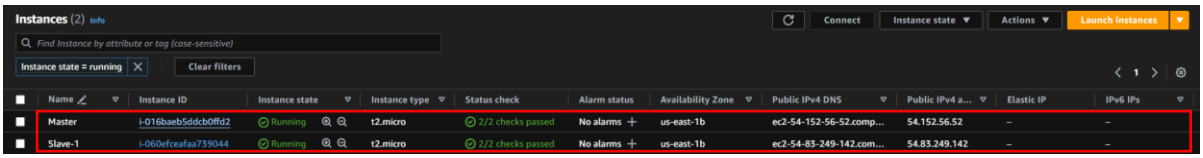
Jenkins Master Slave Architecture

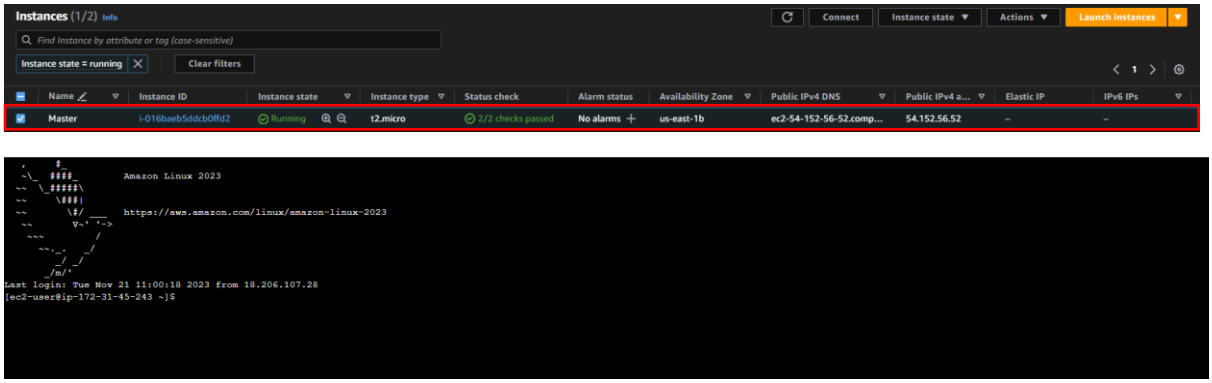
Master to Slaves

Step-1 : Create 2 instance in AWS Linux.

1. EC2-1 is Master
2. EC2-2 is Slave



Step-2 : connect Master



Step-3 : Install java and Jenkins in Master.

to install : java

```
sudo yum install java-17-amazon-corretto-devel
```

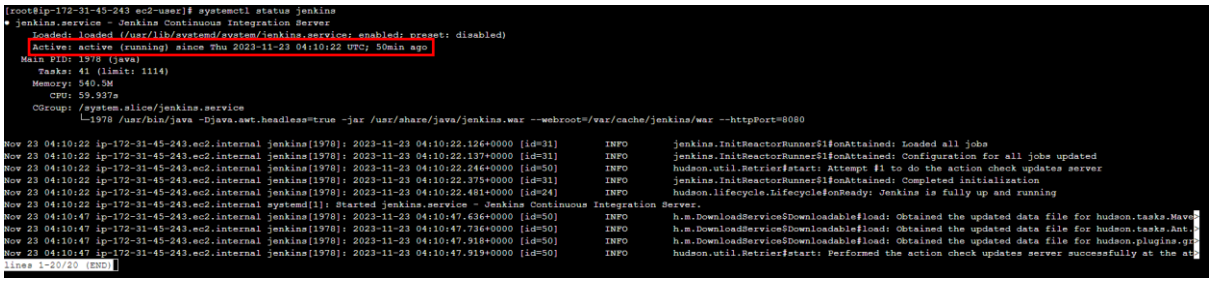
to Check : version of java whether installed or not. cmd : (java -version)

to install : Jenkins

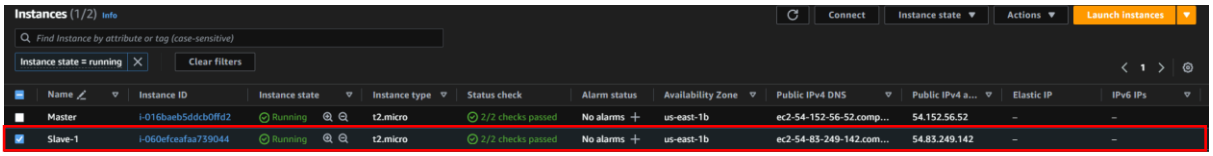
```
sudo yum update -y
sudo wget -O /etc/yum.repos.d/jenkins.repo http://pkg.jenkins.io/redhat/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat/jenkins.io-2023.key
sudo yum install jenkins -y
systemctl daemon-reload
sudo systemctl start jenkins
sudo systemctl enable Jenkins
systemctl status jenkins
```

to Check : version of Jenkins whether installed or not. cmd : (jenkins --version)

- Jenkins status is running.



Step-4 : Connect Slave-1



Step-5 : install java, git.

to install

```
sudo yum install java-17-amazon-corretto-devel
```

to Check : version of java whether installed or not. cmd : (java -version)

