Scripted pipeline

More flexible but harder to maintain comparing to declarative approach.
Because it is more focused on code.

```
node{
      stage('1'){
            checkout scm
      }
      stage('2'){
            clean install
      }
      stage('3'){
            test
      }
      stage('4'){
            deploy
      }
}
```

Summary

Up Stream Job: The job that triggered another JOB
DownStream Job: The Job that is triggered by another

pipeline {

```
 agent any

 stages {
 stage('Build UpStream') {
 steps {
 echo 'Runing Upstream Job.....'
 }
 }
 stage('Trigger DownStream') {
```

```
steps {
build job: 'Job1' , wait:true
}
}
}
}
```

In this above example Job1 is the Downstream Job
and the script written in which its called upstream Job
How to do this??

If you are creating pipeline then you can join by using build
If its free style project then you can do it by using post-build action

Jenkins Master-Agent Setup

Go to AWS Console Screen
Launch Instance
Give name: JenkinsMaster
select ubuntu
instance type: t2.micro
network setting:
launch new security group with 2 rules
ssh :: 22 port (which is coming by default)
custom tcp: type port 8080
select everywhere: 0.0.0.0/0
storage size: 8 GB
Launch Instance

Connect to your instance:
follow the process of installing Jenkins here
install JDK first:

```
sudo apt update
sudo apt install fontconfig openjdk-21-jre
java -version
openjdk version "21.0.3" 2024-04-16
OpenJDK Runtime Environment (build 21.0.3+11-Debian-2)
OpenJDK 64-Bit Server VM (build 21.0.3+11-Debian-2, mixed mode, sharing)
```

Then install Jenkins:

```
sudo wget -O /etc/apt/keyrings/jenkins-keyring.asc \
https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
echo "deb [signed-by=/etc/apt/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
```

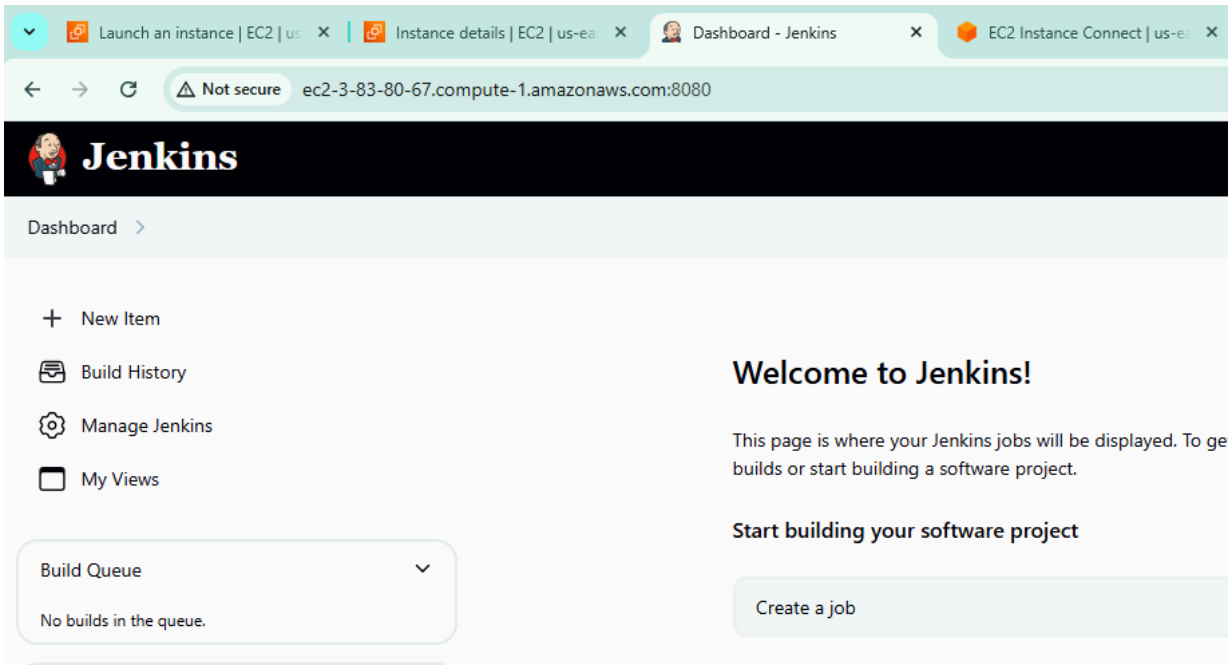
Then enable, start and check status:

```
sudo systemctl enable jenkins

sudo systemctl start jenkins

sudo systemctl status jenkins
```

