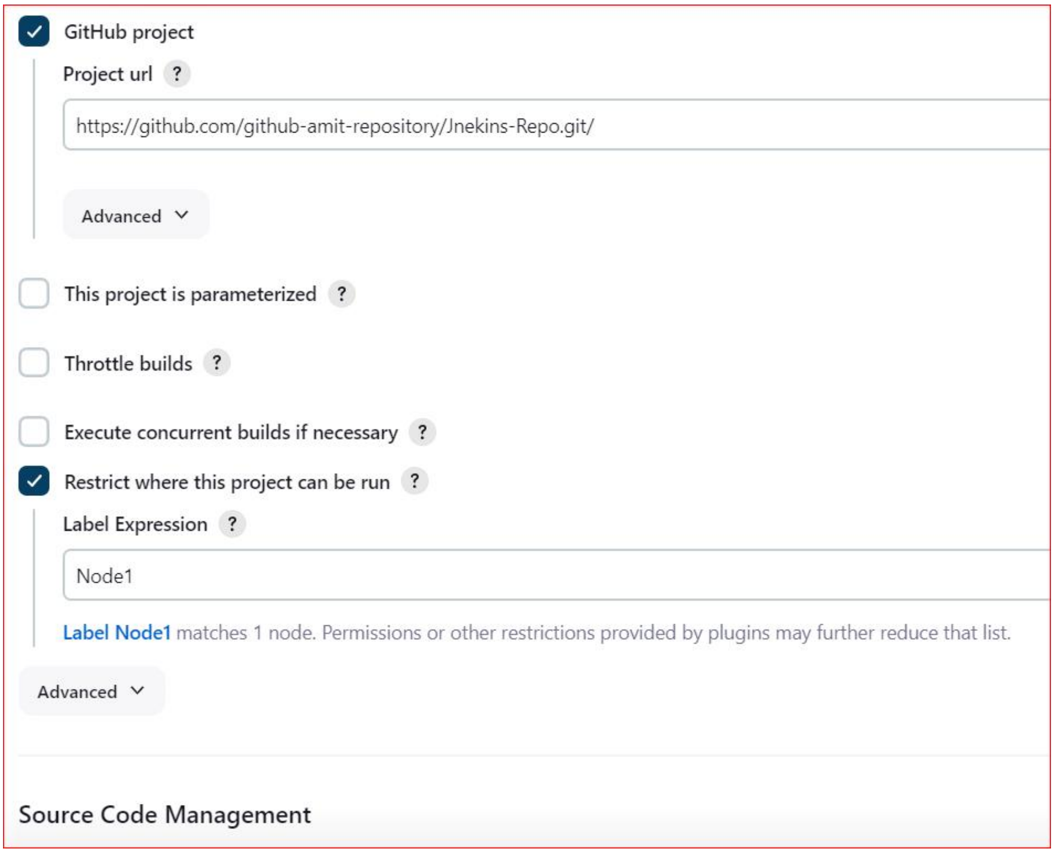
Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any S for something other than software build.

Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for and/or organizing complex activities that do not easily fit in free-style job type.

* Insert url:- Github project -> project url -> paste repo url -> restrict where this project can be run -> Label expression -> Jenkins-Node



☑ GitHub project

Project url ?

https://github.com/github-amit-repository/Jnekins-Repo.git/

Advanced ⌄

□ This project is parameterized ?

□ Throttle builds ?

□ Execute concurrent builds if necessary ?

☑ Restrict where this project can be run ?

Label Expression ?

