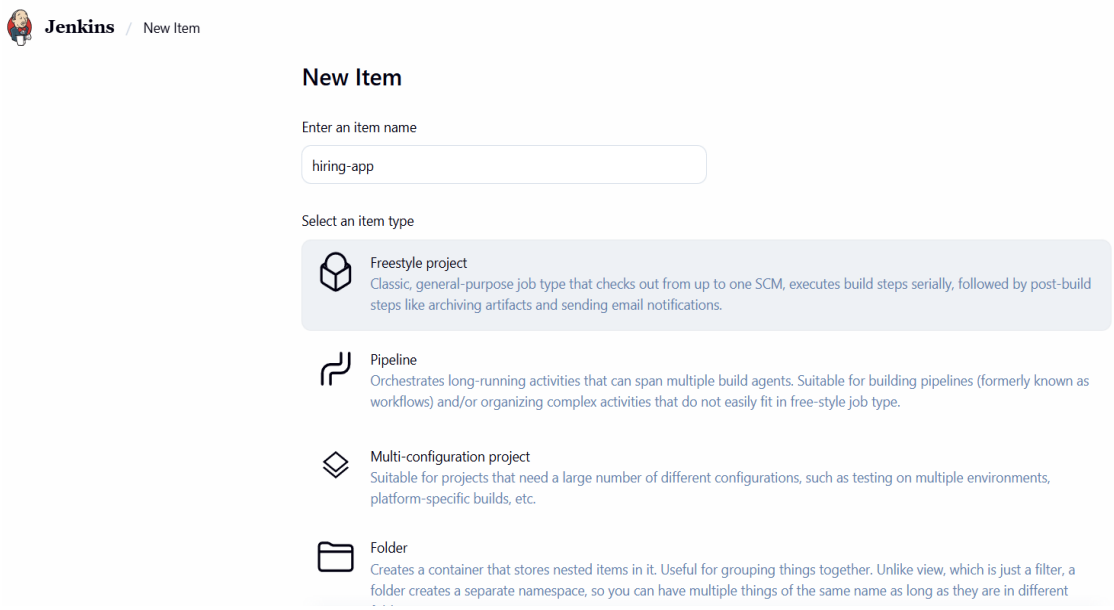
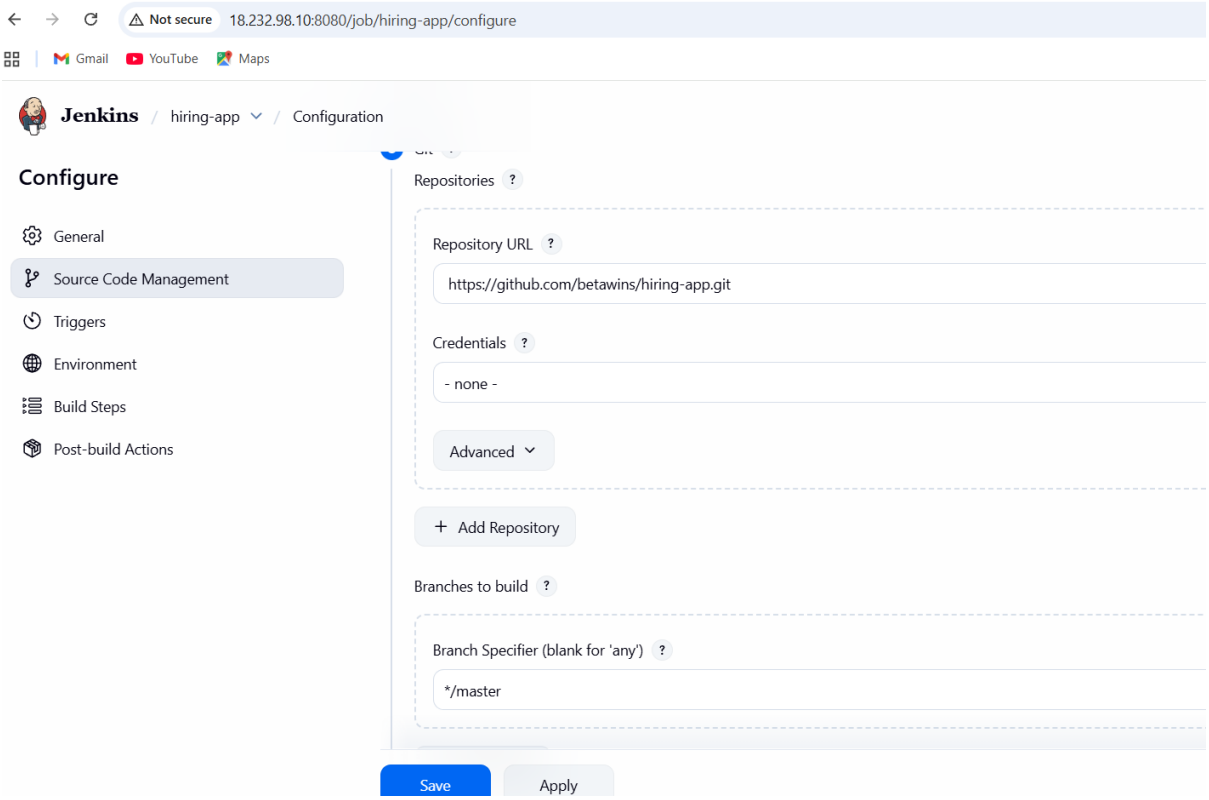
1. **Setup Jenkins CICD pipeline using freestyle job using Docker containers using below code. https://github.com/betawins/hiring-app.git Stages:**
    a. **Git Clone**
    b. **SonarQube Integration**
    c. **Maven Compilation**
    d. **Nexus Artifactory**
    e. **Slack Notification**
    f. **Deploy On tomcat**

a. Git Clone:

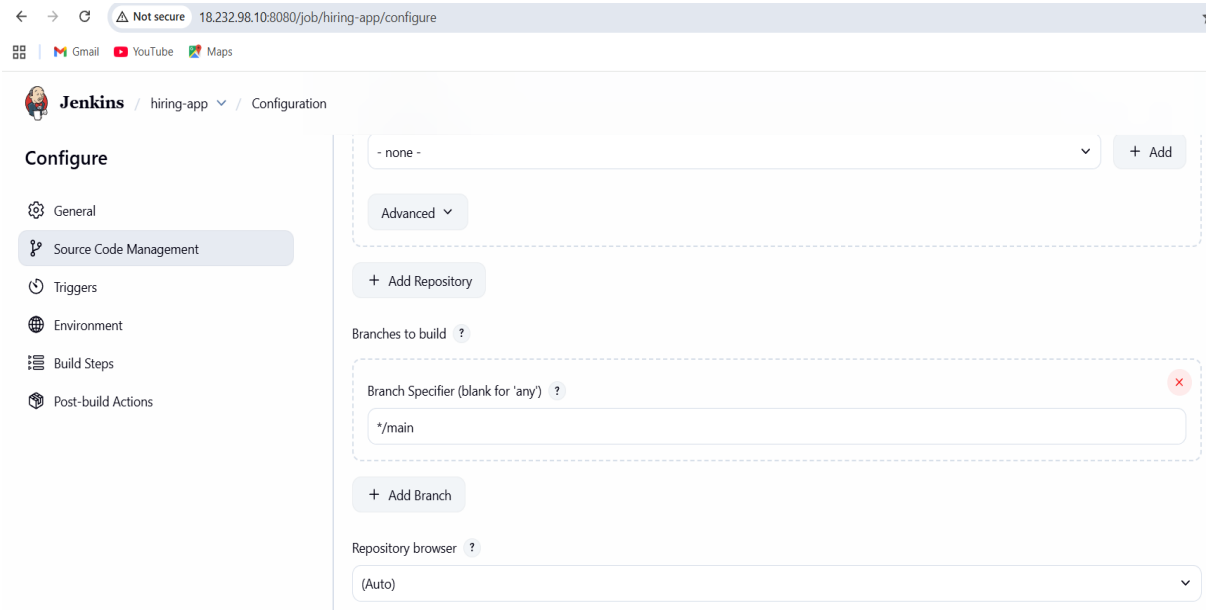
go to items and give the name and select freestyle job.



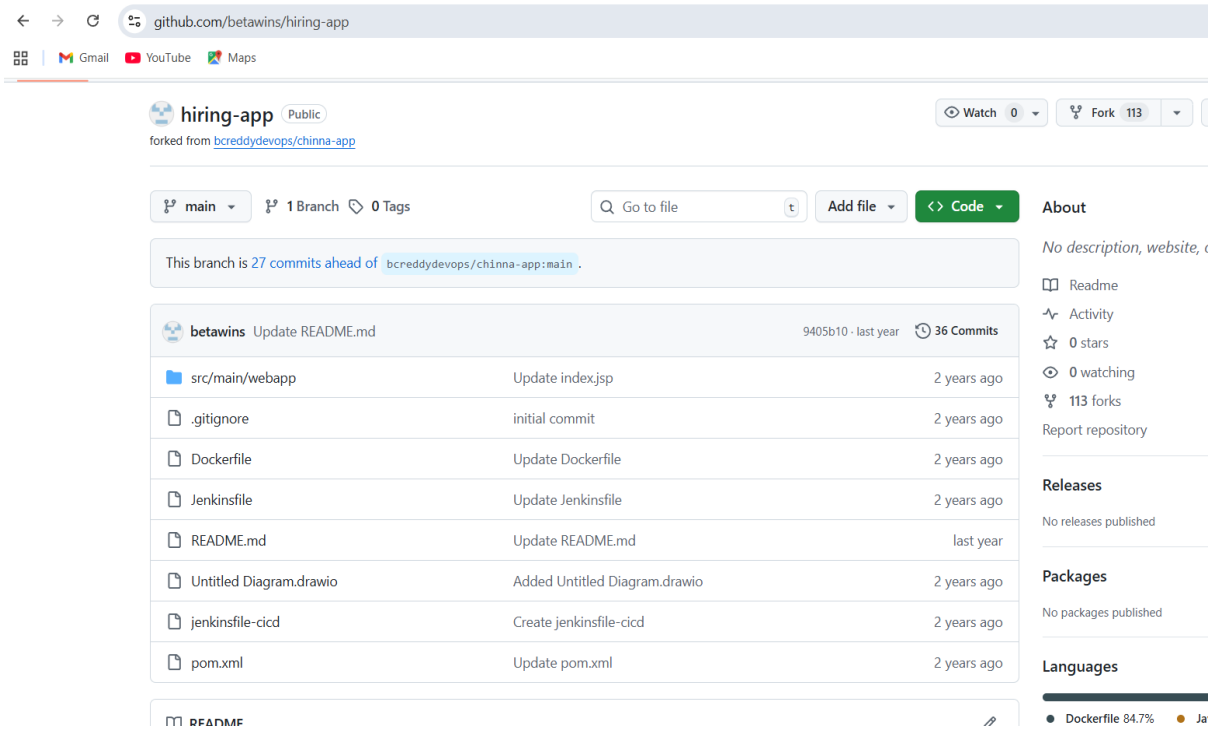
In the git repository give your repository https://github.com/betawins/hiring-app.git and save it.



In branch select as main and save it.



Click on build now you will see your repo will clone to local.




```
[root@ip-172-31-50-64 ~]# cd /var/lib/jenkins
[root@ip-172-31-50-64 jenkins]# ls
config.xml                          jenkins.install.UpgradeWizard.state          nodeMonitors.xml         secrets
hudson.model.UpdateCenter.xml       jenkins.model.JenkinsLocationConfiguration.xml plugins                updates
hudson.plugins.git.GitTool.xml      jenkins.telemetry.Correlator.xml             queue.xml                userContent
identity.key.enc                    jobs                                         secret.key               users
jenkins.install.InstallUtil.lastExecVersion logs                                 secret.key.not-so-secret workspace
[root@ip-172-31-50-64 jenkins]# cd workspace
[root@ip-172-31-50-64 workspace]# ls
hiring-app
[root@ip-172-31-50-64 workspace]# cd hiring-app/
[root@ip-172-31-50-64 hiring-app]# ls
 Dockerfile   Jenkinsfile   README.md  'Untitled Diagram.drawio'   jenkinsfile-cicd   pom.xml   src
[root@ip-172-31-50-64 hiring-app]#
```

b. SonarQube Integration:

* yum install docker -y

```
[root@ip-172-31-16-189 ~]# sudo dnf install docker -y
Amazon Linux 2023 Kernel Livepatch repository
Dependencies resolved.
================================================================================================
 Package                                    Architecture                  Version
================================================================================================
Installing:
 docker                                     x86_64                        25.0.13-1.amzn2023
Installing dependencies:
 container-selinux                          noarch                        4:2.242.0-1.amzn20
 containerd                                 x86_64                        2.1.4-1.amzn2023.0
 iptables-libs                              x86_64                        1.8.8-3.amzn2023.0
 iptables-nft                               x86_64                        1.8.8-3.amzn2023.0
 libcgroup                                  x86_64                        3.0-1.amzn2023.0.1
 libnetfilter_conntrack                     x86_64                        1.0.8-2.amzn2023.0
 libnfnetlink                               x86_64                        1.0.1-19.amzn2023.
 libnftnl                                   x86_64                        1.2.2-2.amzn2023.0
 pigz                                       x86_64                        2.5-1.amzn2023.0.3
 runc                                       x86_64                        1.3.1-1.amzn2023.0

Transaction Summary
================================================================================================
Install  11 Packages

Total download size: 74 M
```

- systemctl start docker
- docker network create sonarnet

  start postgres container by using

  docker run -d \

  --name postgres \

  --network sonarnet \

  -e POSTGRES_USER=sonar \

  -e POSTGRES_PASSWORD=sonar \

  -e POSTGRES_DB=sonarqube \

  -v postgres_data:/var/lib/postgresql/data \

  postgres:15

```
Complete!
[root@ip-172-31-16-189 ~]# systemctl start docker
[root@ip-172-31-16-189 ~]# docker --version
Docker version 25.0.13, build 0bab007
[root@ip-172-31-16-189 ~]# docker network create sonarnet
bf42a5d161ea7a2fddbf5dd9060d081eb10337be63c7ec8f5923d08898ac42aa
[root@ip-172-31-16-189 ~]# docker run -d \
  --name postgres \
  --network sonarnet \
  -e POSTGRES_USER=sonar \
  -e POSTGRES_PASSWORD=sonar \
  -e POSTGRES_DB=sonarqube \
  -v postgres_data:/var/lib/postgresql/data \
  postgres:15
Unable to find image 'postgres:15' locally
15: Pulling from library/postgres
38513bd72563: Pull complete
7cf8703a60ae: Pull complete
1c3c3432cfaa: Pull complete
d5997a3d3e9c: Pull complete
4b34d7a7ab2a: Pull complete
0cdc9eb2a585: Pull complete
f0e812253d9a: Pull complete
70a82bc39e86: Pull complete
b12e04c2e850: Pull complete
bb86ed266eaf: Pull complete
9be150812fa2: Pull complete
```

- start sonarqube container by using

docker run -d \

  --name sonarqube \

  --network sonarnet \

  -p 9000:9000 \

  -e SONAR_JDBC_URL=jdbc:postgresql://postgres:5432/sonarqube \

  -e SONAR_JDBC_USERNAME=sonar \

  -e SONAR_JDBC_PASSWORD=sonar \

-v sonarqube_data:/opt/sonarqube/data \

-v sonarqube_extensions:/opt/sonarqube/extensions \

sonarqube:lts-community

```
[root@ip-172-31-16-189 ~]# docker run -d \
  --name sonarqube \
  --network sonarnet \
  -p 9000:9000 \
  -e SONAR_JDBC_URL=jdbc:postgresql://postgres:5432/sonarqube \
  -e SONAR_JDBC_USERNAME=sonar \
  -e SONAR_JDBC_PASSWORD=sonar \
  -v sonarqube_data:/opt/sonarqube/data \
  -v sonarqube_extensions:/opt/sonarqube/extensions \
  sonarqube:lts-community
Unable to find image 'sonarqube:lts-community' locally
lts-community: Pulling from library/sonarqube
60d98d907669: Pull complete
e24a8b9e652f: Pull complete
f3929ce9ef98: Pull complete
1df735f481ad: Pull complete
5d5a1fad7028: Pull complete
eb27e3a98da1: Pull complete
c7ad1fe61e07: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:f709975ab31d2d08f5a3ae2dc73a31ee011afc8cf28845082c17c55d45df9df5
Status: Downloaded newer image for sonarqube:lts-community
```

Docker ps -a

```
[root@ip-172-31-16-189 ~]# docker ps -a
CONTAINER ID   IMAGE                     COMMAND                  CREATED          STATUS                      PORTS      NAMES
f6bd39c9bfaf   sonarqube:lts-community   "/opt/sonarqube/dock…"   20 seconds ago   Exited (0) 11 seconds ago              sonarqube
4aebb18858a6   postgres:15               "docker-entrypoint.s…"   38 seconds ago   Up 37 seconds               5432/tcp   postgres
[root@ip-172-31-16-189 ~]# docker logs sonarqube
2025.10.30 07:19:47 INFO  app[][o.s.a.AppFileSystem] Cleaning or creating temp directory /opt/sonarqube/temp
2025.10.30 07:19:47 INFO  app[][o.s.a.es.EsSettings] Elasticsearch listening on [HTTP: 127.0.0.1:9001, TCP: 127.0.0.1:39785]
2025.10.30 07:19:47 INFO  app[][o.s.a.ProcessLauncherImpl] Launch process[ELASTICSEARCH] from [/opt/sonarqube/elasticsearch]: /opt/sona
rch
2025.10.30 07:19:47 INFO  app[][o.s.a.SchedulerImpl] Waiting for Elasticsearch to be up and running
2025.10.30 07:19:49 INFO  es[][o.e.n.Node] version[7.17.15], pid[26], build[default/tar/0b8ecfb4378335f4689c4223d1f1115f16bef3ba/2023-1
```

If the sonarqube exited with zero then increase the size of the instance by using

sudo sysctl -w vm.max_map_count=262144

sudo sysctl -w fs.file-max=65536

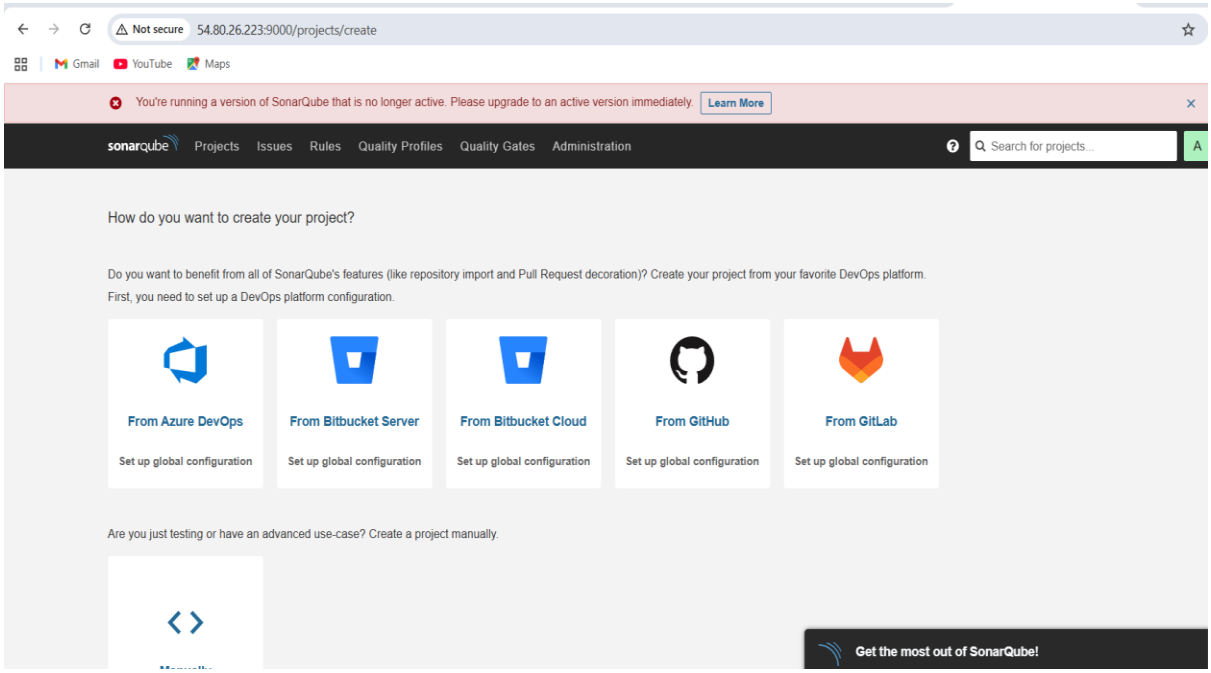
ulimit -n 65536

ulimit -u 4096

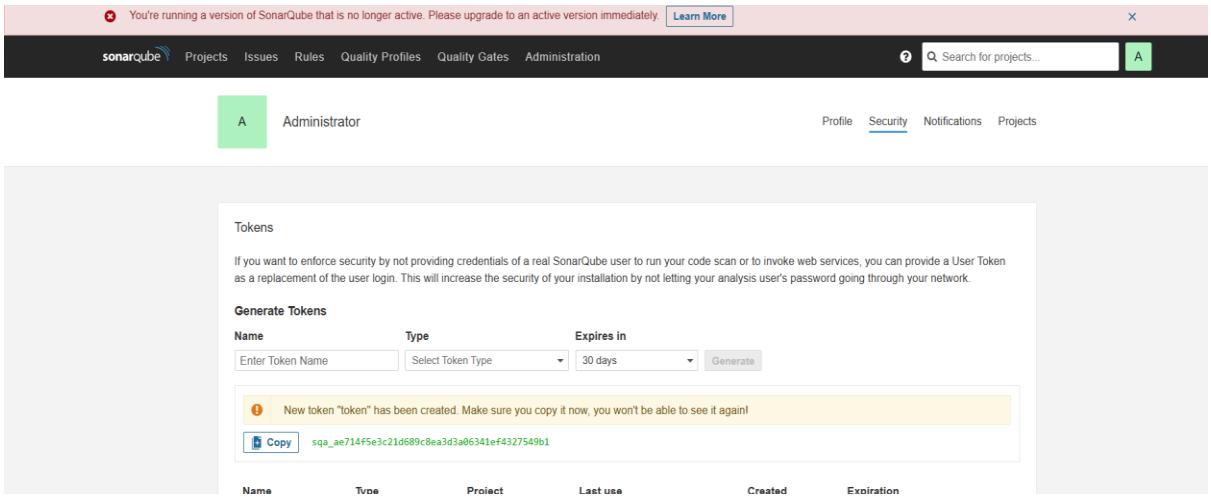
then docker start sonarqube

```
[root@ip-172-31-16-189 ~]# sudo sysctl -w vm.max_map_count=262144
sudo sysctl -w fs.file-max=65536
ulimit -n 65536
ulimit -u 4096
vm.max_map_count = 262144
fs.file-max = 65536
[root@ip-172-31-16-189 ~]# docker start sonarqube
sonarqube
[root@ip-172-31-16-189 ~]# docker start sonarqube
sonarqube
[root@ip-172-31-16-189 ~]# docker ps -a
CONTAINER ID   IMAGE                    COMMAND                  CREATED         STATUS          PORTS                                       NAMES
f6bd39c9bfaf   sonarqube:lts-community  "/opt/sonarqube/dock…"   2 minutes ago   Up 35 seconds   0.0.0.0:9000->9000/tcp, :::9000->9000/tcp   sonarqube
4aebb18858a6   postgres:15              "docker-entrypoint.s…"   2 minutes ago   Up 2 minutes    5432/tcp                                    postgres
[root@ip-172-31-16-189 ~]#
```

Check with ip in the browser

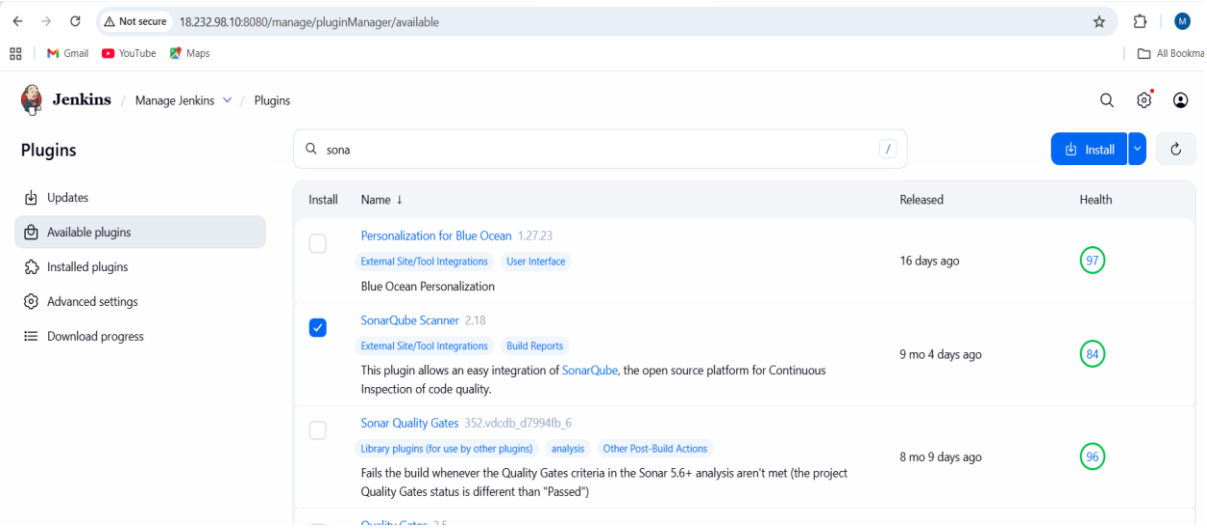


Create a token in sonarqube by going to account, security and create token

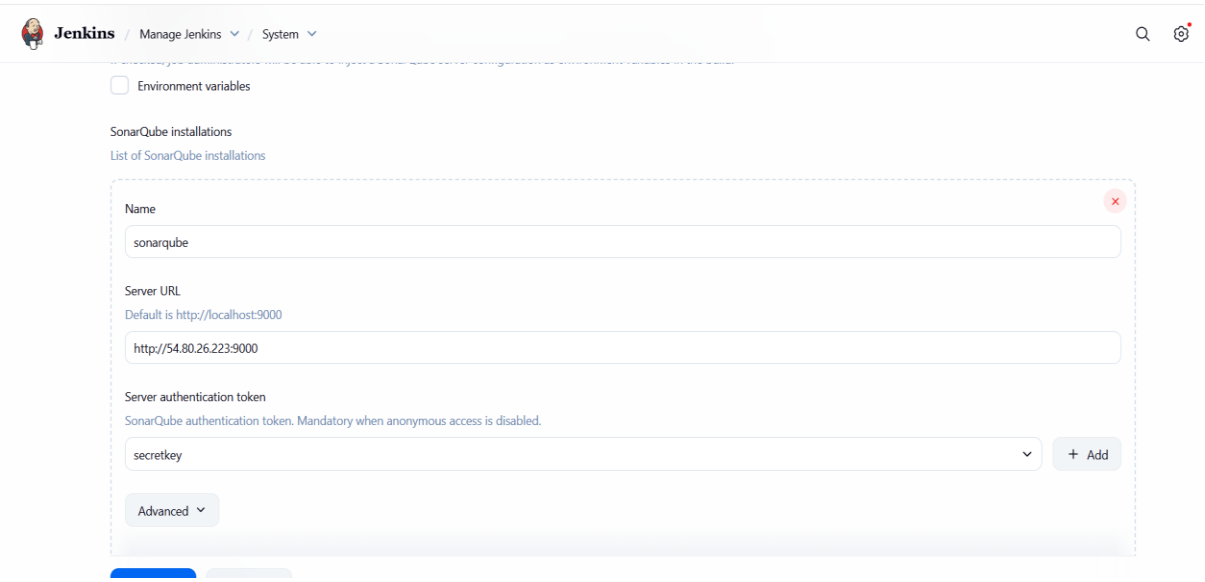


Give the integration in managed Jenkins tools, system,and in the job configure.

