Jenkins

- Jenkins is an open-source project written in Java that runs on Windows, MacOS and Other Unix-like Operating Systems. It is free, Community supported and might be your first choice tool for CI.
- Jenkins automates the entire software development life cycle.
- Jenkins was Originally developed by Sun Microsystem in 2004 under the name hudson.
- The project was later named Jenkins when Oracle bought Microsystems.
- It can run on any major platform without any compatibility issues.
- Whenever developers write code, we integrate all that code of all developers at that point of time and we build, test and deliver/deploy to the client. This process is called CI/CD.
- Jenkins helps us to achieve this.
- Because of CI, Now bugs will be reported fast and get rectified fast, So the entire software development happens fast.

Workflow of Jenkins

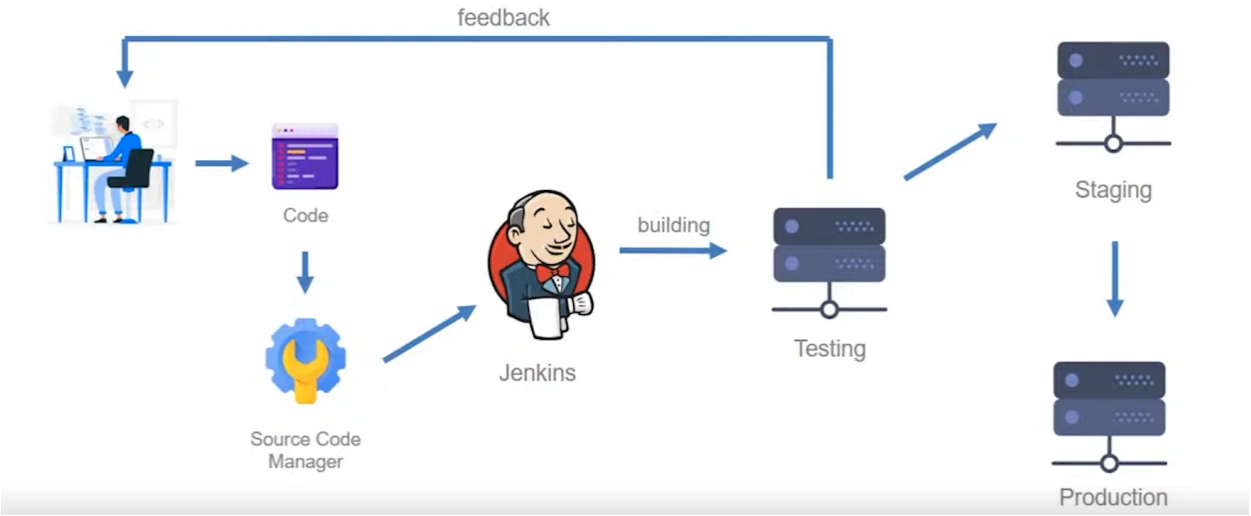
- We can attach Git, Maven, Selenium and Artifactory plugins to Jenkins.
- Once developers put code in github jenkins pull the code and send it to Maven for Build.
- Once build is done, Jenkins pulls that code and sends it to Selenium for testing.
- Once testing is done, then jenkins will pull that code and send it to the Artifactory as per requirement and so on.
- We can also deploy with jenkins.

Advantages of Jenkins

- It has lots of plugins available.
- You can write your own plugin.
- You can use community plugins.
- Jenkins is not just a tool. It is a Framework i.e You can do whatever you want. All you need is plugins.
- We can attach slaves (Nodes) to jenkins master. It instructs others (Nodes) to do jobs. If nodes are not available, jenkins itself does the job.

- ➢ Jenkins also behaves as server Replacement. i.e can do scheduled task.
- ➢ It can create labels.

Jenkins Pipeline



Jenkins & CICD Pipeline Project

Tools need to install on local system:

1. Git
2. Java
3. Maven
4. Jenkins

Git Installation Steps

- ➢ Go to Chrome Browser, Search ‘git download’
- ➢ https://git-scm.com/download/win
- ➢ latest (**2.42.0**) **64-bit** version of **Git for Windows** - Click on Download
- ➢ Open the download file
- ➢ Preamble > C:\Program Files\Git

- ➢ select components
- ➢ Select start menu folder
- ➢ Choosing the default editor vim
- ➢ Let git decide
- ➢ Git from the command line and also from the 3rd party software
- ➢ Use the openssl library
- ➢ Checkout as-is, Commit unix-style line encodings
- ➢ Use MinTTY
- ➢ Choose default behavior
- ➢ Git credentials Manager Core
- ➢ Check Enabled file system caching
- ➢ Install
- ➢ Go to command Prompt in laptop
- ➢ # git config --global user.name “prince.jaiswal”
- ➢ # git config --global user.email “prince.jaiswal0007@gmail.com”
- ➢ # git config --list

```
C:\Users\princ>git config --global user.name "prince.jaiswal"

C:\Users\princ>git config --global user.email "princejaiswal0007@gmail.com"

C:\Users\princ>git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt
core.autocrlf=input
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.name=prince.jaiswal
user.mail=princejaiswal0007@gmail.com
user.email=princejaiswal0007@gmail.com
```

Download & Installation of JDK (Java)

- Go to google chrome
- Search “Java Development Kit Download”
- URL : https://www.oracle.com/in/java/technologies/downloads/#jdk17-windows
- Download x64 Installer **JDK Development Kit 17.0.8 downloads**
- Run and follow the steps to install
- After installation go to cmd and check version

```
C:\Users\princ>java --version
java 17.0.8 2023-07-18 LTS
Java(TM) SE Runtime Environment (build 17.0.8+9-LTS-211)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.8+9-LTS-211, mixed mode, sharing)
```

- Now, go to C drive
- Program Files
- C:\Program Files\Java\jdk-17 -> Select Path & Copy it
- Go to Windows Search “Edit System Environment Variables” in Laptop.
- Go to User Variables -> New
- Variable Name -> JAVA_HOME
- Variable Value -> Paste the Java Path here
- Now, Go to System Variable -> New
- Variable Name -> JAVA_HOME
- Variable Value -> Paste the Java Path here

- Now, Go to inside C:\Program Files\Java\jdk-17\bin -> copy the path
- Again go to “Edit System Environment Variables” -> System Variable -> Path -> New -> Paste Path
- Now verify in Command Prompt
- echo %JAVA_HOME%

```
C:\Users\princ>echo %JAVA_HOME%
C:\Program Files\Java\jdk-17
```

Maven Download & Configure

- ➢ Go to Google Chrome
- ➢ Search maven.apache.org
- ➢ URL : https://maven.apache.org/download.cgi
- ➢ Downloads Binary Zip archive