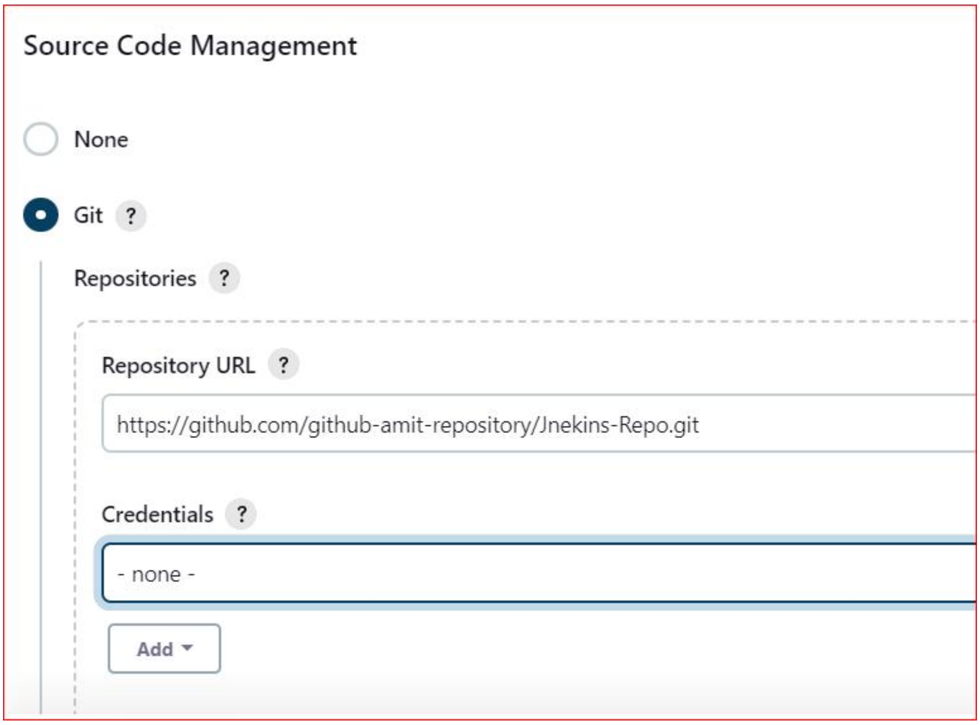
Node1

Label Node1 matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.

Advanced ⌄

Source Code Management

* Now insert url:- source code managemnt -> git -> repo url -> paste repo url



Source Code Management

○ None

● Git ?

Repositories ?

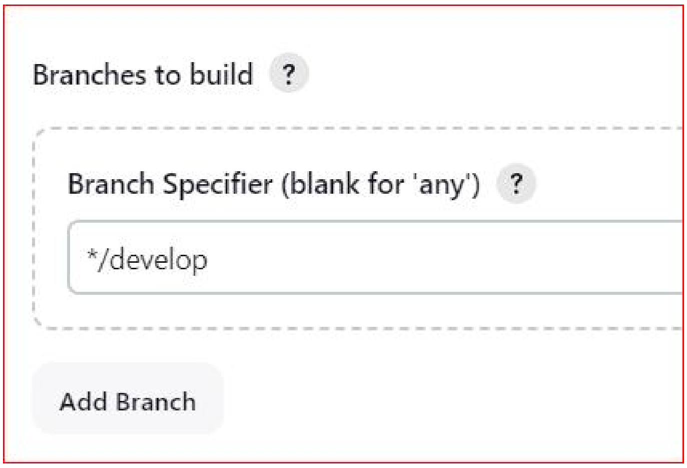
Repository URL ?

https://github.com/github-amit-repository/Jnekins-Repo.git

Credentials ?

- none -

* branch -> */develop

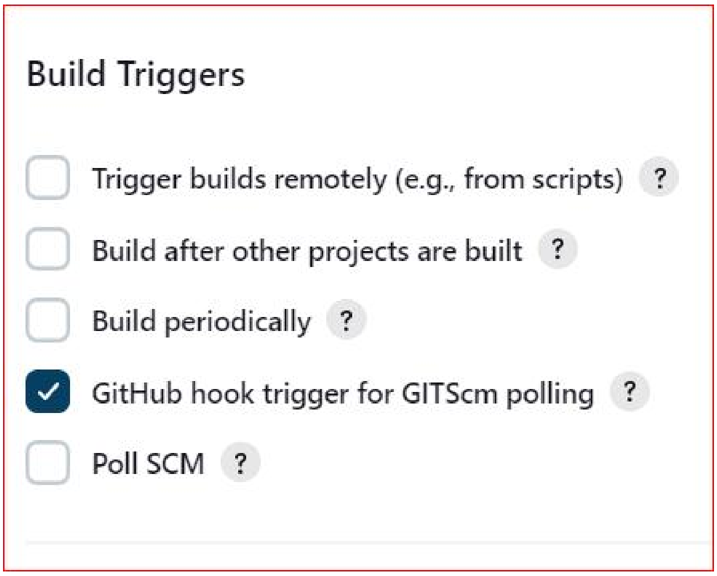


Branches to build ?

Branch Specifier (blank for 'any') ?

*/develop

* Select:- build triggers -> github hook trigger for gitscm polling



Build Triggers

□ Trigger builds remotely (e.g., from scripts) ?