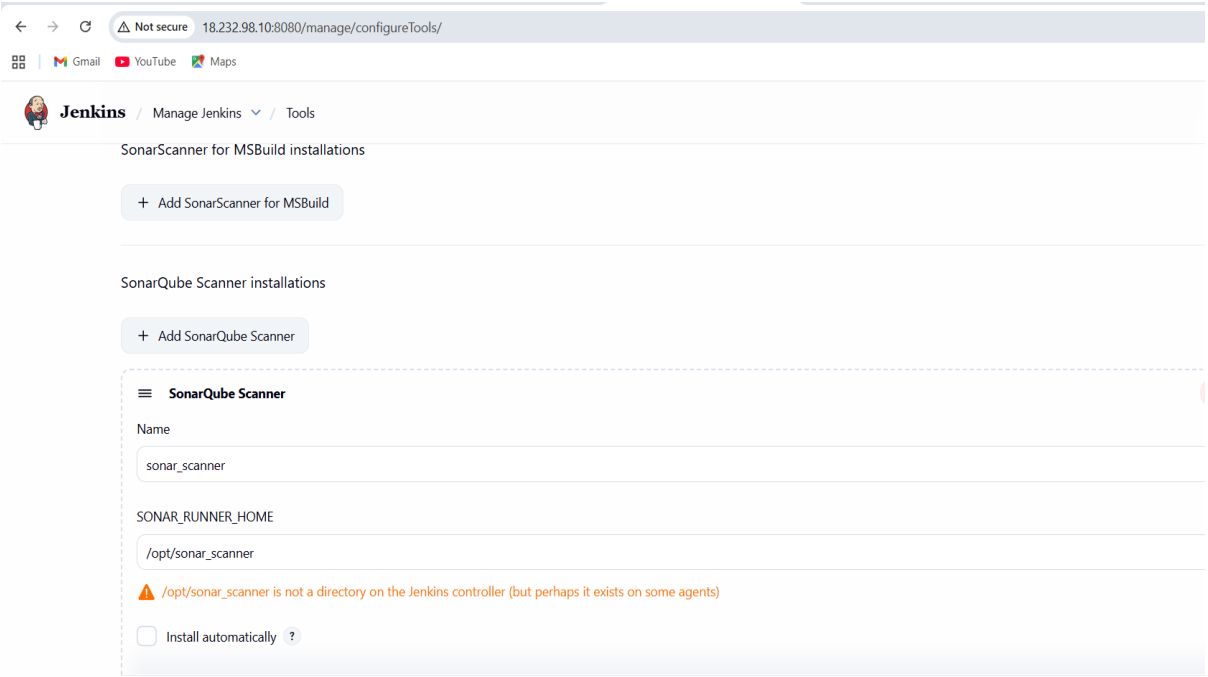
- Install sonarqube scanner



- In system



- In tools



Go to Jenkins server cd /opt and install sonar_scanner

wget https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-4.6.2.2472-linux.zip

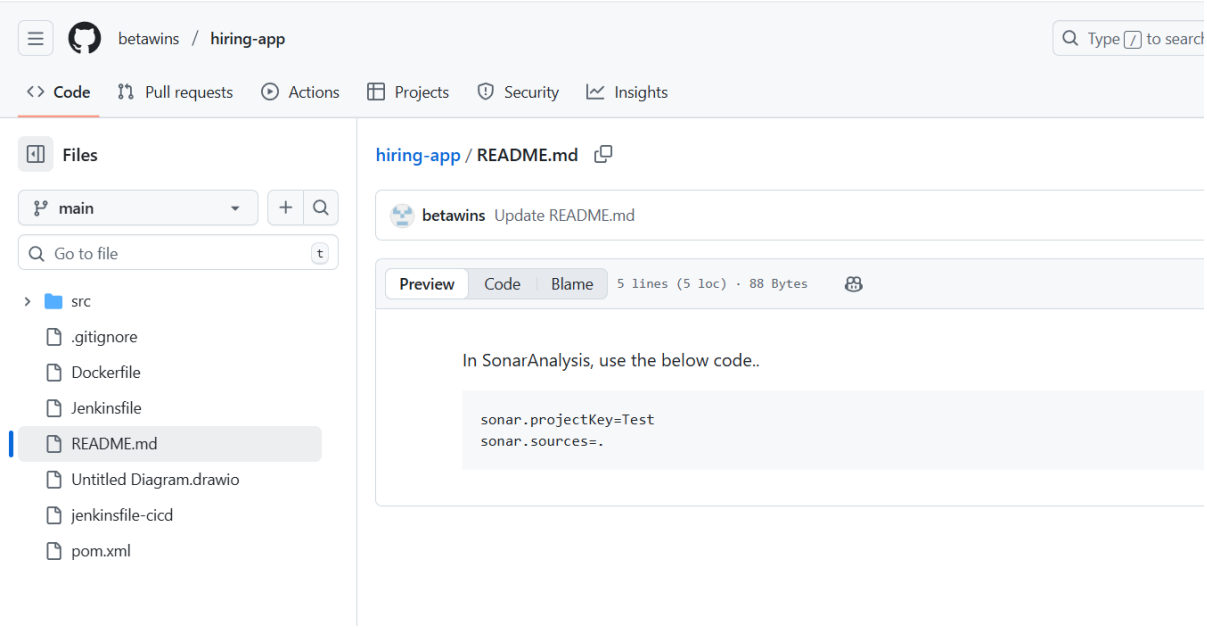
- unzip sonar-scanner-cli-4.6.2.2472-linux.zip
- mv sonar-scanner-4.6.2.2472-linux /opt/sonar_scanner

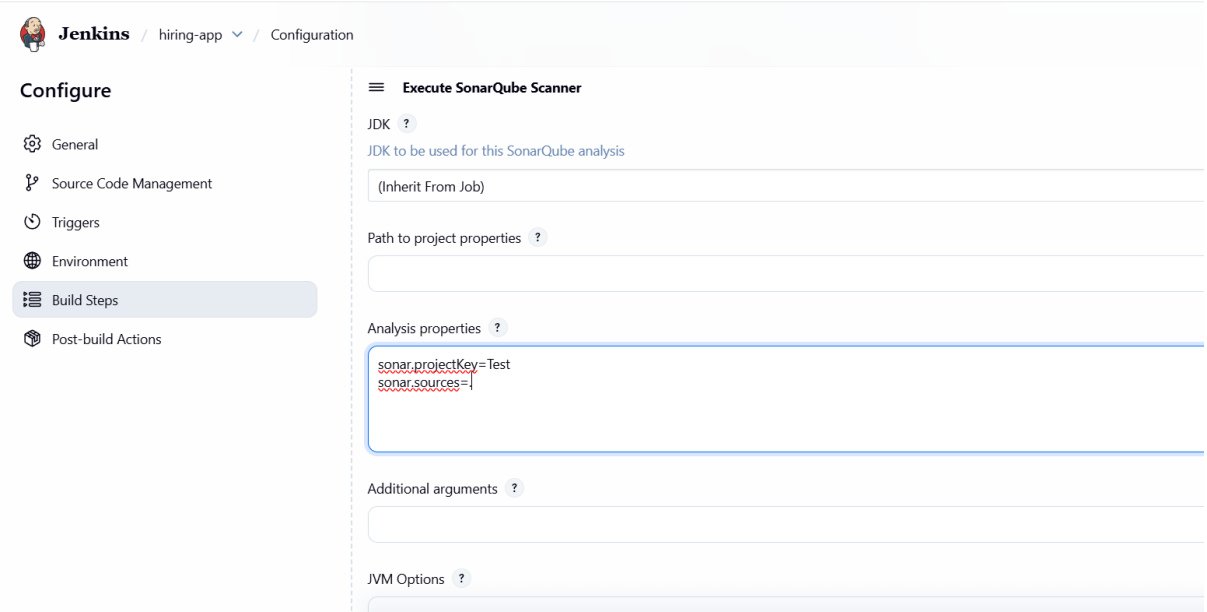
```
[root@ip-172-31-50-64 ~]# cd /opt
[root@ip-172-31-50-64 opt]# ls
aws
[root@ip-172-31-50-64 opt]# wget https://binaries.sonarsource.com/Dist
--2025-10-30 07:46:14--  https://binaries.sonarsource.com/Distributior
Resolving binaries.sonarsource.com (binaries.sonarsource.com)... 99.84
Connecting to binaries.sonarsource.com (binaries.sonarsource.com)|99.8
HTTP request sent, awaiting response... 200 OK
Length: 43099390 (41M) [application/zip]
Saving to: 'sonar-scanner-cli-4.6.2.2472-linux.zip'

sonar-scanner-cli-4.6.2.2472-linu 100%[==============================

2025-10-30 07:46:15 (47.5 MB/s) - 'sonar-scanner-cli-4.6.2.2472-linux.

[root@ip-172-31-50-64 opt]# unzip sonar-scanner-cli-4.6.2.2472-linux.z
Archive:  sonar-scanner-cli-4.6.2.2472-linux.zip
   creating: sonar-scanner-4.6.2.2472-linux/
   creating: sonar-scanner-4.6.2.2472-linux/jre/
   creating: sonar-scanner-4.6.2.2472-linux/jre/conf/
   creating: sonar-scanner-4.6.2.2472-linux/jre/conf/management/
   creating: sonar-scanner-4.6.2.2472-linux/jre/conf/security/
   creating: sonar-scanner-4.6.2.2472-linux/jre/conf/security/policy/
```

Go to job configure give this code

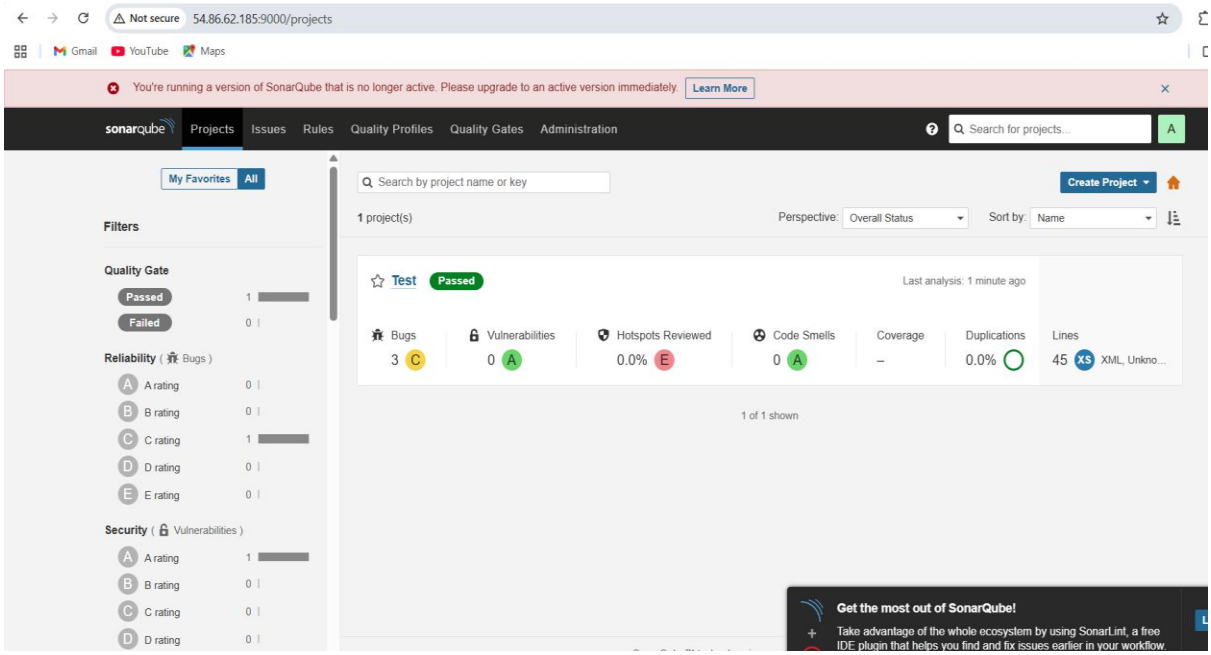
sonar.projectKey=Test

sonar.sources=.





Then build the job.



c. Maven Compilation:

In Jenkins server go to cd /opt and download maven

- Go to browser and search maven download and copy link address and by wget <link adress>

```
[root@ip-172-31-50-64 sonar_scanner]# cd /opt
[root@ip-172-31-50-64 opt]# wget https://dlcdn.apache.org/maven/maven-3/3.9.11/binaries/apache-maven-3.9.11-bin.ta
--2025-10-30 09:49:30--  https://dlcdn.apache.org/maven/maven-3/3.9.11/binaries/apache-maven-3.9.11-bin.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9160848 (8.7M) [application/x-gzip]
Saving to: ‘apache-maven-3.9.11-bin.tar.gz’

apache-maven-3.9.11-bin.tar.gz    100%[===========================================================>]   8.74M  --.

2025-10-30 09:49:30 (347 MB/s) - ‘apache-maven-3.9.11-bin.tar.gz’ saved [9160848/9160848]

[root@ip-172-31-50-64 opt]#
```

- tar xvf apache-maven-3.9.11-bin.tar.gz

```
[root@ip-172-31-50-64 opt]# ls
apache-maven-3.9.11-bin.tar.gz  aws  sonar-scanner-cli-4.6.2.2472-linux.zip  son
[root@ip-172-31-50-64 opt]# tar xvf apache-maven-3.9.11-bin.tar.gz
apache-maven-3.9.11/README.txt
apache-maven-3.9.11/LICENSE
apache-maven-3.9.11/NOTICE
apache-maven-3.9.11/lib/
apache-maven-3.9.11/lib/aopalliance.license
apache-maven-3.9.11/lib/asm.license
apache-maven-3.9.11/lib/commons-cli.license
apache-maven-3.9.11/lib/commons-codec.license
apache-maven-3.9.11/lib/error_prone_annotations.license
apache-maven-3.9.11/lib/failureaccess.license
apache-maven-3.9.11/lib/gson.license
apache-maven-3.9.11/lib/guava.license
apache-maven-3.9.11/lib/guice.license
apache-maven-3.9.11/lib/httpclient.license
apache-maven-3.9.11/lib/httpcore.license
apache-maven-3.9.11/lib/jansi.license
apache-maven-3.9.11/lib/javax.annotation-api.license
apache-maven-3.9.11/lib/javax.inject.license
apache-maven-3.9.11/lib/jcl-over-slf4j.license
apache-maven-3.9.11/lib/jspecify.license
apache-maven-3.9.11/lib/org.eclipse.sisu.inject.license
apache-maven-3.9.11/lib/org.eclipse.sisu.plexus.license
```

Go to bin directory and copy the path /opt/apache-maven3.9.11/bin and paste the address in vi .bash_profile

```
[root@ip-172-31-50-64 opt]# cd apache-maven-3.9.11/
[root@ip-172-31-50-64 apache-maven-3.9.11]# ls
LICENSE  NOTICE  README.txt  bin  boot  conf  lib
[root@ip-172-31-50-64 apache-maven-3.9.11]# cd bin
[root@ip-172-31-50-64 bin]# ls
m2.conf  mvn  mvn.cmd  mvnDebug  mvnDebug.cmd  mvnyjp
[root@ip-172-31-50-64 bin]# pwd
/opt/apache-maven-3.9.11/bin
[root@ip-172-31-50-64 bin]# cd
[root@ip-172-31-50-64 ~]# ls
[root@ip-172-31-50-64 ~]# vi .bash_profile
```

Paste this three lines in that file and save it.

MVN_HOME=/opt/apache-maven-3.9.11/bin

PATH=$PATH:$HOME/bin:$MVN_HOME

export PATH

```
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
        . ~/.bashrc
fi

# User specific environment and startup programs
MVN_HOME=/opt/apache-maven-3.9.11/bin
PATH=$PATH:$HOME/bin:$MVN_HOME
export PATH
~
~
~
~
~
~
```

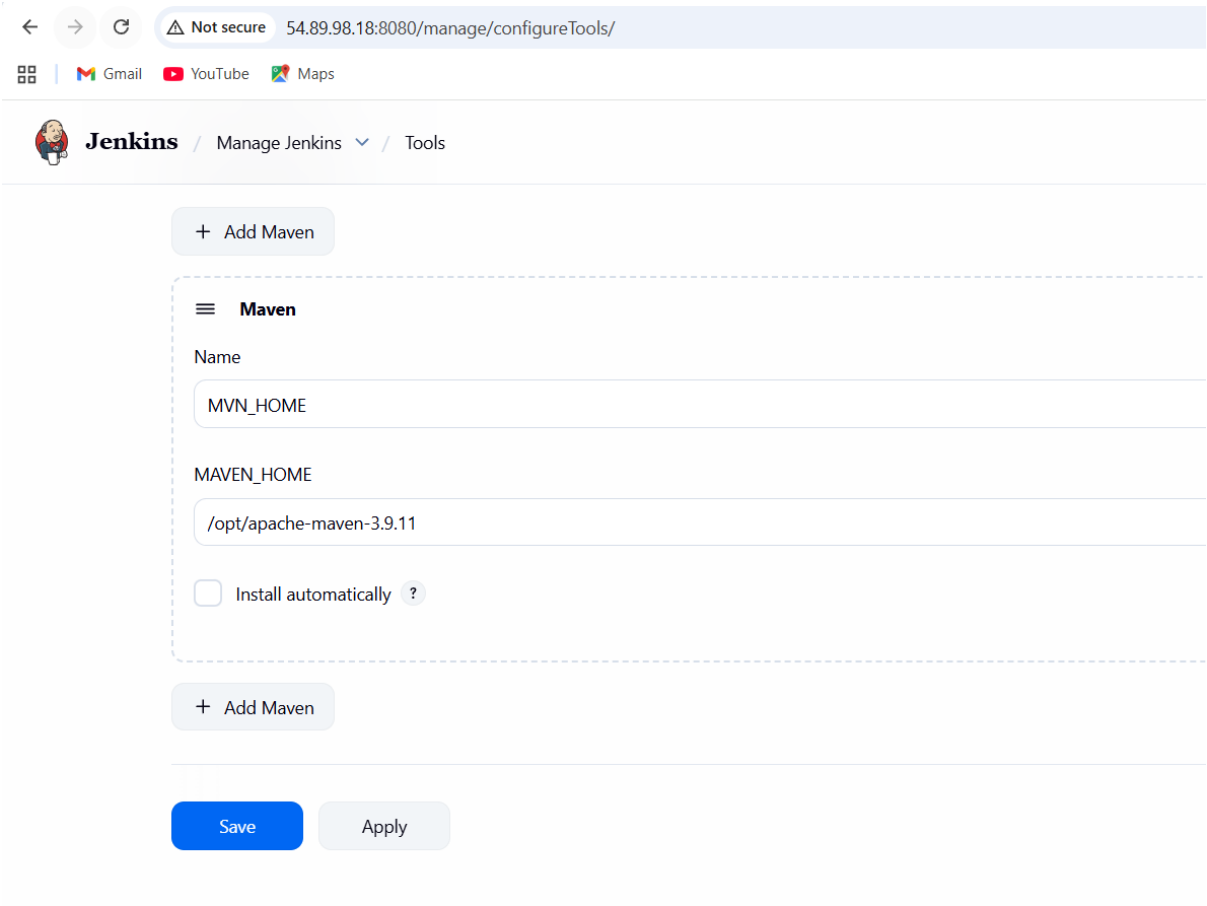
Restart the file with this command

- source ~/.bash_profile

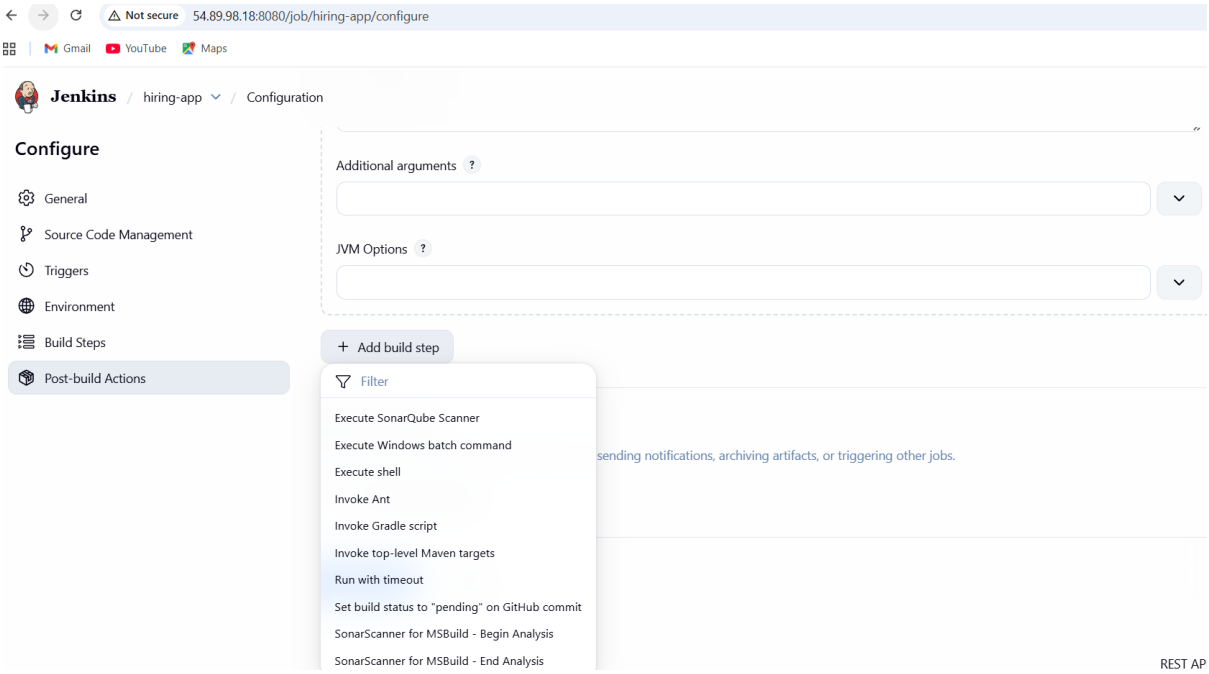
```
[root@ip-172-31-50-64 ~]# source ~/.bash_profile
[root@ip-172-31-50-64 ~]# echo $MVN_HOME
/opt/apache-maven-3.9.11/bin
[root@ip-172-31-50-64 ~]#
```

Go to Jenkins server and go to mange Jenkins,tools in that we are having add maven there you will give name and path