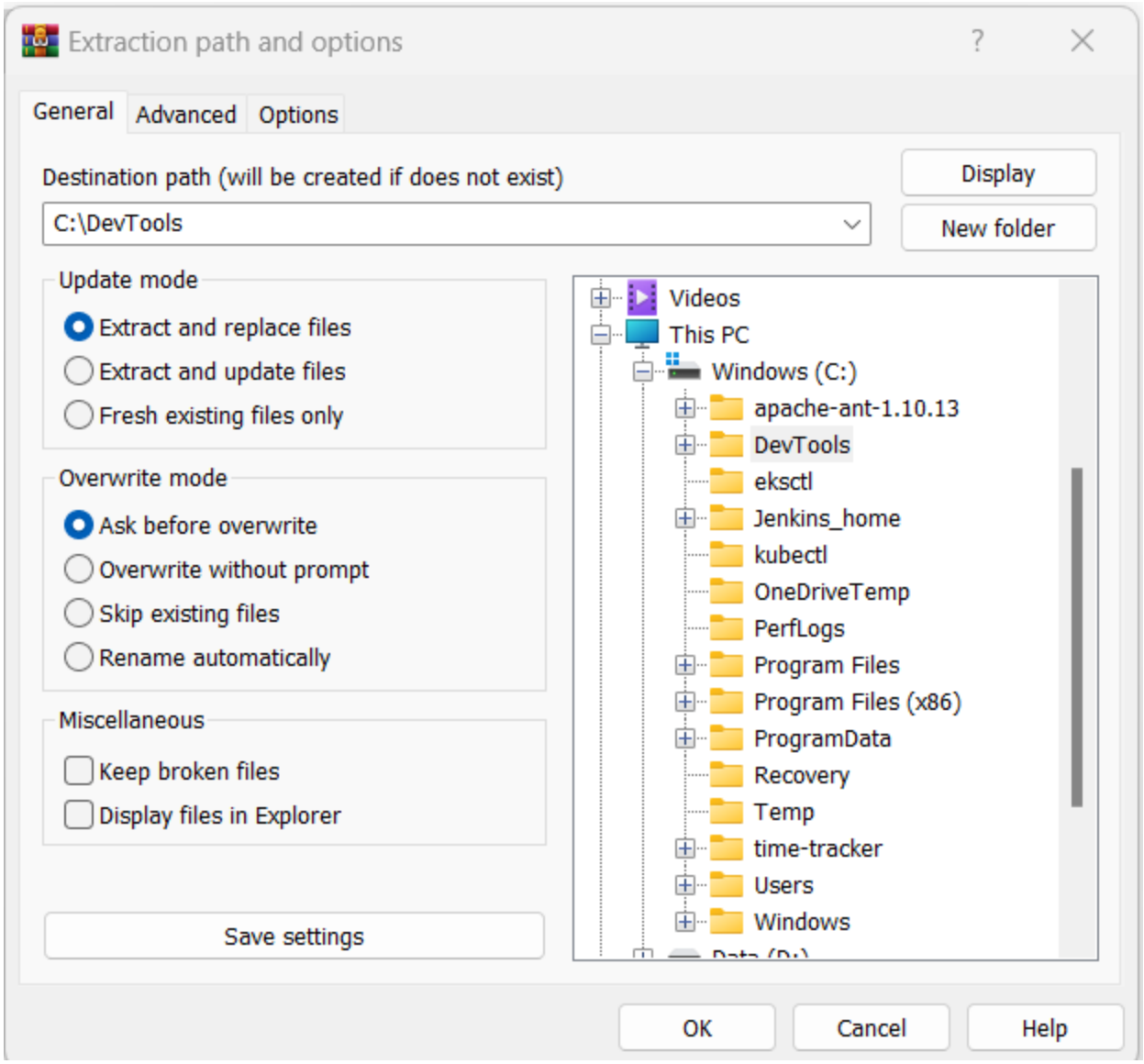
Files

Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the installation instructions. Use a source archive if you intend to build Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to verify the signature of the release bundles against the public KEYS used by the Apache Maven developers.

	Link	Checksums	Signature
Binary tar.gz archive	apache-maven-3.9.4-bin.tar.gz	apache-maven-3.9.4-bin.tar.gz.sha512	apache-maven-3.9.4-bin.tar.gz.asc
Binary zip archive	apache-maven-3.9.4-bin.zip	apache-maven-3.9.4-bin.zip.sha512	apache-maven-3.9.4-bin.zip.asc

- ➢ Extract Files -> C:\DevTools



- ➢ Go to C:\DevTools\apache-maven-3.9.4 –- Copy the path
- ➢ Now Search “Edit System Environment Variables”
- ➢ System Variable -> New
- ➢ Variable Name = M2_HOME
- ➢ Variable Value = Paste the path
- ➢ Now, Go inside C:\DevTools\apache-maven-3.9.4\bin —--Copy the path
- ➢ Now again go to “Edit Environment Variables”
- ➢ System Variable -> Path -> New -> Paste the path
- ➢ Now open Command Prompt

➢ # maven --version

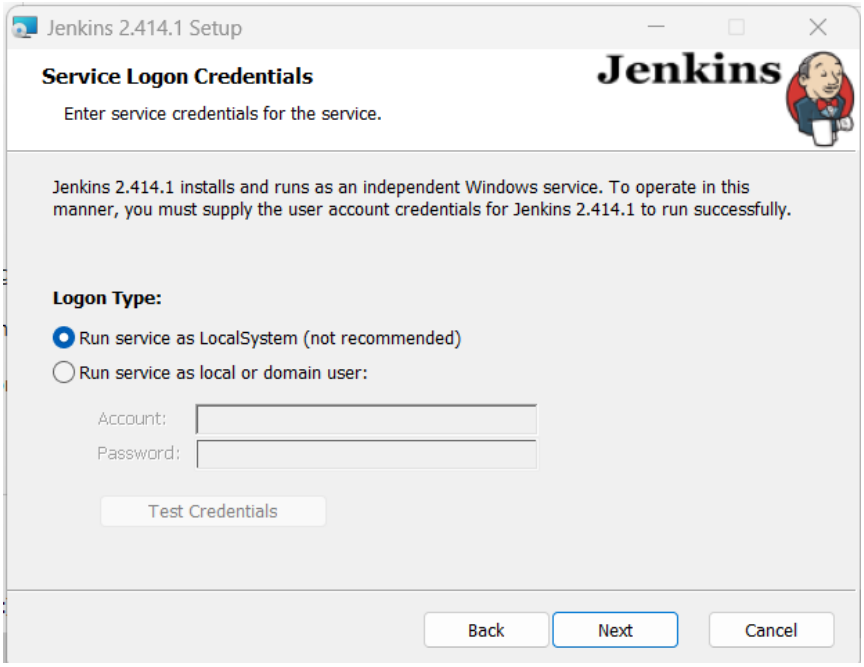
```
C:\Users\princ>mvn --version
Apache Maven 3.9.4 (dfbb324ad4a7c8fb0bf182e6d91b0ae20e3d2dd9)
Maven home: C:\DevTools\apache-maven-3.9.4
Java version: 20.0.2, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk-20
Default locale: en_IN, platform encoding: UTF-8
OS name: "windows 11", version: "10.0", arch: "amd64", family: "windows"
```

➢ Now Restart the Laptop

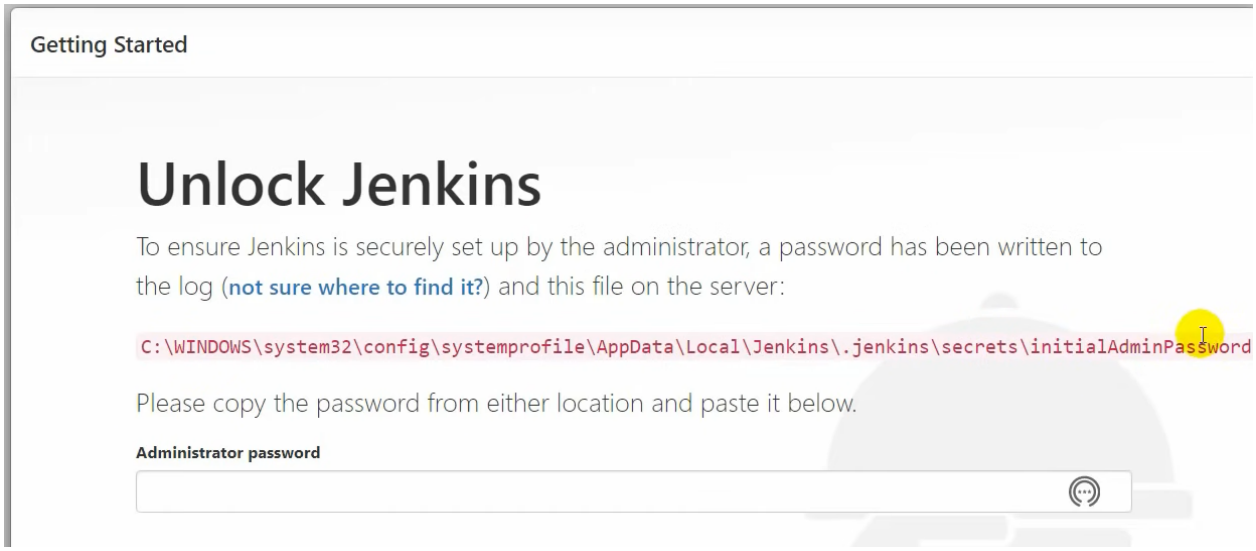
Jenkins Installation

➢ Go to Google Chrome
➢ Search jenkins.io
➢ URL : https://www.jenkins.io/
➢ Go to Download : https://www.jenkins.io/download/
➢ **Download Jenkins 2.414.1 LTS for:**
➢ Select LTS -> Windows - > Download
➢ URL : https://www.jenkins.io/download/thank-you-downloading-windows-installer-stable/