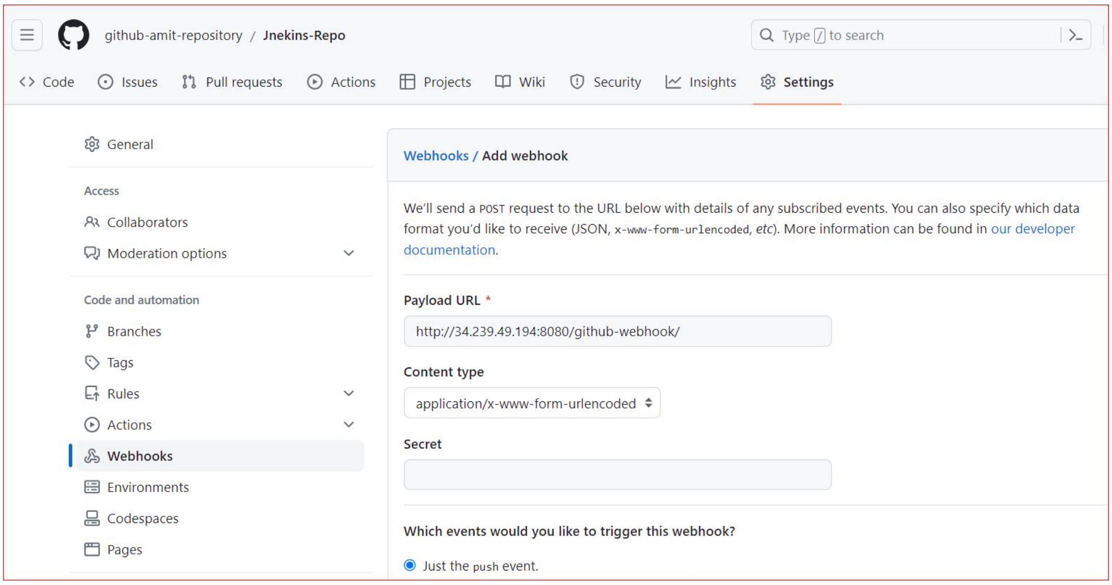
□ Build after other projects are built ?

□ Build periodically ?

☑ GitHub hook trigger for GITScm polling ?

□ Poll SCM ?

* Create webhook connection:- Github -> copy 00:56:00 -> copy jenkins url and paste in -> github repo -> settings -> webhooks -> add webhook -> payload url -> http://jenkins url till port/github-webhook/ -> add webhook -> save



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://34.239.49.194:8080/github-webhook/

Content type

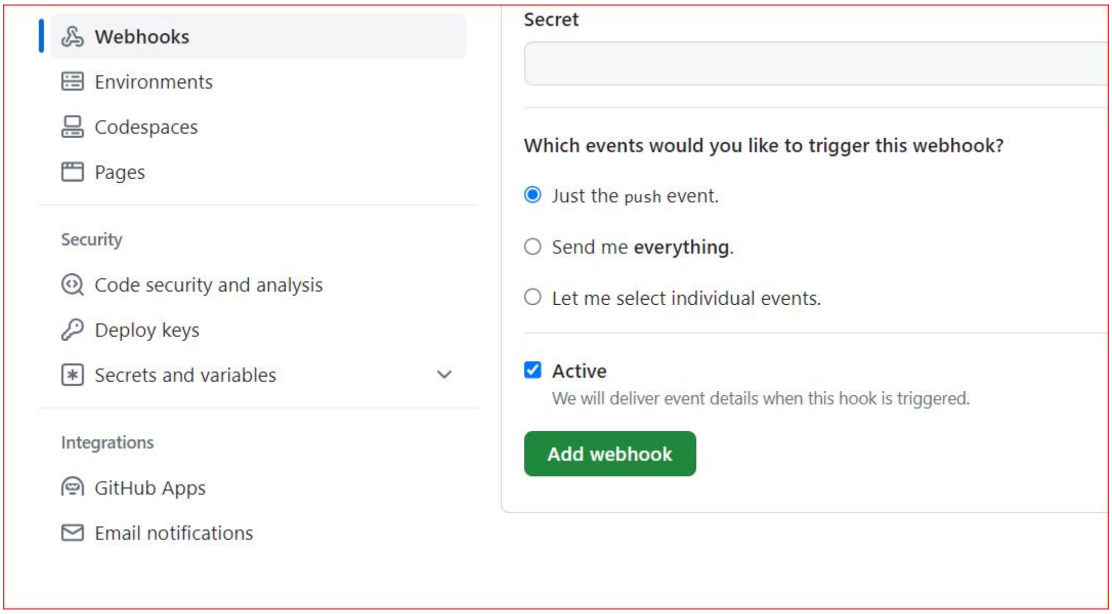
application/x-www-form-urlencoded

Secret

Which events would you like to trigger this webhook?