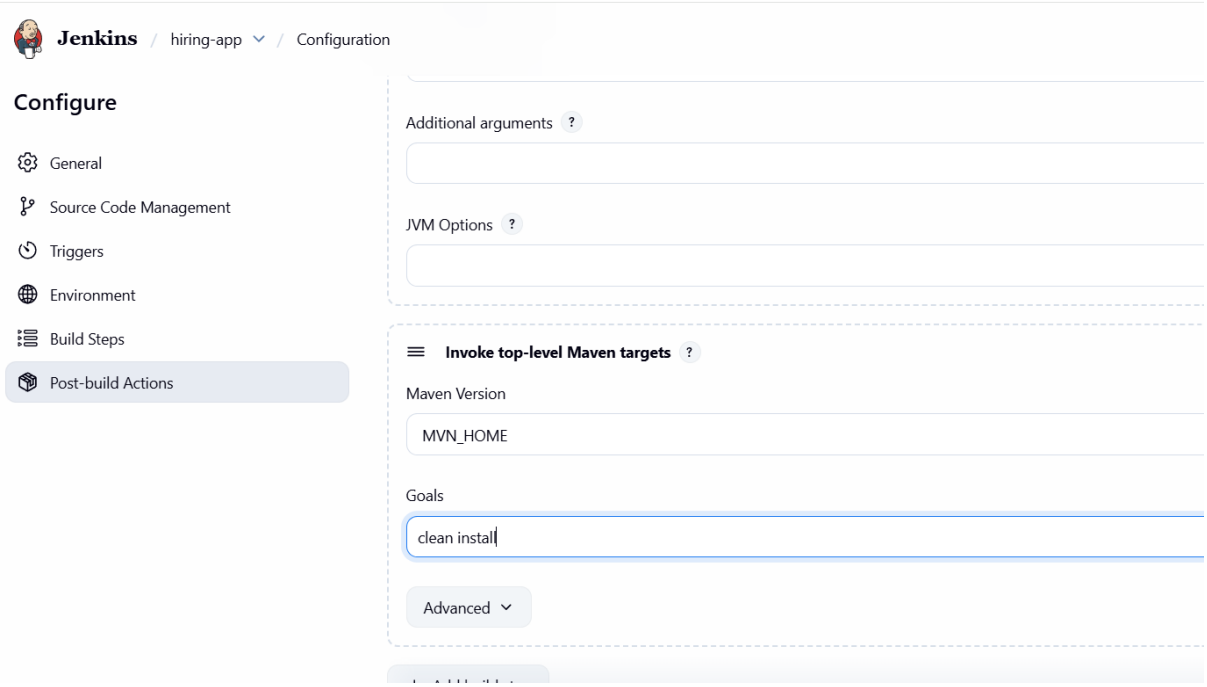
- MVN_HOME
- /opt/apache-maven-3.9.11



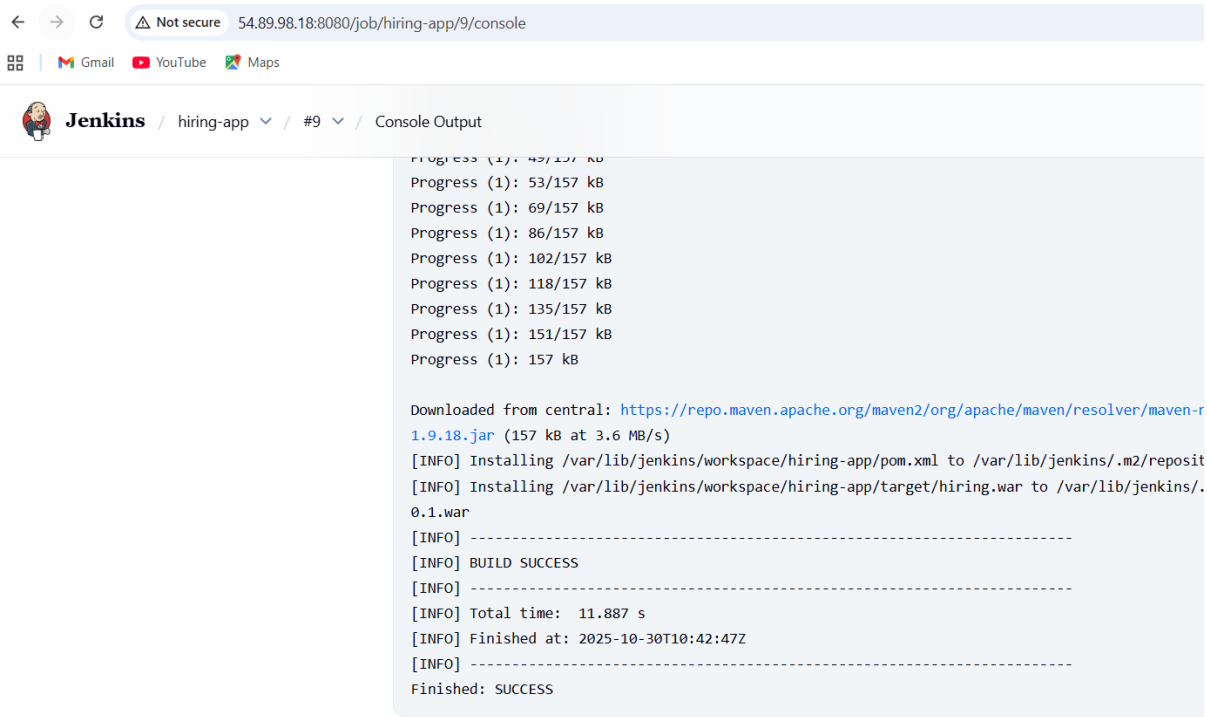
Go to build steps there youwill find invoke top-level Maven targets.



Give the goal as clean install and save it.



Click on build now then our package will be created.



When I go to cd /var/lib/Jenkins/workspace/hiringapp

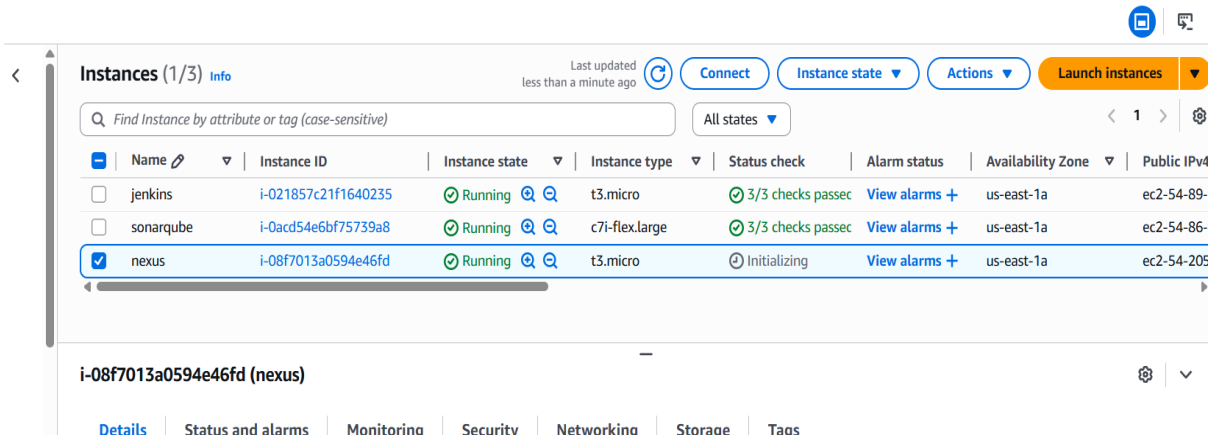
I can see target directory which was newly created to store packages in that directory I can see my war file

```
[root@ip-172-31-50-64 jenkins]# cd /workspace
-bash: cd: /workspace: No such file or directory
[root@ip-172-31-50-64 jenkins]# cd workspace
[root@ip-172-31-50-64 workspace]# ls
hiring-app
[root@ip-172-31-50-64 workspace]# cd hiring-app/
[root@ip-172-31-50-64 hiring-app]# ls
 Dockerfile   Jenkinsfile   README.md  'Untitled Diagram.drawio'   jenkinsfile-cicd   pom.xml   src   target
[root@ip-172-31-50-64 hiring-app]# cd target
[root@ip-172-31-50-64 target]# ls
hiring  hiring.war  maven-archiver
[root@ip-172-31-50-64 target]#
```

d. Nexus Artifactory:

create an ec2-instance with the name nexus.

- use c7-flexlarge instance type.
- Use the memory as 20 gb



- Install java in that server by using yum install java

Go to opt directory and download nexus and extract and rename it.

- cd /opt
- wgetwget https://download.sonatype.com/nexus/3/nexus-3.68.1-02-unix.tar.gz

- tar -xvzf nexus-3.68.1-02-unix.tar.gz
- mv nexus-3.68.1-02 nexus

```
root@ip-172-31-45-191 opt]# tar -xvzf nexus-3.68.1-02-unix.tar.gz
exus-3.68.1-02/.install4j/9d17dc87.lprop
exus-3.68.1-02/.install4j/MessagesDefault
exus-3.68.1-02/.install4j/build.uuid
exus-3.68.1-02/.install4j/e4ada6b7.lprop
exus-3.68.1-02/.install4j/i4j_extf_0_17is1ik.utf8
exus-3.68.1-02/.install4j/i4j_extf_10_17is1ik_10358jn.png
exus-3.68.1-02/.install4j/i4j_extf_11_17is1ik_1gne9sv.png
exus-3.68.1-02/.install4j/i4j_extf_12_17is1ik_sc8j43.png
exus-3.68.1-02/.install4j/i4j_extf_13_17is1ik_10nxrsm.png
exus-3.68.1-02/.install4j/i4j_extf_14_17is1ik_yd7am4.png
exus-3.68.1-02/.install4j/i4j_extf_15_17is1ik_vu6hgs.png
exus-3.68.1-02/.install4j/i4j_extf_16_17is1ik_1g1wykh.png
exus-3.68.1-02/.install4j/i4j_extf_17_17is1ik_18gg8kx.png
exus-3.68.1-02/.install4j/i4j_extf_17_17is1ik_18gg8kx@2x.png
exus-3.68.1-02/.install4j/i4j_extf_18_17is1ik_11g5ail.png
exus-3.68.1-02/.install4j/i4j_extf_19_17is1ik_fyoktp.png
exus-3.68.1-02/.install4j/i4j_extf_1_17is1ik.properties
exus-3.68.1-02/.install4j/i4j_extf_20_17is1ik_11g5ail.png
```

Change the directory to cd opt/nexus/bin/vi nexus.rc

Write the user as ec2-user in that file.

```
#run_as_user="ec2-user"
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

Change the ownership permission to nexus

- sudo chown -R ec2-user:ec2-user nexus
- sudo mkdir -p /opt/sonatype-work/nexus3/log
- sudo chown -R ec2-user:ec2-user /opt/nexus /opt/sonatype-work
- sudo chmod -R 755 /opt/nexus /opt/sonatype-work

```
[ec2-user@ip-172-31-45-191 opt]$ ll
total 223364
drwxr-xr-x.  4 root root        33 Oct 23 00:08 aws
drwxr-xr-x. 10 root root       181 Oct 30 12:20 nexus
-rw-r--r--.  1 root root 228724143 May 16  2024 nexus-3.68.1-02-unix.tar.gz
drwxr-xr-x.  3 root root        20 Oct 30 12:20 sonatype-work
[ec2-user@ip-172-31-45-191 opt]$ sudo chown -R ec2-user:ec2-user nexus
[ec2-user@ip-172-31-45-191 opt]$ ll
total 223364
drwxr-xr-x.  4 root     root            33 Oct 23 00:08 aws
drwxr-xr-x. 10 ec2-user ec2-user       181 Oct 30 12:20 nexus
-rw-r--r--.  1 root     root     228724143 May 16  2024 nexus-3.68.1-02-unix.tar.gz
drwxr-xr-x.  3 root     root            20 Oct 30 12:20 sonatype-work
[ec2-user@ip-172-31-45-191 opt]$
```

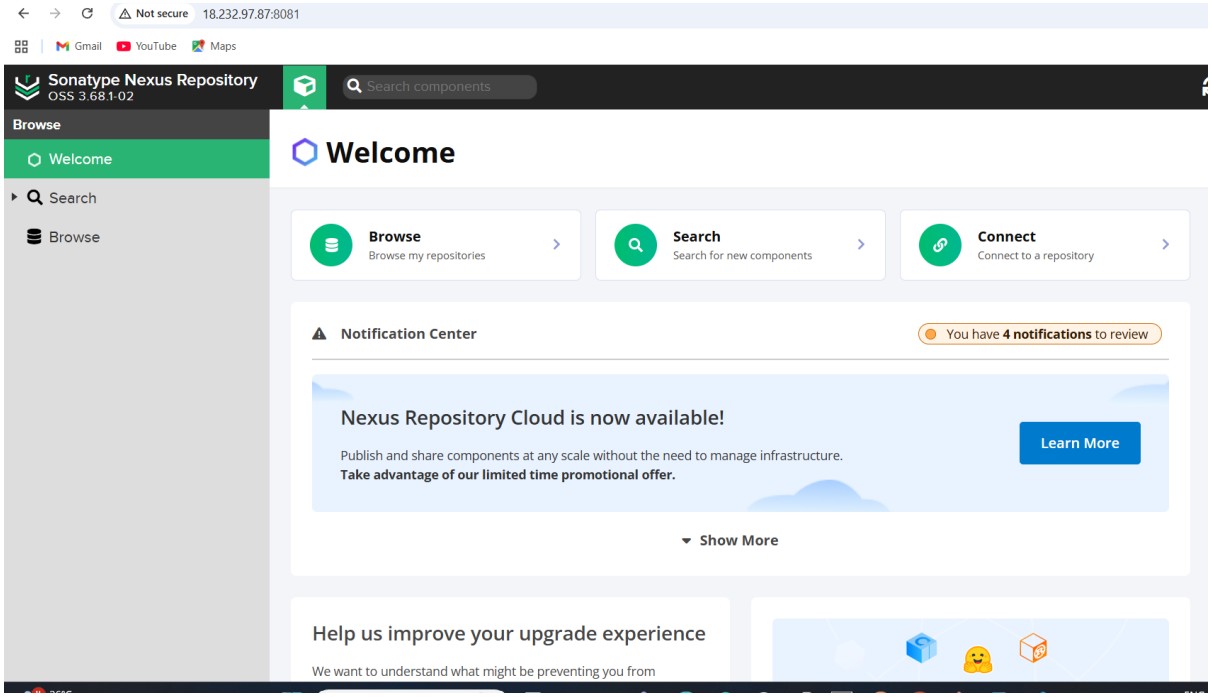
Go to cd /opt/nexus/bin

- ./nexus start
- ./nexus status

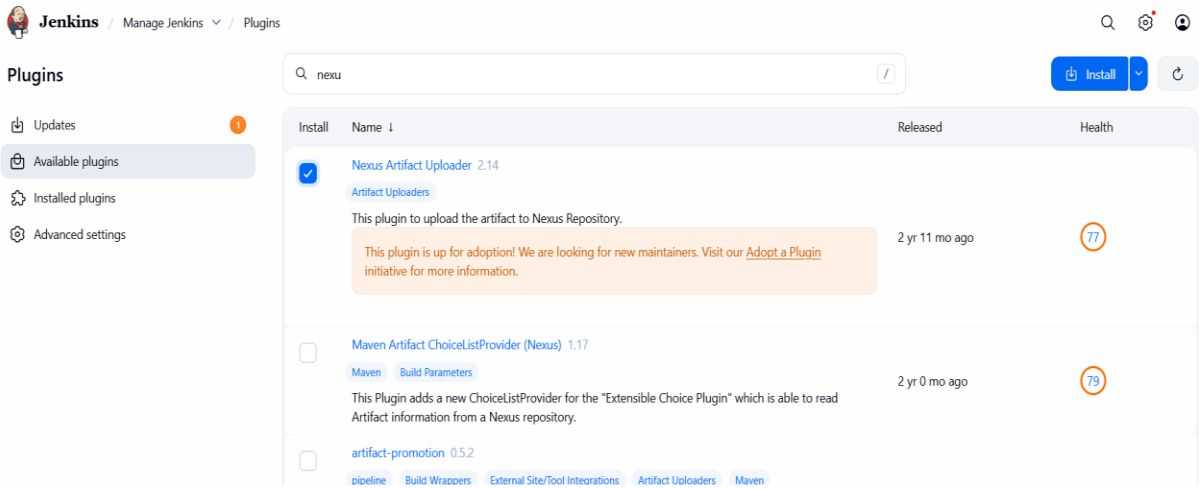
```
[ec2-user@ip-172-31-45-191 bin]$ ./nexus start
Starting nexus
[ec2-user@ip-172-31-45-191 bin]$ ./nexus status
nexus is stopped.
[ec2-user@ip-172-31-45-191 bin]$ free -h
              total        used        free      shared  buff/cache   available
Mem:           904Mi       215Mi       310Mi       0.0Ki       378Mi       543Mi
Swap:             0B          0B          0B
[ec2-user@ip-172-31-45-191 bin]$ sudo fallocate -l 2G /swapfile
[ec2-user@ip-172-31-45-191 bin]$ sudo chmod 600 /swapfile
[ec2-user@ip-172-31-45-191 bin]$ sudo mkswap /swapfile
Setting up swapspace version 1, size = 2 GiB (2147479552 bytes)
no label, UUID=e49c4d6b-c8e5-43c9-b633-78fd41c93afe
[ec2-user@ip-172-31-45-191 bin]$ sudo swapon /swapfile
[ec2-user@ip-172-31-45-191 bin]$ ./nexus start
Starting nexus
[ec2-user@ip-172-31-45-191 bin]$ ./nexus status
nexus is running.
[ec2-user@ip-172-31-45-191 bin]$
```

With the ip address go to browser

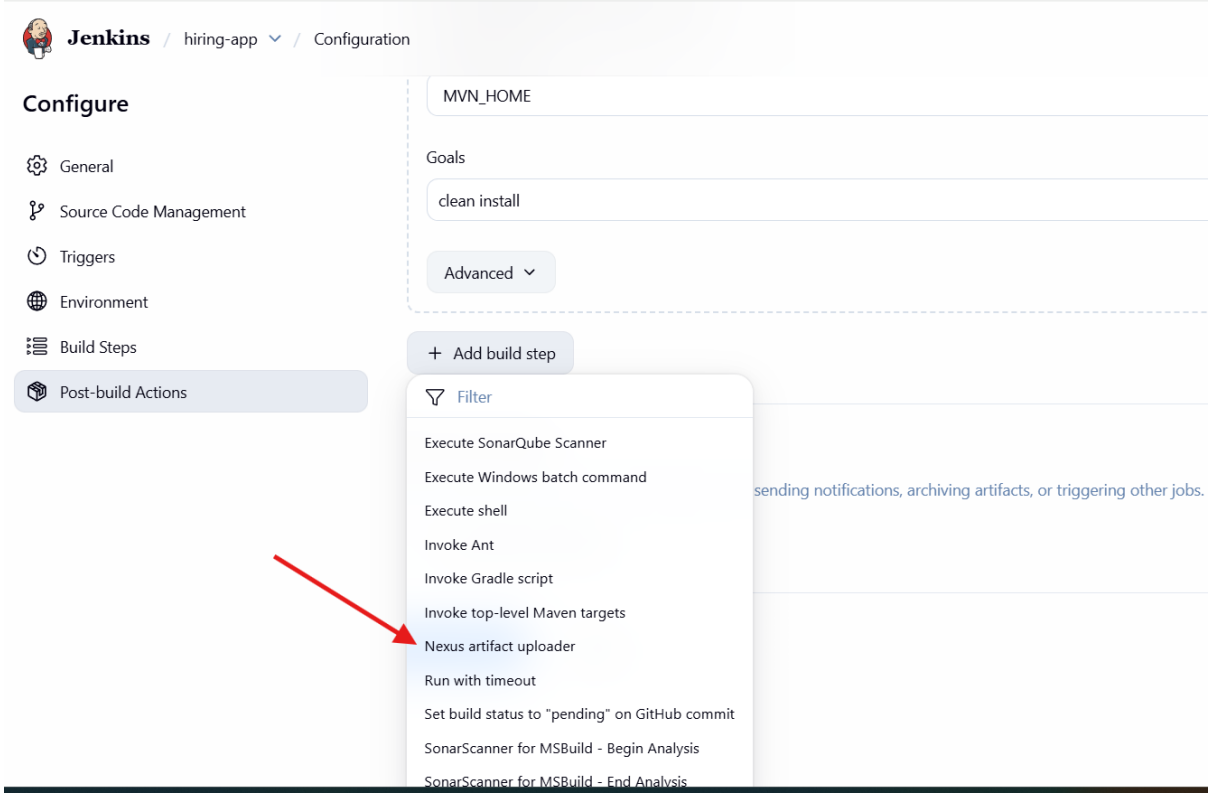
- Ip:8081



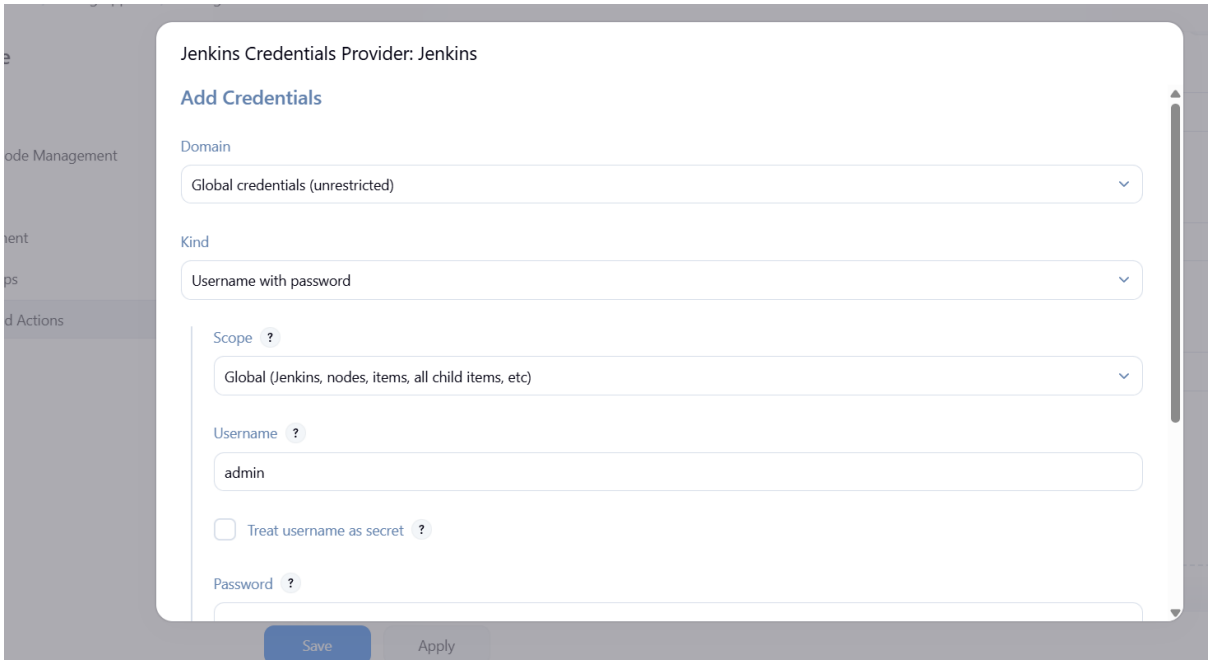
Go to plugins and install nexus artifactory plugin.



Go to your job in that in build steps you will find nexus artifact uploader.



In that fill those and give the credentials.



Go to pom.xml you find this id's like

Files

main

Go to file

- src
- .gitignore
- Dockerfile
- Jenkinsfile
- README.md
- Untitled Diagram.drawio
- jenkinsfile-cicd
- pom.xml

hiring-app / pom.xml

bcreddydevops Update pom.xml 06349c9

Code Blame 28 lines (28 loc) · 890 Bytes Raw