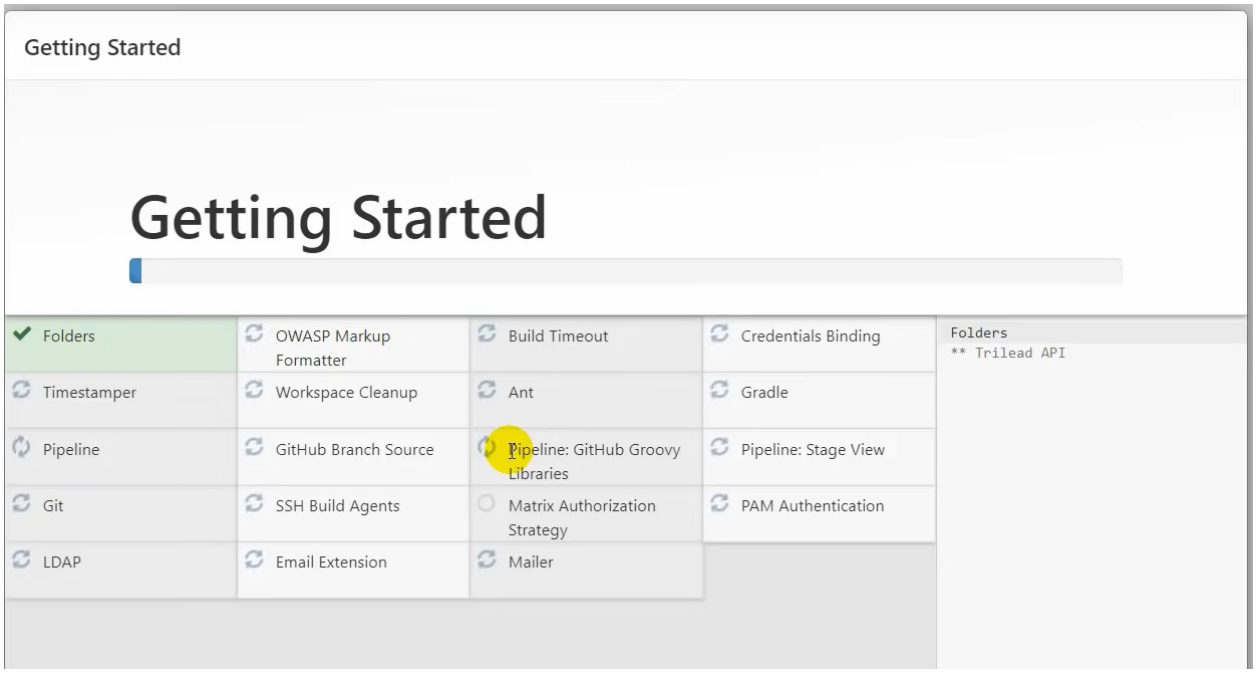
➢ Open Downloaded file -> Run and Install



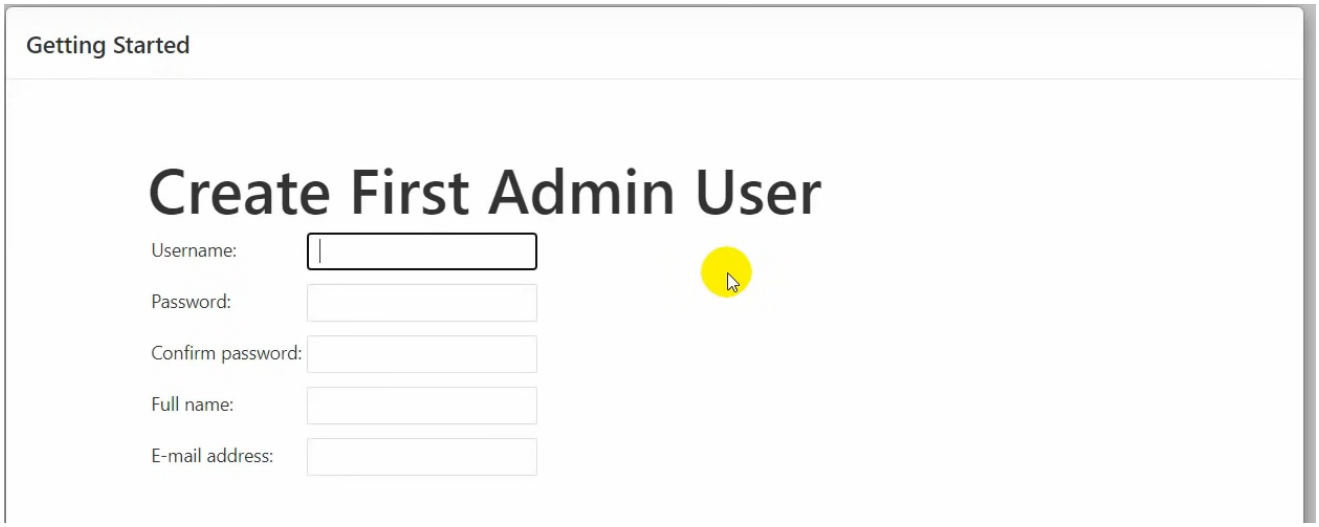
➢ After Installation it automatically open as localhost:8080



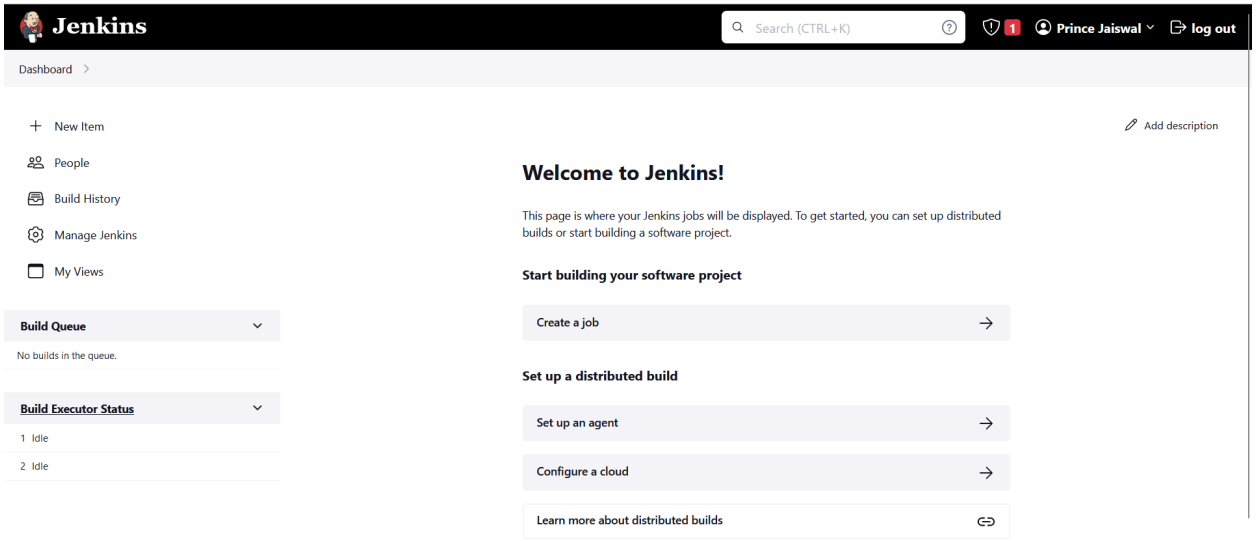
➢ Unlock the page by using password
➢ Now install suggested plugins