● Just the push event.

* Add webhook:-



Webhooks

Environments

Codespaces

Pages

Security

Code security and analysis

Deploy keys

Secrets and variables

Integrations

GitHub Apps

Email notifications

Secret

Which events would you like to trigger this webhook?

● Just the push event.

○ Send me **everything**.

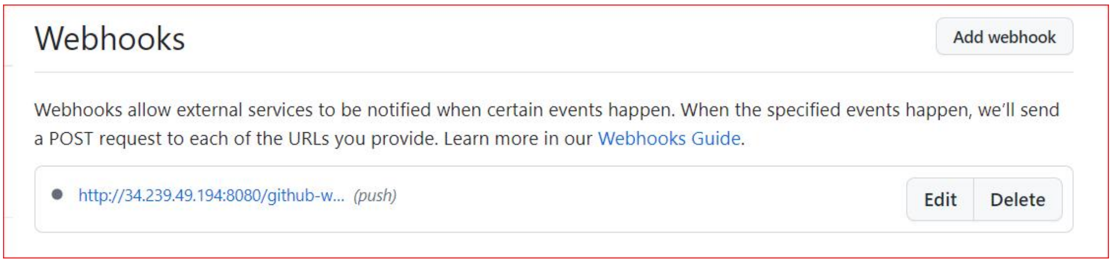
○ Let me select individual events.

☑ Active

We will deliver event details when this hook is triggered.

Add webhook

* Go to:- add webhook -> webhooks/manage webhook/ recent deliveries -> ping

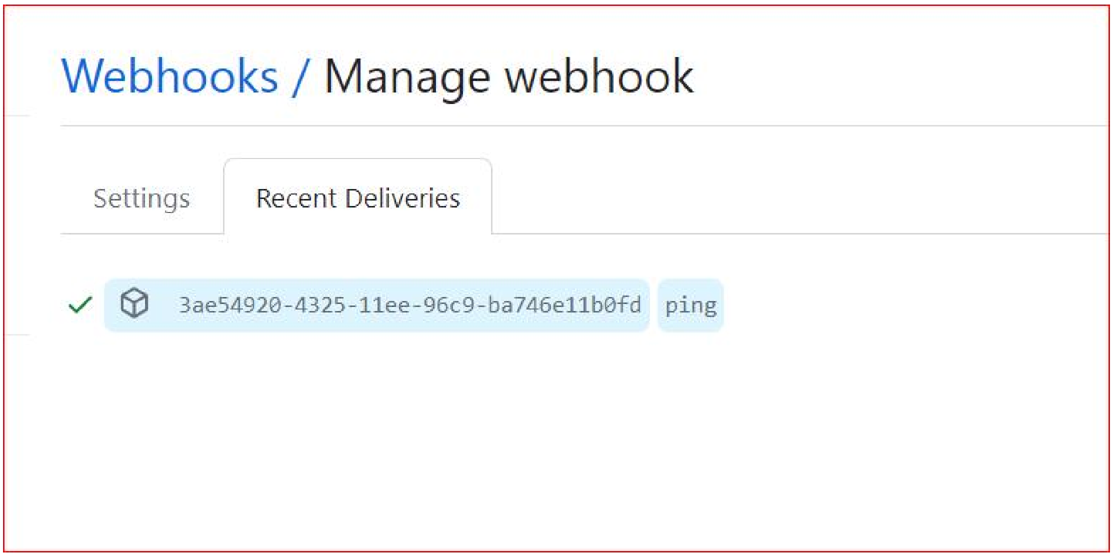


Webhooks

Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our Webhooks Guide.