to install

```
sudo ymu install git -y
```

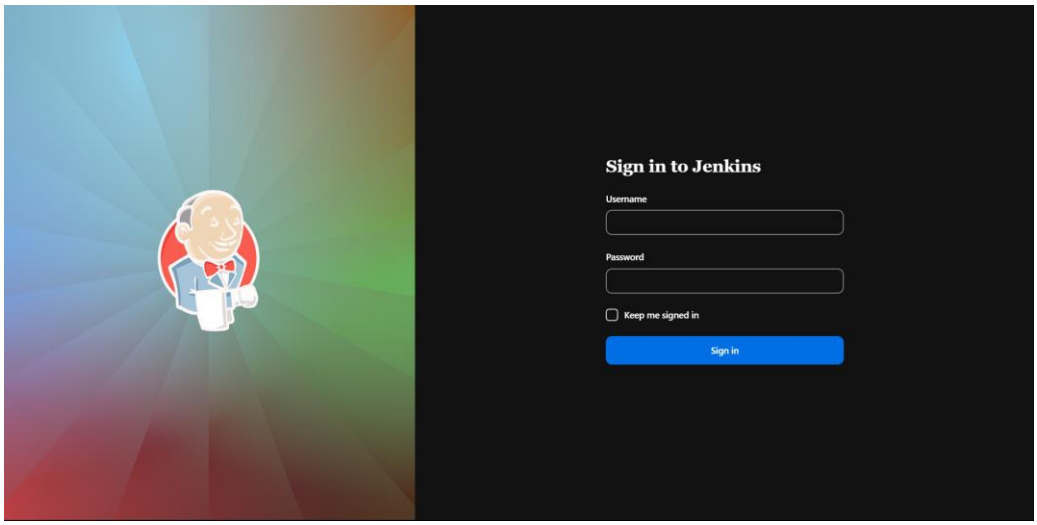
to Check : version of git whether installed or not. cmd : (git --version)

Step-6 : Create 1 directory (Slave-1), go inside, copy current absolute path.

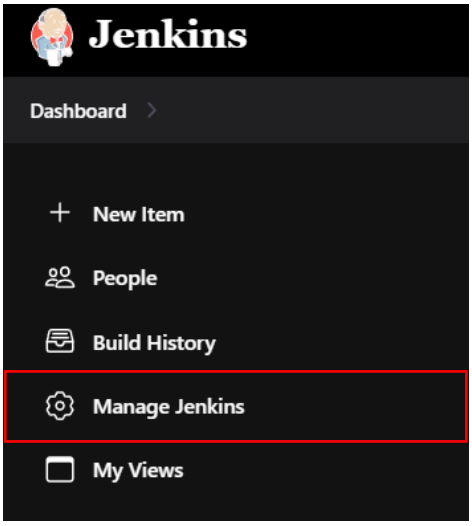
```
[ec2-user@ip-172-31-45-243 ~]$ mkdir Slave1
[ec2-user@ip-172-31-45-243 ~]$ cd Slave1/
[ec2-user@ip-172-31-45-243 Slave1]$ pwd
/home/ec2-user/Slave1
[ec2-user@ip-172-31-45-243 Slave1]$
```

Step-7 : connect to Jenkins.

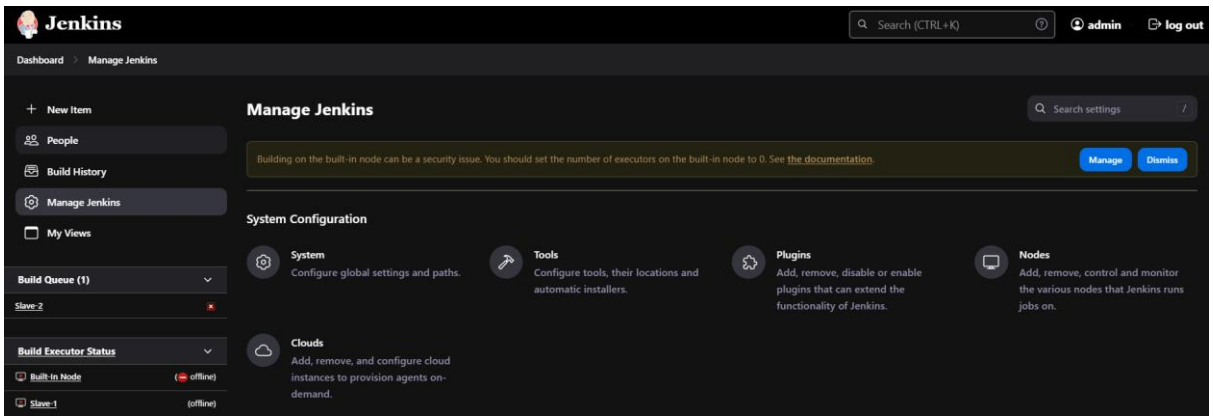
(with the help of Master instance public IP with :8080)



Step-8 : Create Node and connect to slave-1.

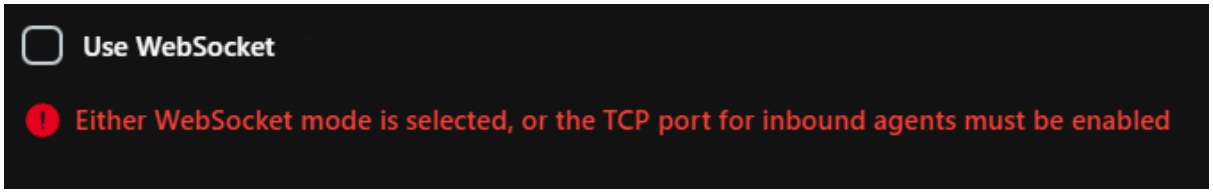


1. Go to manage Jenkins.
2. Node, New node.

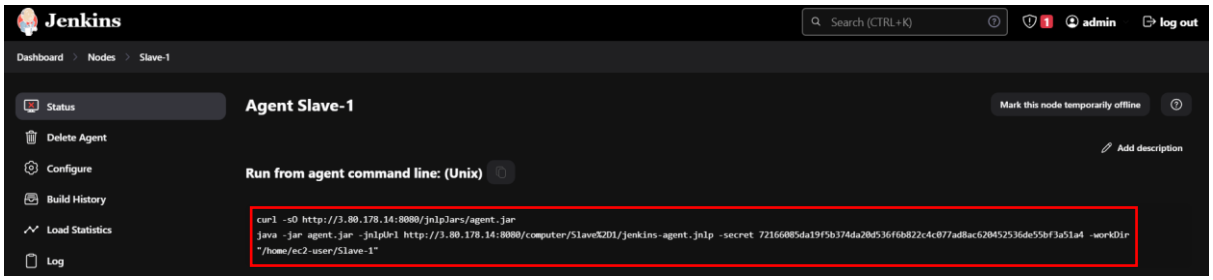


3. in Remote Root Directory give path which we have created in slave-1.