```xml
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>in.javahome</groupId>
  <artifactId>hiring</artifactId>
  <packaging>war</packaging>
  <version>0.1</version>
  <name>hiring Maven Webapp</name>
  <url>http://maven.apache.org</url>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-war-plugin</artifactId>
        <version>3.3.1</version>
      </plugin>
```

- Groupid= in.javahome
- Version= 0.1

Go to nexus server and go to settings, repository, create repository. select maven2hosted.

Not secure 3.84.229.236:8081/#admin/repository/repositories

Gmail YouTube Maps

Sonatype Nexus Repository Manager
OSS 3.19.1-01

Search components

Administration

- Repository
  - Repositories
  - Blob Stores
  - Content Selectors
  - Cleanup Policies
  - Routing Rules
- Security
  - Privileges
  - Roles
  - Users
  - Anonymous
  - LDAP
  - Realms
  - SSL Certificates
- IQ Server
- Support
- System

Repositories / Select Recipe

Recipe
bower (proxy)
cocoapods (proxy)
conda (proxy)
docker (group)
docker (hosted)
docker (proxy)
gitlfs (hosted)
go (group)
go (proxy)
maven2 (group)
maven2 (hosted)
maven2 (proxy)
npm (group)
npm (hosted)
npm (proxy)
nuget (group)
nuget (hosted)
nuget (proxy)
pypi (group)
pypi (hosted)
pypi (proxy)
raw (group)
raw (hosted)

Sonatype Nexus Repository Manager
OSS 3.19.1-01

Search components

Administration

- Repository
  - Repositories
  - Blob Stores
  - Content Selectors
  - Cleanup Policies
  - Routing Rules
- Security
  - Privileges
  - Roles
  - Users
  - Anonymous
  - LDAP
  - Realms
  - SSL Certificates
- IQ Server
- Support

Repositories / Select Recipe / Create Repository: maven2 (hosted)

Maven 2

Version policy:

What type of artifacts does this repository store?

Release

Layout policy:

Validate that all paths are maven artifact or metadata paths

Permissive

Storage

Blob store:

Blob store used to store repository contents

default

Strict Content Type Validation:

☑ Validate that all content uploaded to this repository is of a MIME type appropriate for the repository format

Hosted

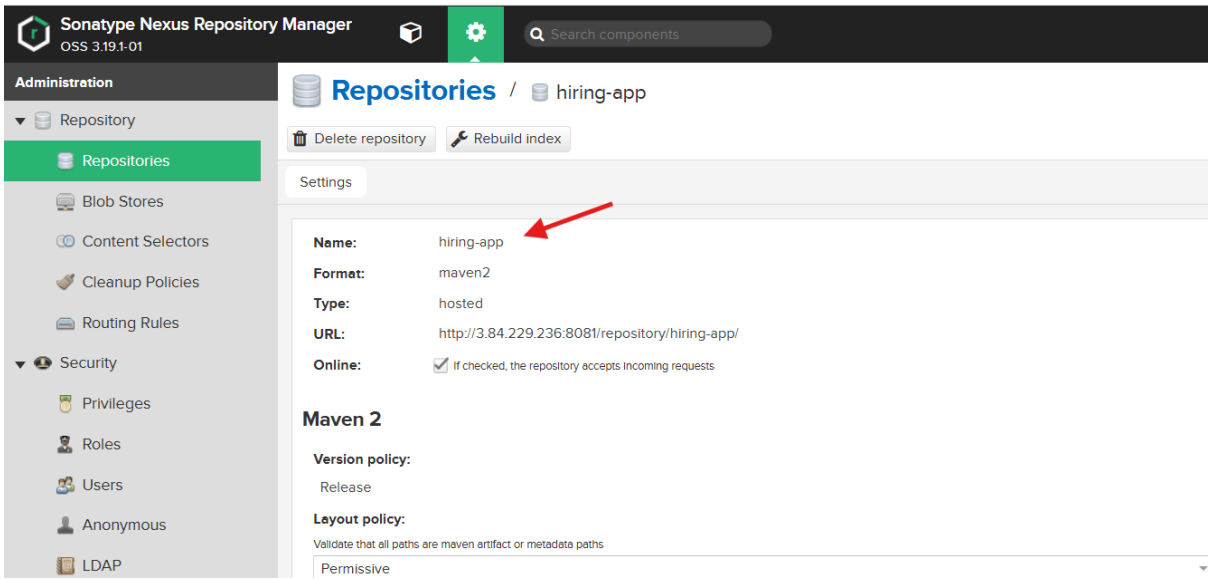
Deployment policy:

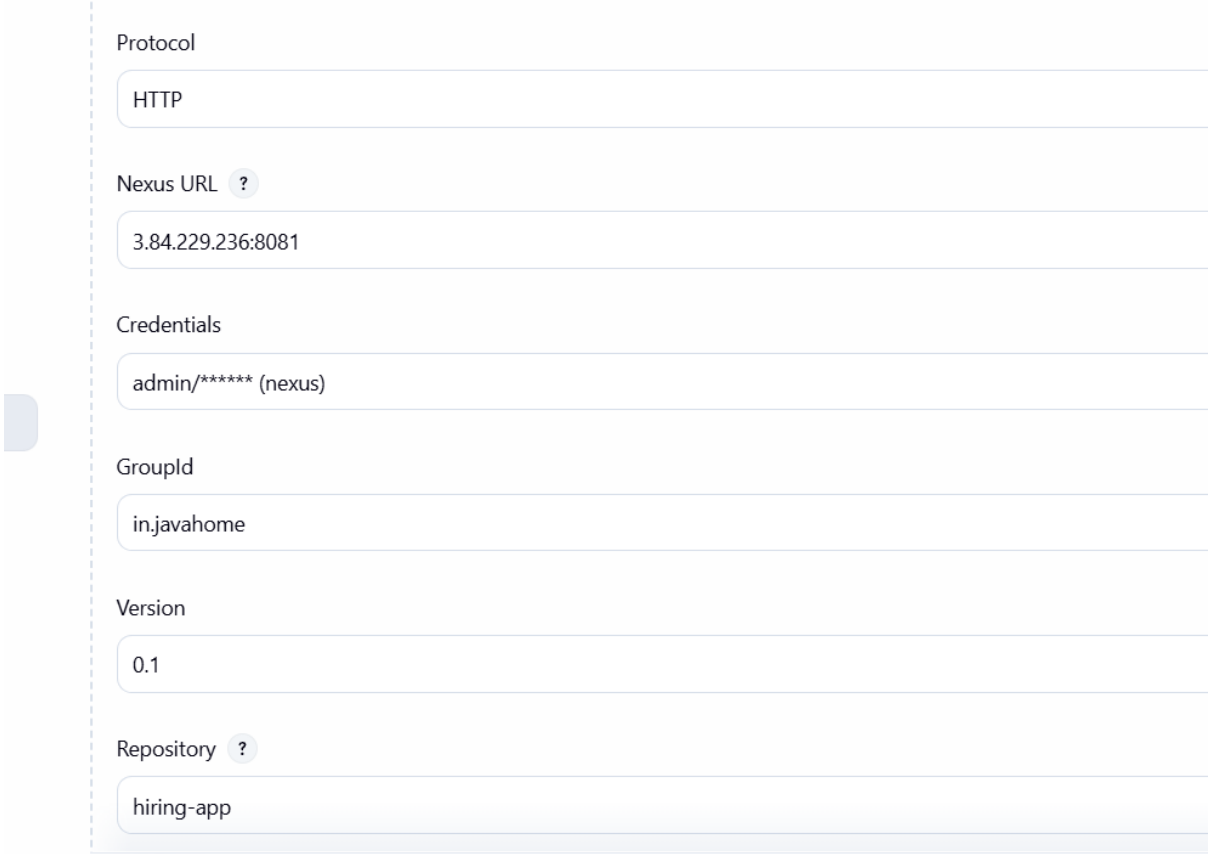
Controls if deployments of and updates to artifacts are allowed

Allow redeploy

Cleanup

You need to pass this repository.





on

Artifact

ArtifactId

hiring

Type ?

war

Classifier ?

File ?

/var/lib/jenkins/workspace/hiring-app/target/hiring.war

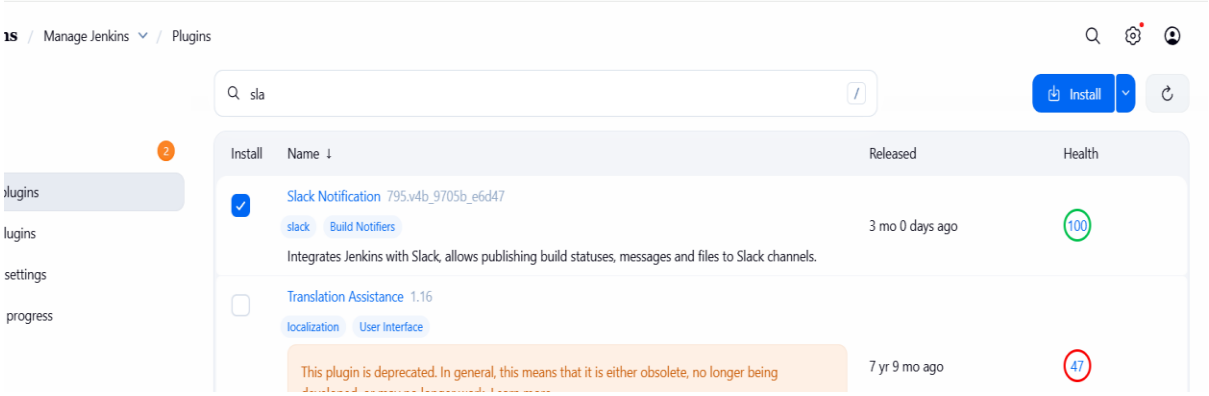
Click on build now.

Jenkins / hiring-app / #21 / Console Output

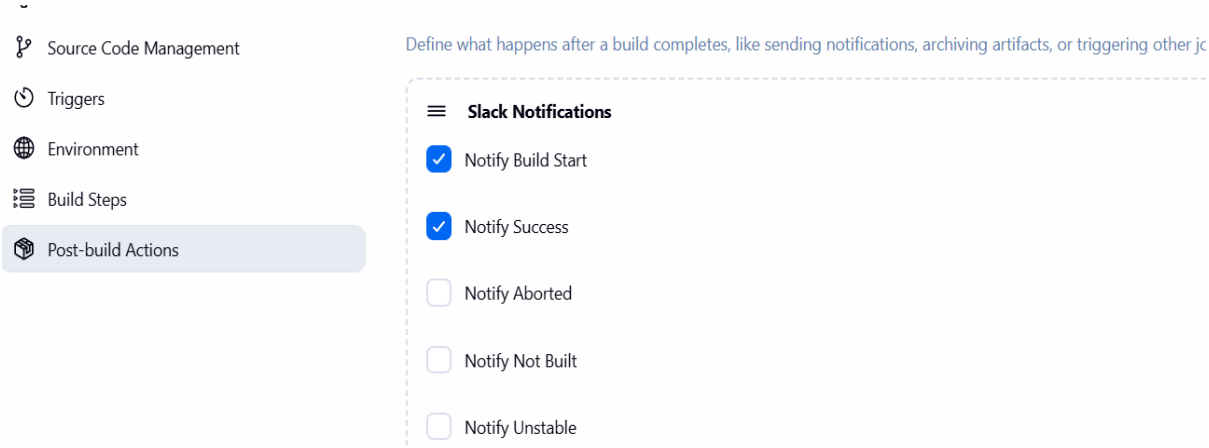
```
[INFO] Installing /var/lib/jenkins/workspace/hiring-app/target/hiring.war to /var/lib/jenkins/.m2/repository/in/javahome/hiring/0.1/hiring-0.1.war
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  2.633 s
[INFO] Finished at: 2025-11-03T06:25:07Z
[INFO] ------------------------------------------------------------------------
Uploading artifact hiring.war started....
GroupId: in.javahome
ArtifactId: hiring
Classifier:
Type: war
Version: 0.1
File: hiring.war
Repository:hiring-app
Uploading: http://13.221.174.86:8081/repository/hiring-app/in/javahome/hiring/0.1/hiring-0.1.war
100 % completed (1.7 kB / 1.7 kB).
Uploaded: http://13.221.174.86:8081/repository/hiring-app/in/javahome/hiring/0.1/hiring-0.1.war (1.7 kB at 5.9 kB/s)
Uploading artifact hiring.war completed.
Finished: SUCCESS
```

e. Slack Notification:

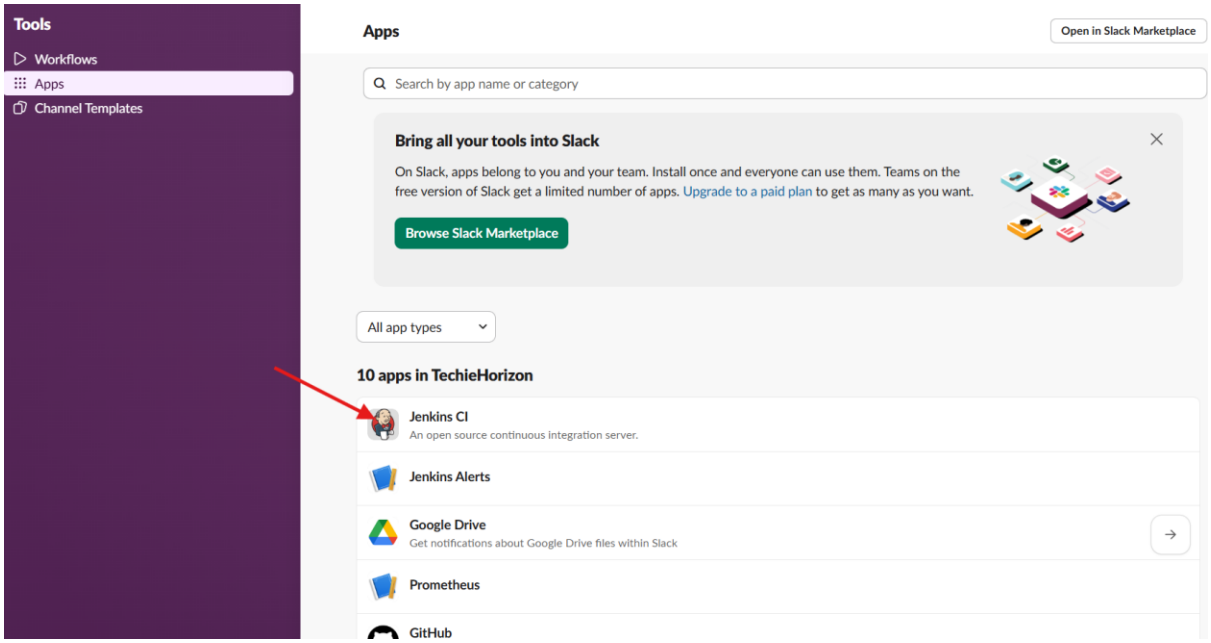
go to mange Jenkins go to plugin and install slack notification plugin.



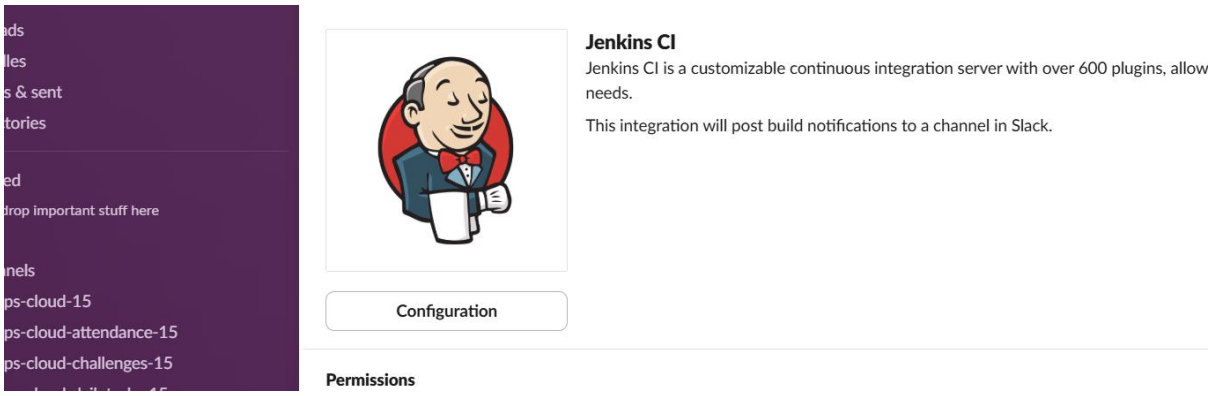
Go to your job go to configure click on post build actions, and select slack notification.and save it.

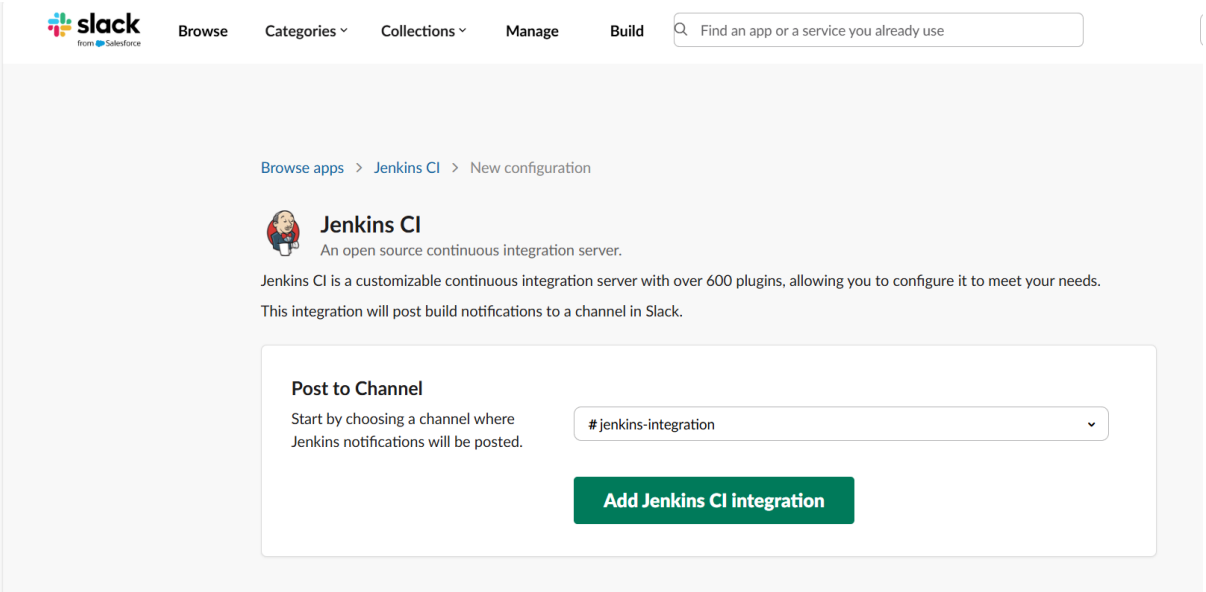


Go to slack and select apps, click on Jenkins app



Click on configuration.





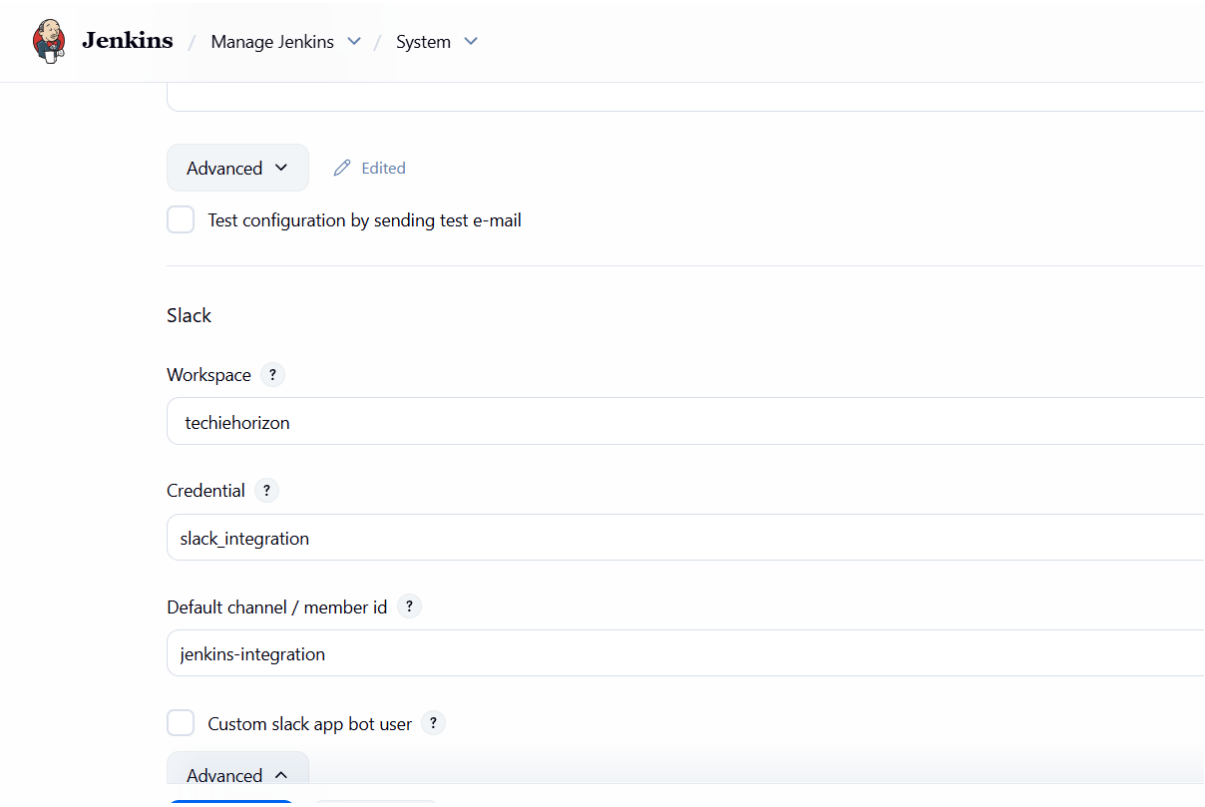
Go to manage Jenkins, system and add this details.

Step 3

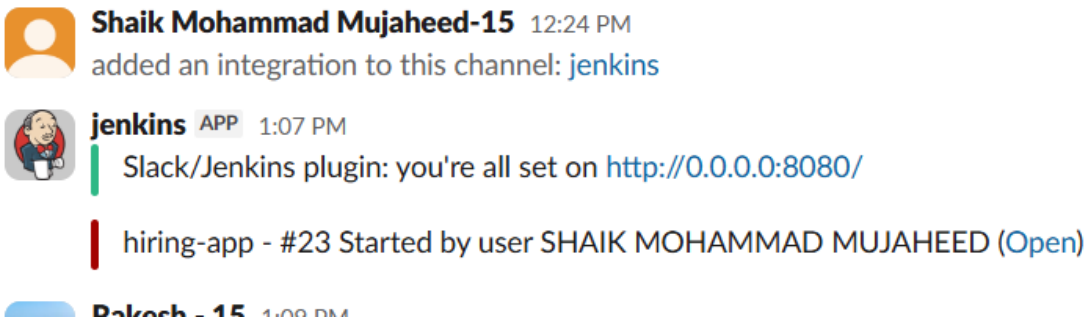
After it's installed, click on **Manage Jenkins** again in the left navigation, and then go to **Configure System**. Find the **Global Slack Notifier Settings** section and add the following values:

- **Team Subdomain:** `techiehorizon`
- **Integration Token Credential ID:** Create a secret text credential using `DQ004OF3vmAIFEhft5Z8dzo8` as the value

The other fields are optional. You can click on the question mark icons next to them for more information. Press **Save** when you're done.



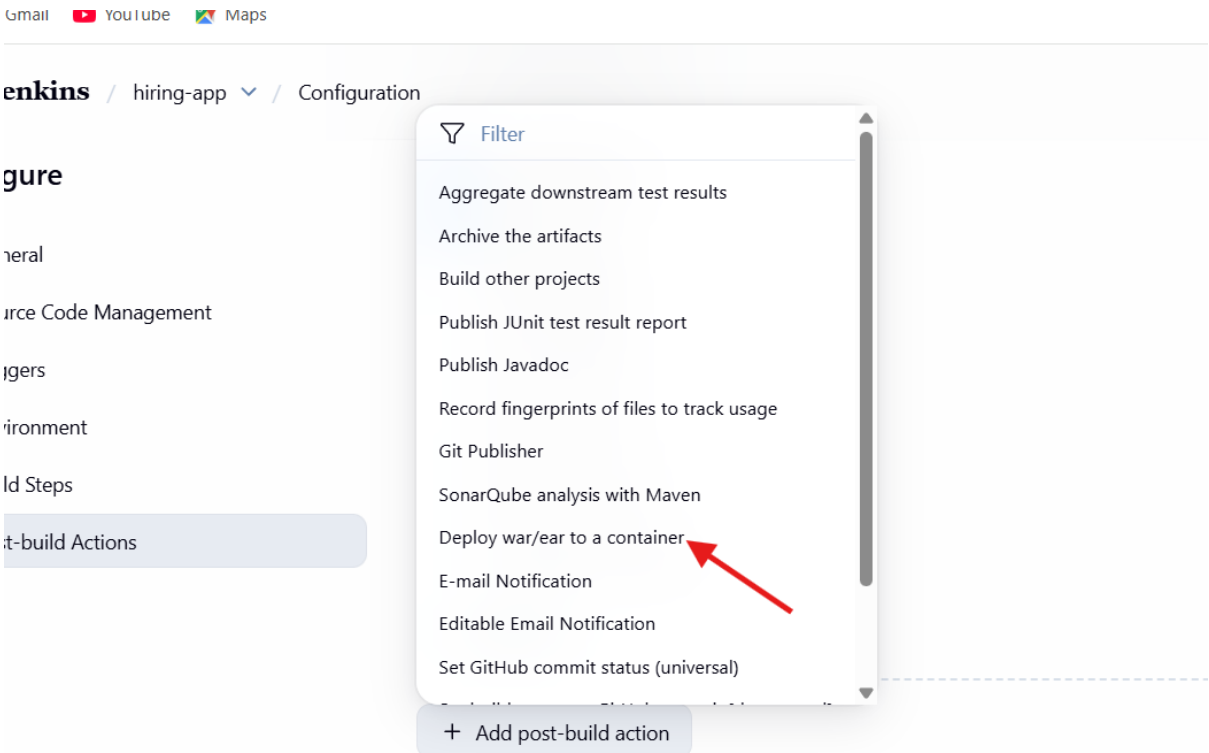
If you go to channel then you will notify .



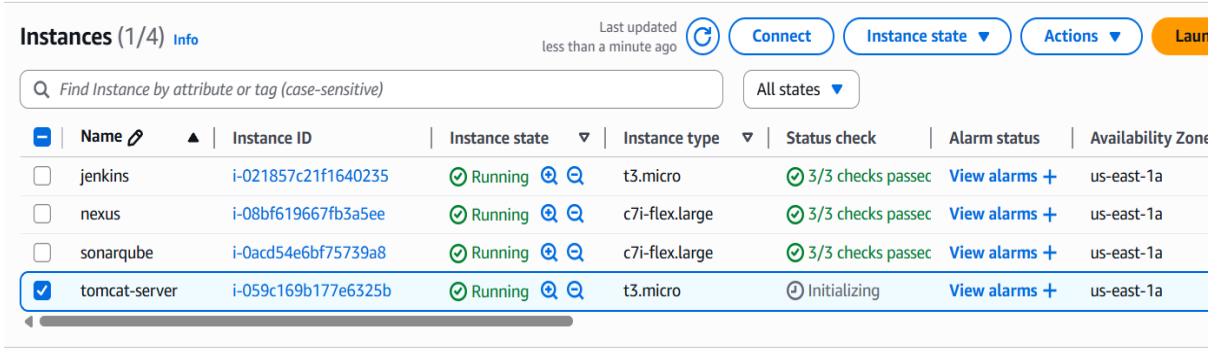
f. Deploy On tomcat:

for this we need to install a plugin called deploy to container.

Go to job, configure and select post build actions select deploy war/ear to a container.

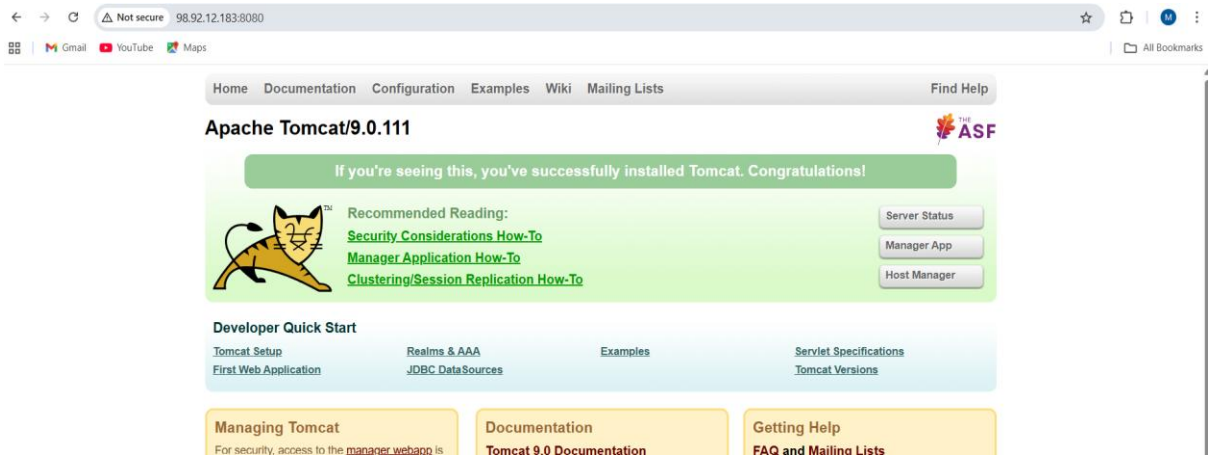


Create an instance for tomcat.



- Install java
- Wget tomcat
- tar xvf
- start tomcat

```
[root@ip-172-31-111-184 opt]# ls
apache-tomcat-9.0.111.tar.gz  aws
[root@ip-172-31-111-184 opt]# tar xvf apache-tomcat-9.0.111.tar.gz
apache-tomcat-9.0.111/conf/
apache-tomcat-9.0.111/conf/catalina.policy
apache-tomcat-9.0.111/conf/catalina.properties
apache-tomcat-9.0.111/conf/context.xml
apache-tomcat-9.0.111/conf/jaspic-providers.xml
apache-tomcat-9.0.111/conf/jaspic-providers.xsd
apache-tomcat-9.0.111/conf/logging.properties
apache-tomcat-9.0.111/conf/server.xml
```



We need to comment to create an user in tomcat

- find / -name context.xml
- edit files where webapps will be there.