➢ Ask for username & password



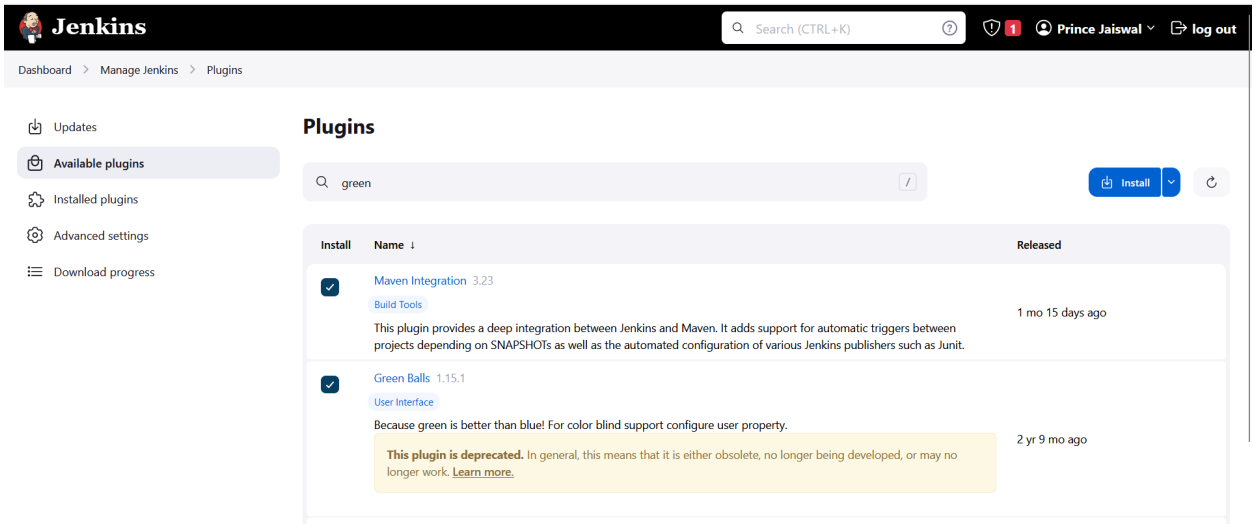
➢ Username : admin
➢ Password : admin
➢ Name: Prince Jaiswal
➢ Email : princejaiswal0007@gmail.com
➢ Save & Continue
➢ Start using Jenkins



Plugins: Plugins are small libraries that add new abilities to jenkins and can provide integration points to other tools.

Maven Job, Schedule Task, Poll SCM

- Go to google chrome -> localhost:8080 -> login
- Go to Manage Jenkins on left side of Jenkins Dashboard
- Manage Plugins
- Available
- Select Maven Integration & Green balls
- Install without Restart



- Go to New Item
- Maven Project

- Now go to Manage Jenkins
- Tools
- Go to Add JDK
- [] Uncheck this install Automatically option
- NAME = JAVA
- JAVA_HOME = c:\programfiles\java\jdk
- Now go to MAVEN
- Name = MAVEN
- MAVEN_HOME = c:\DevTools\apache-maven

MAVEN PROJECT (By Maven)

- Go to https://github.com/technicalguftgu/time-tracker
- Click on time-tracker repo
- “Fork” to copy this repo
- Sign-in into your github account
- Click on time-tracker repo
- Clone
- To to C drive
- # git clone <url of time tracker repo>

```
C:\Users\bhupi>CD ../..

C:\>git clone https://github.com/technicalguftgu/time-tracker.git
Cloning into 'time-tracker'...
remote: Enumerating objects: 318, done.
remote: Total 318 (delta 0), reused 0 (delta 0), pack-reused 318 eceiving objects:  67% (214/318)
Receiving objects: 100% (318/318), 77.69 KiB | 2.04 MiB/s, done.
Resolving deltas: 100% (97/97), done.
```

- # cd time-tracker
- # c:\time-tracker > mvn clean package
-

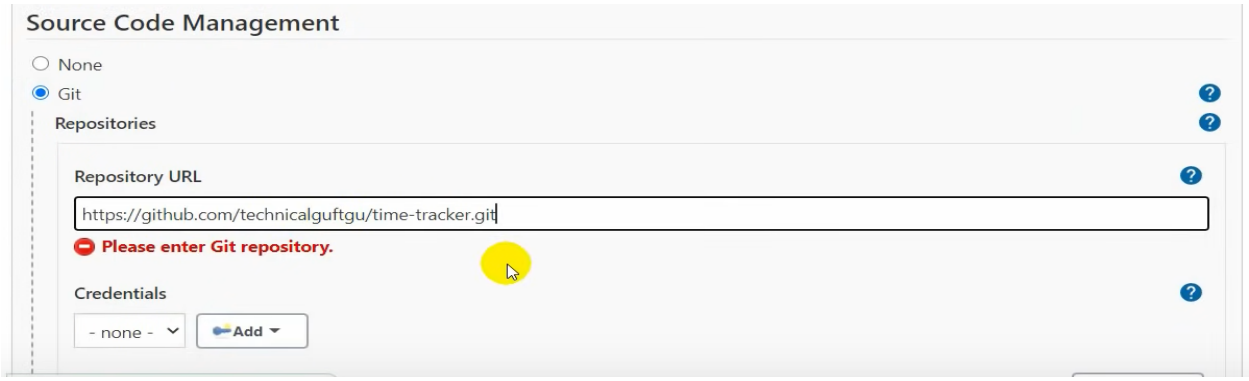
```
C:\>cd time-tracker

C:\time-tracker>mvn clean package
[INFO] Scanning for projects...
[INFO] ------------------------------------------------------------------------
[INFO] Reactor Build Order:
[INFO]
[INFO] Time Tracker (Parent)                                              [pom]
[INFO] Time Tracker (Core)                                                [jar]
[INFO] Time Tracker (Web)                                                 [war]
[INFO]
[INFO] ----------< training.taylor.time-tracker:time-tracker-parent >----------
[INFO] Building Time Tracker (Parent) 0.5.0-SNAPSHOT                      [1/3]
[INFO] --------------------------------[ pom ]---------------------------------
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/m
```

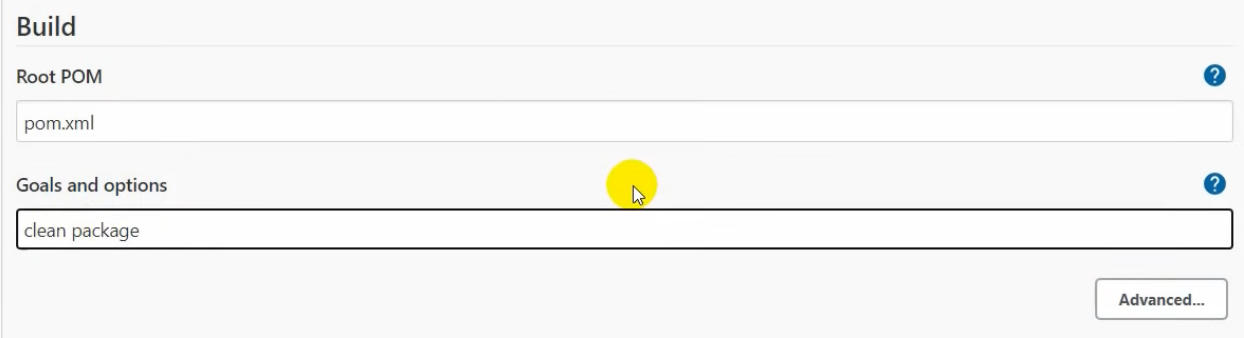
MAVEN PROJECT (By Jenkins)