4. Check Websocket.



5. Save.



6. Copy unix command and paste in Slave-1 (paste here: /home/ec2-user/Slave-1)
7. After paste agent.jar, remoting, secret-file, workspace will be create in that directory.

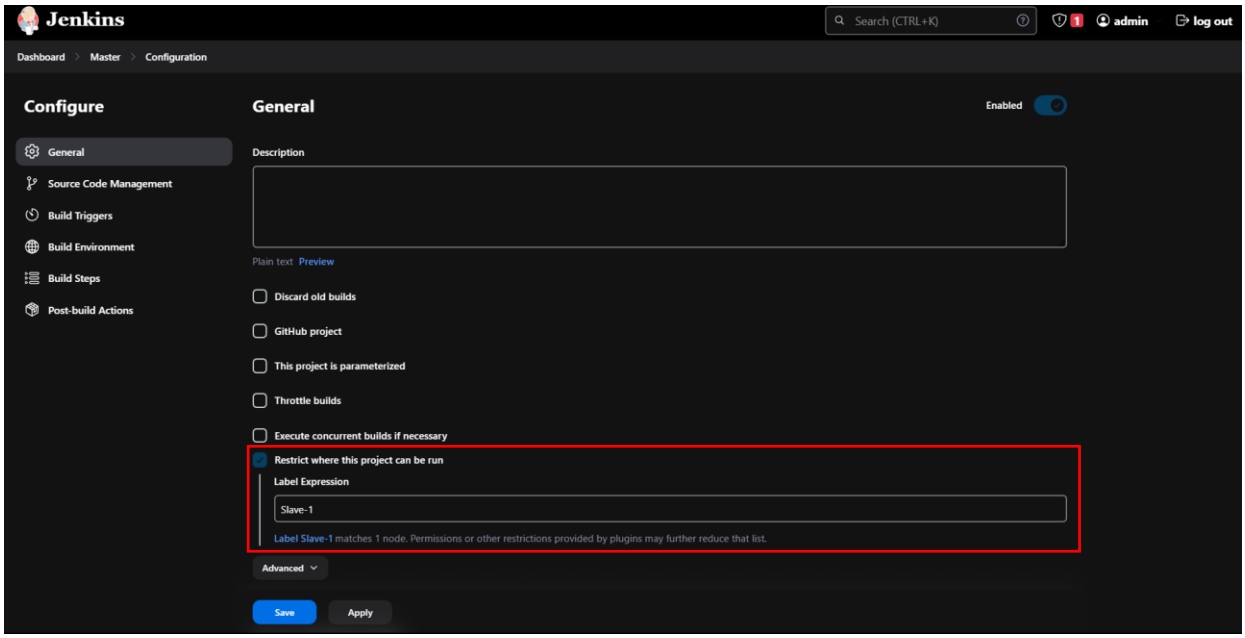
```
[root@ip-172-31-41-200 Slave-1]# ls
agent.jar  remoting  secret-file  workspace
[root@ip-172-31-41-200 Slave-1]# pwd
```

Note : Make sure your slave-1 is in online.

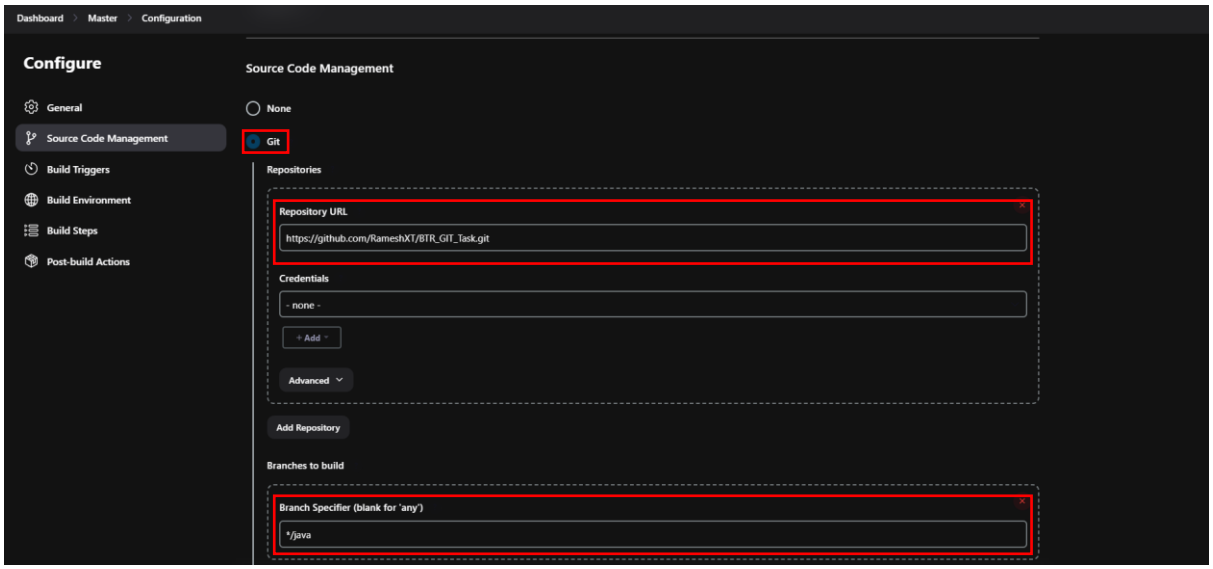
Step-9 : Create a Freestyle Project in Jenkins.

After creating a Jenkins project we need to configure that.