```
[root@ip-172-31-111-184 bin]# find / -name context.xml
/opt/apache-tomcat-9.0.111/conf/context.xml
/opt/apache-tomcat-9.0.111/webapps/docs/META-INF/context.xml
/opt/apache-tomcat-9.0.111/webapps/examples/META-INF/context.xml
/opt/apache-tomcat-9.0.111/webapps/host-manager/META-INF/context.xml
/opt/apache-tomcat-9.0.111/webapps/manager/META-INF/context.xml
[root@ip-172-31-111-184 bin]#
```

Need to comment like this at valve.

```
-->
<Context antiResourceLocking="false" >
        <!--  <Valve className="org.apache.catalina.valves.RemoteCIDRValve"
        allow="127.0.0.0/8,::1/128" /> -->
</Context>
~
```

```
[root@ip-172-31-111-184 bin]# vi /opt/apache-tomcat-9.0.111/webapps/docs/META-INF/context.xml
[root@ip-172-31-111-184 bin]# vi /opt/apache-tomcat-9.0.111/webapps/examples/META-INF/context.xml
[root@ip-172-31-111-184 bin]# /opt/apache-tomcat-9.0.111/webapps/host-manager/META-INF/context.xml
-bash: /opt/apache-tomcat-9.0.111/webapps/host-manager/META-INF/context.xml: Permission denied
[root@ip-172-31-111-184 bin]# vi /opt/apache-tomcat-9.0.111/webapps/host-manager/META-INF/context.xml
[root@ip-172-31-111-184 bin]# vi /opt/apache-tomcat-9.0.111/webapps/manager/META-INF/context.xml
[root@ip-172-31-111-184 bin]#
```

- then go to conf directory and vi tomcat-users.xml add this.

<role rolename="manager-gui"/>

<role rolename="manager-script"/>

<role rolename="manager-jmx"/>

<role rolename="manager-status"/>

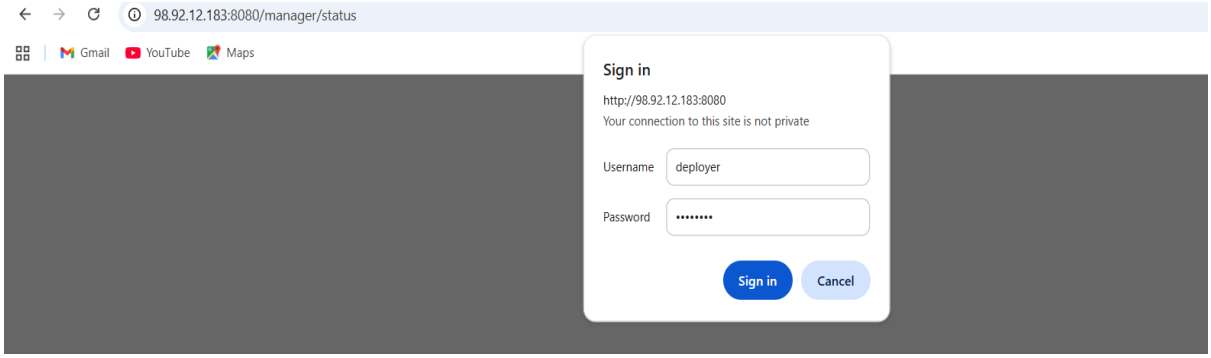
<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>

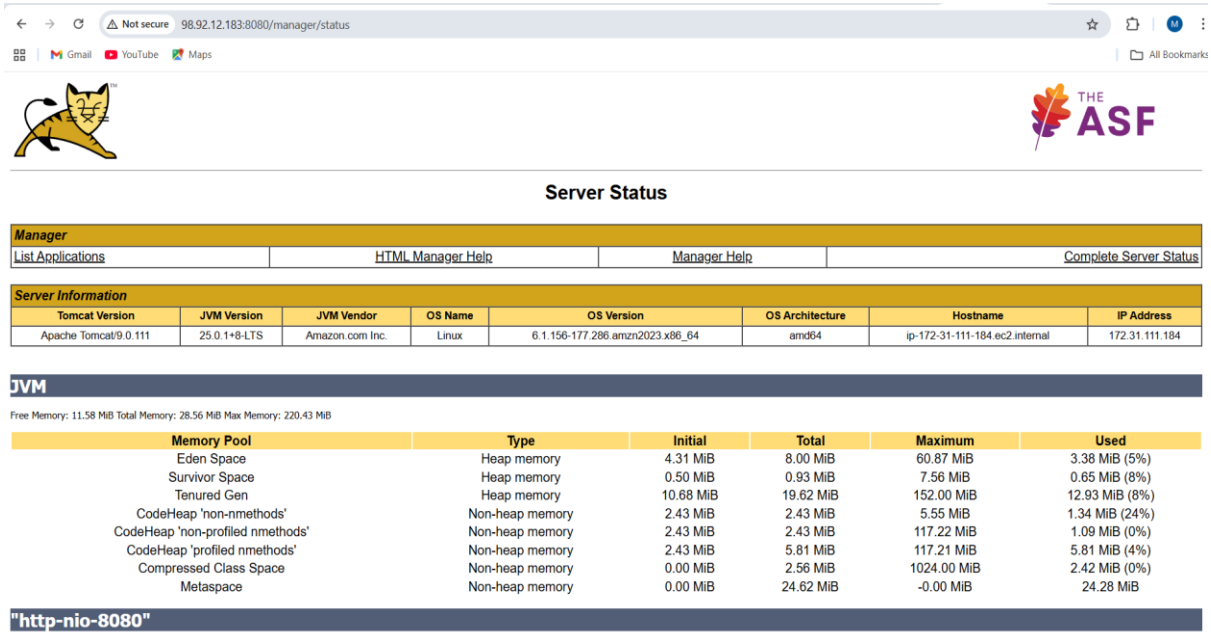
<user username="deployer" password="deployer" roles="manager-script"/>

<user username="tomcat" password="s3cret" roles="manager-gui"/>

```
  <user username="tomcat" password="<must-be-changed>" roles="tomcat"/>
  <user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
  <user username="role1" password="<must-be-changed>" roles="role1"/>
-->
  <role rolename="manager-gui"/>
        <role rolename="manager-script"/>
        <role rolename="manager-jmx"/>
        <role rolename="manager-status"/>
        <user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>
        <user username="deployer" password="deployer" roles="manager-script"/>
        <user username="tomcat" password="s3cret" roles="manager-gui"/>
</tomcat-users>
-- INSERT --
```

Click on server status it will ask username and password.





Give the configurations in the job and save it.

Jenkins Credentials Provider: Jenkins

Scope ?

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

Username ?

deployer

Treat username as secret ?

Password ?

••••••••

ID ?

tomcat-server

Description ?

tomcat-server

hiring-app / Configuration

Management

tions

Deploy war/ear to a container

WAR/EAR files ?

**/*.war

War/ear files to deploy. Relative to the workspace root. You can also specify Ant-style GLOBs, like "**/*.war"

(from Deploy to container Plugin)

Context path ?

Containers

Tomcat 9.x Remote

Credentials

deployer/****** (tomcat-server)

+ Add

Tomcat URL ?

http://98.92.12.183:8080

Save Apply

```
-app / #25 / Console Output

[INFO] Finished at: 2025-11-03T09:55:36Z
[INFO] ------------------------------------------------------------------------
Uploading artifact hiring.war started....
GroupId: in.javahome
ArtifactId: hiring
Classifier:
Type: war
Version: 0.1
File: hiring.war
Repository:hiring-app
Uploading: http://54.211.68.219:8081/repository/hiring-app/in/javahome/hiring/0.1/hiring-0.1.war
100 % completed (1.7 kB / 1.7 kB).
Uploaded: http://54.211.68.219:8081/repository/hiring-app/in/javahome/hiring/0.1/hiring-0.1.war (1.7 kB at 7.1
Uploading artifact hiring.war completed.
[DeployPublisher][INFO] Attempting to deploy 1 war file(s)
[DeployPublisher][INFO] Deploying /var/lib/jenkins/workspace/hiring-app/target/hiring.war to container Tomcat
  [/var/lib/jenkins/workspace/hiring-app/target/hiring.war] is not deployed. Doing a fresh deployment.
  Deploying [/var/lib/jenkins/workspace/hiring-app/target/hiring.war]
[Slack Notifications] found #24 as previous completed, non-aborted build
[Slack Notifications] will send OnSuccessNotification because build matches and user preferences allow it
Finished: SUCCESS
```

In your tomcat server your package will be deployed.

```
[root@ip-172-31-111-184 apache-tomcat-9.0.111]# cd webapps
[root@ip-172-31-111-184 webapps]# ls
ROOT  docs  examples  hiring  hiring.war  host-manager  manager
[root@ip-172-31-111-184 webapps]#
```

Not secure 98.87.16.156:8080/hiring/

Gmail YouTube Maps

Welcome to Techie Horizon

2. Setup a Jenkins CICD pipeline using Declarative pipeline using feature-1.1

branch. https://github.com/betawins/sabear_simplecutomerapp/tree/feature-1.1

stages:

a. Git Clone

b. SonarQube Integration

c. Maven Compilation

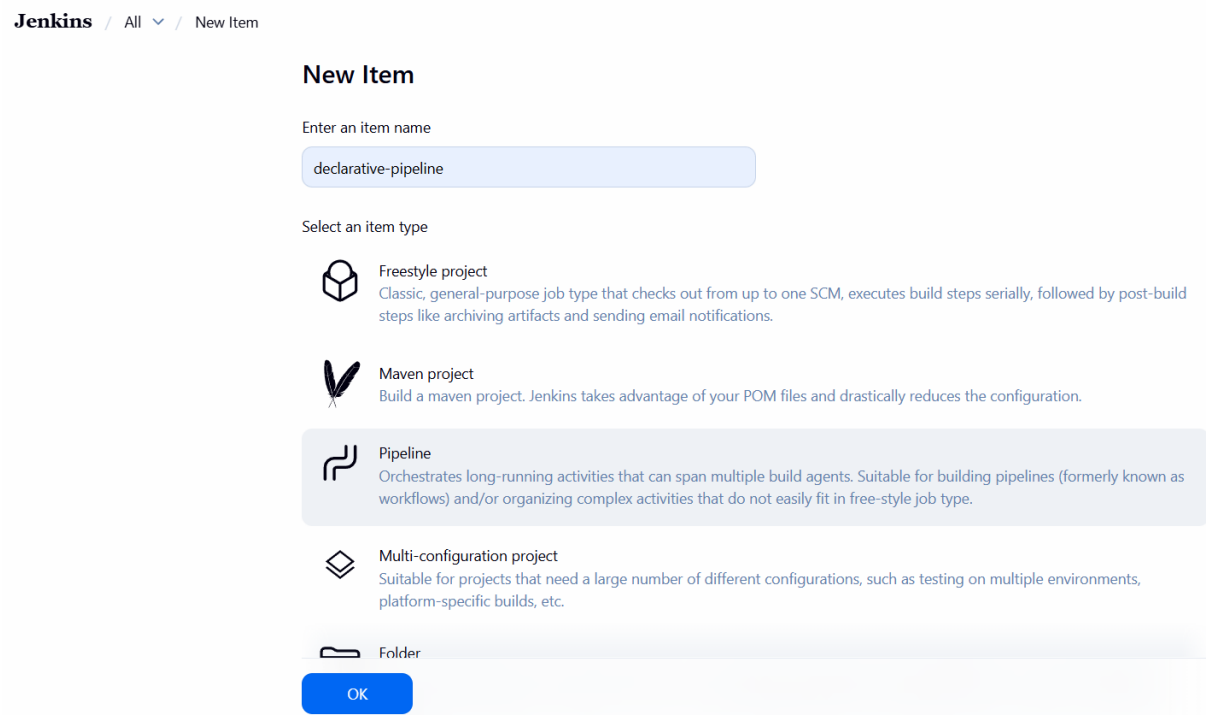
d. Nexus Artifactory

e. Slack Notification

f. Deploy On tomcat

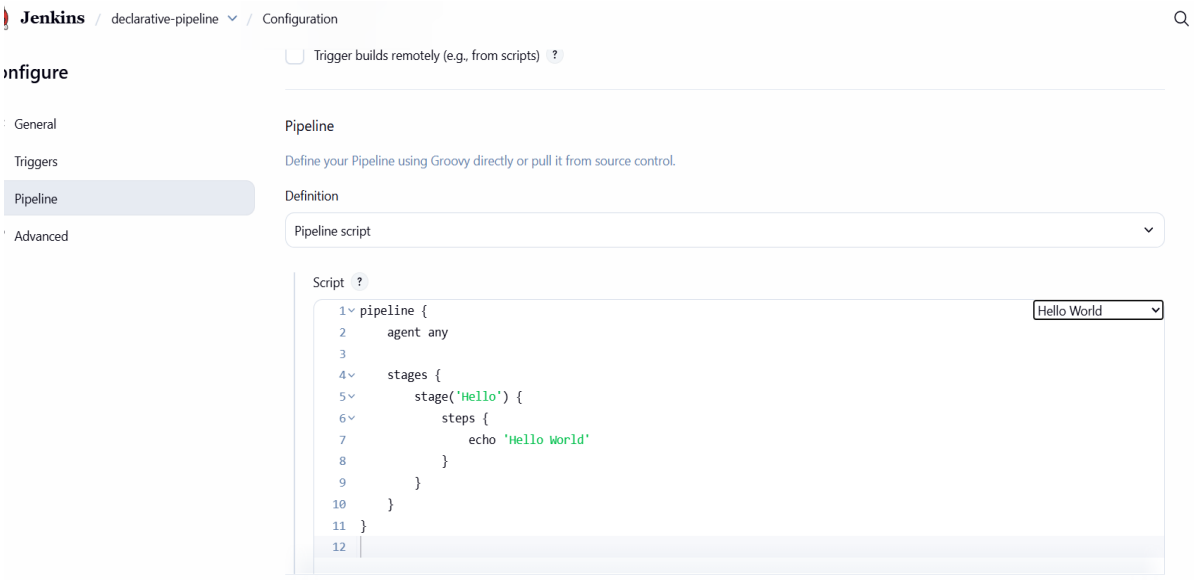
a. Git clone:

Click on new item enter name as declarative-pipeline and select type as pipeline.



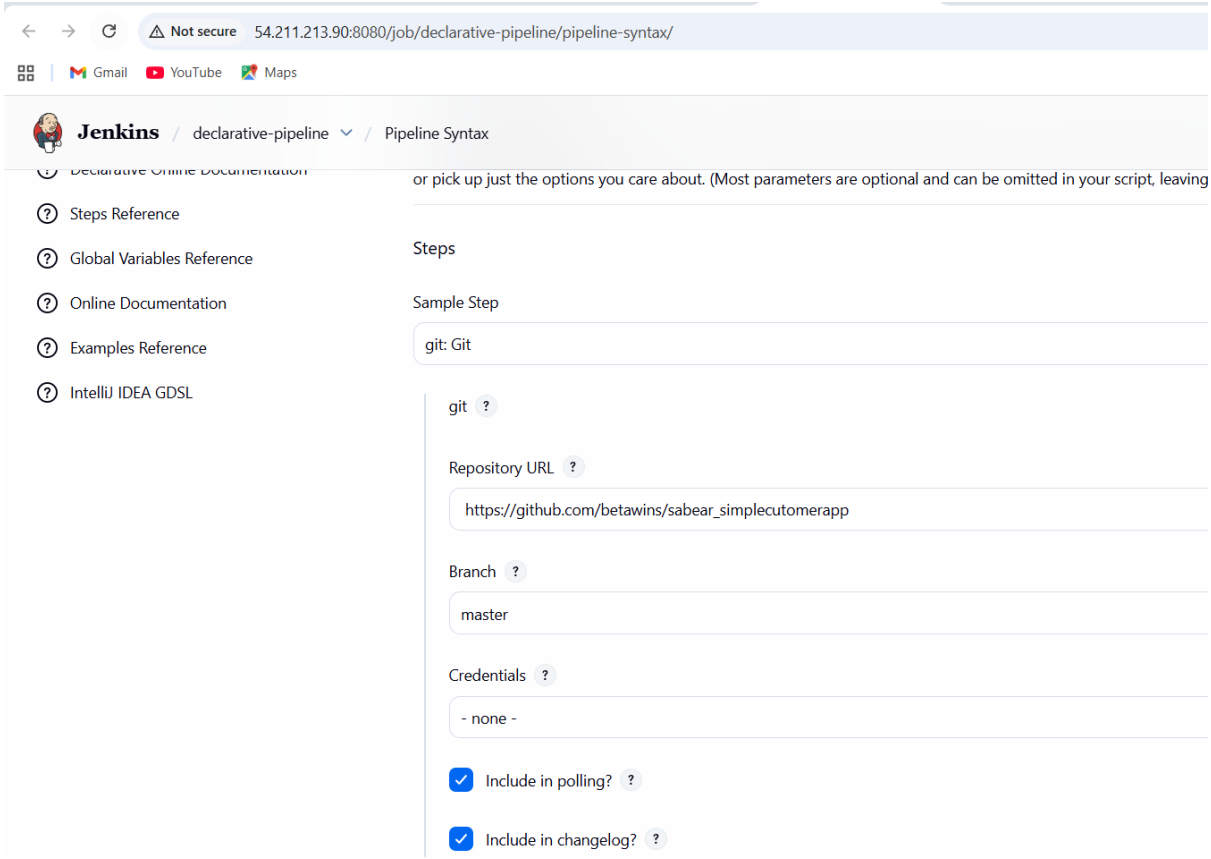
- Select pipeline for pipeline script and select helloworld.

- In scm select git and give your repository and click on save.

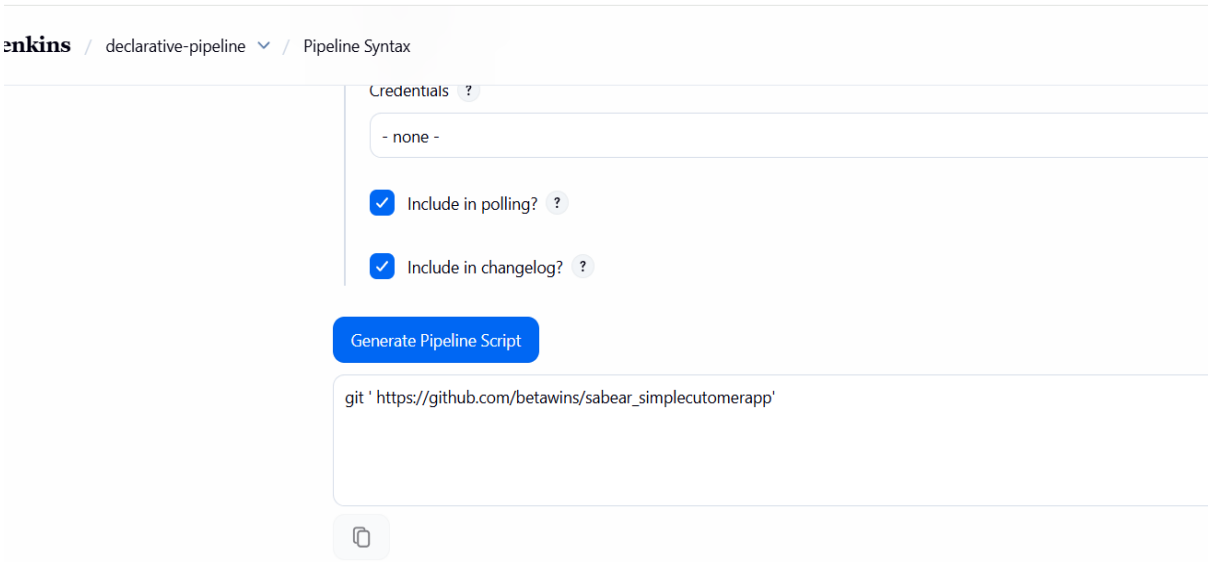


Click on pipeline syntax

- select git Git
- Give repository URL

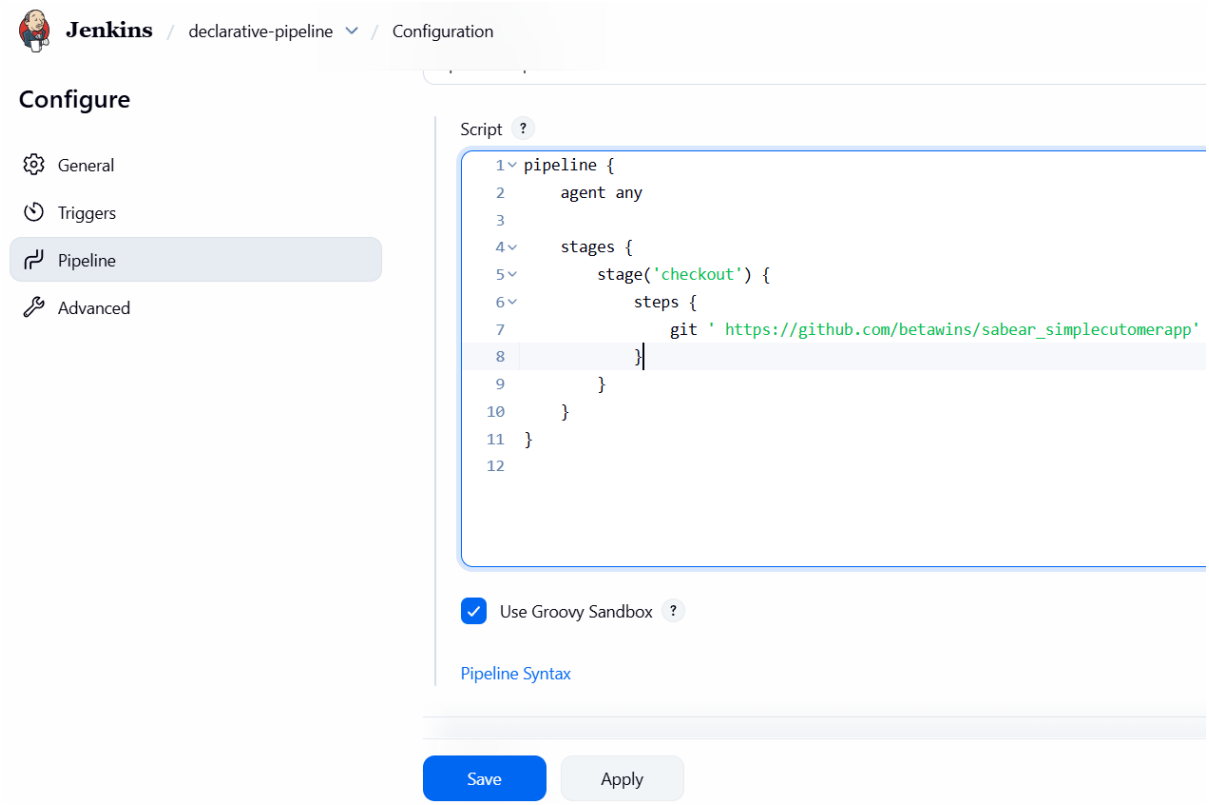


Click on generate pipeline script.it will create a link.

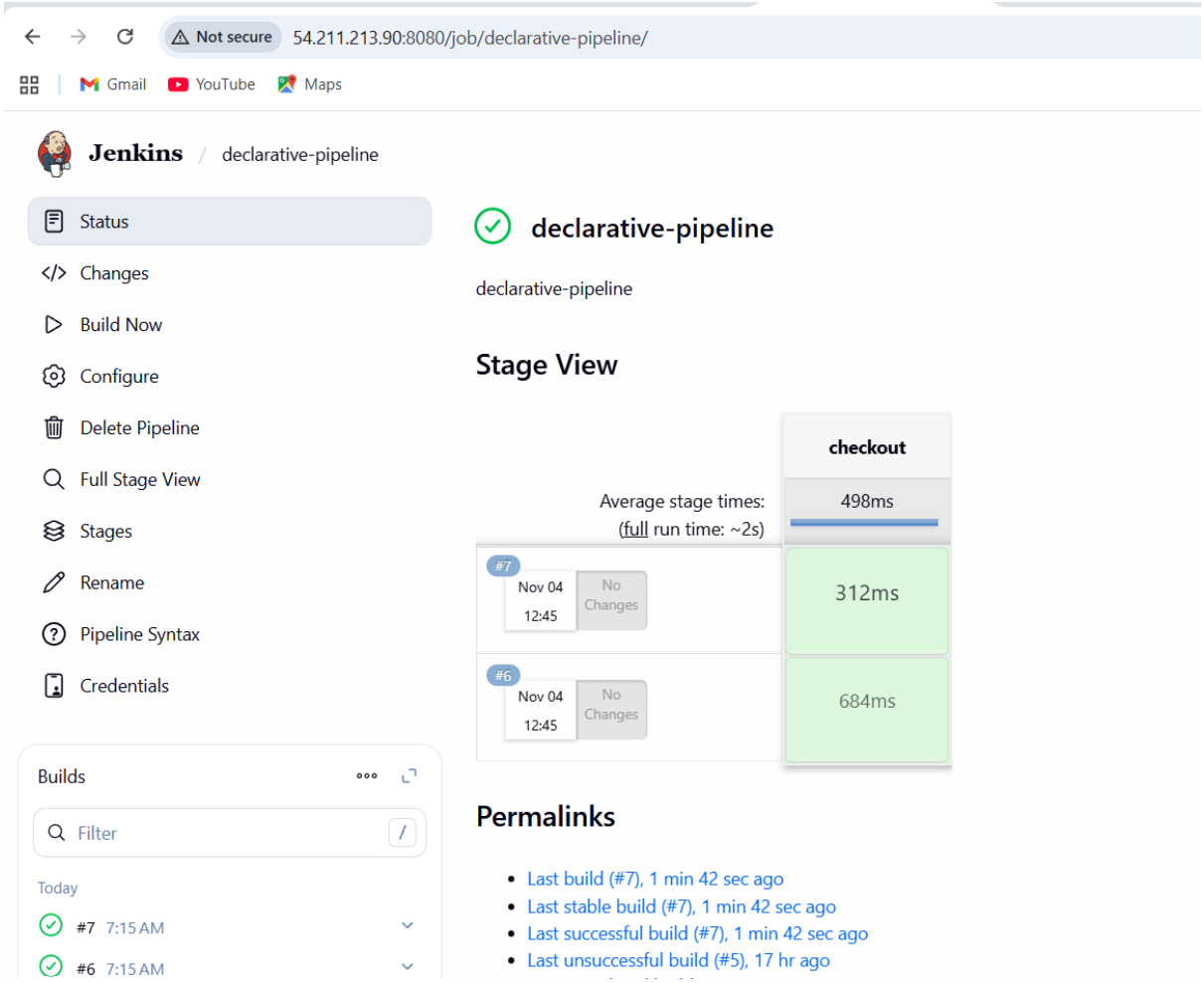


git ' https://github.com/betawins/sabear_simplecutomerapp'

add this link in the script.



Click on build now.



Git has cloned successfully.