● http://34.239.49.194:8080/github-w... *(push)* Edit Delete

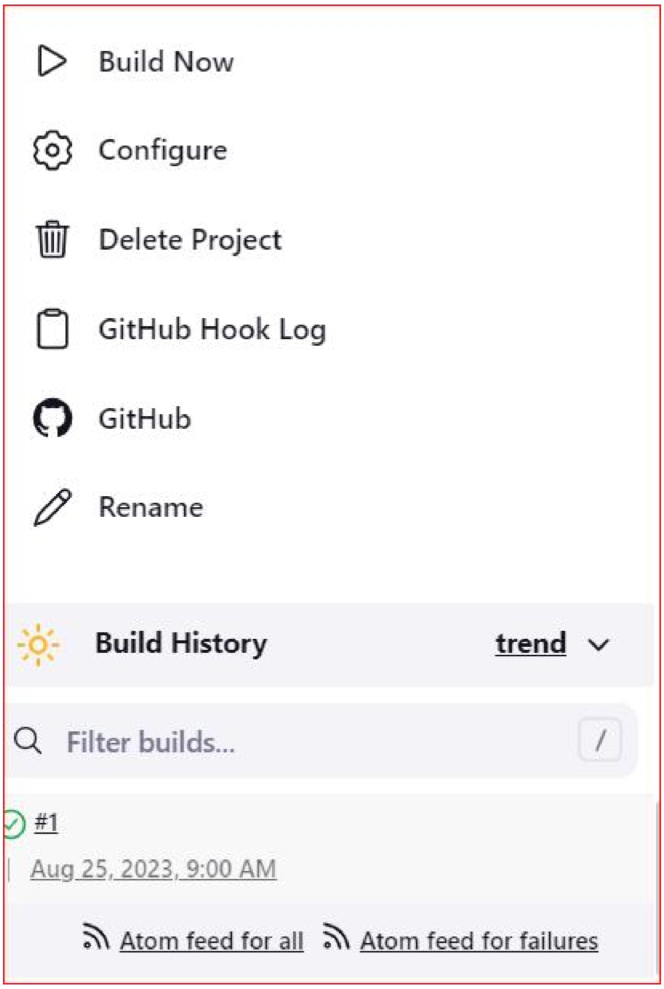


Webhooks / Manage webhook

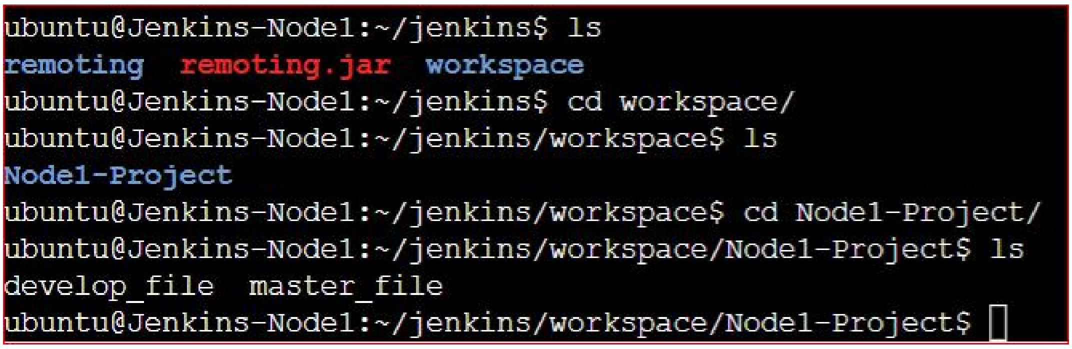
Settings | Recent Deliveries

✓ 3ae54920-4325-11ee-96c9-ba746e11b0fd ping

* Now run build:- click on Build Now

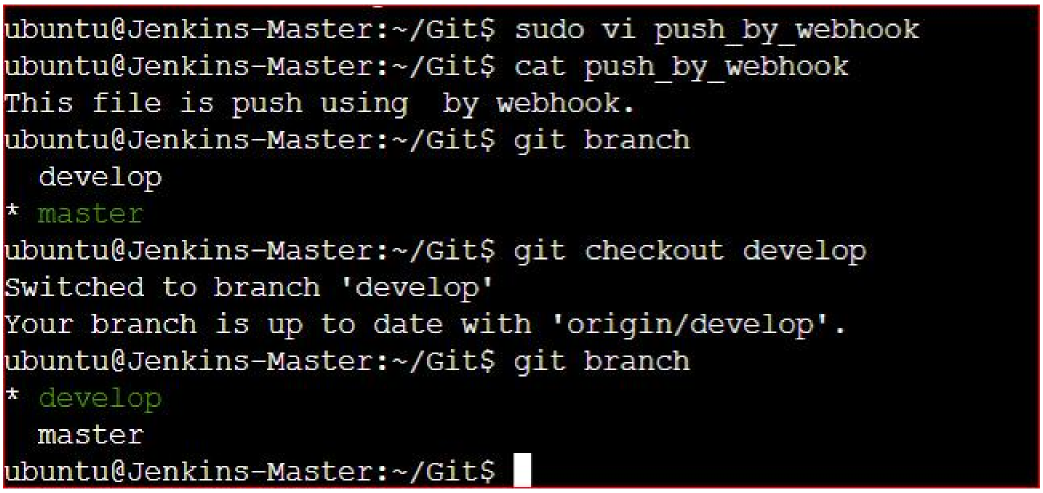


$ ls
$ cd workspace
$ ls
$ cd Node1-Project
$ ls



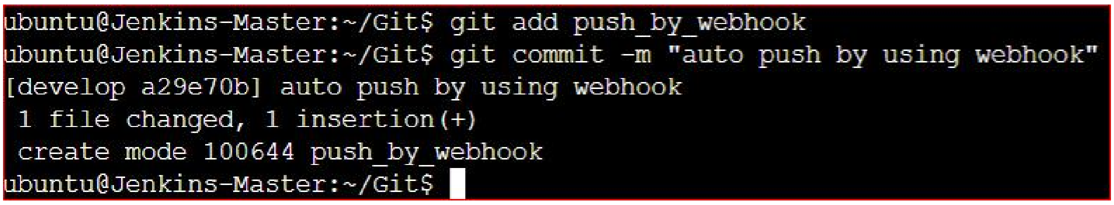
```
ubuntu@Jenkins-Node1:~/jenkins$ ls
remoting  remoting.jar  workspace
ubuntu@Jenkins-Node1:~/jenkins$ cd workspace/
ubuntu@Jenkins-Node1:~/jenkins/workspace$ ls
Node1-Project
ubuntu@Jenkins-Node1:~/jenkins/workspace$ cd Node1-Project/
ubuntu@Jenkins-Node1:~/jenkins/workspace/Node1-Project$ ls