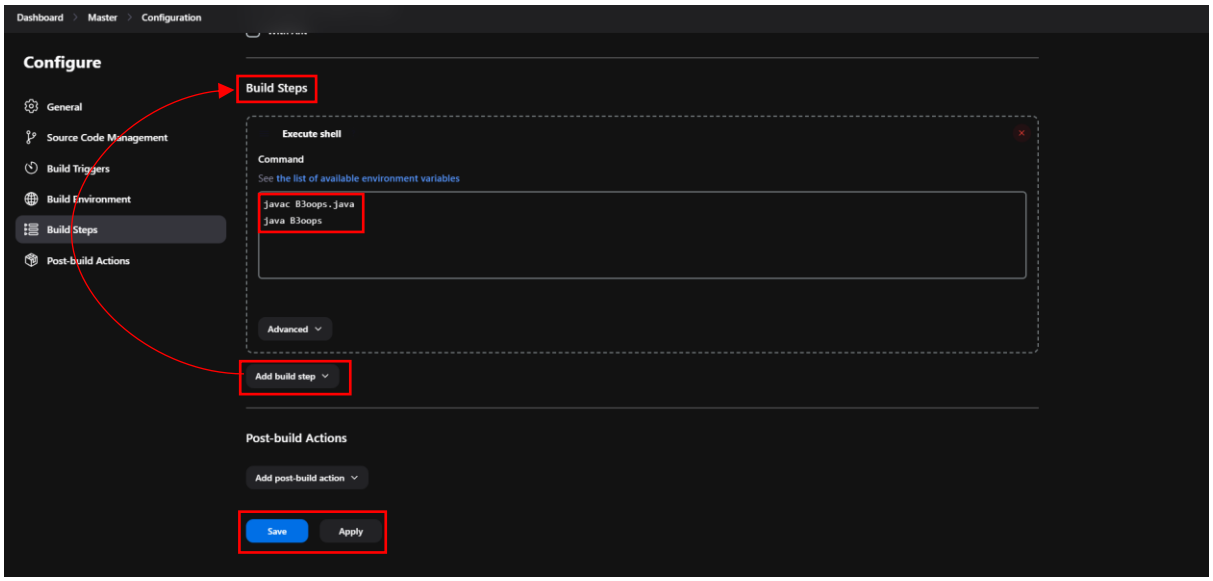
1. Restrict where this project can be run. (label Slave-1)



2. Click git (Source code management)
3. Add you repository link there.
4. Add branch.



5. in add build steps
6. execute shell command

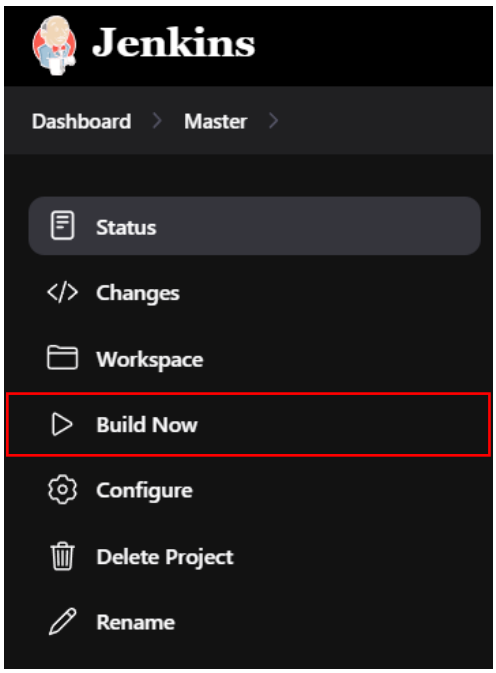


7. to compile java file, to run java file

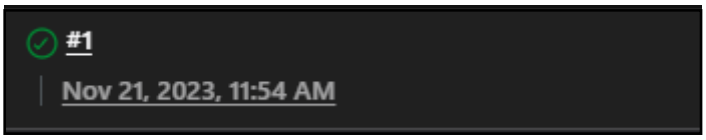
```
javac B3oops.java
java B3oops
```

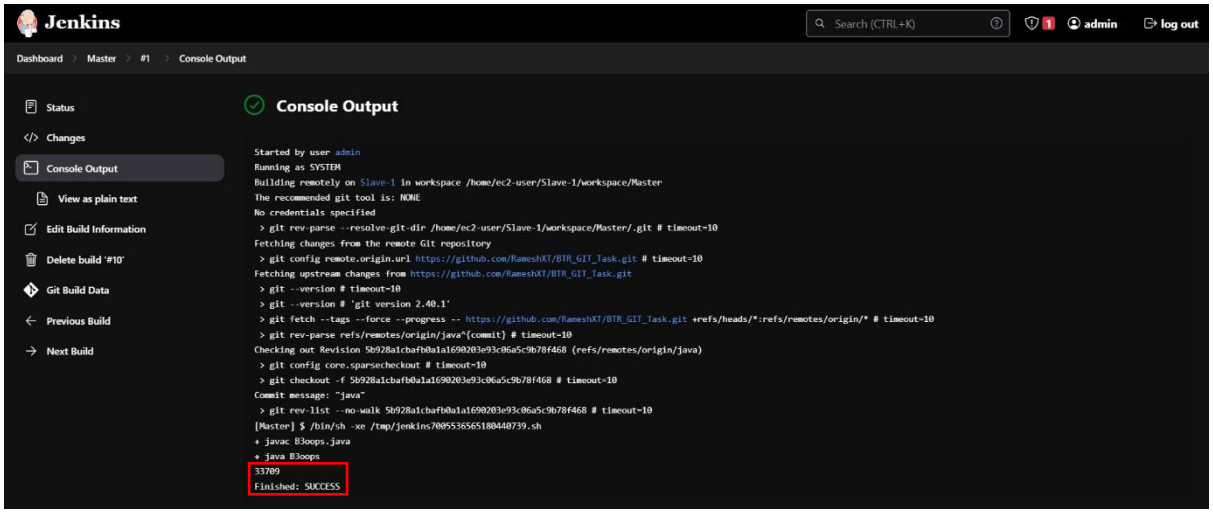
8. save, and apply.

Step-10 : once finish with configuration build our project in project dashboard.



Step-11 : if our Build is success output will be shown in below.



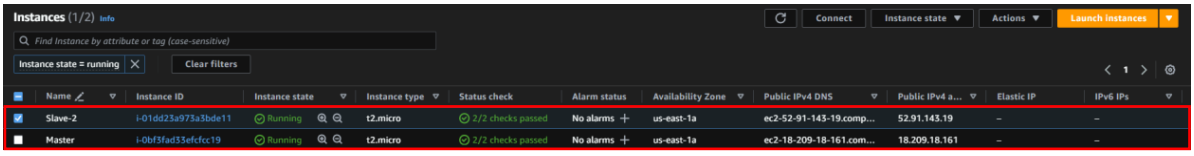


Successfully java Program run.