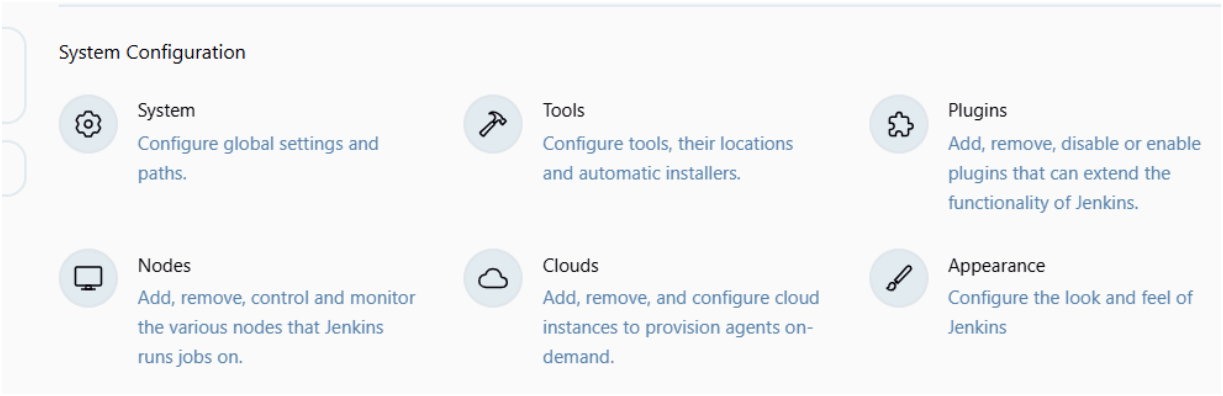
Last once you copy password from status
copy the public ip of your instance or public DNS after that attach :8080
you can see the Jenkins setup screen
follow the same installation process and you can see the Jenkins dashboard.
(make sure you access it using http not https)