```
[root@ip-172-31-50-64 workspace]# ls
declarative-pipeline  declarative-pipeline@tmp  hiring-app
[root@ip-172-31-50-64 workspace]# cd declarative-pipeline
[root@ip-172-31-50-64 declarative-pipeline]# ls
Jenkinsfile  Service-API.dockerfile  XMLSave.txt  build.xml  sonar-project.properties
README.md    WebContent              build        pom.xml    src
[root@ip-172-31-50-64 declarative-pipeline]#
```

b. Maven Compilation:

give mvn location script should be like.

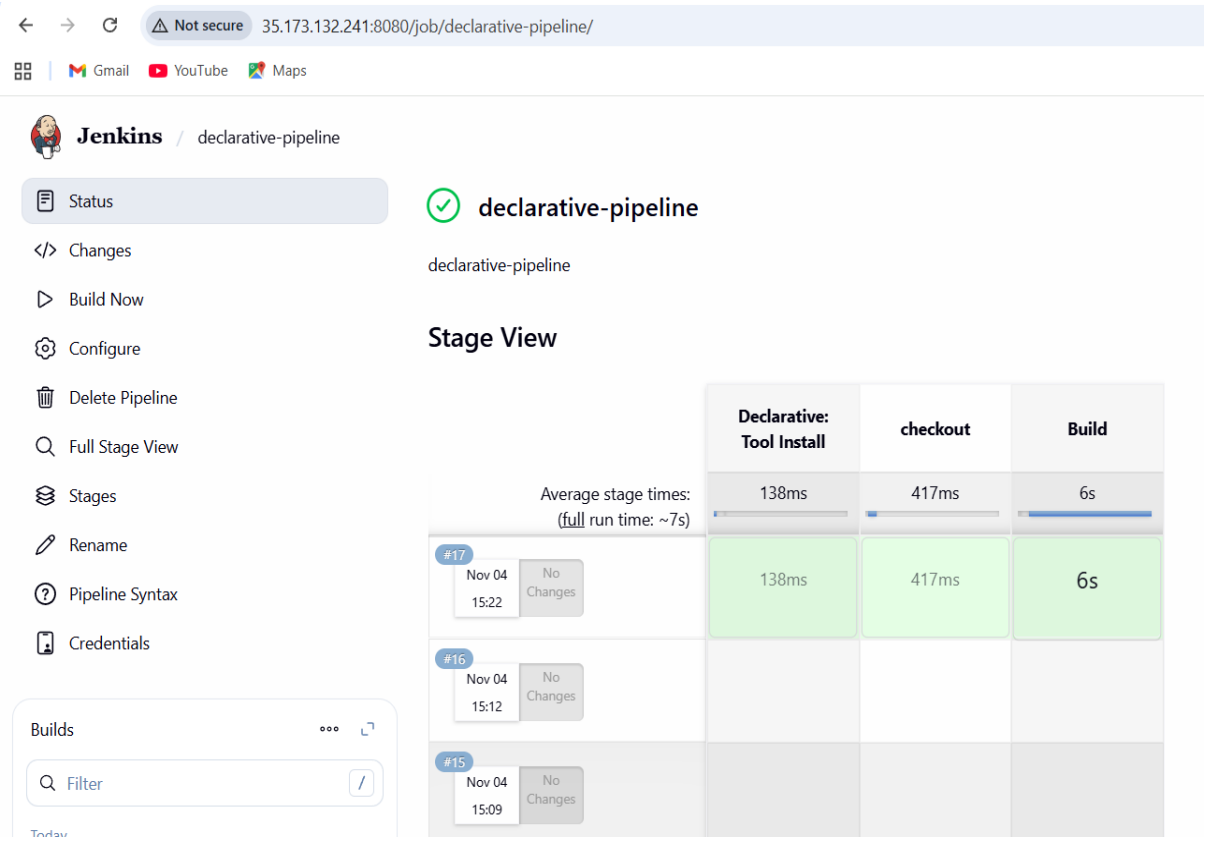
```
pipeline {
    agent any
    tools{
        maven "MVN_HOME"
    }


    stages {
        stage('checkout') {
            steps {
                git '
https://github.com/betawins/sabear_simplecutomerapp'
            }
        }
        stage('build'){
            steps{
                sh  "mvn clean install -f pom.xml"
            }
            }
}
}
```

Script ?

```
pipeline {
    agent any
    tools {
        maven "MVN_HOME"
    }
    stages {
        stage('checkout'){
            steps {
                git ' https://github.com/betawins/sabear_simplecutomerapp'
            }
        }
        stage ("Build"){
            steps{
                sh "mvn clean install -f pom.xml"
            }
```

☑ Use Groovy Sandbox ?

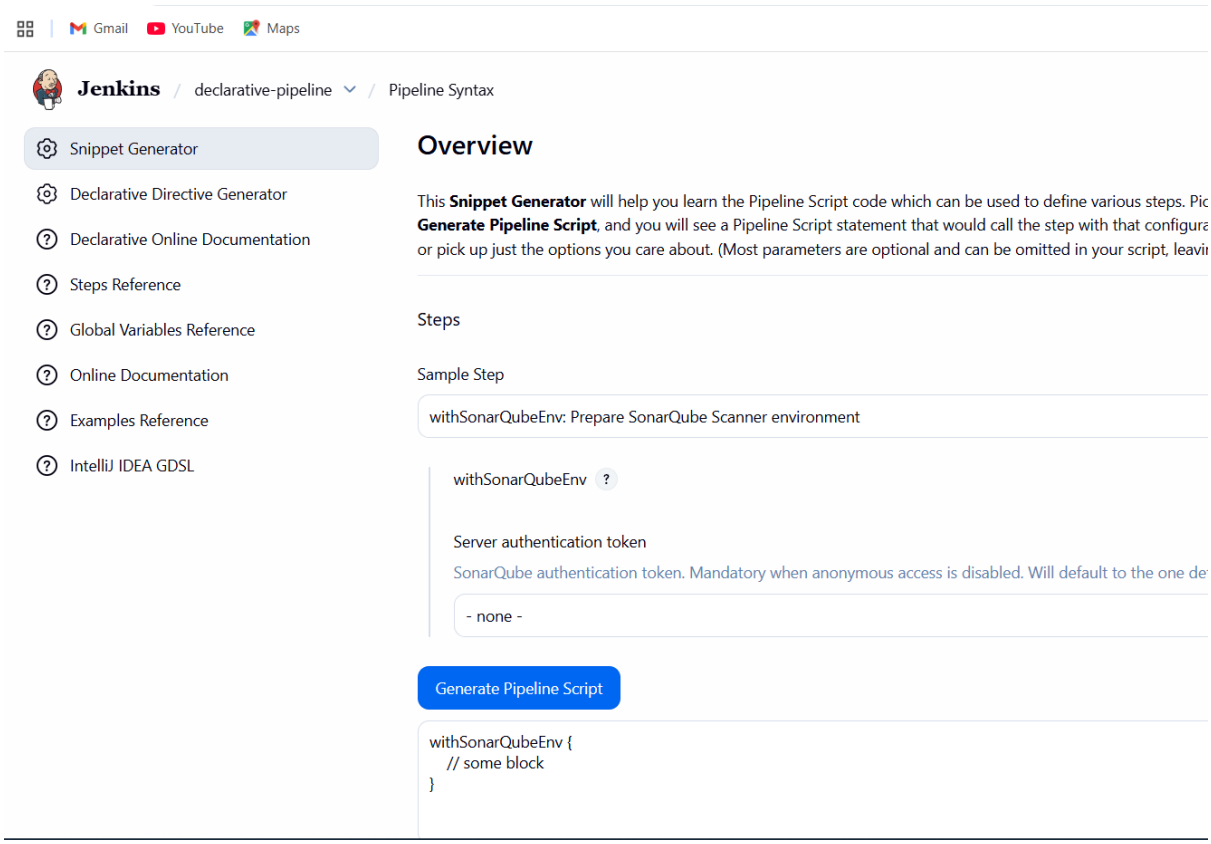
Click on build now.



```
[root@ip-172-31-50-64 jenkins]# cd workspace
[root@ip-172-31-50-64 workspace]# ls
declarative-pipeline  declarative-pipeline@tmp  hiring-app
[root@ip-172-31-50-64 workspace]# cd declarative-pipeline
[root@ip-172-31-50-64 declarative-pipeline]# ls
Jenkinsfile  Service-API.dockerfile  XMLSave.txt  build.xml  sonar-project.properties  target
README.md    WebContent              build        pom.xml    src
[root@ip-172-31-50-64 declarative-pipeline]# cd target
[root@ip-172-31-50-64 target]# ls
SimpleCustomerApp-18-SNAPSHOT  SimpleCustomerApp-18-SNAPSHOT.war  maven-archiver
[root@ip-172-31-50-64 target]#
```

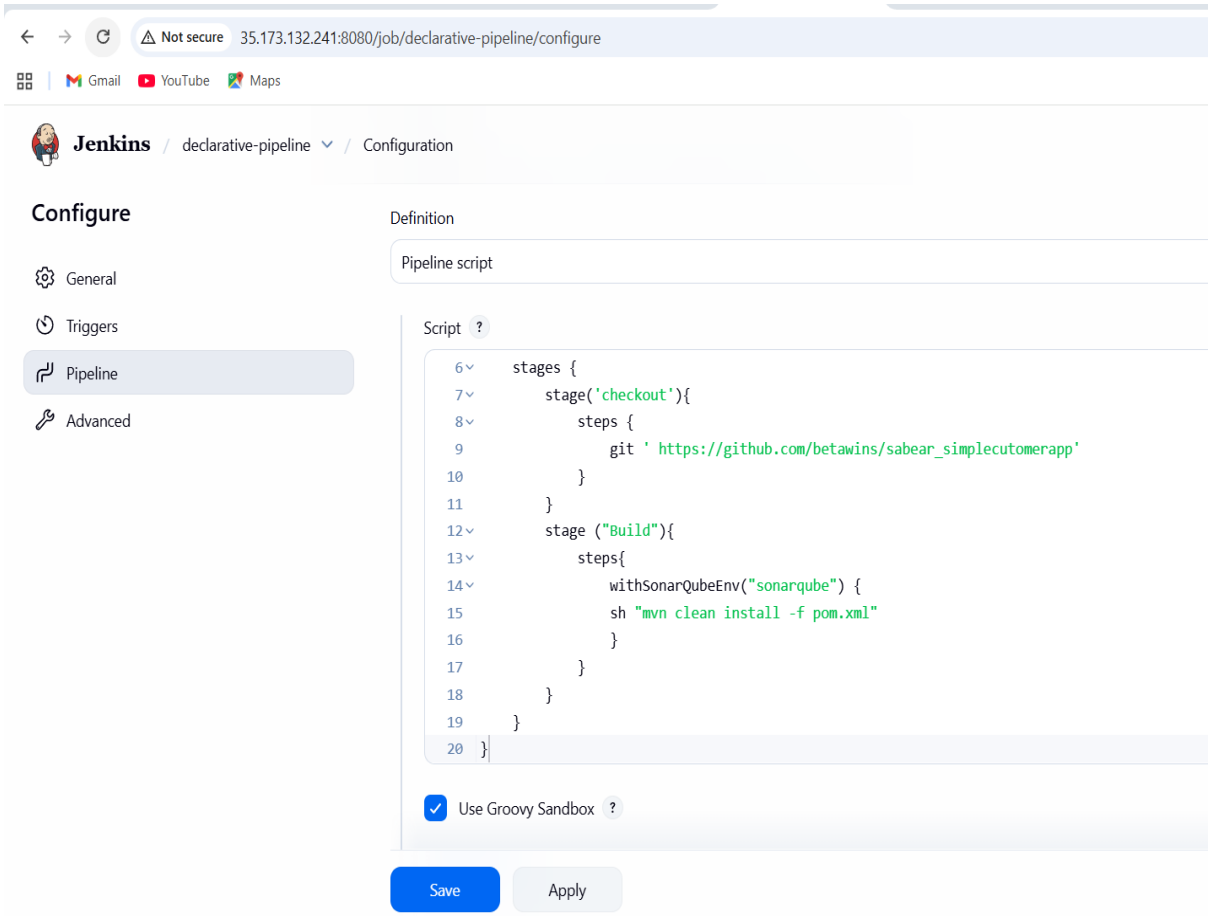
c. Sonar integration:

go to pipeline syntax and select with sonarqube env:prepare sonarqube scanner environment click generate pipline script it will give you script add that in the declarative pipeline script.

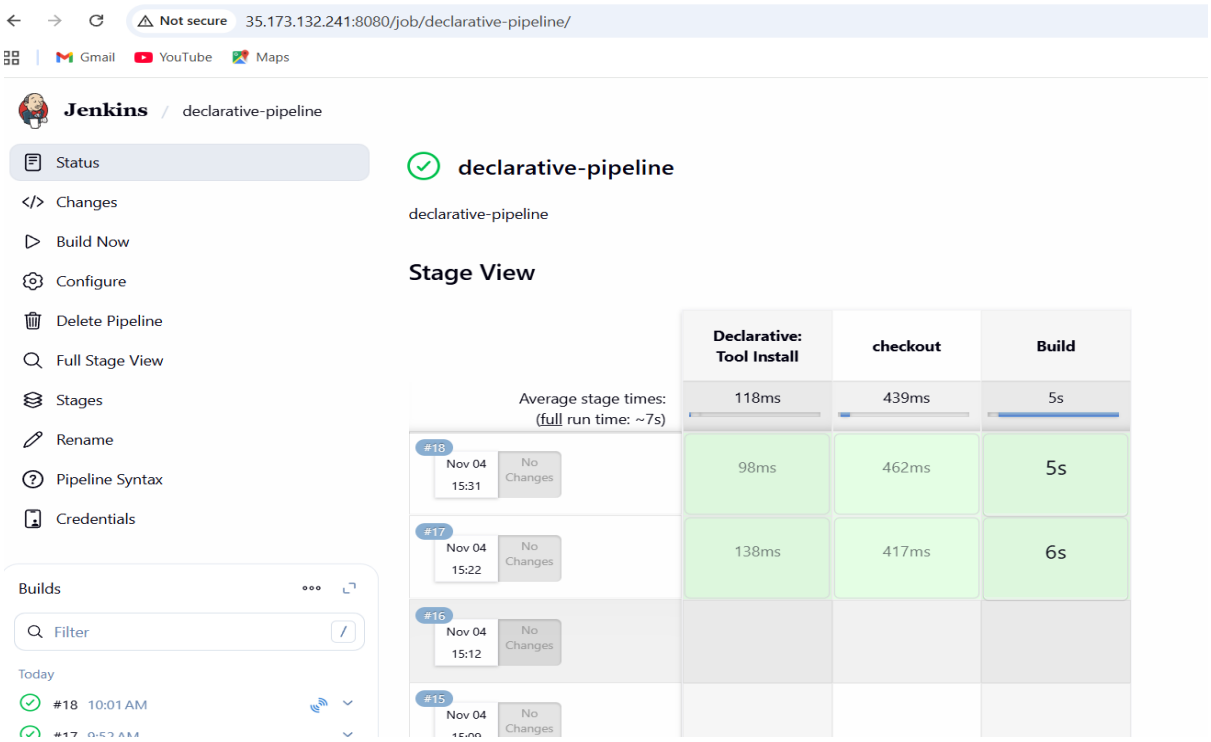