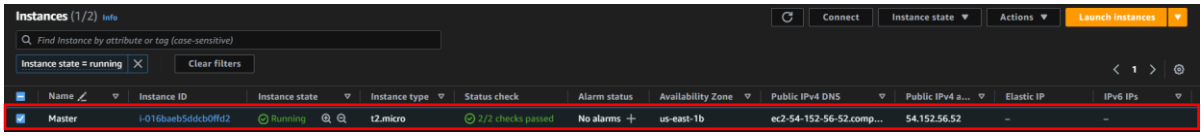
Master to Slaves -2

Step-1 : Create 2 instance in AWS Linux.

1. EC2-1 is Master
2. EC2-2 is Slave



Step-2 : connect Master



Step-3 : Install java and Jenkins in Master.

to install : java

```
sudo yum install java-17-amazon-corretto-devel
```

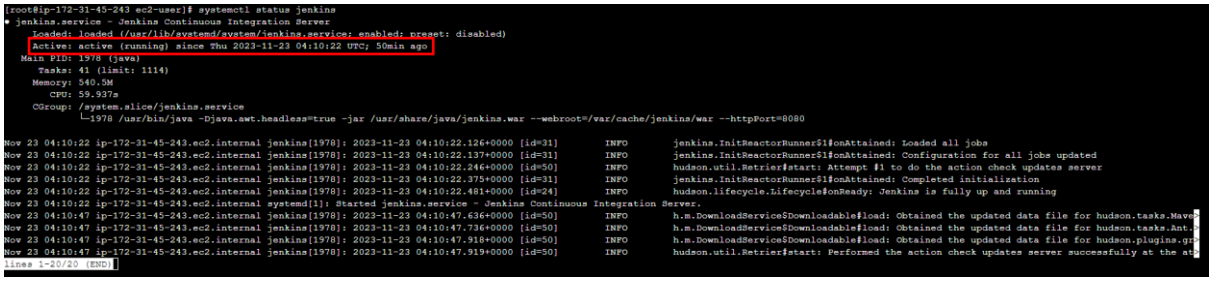
to Check : version of java whether installed or not. cmd : (java -version)

to install : Jenkins

```
sudo yum update -y
sudo wget -O /etc/yum.repos.d/jenkins.repo http://pkg.jenkins.io/redhat/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat/jenkins.io-2023.key
sudo yum install jenkins -y
systemctl daemon-reload
sudo systemctl start jenkins
sudo systemctl enable Jenkins
systemctl status jenkins
```

to Check : version of Jenkins whether installed or not. cmd : (jenkins --version)

- Jenkins status is running.



Step-4 : Connect Slave-1

Step-5 : install apache2(htppd), git.

to install

```
yum install httpd -y
systemctl start httpd
systemctl status httpd
```

to Check : version of apache2(htppd) whether installed or not. cmd : (htppd --version)

to install git

```
sudo ymu install git -y
```

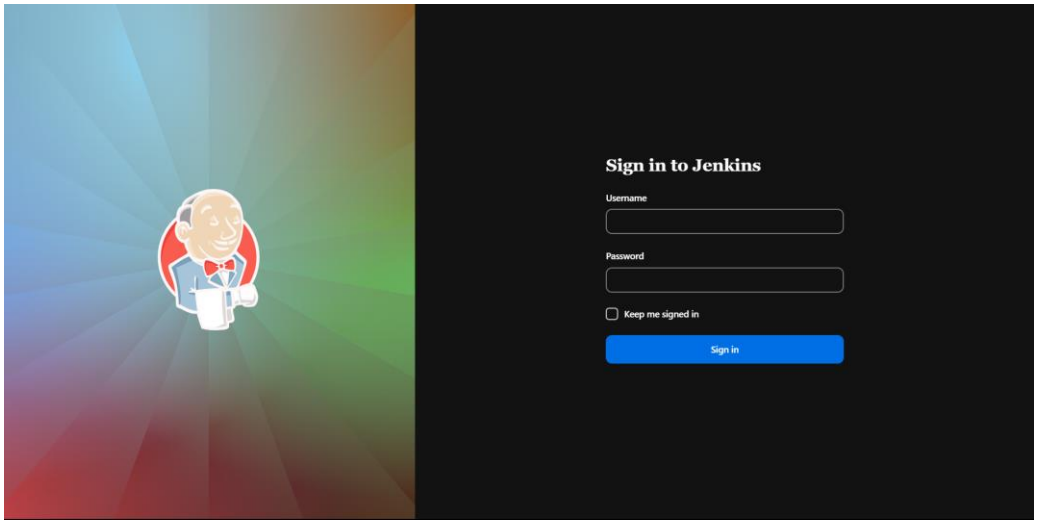
to Check : version of git whether installed or not. cmd : (git --version)

Step-6 : Create 1 directory (Slave-1), go inside, copy current absolute path.

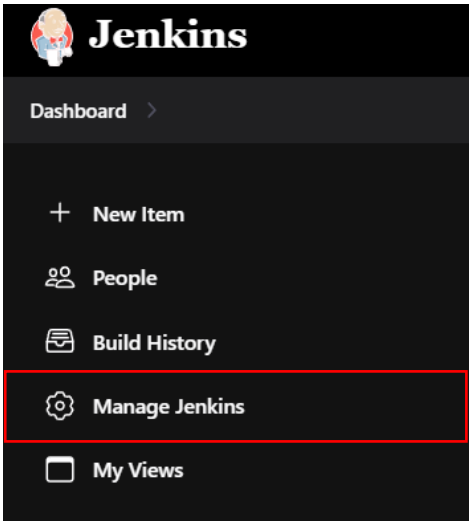
```
[ec2-user@ip-172-31-24-167 ~]$ mkdir Slave-2
[ec2-user@ip-172-31-24-167 ~]$ cd Slave-2/
[ec2-user@ip-172-31-24-167 Slave-2]$ pwd
/home/ec2-user/Slave-2
[ec2-user@ip-172-31-24-167 Slave-2]$
```

Step-7 : connect to Jenkins.