develop_file  master_file
ubuntu@Jenkins-Node1:~/jenkins/workspace/Node1-Project$ 
```

```
$ sudo vi push_by_webhook
$ cat push_by_webhook
$ git branch
$ git checkout develop
$ git branch
```



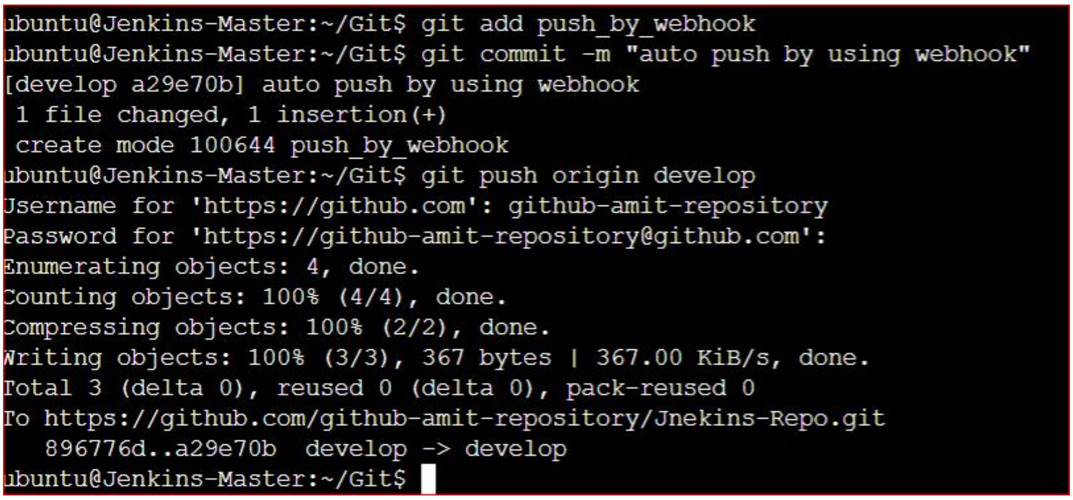
```
ubuntu@Jenkins-Master:~/Git$ sudo vi push_by_webhook
ubuntu@Jenkins-Master:~/Git$ cat push_by_webhook
This file is push using  by webhook.
ubuntu@Jenkins-Master:~/Git$ git branch
  develop
* master
ubuntu@Jenkins-Master:~/Git$ git checkout develop
Switched to branch 'develop'
Your branch is up to date with 'origin/develop'.
ubuntu@Jenkins-Master:~/Git$ git branch
* develop
  master
ubuntu@Jenkins-Master:~/Git$
```

```
$ git add push_by_webhook
$ git commit -m “auto push by using webhook”
```