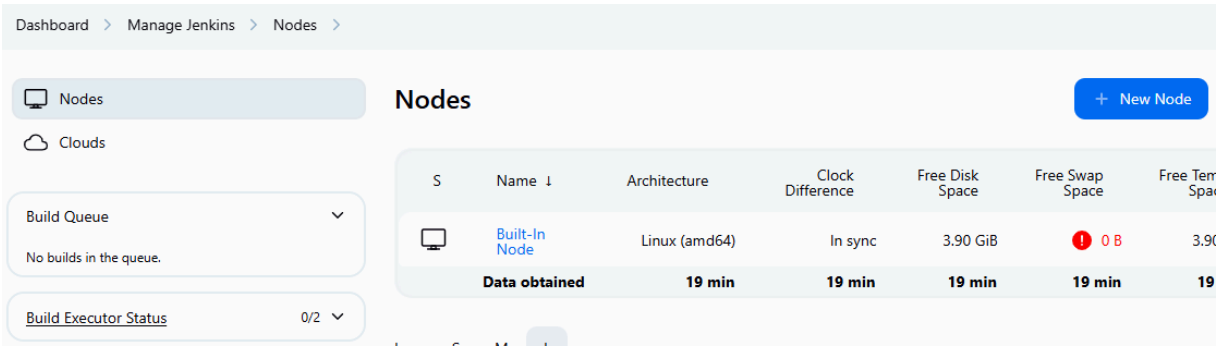


Now Let's Setup Agent Jenkins Node

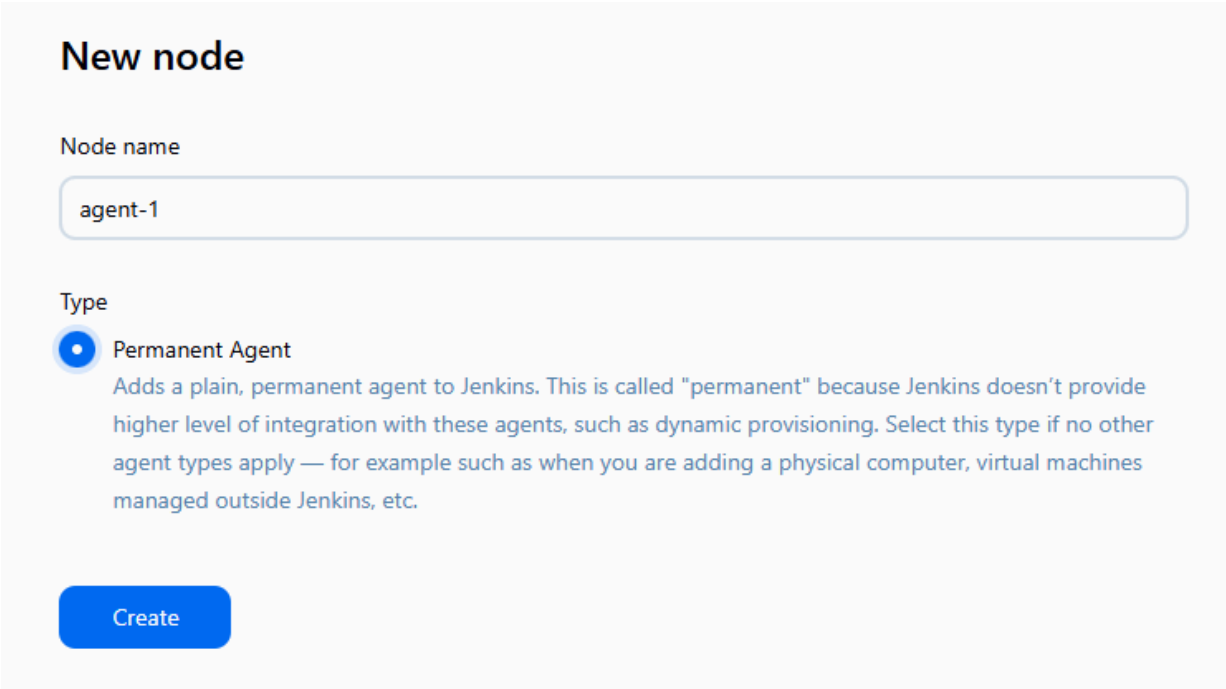
Again create one more EC2 instance using ubuntu or amazon linux AMI.
install same jdk version which is installed on master node