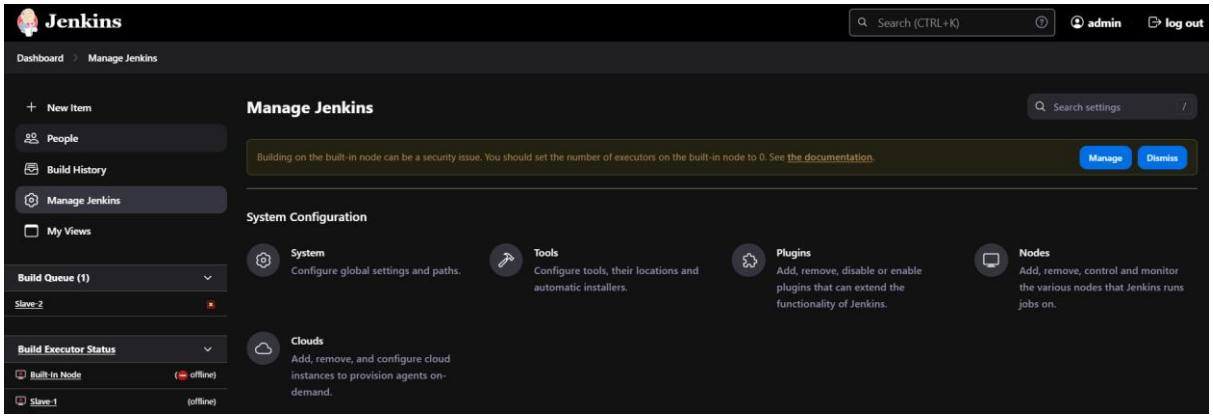
(with the help of Master instance public IP with :8080)



Step-8 : Create Node and connect to slave-2.

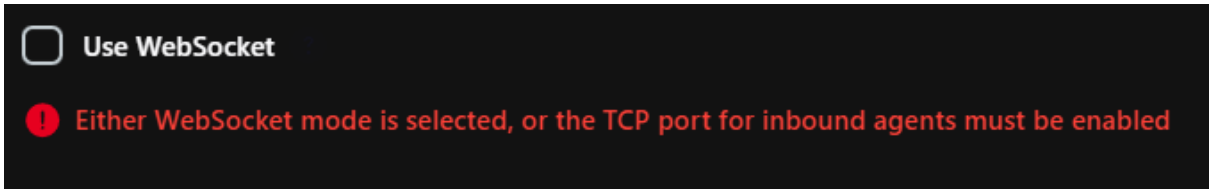


8. Go to manage Jenkins.
9. Node, New node.



10. in Remote Root Directory give path which we have created in slave-1.

11. Check Websocket.



12. Save.

13. Copy unix command and paste in Slave-2 (paste here: /home/ec2-user/Slave-1)
14. After paste agent.jar, remoting, secret-file, workspace will be create in that directory.

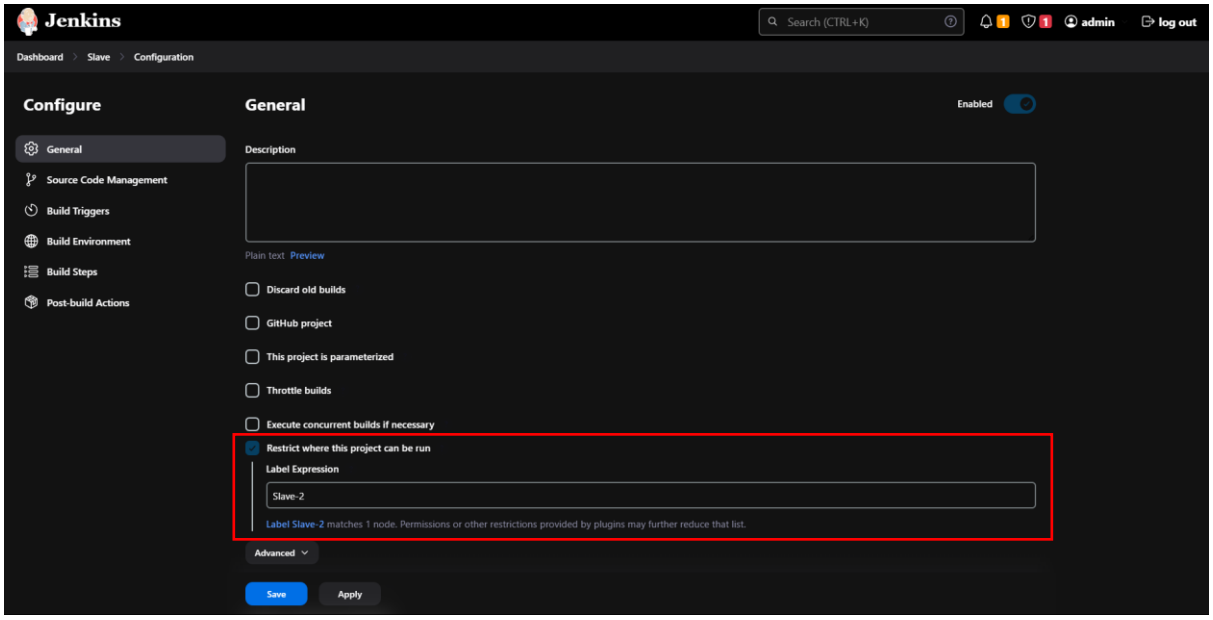
```
[root@ip-172-31-24-167 Slave-2]# ls
agent.jar  remoting  workspace
[root@ip-172-31-24-167 Slave-2]#
```

Note : Make sure your slave-1 is in online.