- Now go to Jenkins
- New Item
- Enter Project Name -> MyMavenProject
- Then select Maven Project - > OK

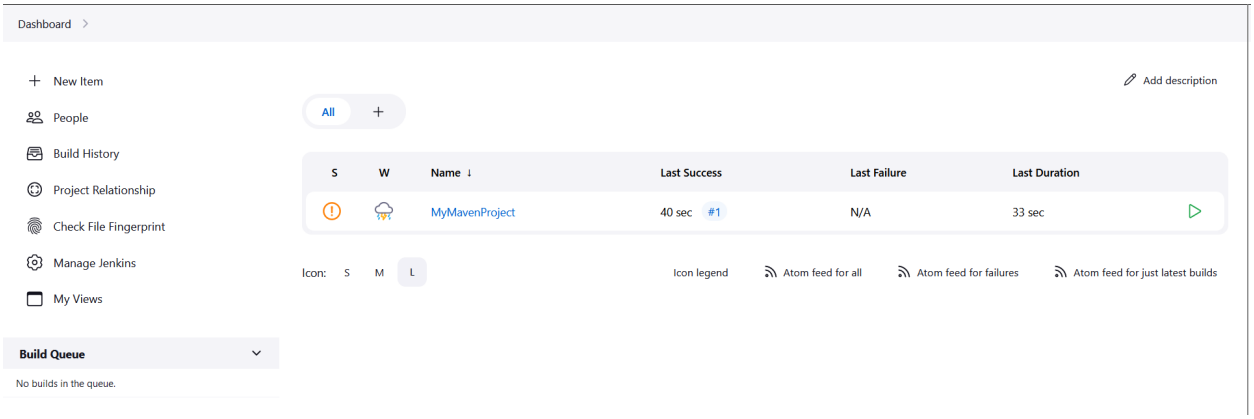
➢ Source Code Management -> git



➢ Repository URL
➢ Build Option -. Root POM -> pom.xml
➢ Goals & Options -> Clean Package -> Save

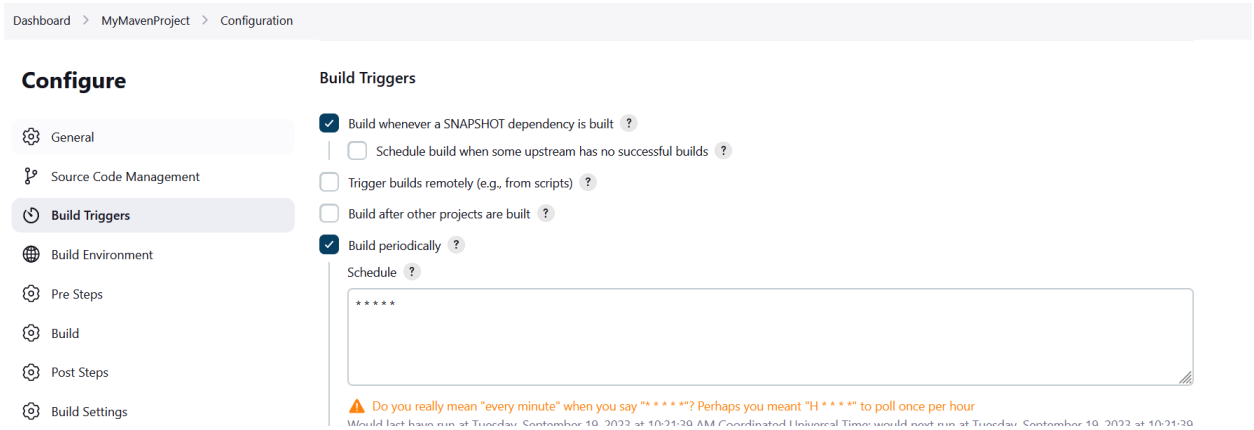


➢ Go to Jenkins home page
➢ Click on MyMavenProject
➢ Build Now



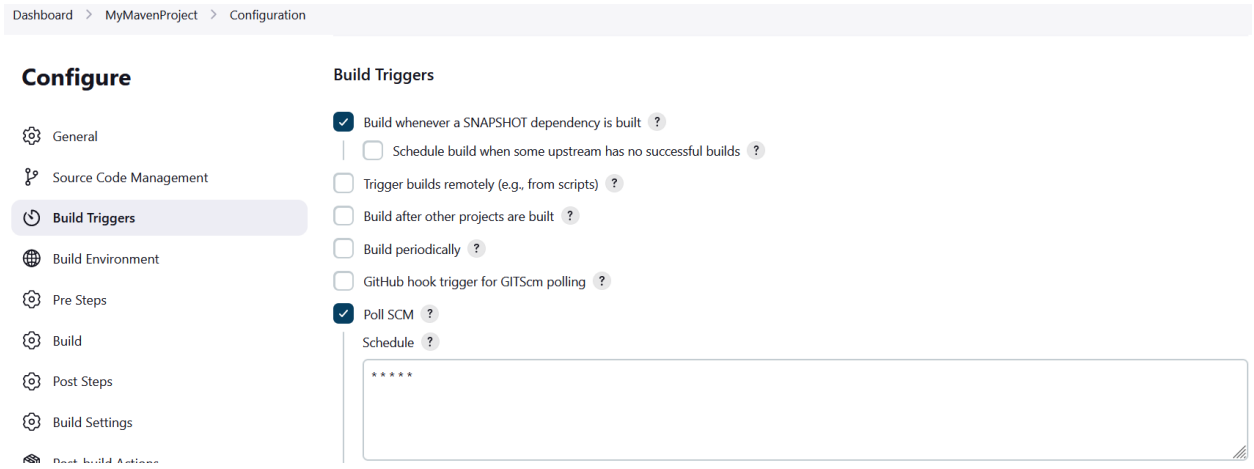
Scheduled Project

- Click on any project
- Configure
- Build triggers
- Build Periodically
- * * * * *
- Save
- Can see automatic builds after every 1 minute
- You can manually trigger build as well



Source Code Polling (Poll SCM)

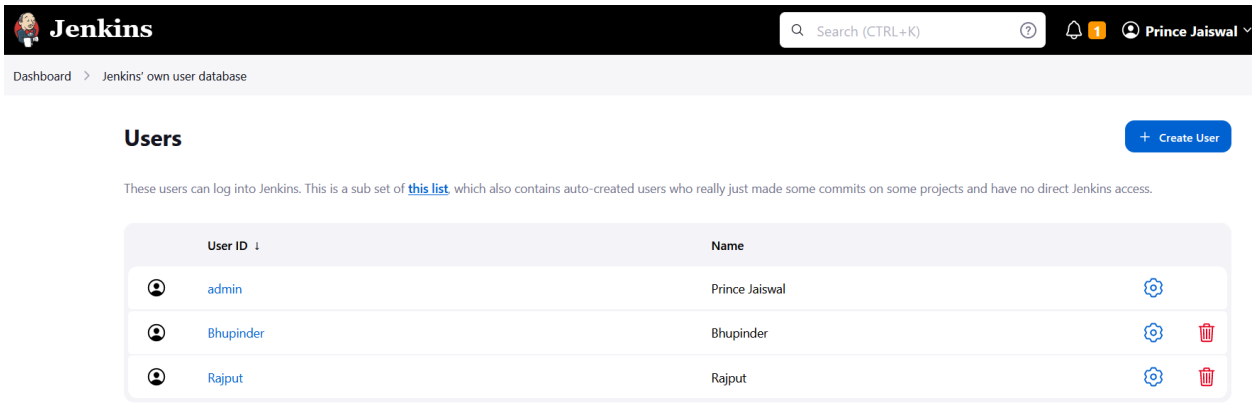
- Now go to Jenkins HomePage
- Go to MyMavenProject
- Configure
- Now go to build trigger
- [V] Poll SCM
- Schedule [* * * * *] –> Save



- Now go to github account > do some changes in README.md > Commit changes
- You can see, after 1 min, It build automatically.