Open Jenkins dashboard
manage Jenkins --> nodes --> new node



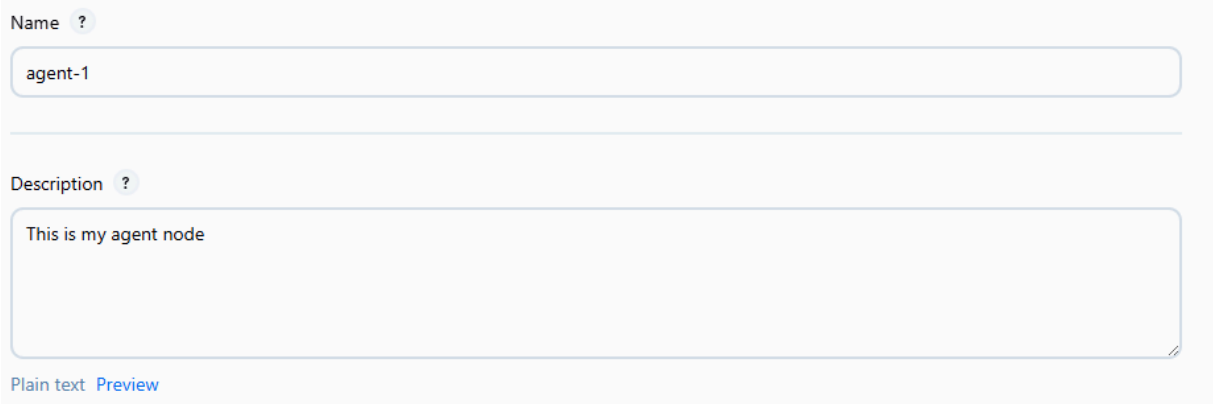


click on new node

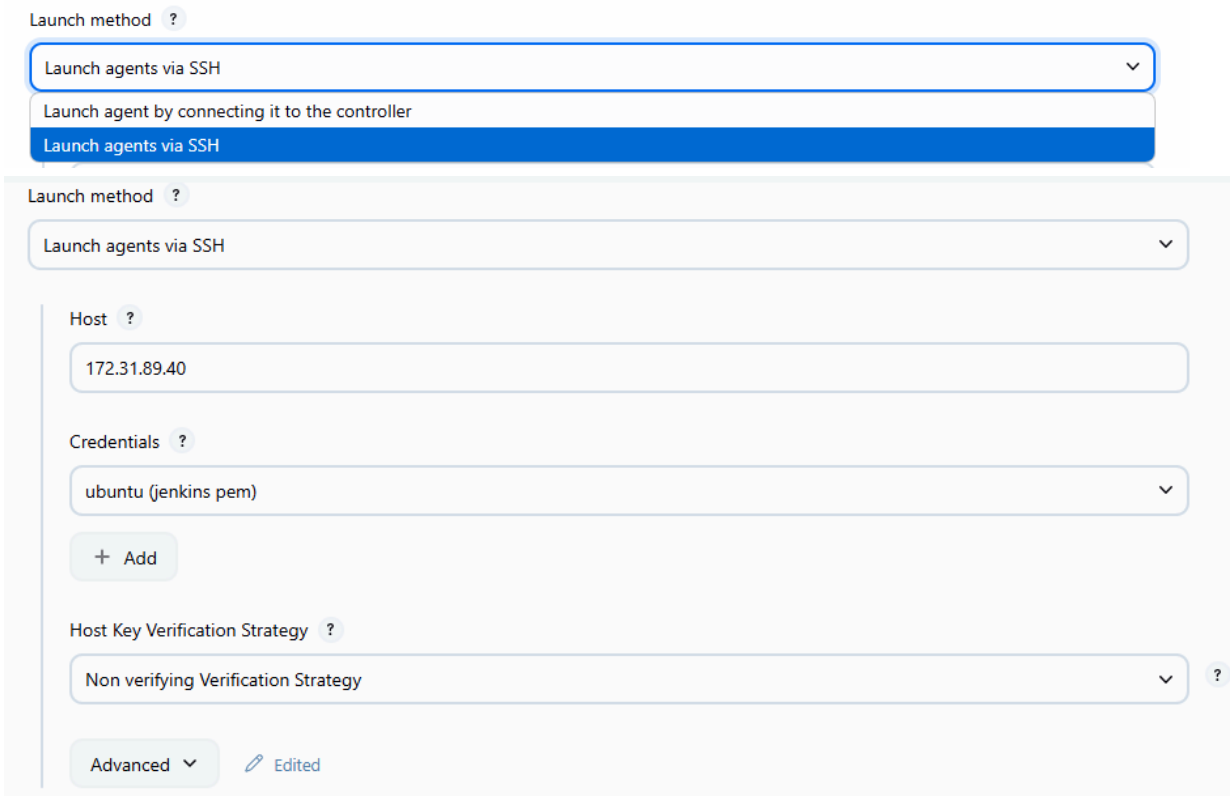


click on create

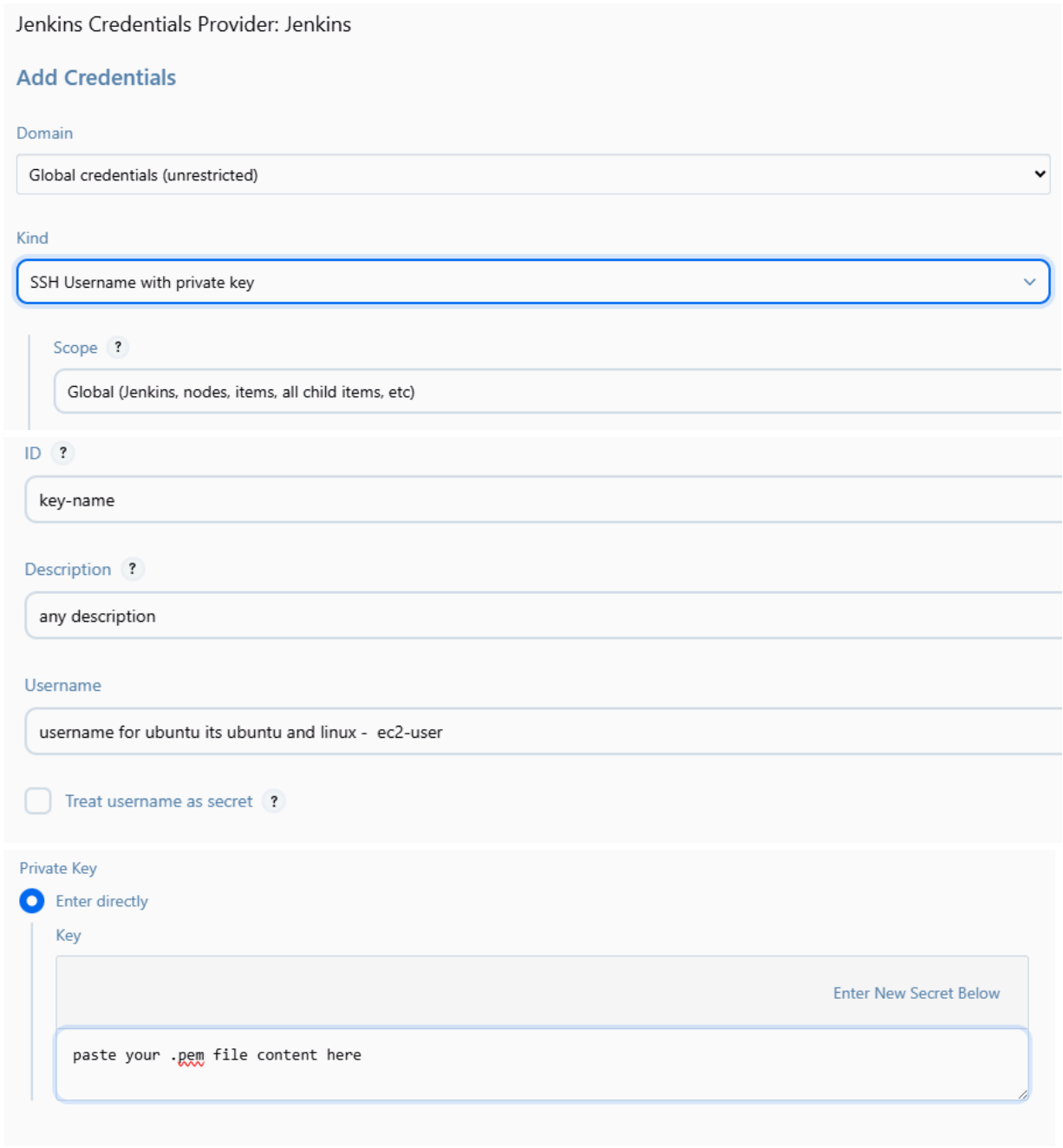
click on configure:



Select Launch via ssh



For credentials you have to click on add



click on Add
after this you can select this created credentials and save.
once you save it will try connecting your instance as agent.
check logs

```
<===[JENKINS REMOTING CAPACITY]===>channel started
Remoting version: 3301.v4363ddcca_4e7
Launcher: SSHLauncher
Communication Protocol: Standard in/out
This is a Unix agent
Agent successfully connected and online
```

Means we are able to create agent-master Jenkins structure.