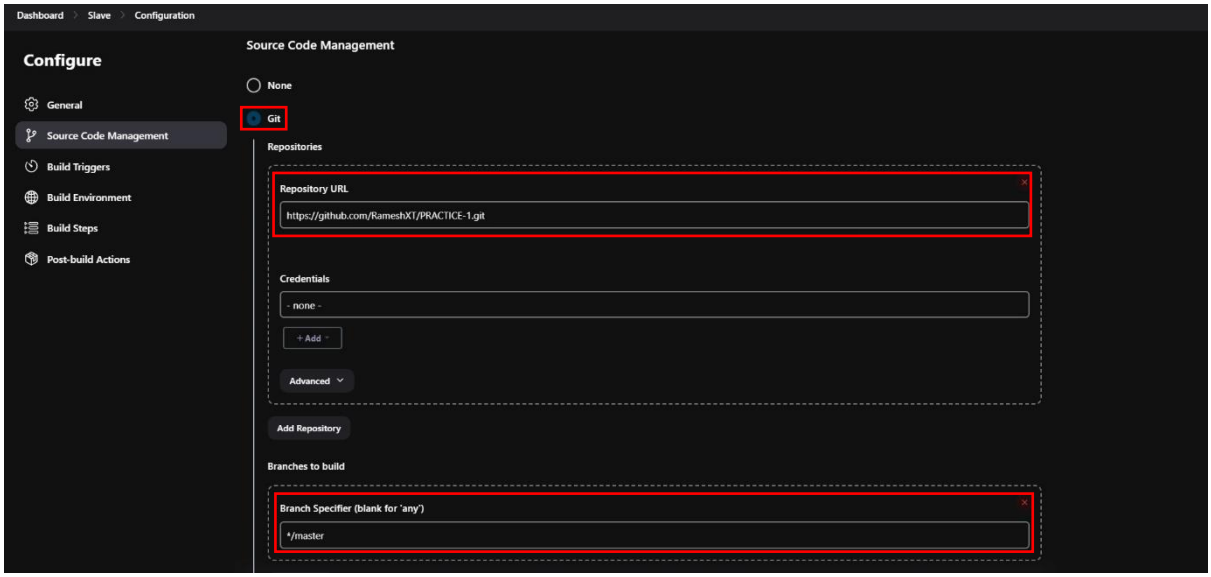
Step-9 : Create a Freestyle Project in Jenkins.

After creating a Jenkins project we need to configure that.

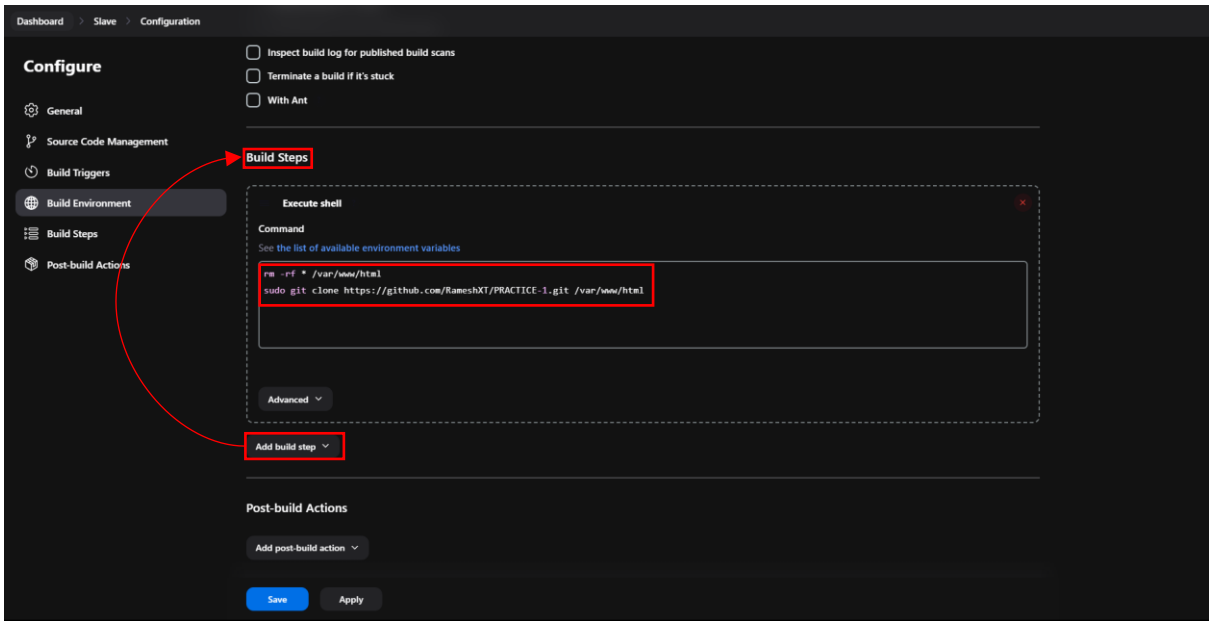
9. Restrict where this project can be run. (label Slave-2)



10. Click git (Source code management)
11. Add you repository link there.
12. Add branch.



13. in add build steps
14. execute shell command



15. 1st we need clean a html directory (path : /var/www/html)

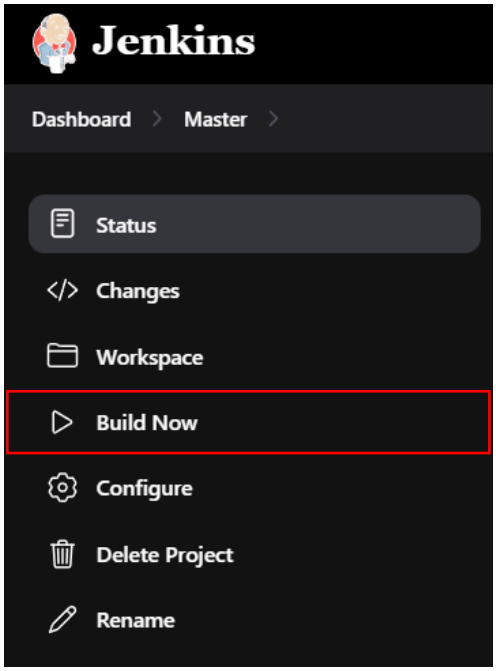
```
rm -rf * /var/www/html
```

16. Clone a project html and move to html path then only it will expose in webpage.

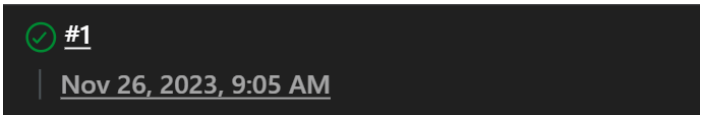
```
sudo git clone https://github.com/RameshXT/PRACTICE-1.git /var/www/html
```