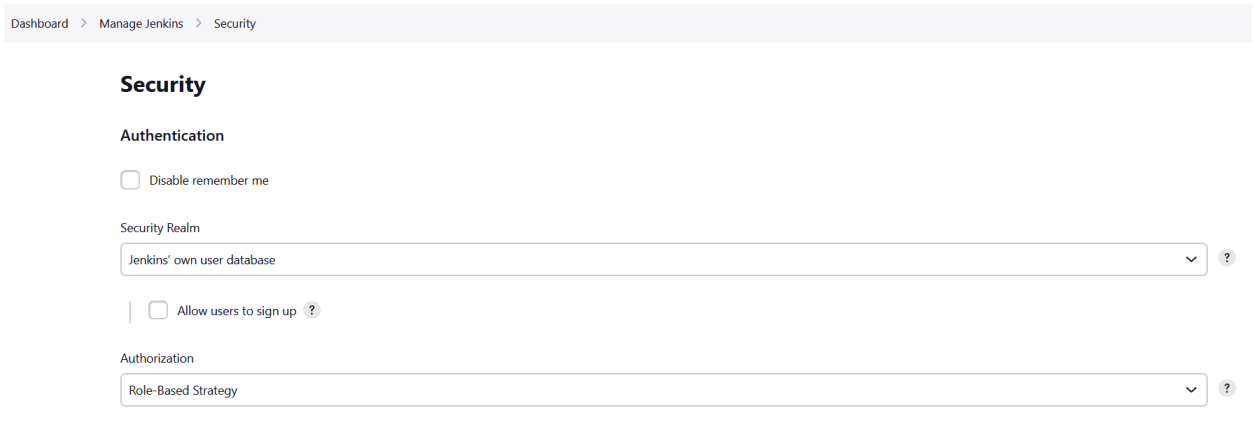
Linked Project, Views, User Management & Master Slave Concept

- Go to Jenkins Homepage
- Manage Jenkins
- User
- Create two users
- Bhupinder & Rajput
- Now login as Bhupinder

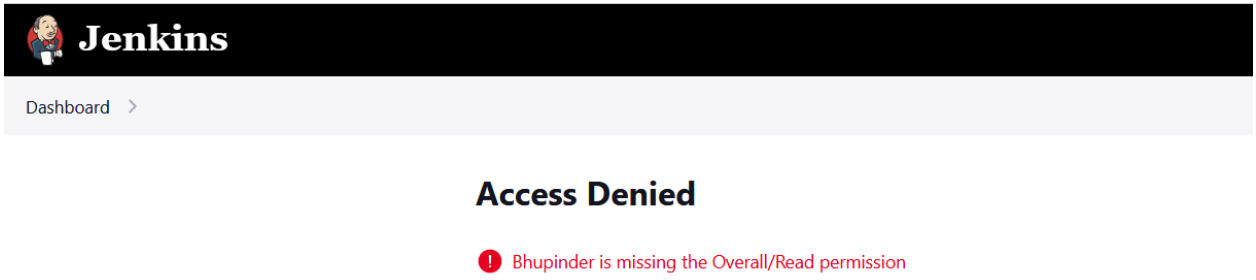


- (By default, you have all the permissions)
- Login as “admin” again
- Go to Manage Jenkins
- Plugins
- Search “Role-based Authorization Strategy”
- Install without restart

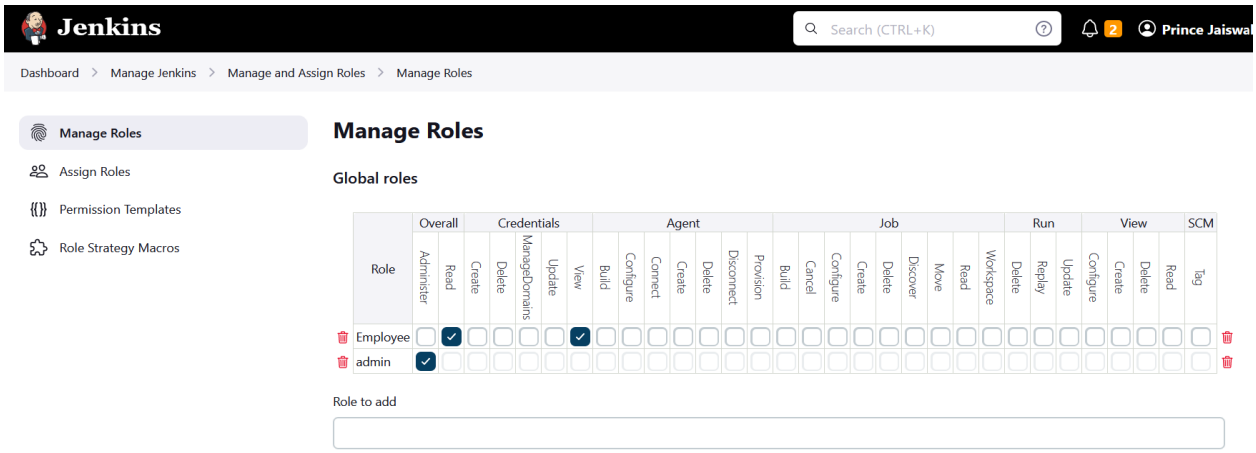
- Go to jenkins home
- Manage Jenkins
- Security
- Select Role Based Strategy
- Save
- Login as “Bhupinder” > Access denied



➢ Login as “Bhupinder” > Access denied



➢ Now, Attach permissions
➢ Go to Jenkins
➢ Manage Jenkins
➢ Manage and Assign Roles
➢ Manage Roles
➢ Role to Add