```
withSonarQubeEnv {
    // some block
```

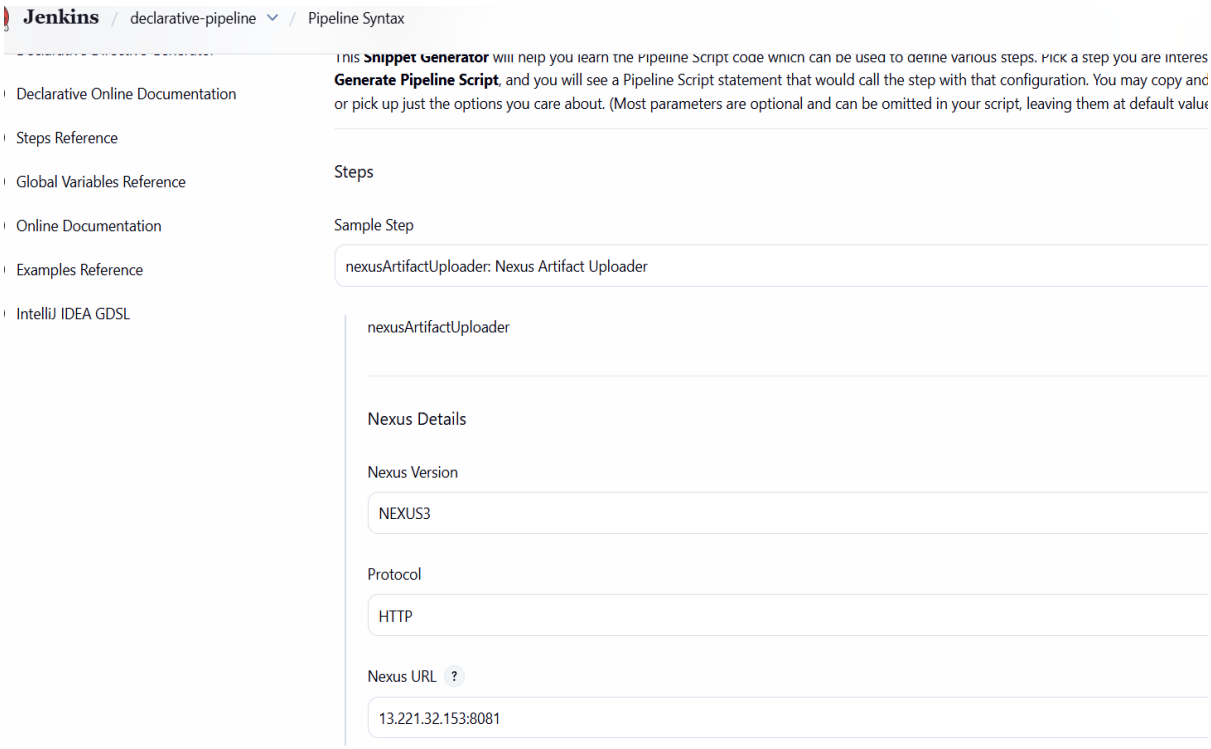
}



click on build now.



d. Nexus Artifactory:



Not secure 35.173.132.241:8080/job/declarative-pipeline/pipeline-syntax/

Gmail YouTube Maps

Jenkins / declarative-pipeline / Pipeline Syntax

13.221.32.153:8081

Credentials

admin/****** (nexus)

GroupId

com.javatpoint

Version

${BUILD_NUMBER}-SNAPSHOT

Repository ?

simple-customer-app

Artifacts

+ Add

Generate Pipeline Script

larative-pipeline / Pipeline Syntax

simple-customer-app

Artifacts

Artifact

ArtifactId

simple-customer-app

Type ?

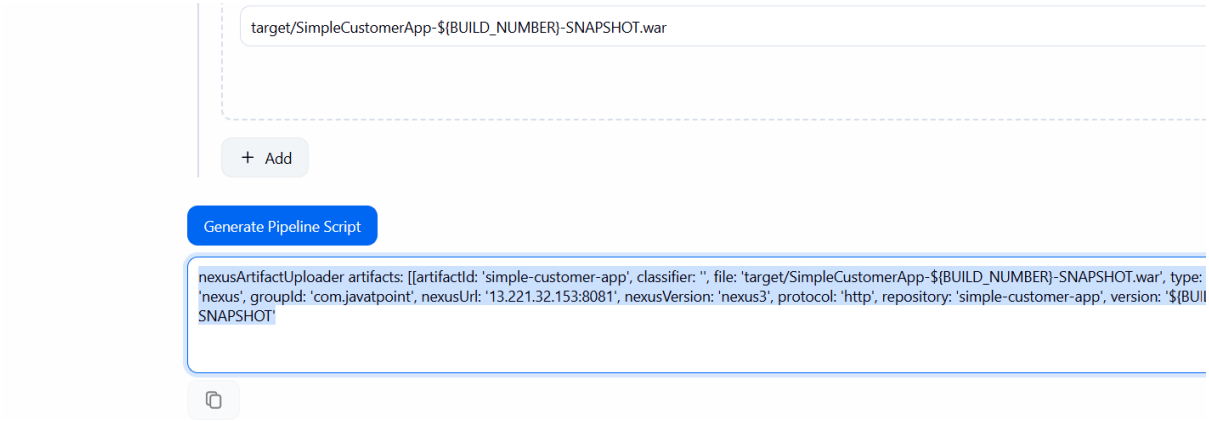
war

Classifier ?

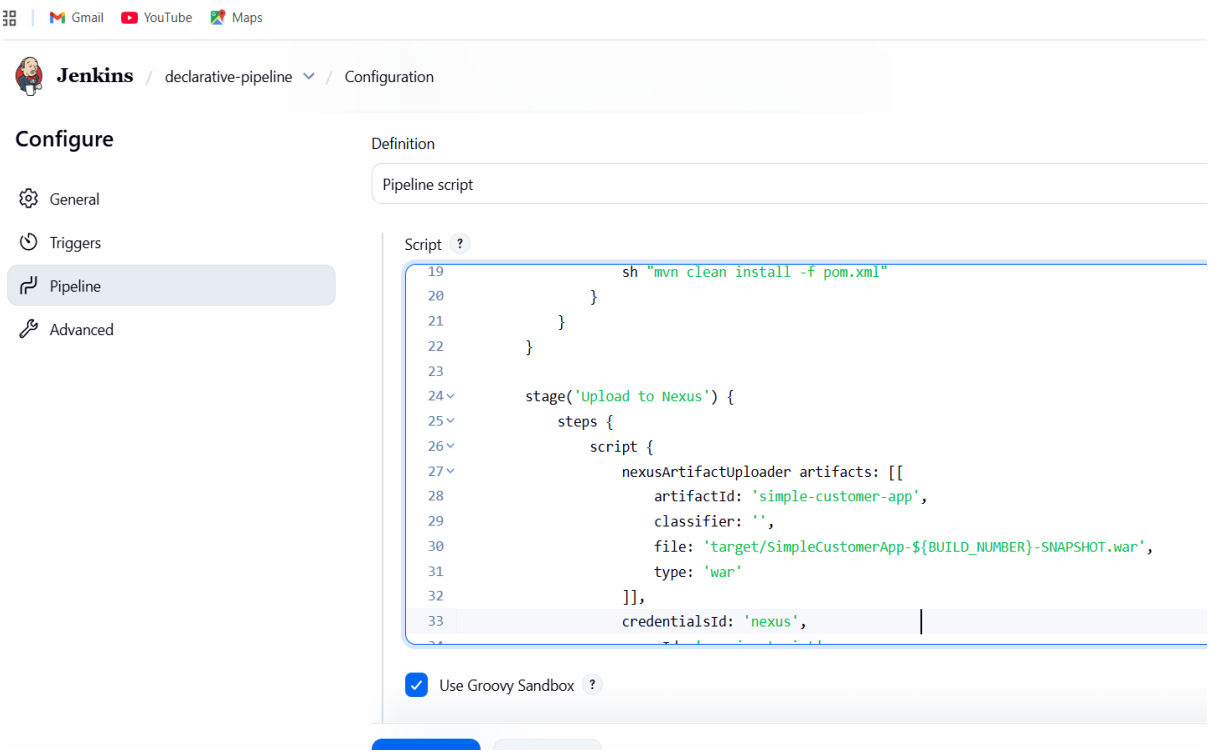
File ?

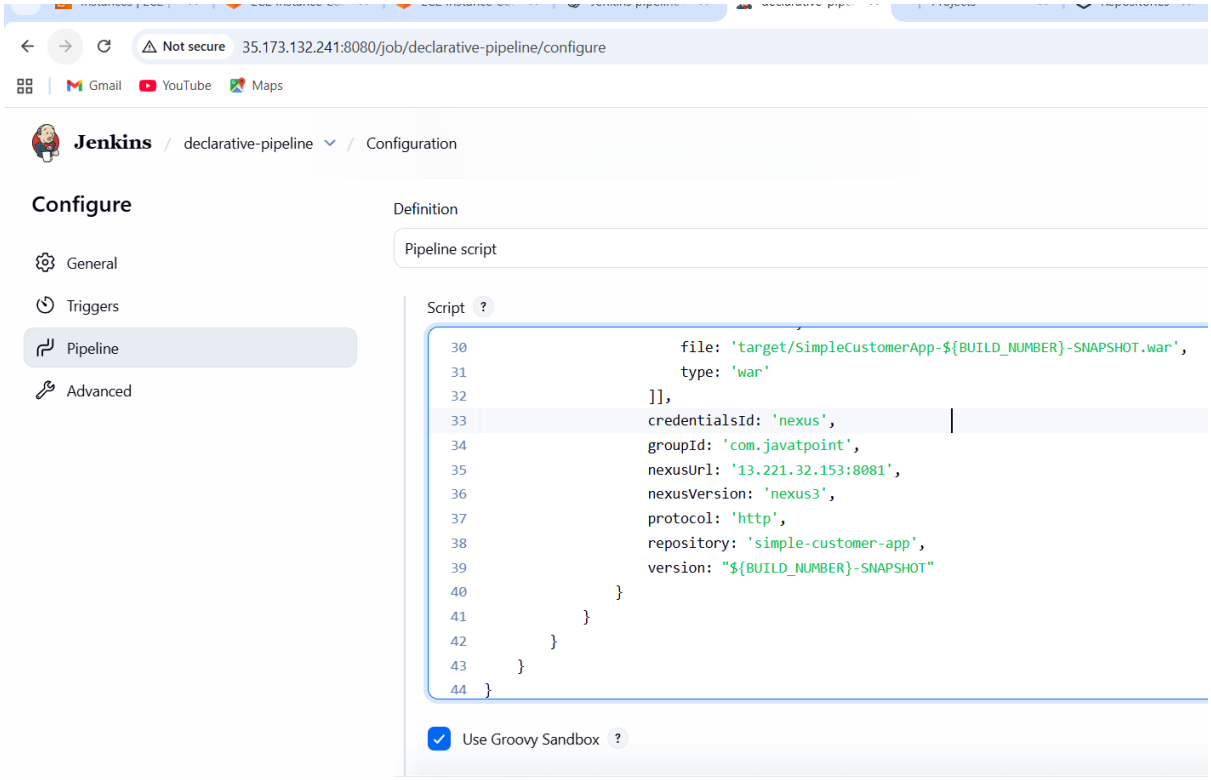
target/SimpleCustomerApp-${BUILD_NUMBER}-SNAPSHOT.war

It will give an script add that in the script.

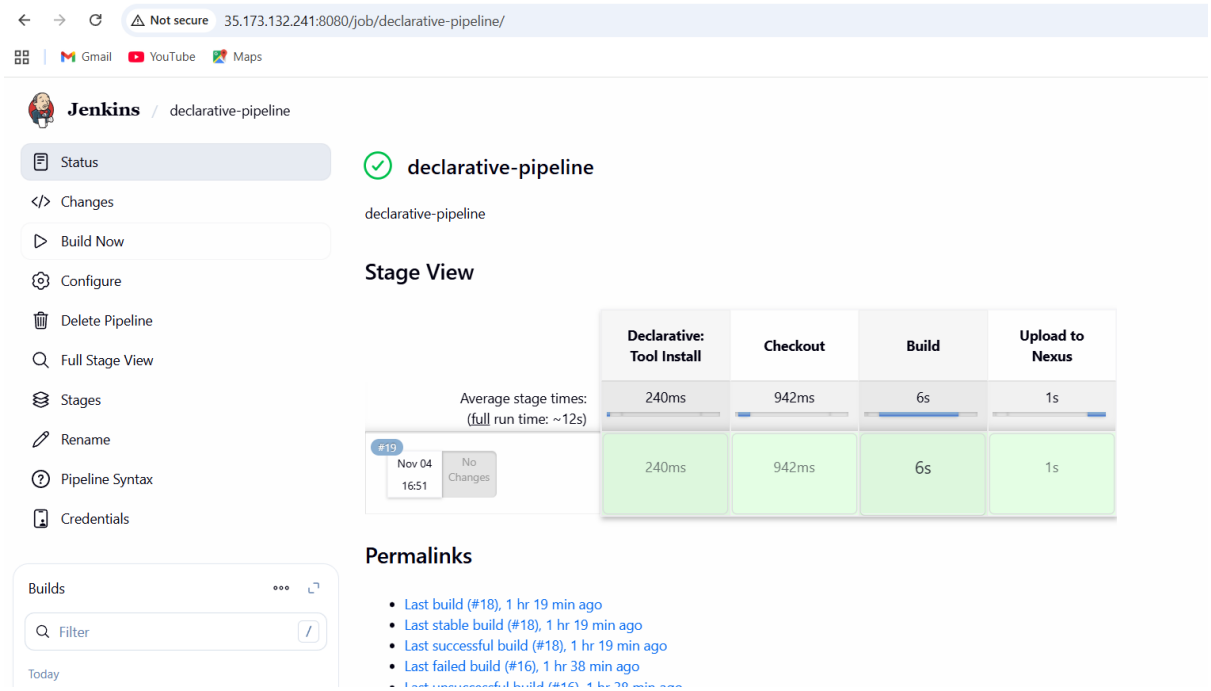


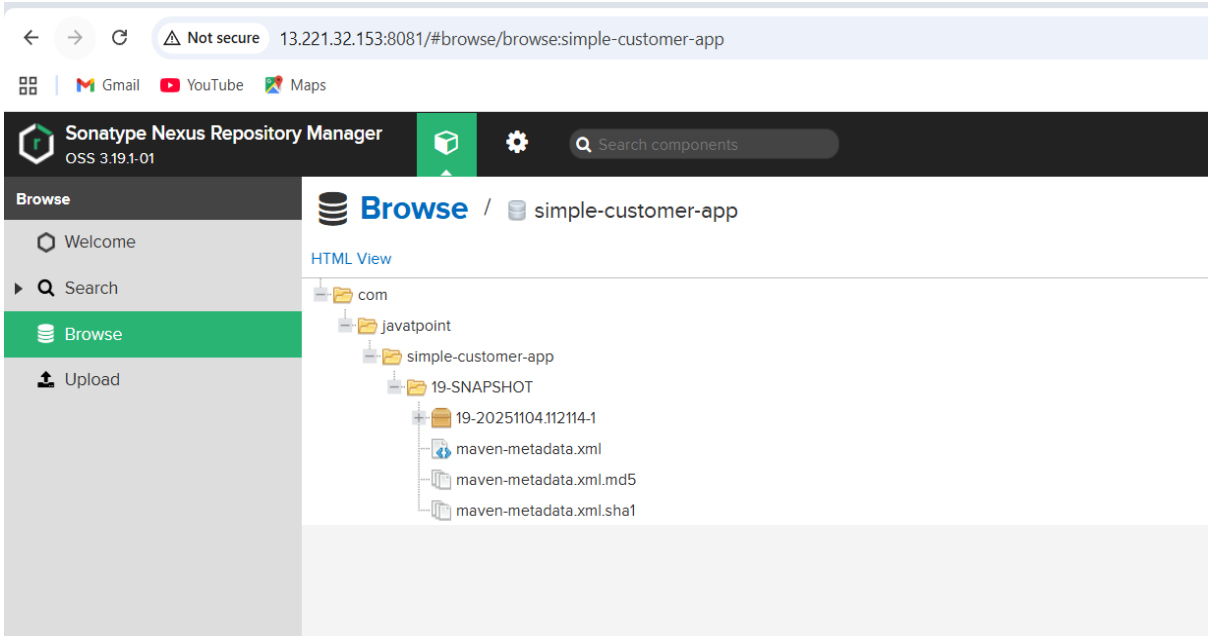
nexusArtifactUploader artifacts: [[artifactId: 'simple-customer-app', classifier: '', file: 'target/SimpleCustomerApp-${BUILD_NUMBER}-SNAPSHOT.war', type: 'war']], credentialsId: 'nexus', groupId: 'com.javatpoint', nexusUrl: '13.221.32.153:8081', nexusVersion: 'nexus3', protocol: 'http', repository: 'simple-customer-app', version: '${BUILD_NUMBER}-SNAPSHOT'



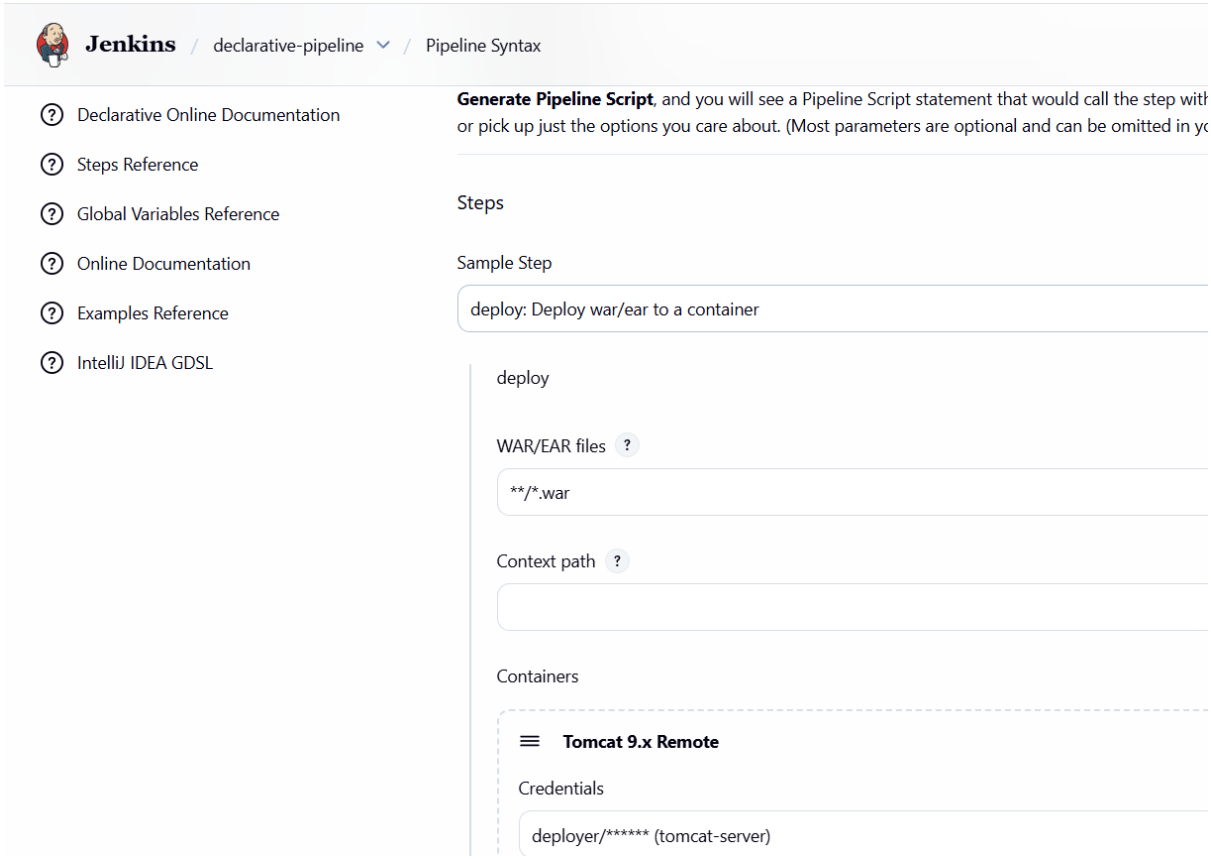


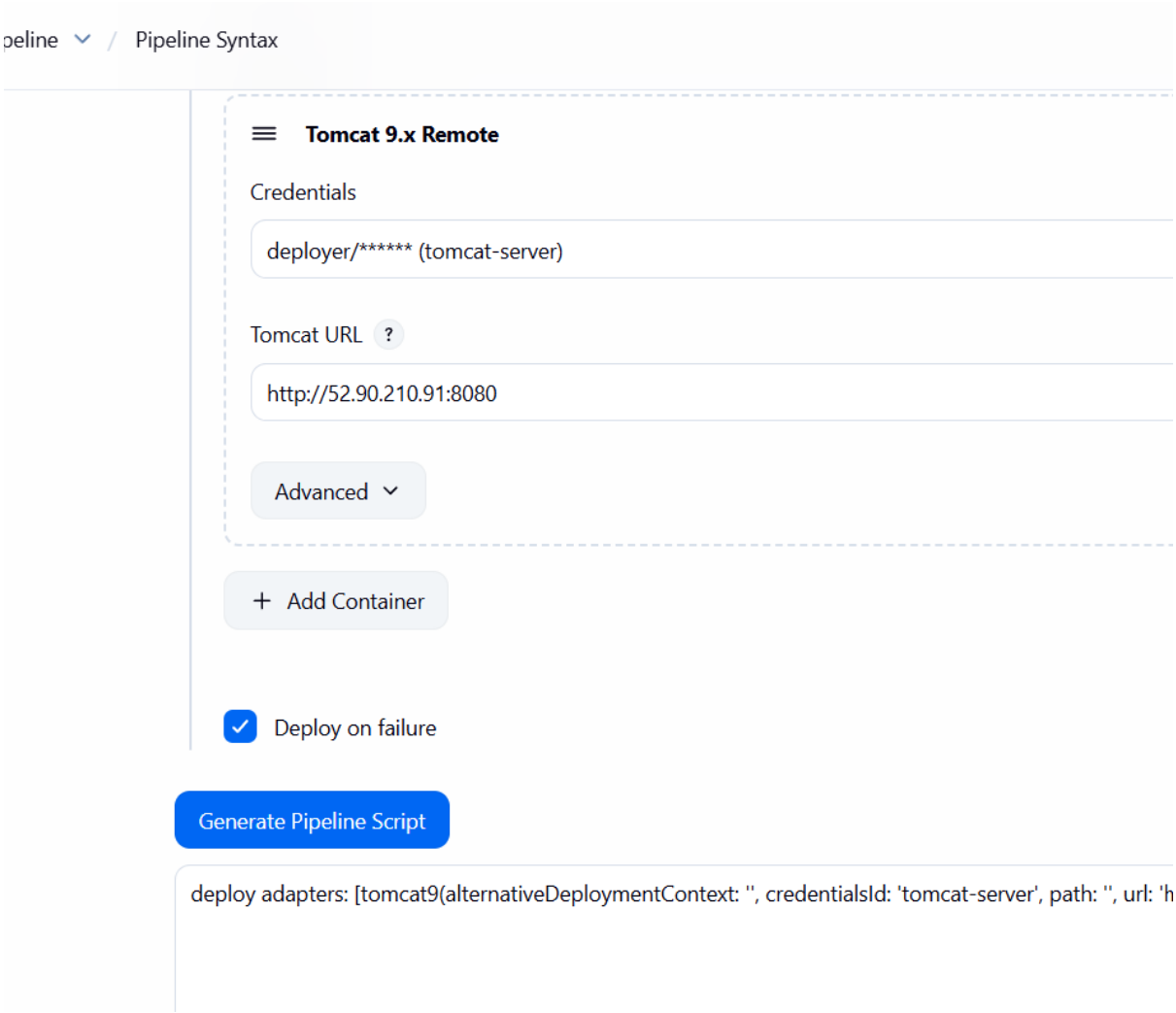
Click on build now.





f . Deploy On tomcat:





deploy adapters: [tomcat9(alternativeDeploymentContext: '', credentialsId: 'tomcat-server', path: '', url: 'http://52.90.210.91:8080')], contextPath: null, war: '**/*.war'

Script