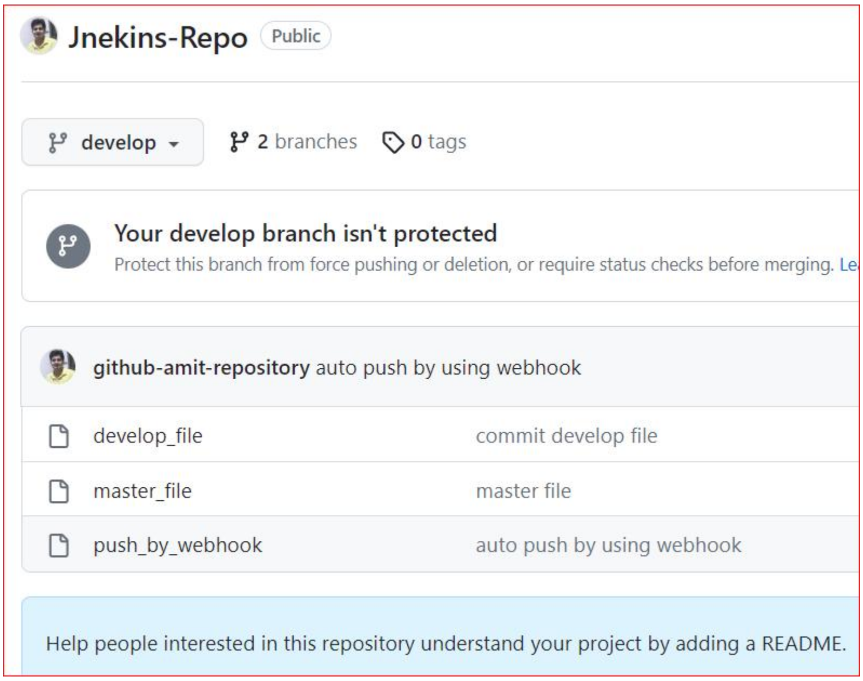
```
ubuntu@Jenkins-Master:~/Git$ git add push_by_webhook
ubuntu@Jenkins-Master:~/Git$ git commit -m "auto push by using webhook"
[develop a29e70b] auto push by using webhook
 1 file changed, 1 insertion(+)
 create mode 100644 push_by_webhook
ubuntu@Jenkins-Master:~/Git$
```

$ git add push_by_webhook
$ git commit -m “auto push by using webhook”
$ git push origin develop
* Enter github username and password.



```
ubuntu@Jenkins-Master:~/Git$ git add push_by_webhook
ubuntu@Jenkins-Master:~/Git$ git commit -m "auto push by using webhook"
[develop a29e70b] auto push by using webhook
 1 file changed, 1 insertion(+)
 create mode 100644 push_by_webhook
ubuntu@Jenkins-Master:~/Git$ git push origin develop
Username for 'https://github.com': github-amit-repository
Password for 'https://github-amit-repository@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 367 bytes | 367.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/github-amit-repository/Jnekins-Repo.git
   896776d..a29e70b  develop -> develop
ubuntu@Jenkins-Master:~/Git$
```

* Now we can see files are present in github:-



Jnekins-Repo Public

develop 2 branches 0 tags

Your develop branch isn't protected

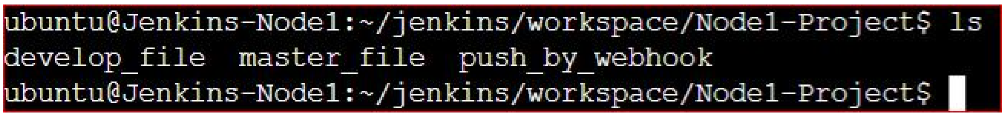
Protect this branch from force pushing or deletion, or require status checks before merging. Le

github-amit-repository auto push by using webhook

develop_file	commit develop file
master_file	master file
push_by_webhook	auto push by using webhook

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

$ ls



```
ubuntu@Jenkins-Node1:~/jenkins/workspace/Node1-Project$ ls
develop_file  master_file  push_by_webhook
ubuntu@Jenkins-Node1:~/jenkins/workspace/Node1-Project$
```