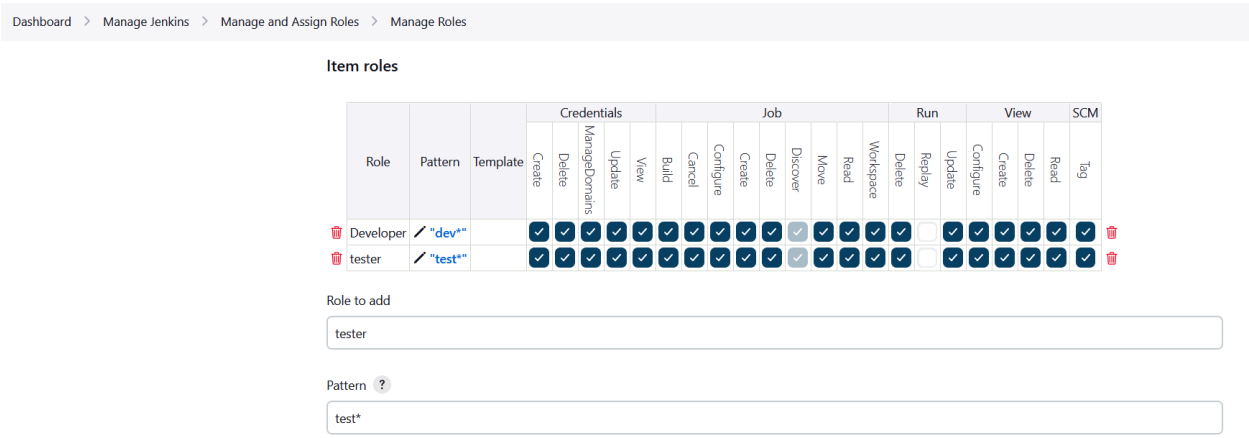


➢ Employee

➢ Scroll down and go to Item roles

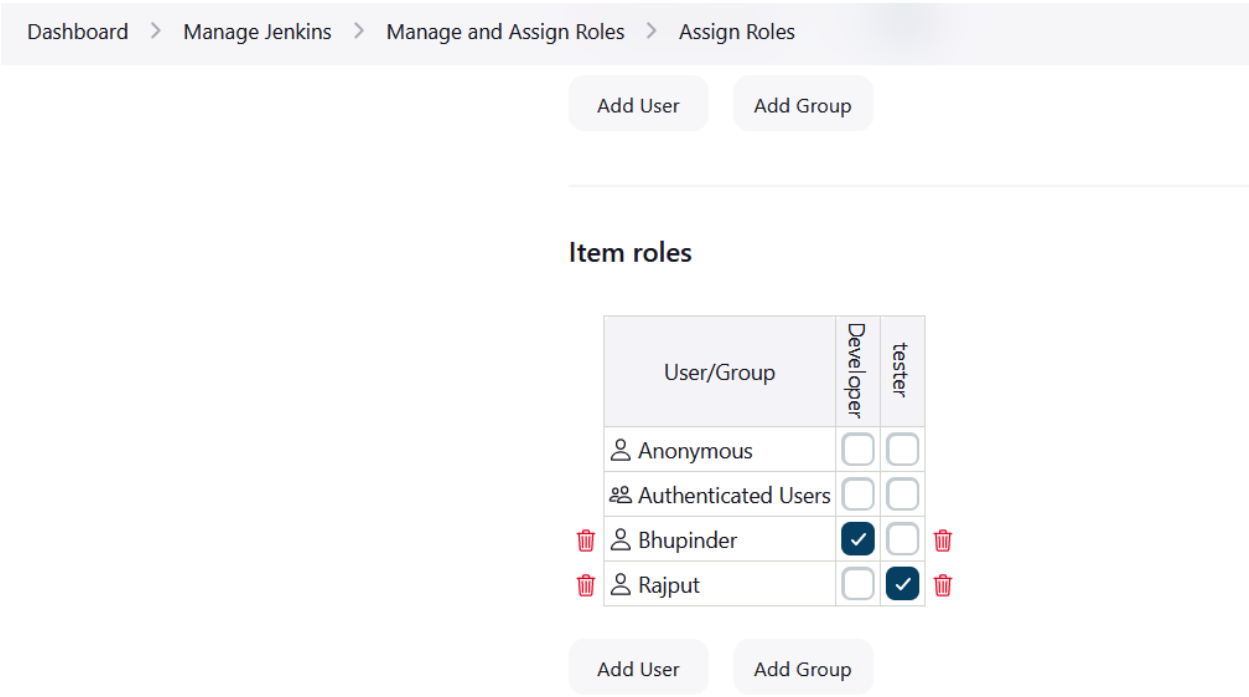
➢ Add Developer & Tester

Pattern—> dev* test*

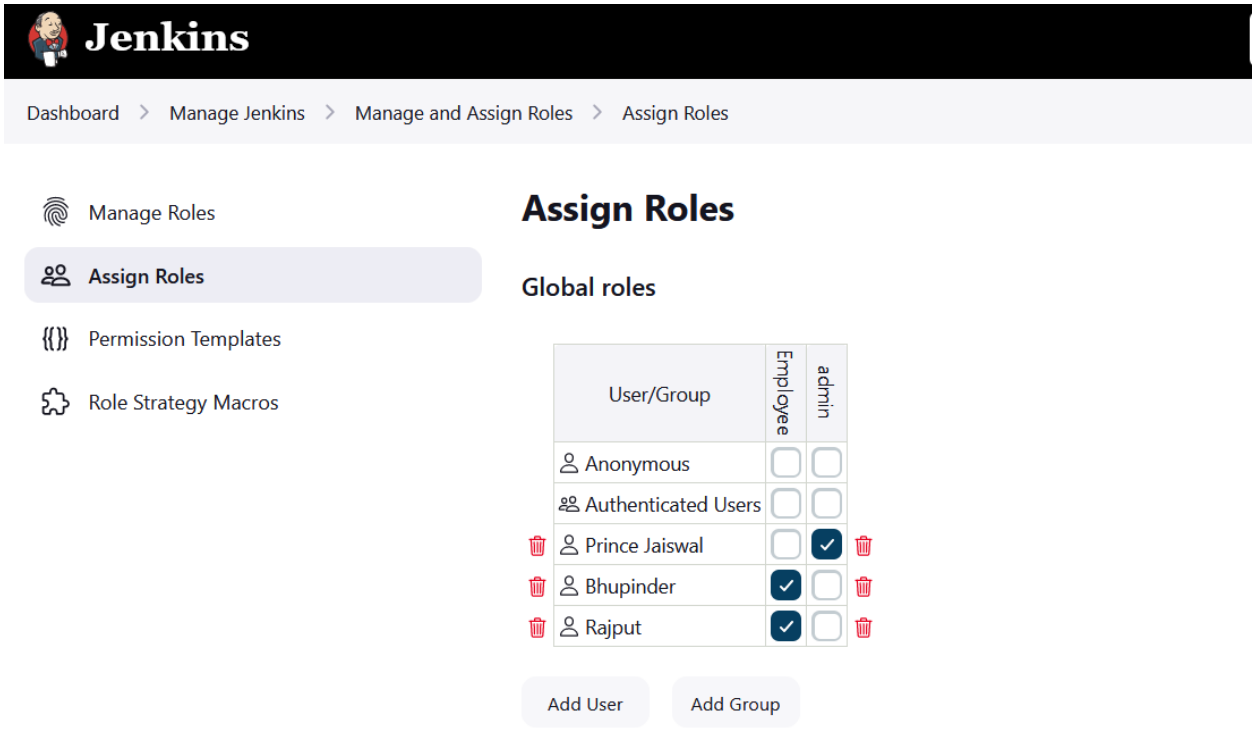


Then Go to Assign Roles

User/group to add Bhupinder & Rajput



Now add Bhupinder 7 Rajput in Global Roles Add user and marked as Employee:



How to Install Jenkins on Ubuntu

- Login to AWS account and create one Ubuntu Instance.
- Access it through Putty and login as ubuntu
- # sudo apt-get update -y
- # sudo apt-cache search openjdk
- # sudo apt-get install openjdk-17-jdk -y
- # java --version

```
ubuntu@ip-172-31-0-138:~$ java --version
openjdk 17.0.8.1 2023-08-24
OpenJDK Runtime Environment (build 17.0.8.1+1-Ubuntu-0ubuntu122.04)
OpenJDK 64-Bit Server VM (build 17.0.8.1+1-Ubuntu-0ubuntu122.04, mixed mode, sharing)
```

- # sudo vi /etc/apt/sources.list
- Paste at the bottom
- deb https://pkg.jenkins.io/debian-stable binary/

```
## of updates from the Ubuntu security team.
deb http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/ jammy-backports main re
# deb-src http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/ jammy-backports r

deb http://security.ubuntu.com/ubuntu jammy-security main restricted
# deb-src http://security.ubuntu.com/ubuntu jammy-security main restricted
deb http://security.ubuntu.com/ubuntu jammy-security universe
# deb-src http://security.ubuntu.com/ubuntu jammy-security universe
deb http://security.ubuntu.com/ubuntu jammy-security multiverse
# deb-src http://security.ubuntu.com/ubuntu jammy-security multiverse
deb https://pkg.jenkins.io/debian-stable binary/
:wq!
```

➢ # sudo apt-get update -y

```
ubuntu@ip-172-31-0-138:~$ sudo apt-get update -y
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign:4 https://pkg.jenkins.io/debian-stable binary/ InRelease
Get:5 https://pkg.jenkins.io/debian-stable binary/ Release [2044 B]
Get:6 https://pkg.jenkins.io/debian-stable binary/ Release.gpg [833 B]
Hit:7 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign:6 https://pkg.jenkins.io/debian-stable binary/ Release.gpg
Reading package lists... Done
W: GPG error: https://pkg.jenkins.io/debian-stable binary/ Release: The following signatures coul
y is not available: NO_PUBKEY 5BA31D57EF5975CA
E: The repository 'https://pkg.jenkins.io/debian-stable binary/ Release' is not signed.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
```