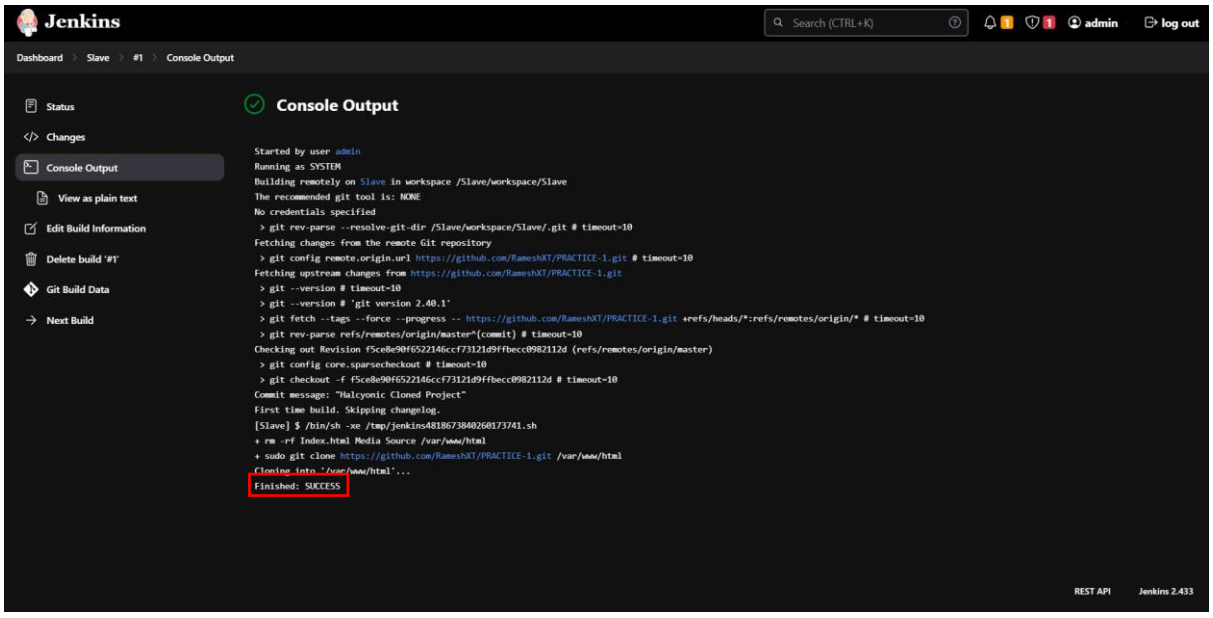
17. Save and apply.

Step-10 : once finish with configuration build our project in project dashboard.

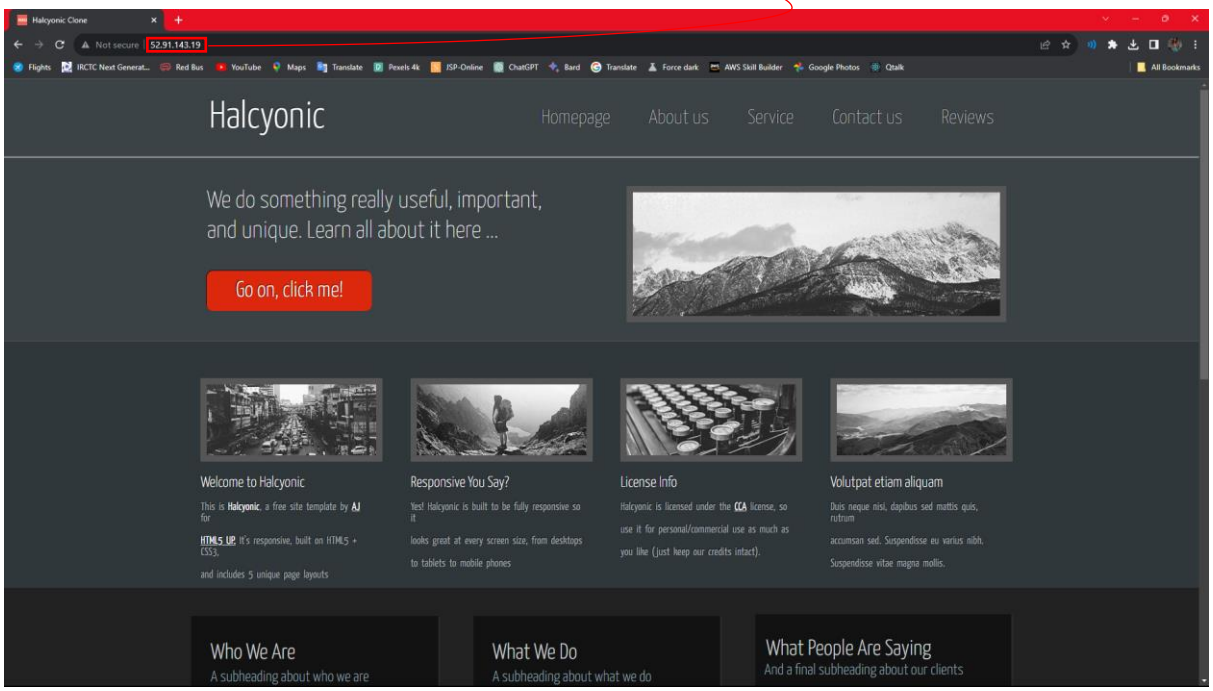


Step-11 : if our Build is success output will be shown in below.





1. (copy Slave instance public IP with :80)



Successfully running our application.