➢ # 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

```
ubuntu@ip-172-31-0-138:~$ 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
ubuntu@ip-172-31-0-138:~$ sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign:4 https://pkg.jenkins.io/debian-stable binary/ InRelease
Get:5 https://pkg.jenkins.io/debian-stable binary/ Release [2044 B]
Get:6 https://pkg.jenkins.io/debian-stable binary/ Release.gpg [833 B]
Hit:7 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:8 https://pkg.jenkins.io/debian-stable binary/ Packages [25.4 kB]
Fetched 28.3 kB in 1s (39.1 kB/s)
Reading package lists... Done
W: Target Packages (Packages) is configured multiple times in /etc/apt/sources.list:51 and /etc/apt/sources.list.d/jenkins.list:1
W: Target Translations (en) is configured multiple times in /etc/apt/sources.list:51 and /etc/apt/sources.list.d/jenkins.list:1
W: Target Packages (Packages) is configured multiple times in /etc/apt/sources.list:51 and /etc/apt/sources.list.d/jenkins.list:1
W: Target Translations (en) is configured multiple times in /etc/apt/sources.list:51 and /etc/apt/sources.list.d/jenkins.list:1
```

➢ # sudo apt-get install jenkins

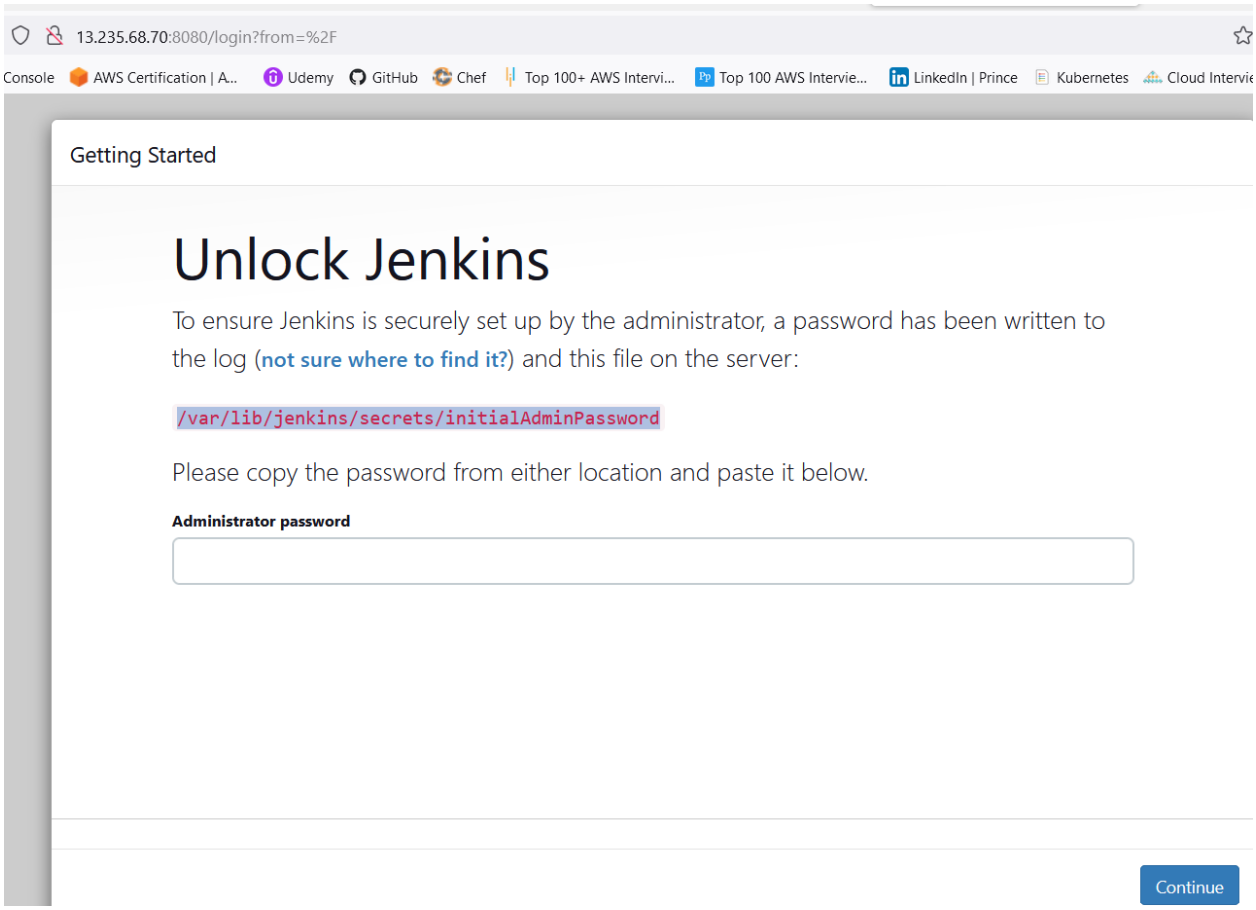
```
ubuntu@ip-172-31-0-138:~$ sudo apt-get install jenkins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  net-tools
The following NEW packages will be installed:
  jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 126 not upgraded.
Need to get 89.1 MB of archives.
After this operation, 90.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 net-tools amd64 1.60+git20181103.0eebece-1ubuntu5 [204 kB]
Get:2 https://pkg.jenkins.io/debian-stable binary/ jenkins 2.414.1 [88.9 MB]
Fetched 89.1 MB in 10s (8637 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 80619 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-1ubuntu5_amd64.deb ...
Unpacking net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Selecting previously unselected package jenkins.
Preparing to unpack .../jenkins_2.414.1_all.deb ...
Unpacking jenkins (2.414.1) ...
Setting up net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Setting up jenkins (2.414.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /lib/systemd/system/jenkins.service.
```

- # sudo service jenkins status
- # q

```
ubuntu@ip-172-31-0-138:~$ sudo service jenkins status
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
     Active: active (running) since Tue 2023-09-19 09:40:58 UTC; 21s ago
   Main PID: 5859 (java)
      Tasks: 44 (limit: 1141)
     Memory: 287.1M
        CPU: 38.303s
     CGroup: /system.slice/jenkins.service
             └─5859 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPor>

Sep 19 09:40:28 ip-172-31-0-138 jenkins[5859]: b190774d47ea47378609db8084fdd342
Sep 19 09:40:28 ip-172-31-0-138 jenkins[5859]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Sep 19 09:40:28 ip-172-31-0-138 jenkins[5859]: ************************************************************
Sep 19 09:40:28 ip-172-31-0-138 jenkins[5859]: ************************************************************
Sep 19 09:40:28 ip-172-31-0-138 jenkins[5859]: ************************************************************
Sep 19 09:40:58 ip-172-31-0-138 jenkins[5859]: 2023-09-19 09:40:58.761+0000 [id=31]        INFO        jenkins.InitReactorRunner$1#onA>
Sep 19 09:40:58 ip-172-31-0-138 jenkins[5859]: 2023-09-19 09:40:58.780+0000 [id=24]        INFO        hudson.lifecycle.Lifecycle#onRe>
Sep 19 09:40:58 ip-172-31-0-138 systemd[1]: Started Jenkins Continuous Integration Server.
Sep 19 09:40:59 ip-172-31-0-138 jenkins[5859]: 2023-09-19 09:40:59.689+0000 [id=46]        INFO        h.m.DownloadService$Downloadabl>
Sep 19 09:40:59 ip-172-31-0-138 jenkins[5859]: 2023-09-19 09:40:59.691+0000 [id=46]        INFO        hudson.util.Retrier#start: Perf>
```

- Copy public ip from aws and paste in chrome browser url with 8080 port.



- Go to instance
- # sudo cat /var/lib/jenkins/secrets/initialAdminPassword

```
ubuntu@ip-172-31-0-138:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
b190774d47ea47378609db8084fdd342
ubuntu@ip-172-31-0-138:~$
```

To install Java and Maven on Ubuntu machine follow below website

https://www.digitalocean.com/community/tutorials/install-maven-linux-ubuntu

Notes Created By: Prince Jaiswal (princejaiswal0007@gmail.com)