```
46                  script {
47                      deploy adapters: [tomcat9(
48                          alternativeDeploymentContext: '',
49                          credentialsId: 'tomcat-server',     // Jenkins credentials for Tomca
50                          path: '',                           // leave empty for ROOT deploymer
51                          url: 'http://52.90.210.91:8080'     // Tomcat server URL
52                      )],
53                      contextPath: null,
54                      war: '**/*.war'
55                  }
56              }
57          }
58      }
59  }
60
```

Use Groovy Sandbox

Save Apply

Search

Click on build now.

Not secure 35.173.132.241:8080/job/declarative-pipeline/

Gmail YouTube Maps

Jenkins / declarative-pipeline

Status
Changes
Build Now
Configure
Delete Pipeline
Full Stage View
Stages
Rename
Pipeline Syntax
Credentials

Builds
Filter
Today

declarative-pipeline

declarative-pipeline

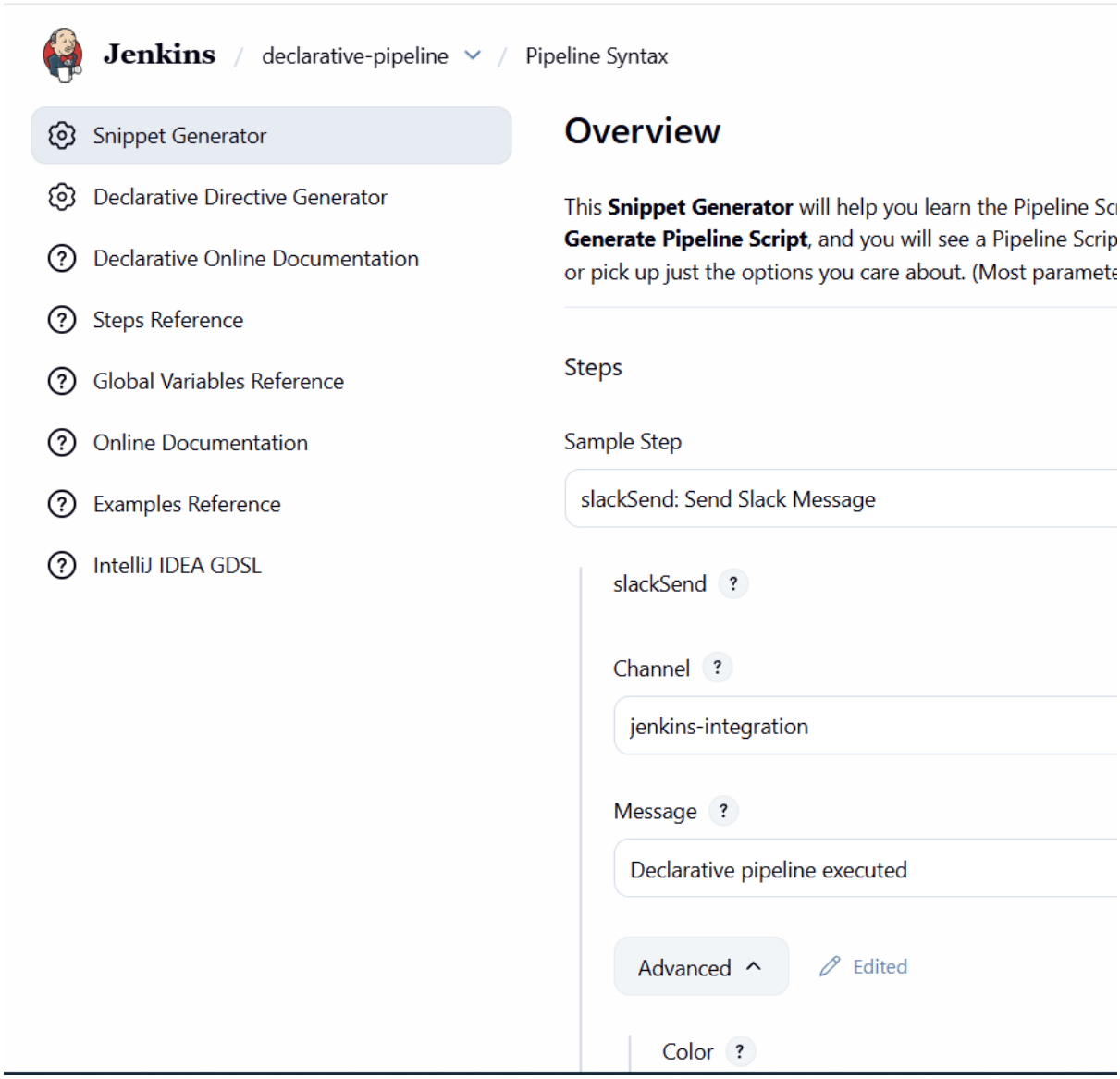
Stage View

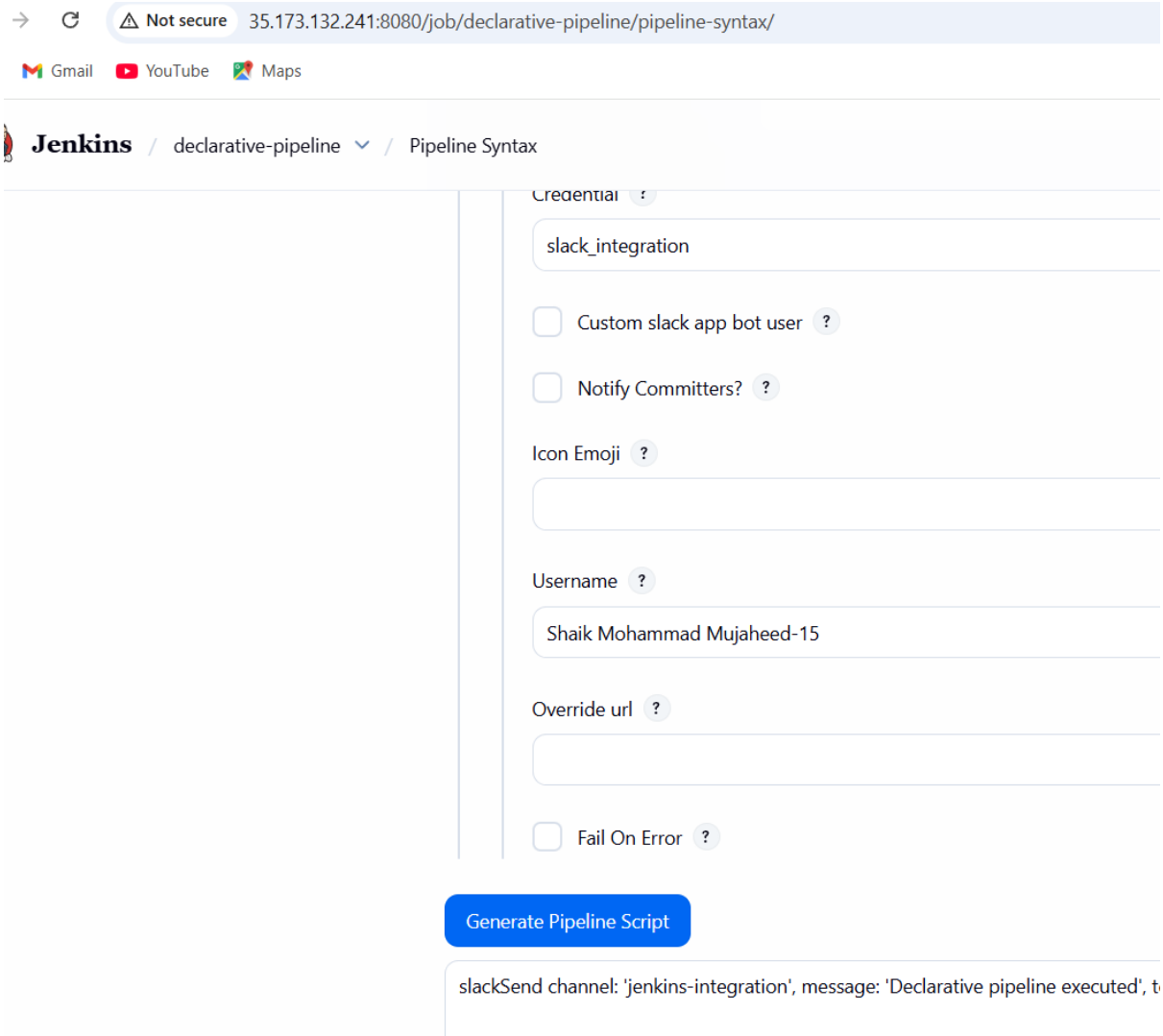
	Declarative: Tool Install	Checkout	Build	Upload to Nexus	Deploy to Tomcat
Average stage times: (full run time: ~10s)	177ms	688ms	6s	922ms	724ms
#20 Nov 04 17:40 No Changes	115ms	435ms	6s	422ms	724ms
#19 Nov 04 16:51 No Changes	240ms	942ms	6s	1s	

Permalinks

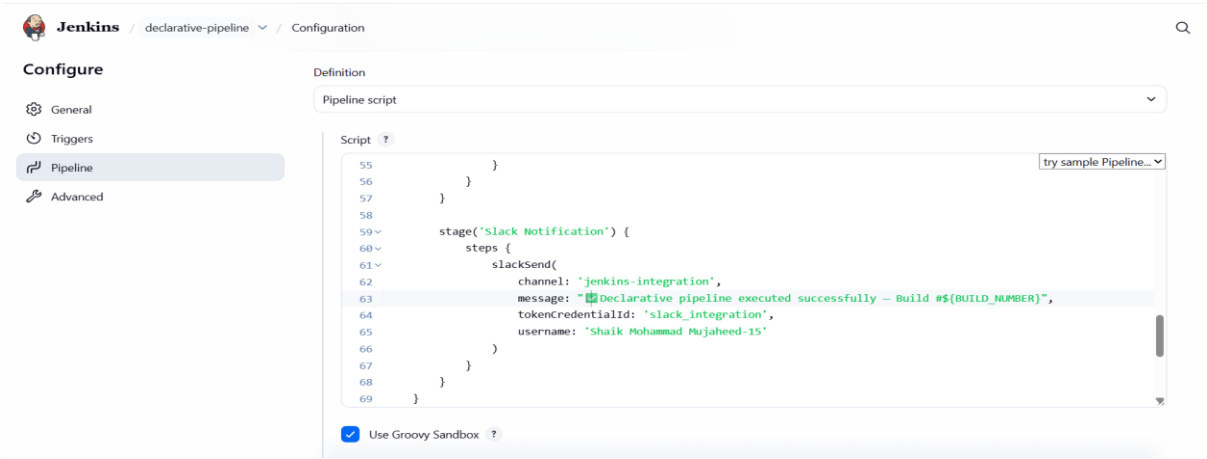
```
[root@ip-172-31-111-184 ~]# cd /opt
[root@ip-172-31-111-184 opt]# ls
apache-tomcat-9.0.111  apache-tomcat-9.0.111.tar.gz  aws
[root@ip-172-31-111-184 opt]# cd apache-tomcat-9.0.111/
[root@ip-172-31-111-184 apache-tomcat-9.0.111]# ls
BUILDING.txt  CONTRIBUTING.md  LICENSE  NOTICE  README.md  RELEASE-NOTES  RUNNING.txt  bin  conf  lib  logs  temp  webapps  work
[root@ip-172-31-111-184 apache-tomcat-9.0.111]# cd webapps
[root@ip-172-31-111-184 webapps]# ls
ROOT  SimpleCustomerApp-20-SNAPSHOT  SimpleCustomerApp-20-SNAPSHOT.war  docs  examples  hiring  hiring.war  host-manager  manager
[root@ip-172-31-111-184 webapps]#
```

f. Slack Notification:

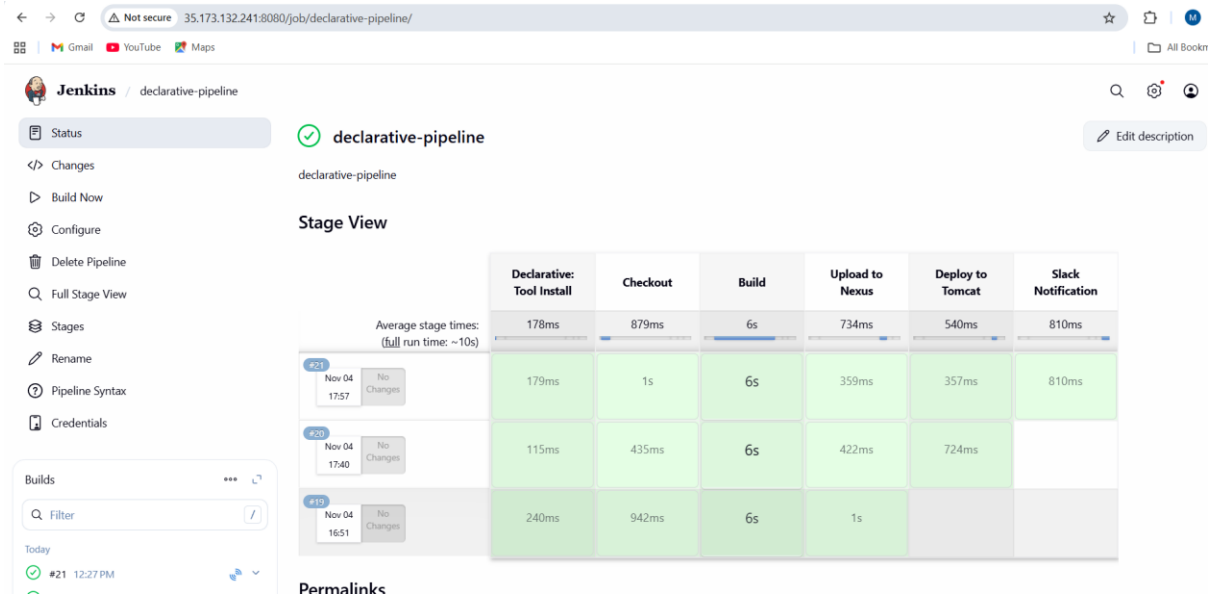


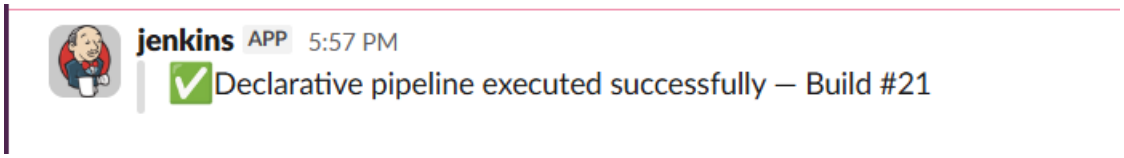


slackSend channel: 'jenkins-integration', message: 'Declarative pipeline executed', tokenCredentialId: 'slack_integration', username: 'Shaik Mohammad Mujaheed-15'



click on build now.





3. Setup a Jenkins CICD pipeline using Scripted pipeline using feature-1.1 branch. https://github.com/betawins/sabear_simplecutomerapp/tree/feature-1.1

stages:

a. Git Clone

b. SonarQube Integration

c. Maven Compilation

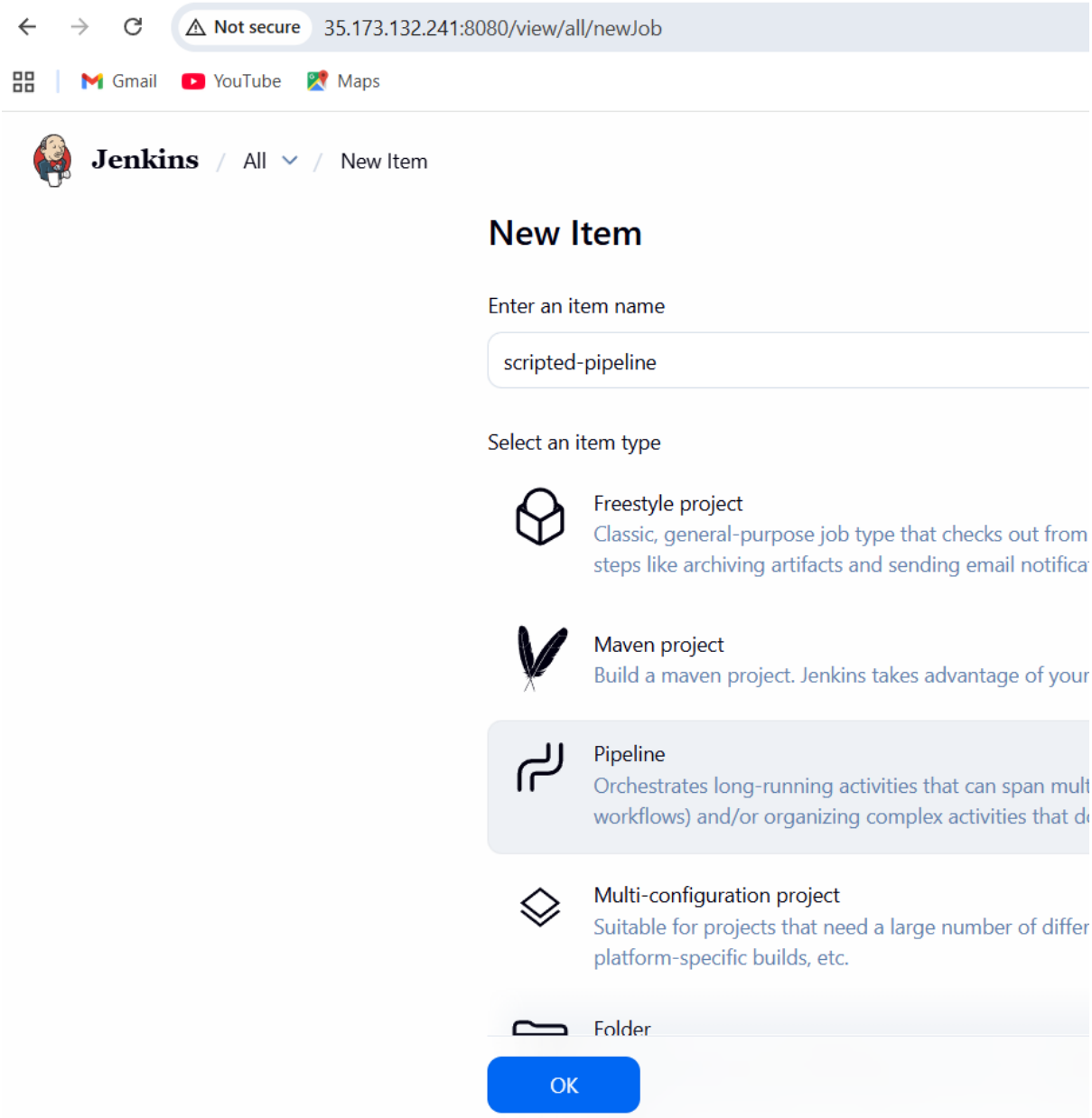
d. Nexus Artifactory

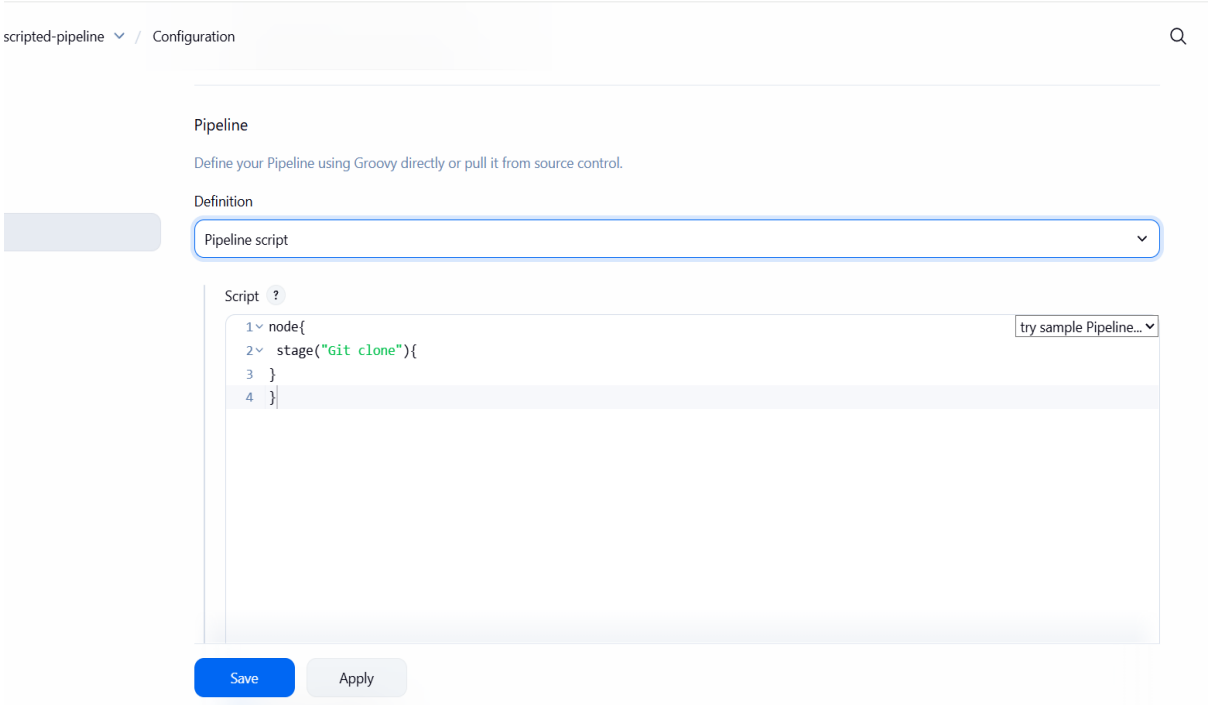
e. Slack Notification

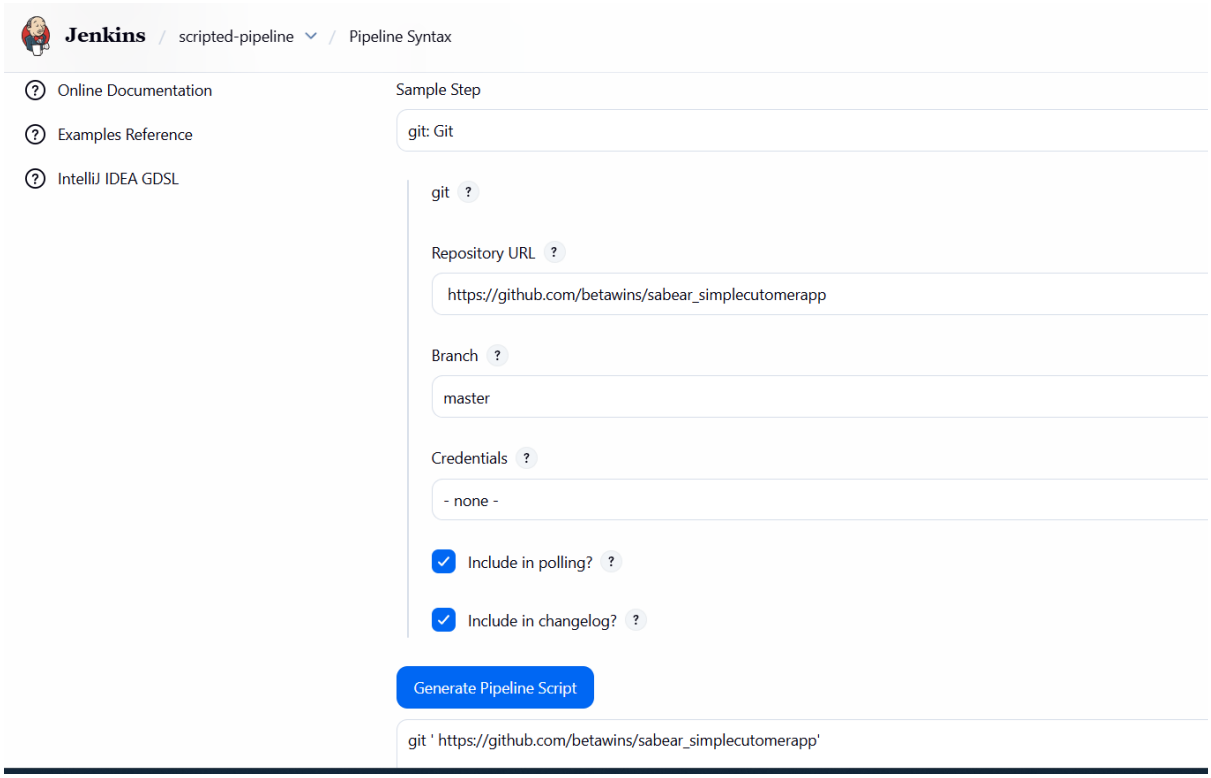
f. Deploy On tomcat

a. git clone:

go to Jenkins add new item name as scripted pipeline select type as pipeline.

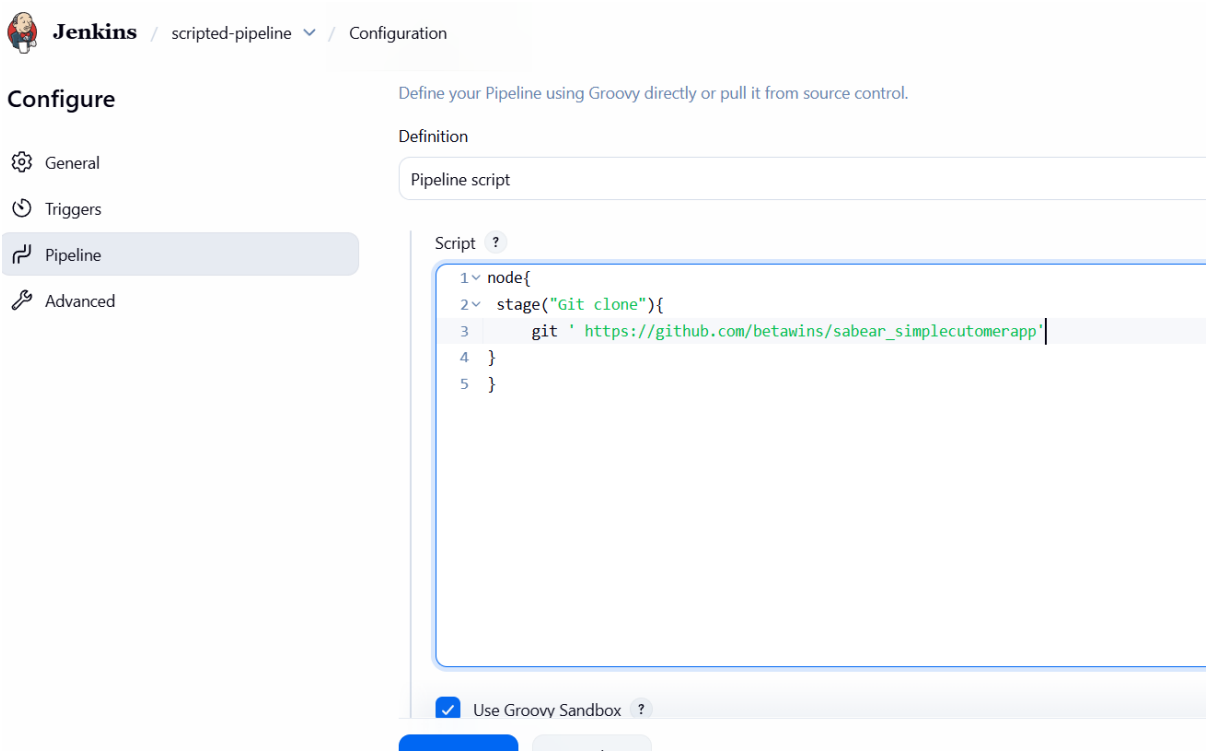


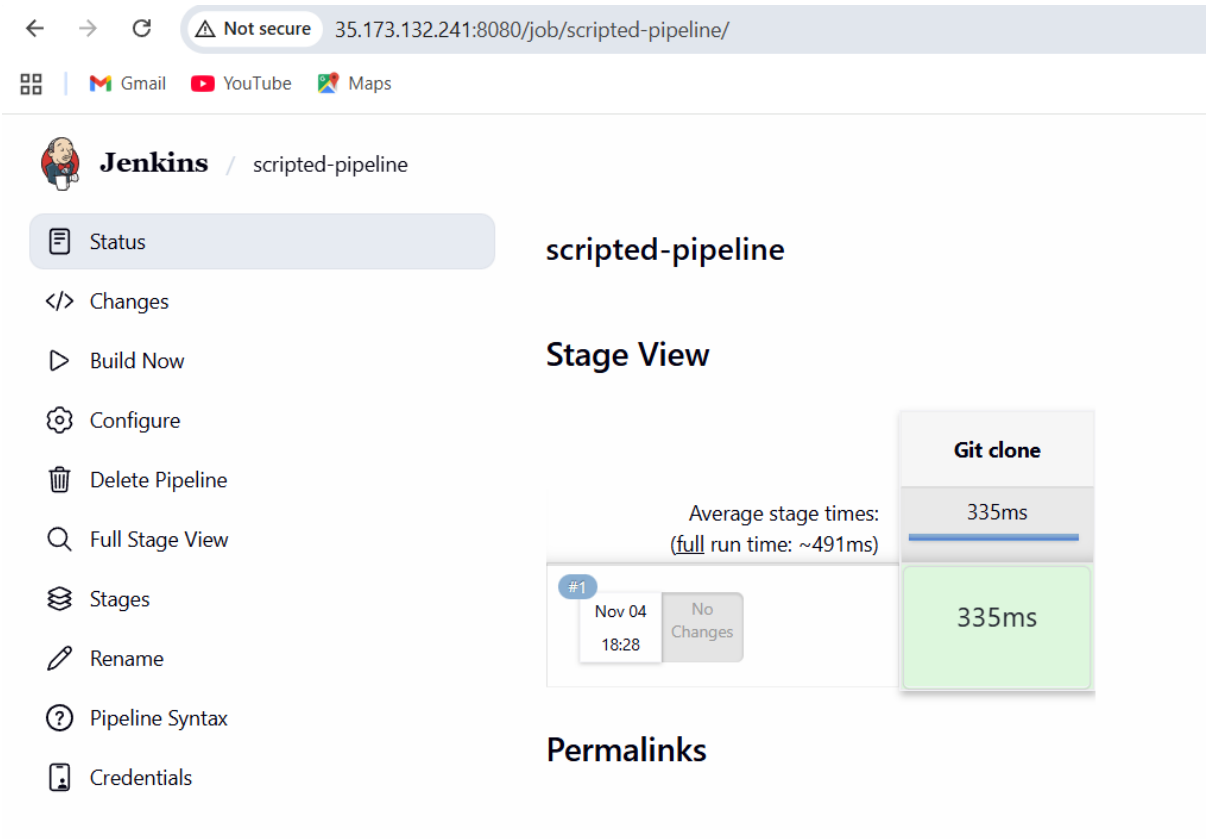




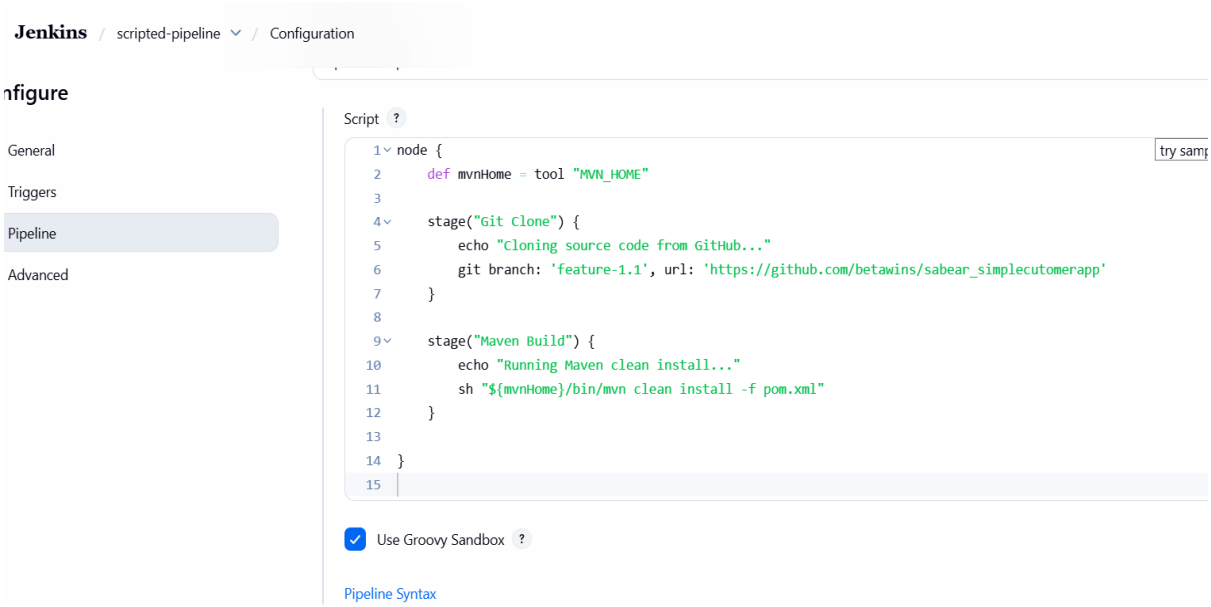
git ' https://github.com/betawins/sabear_simplecutomerapp'

add this in script.

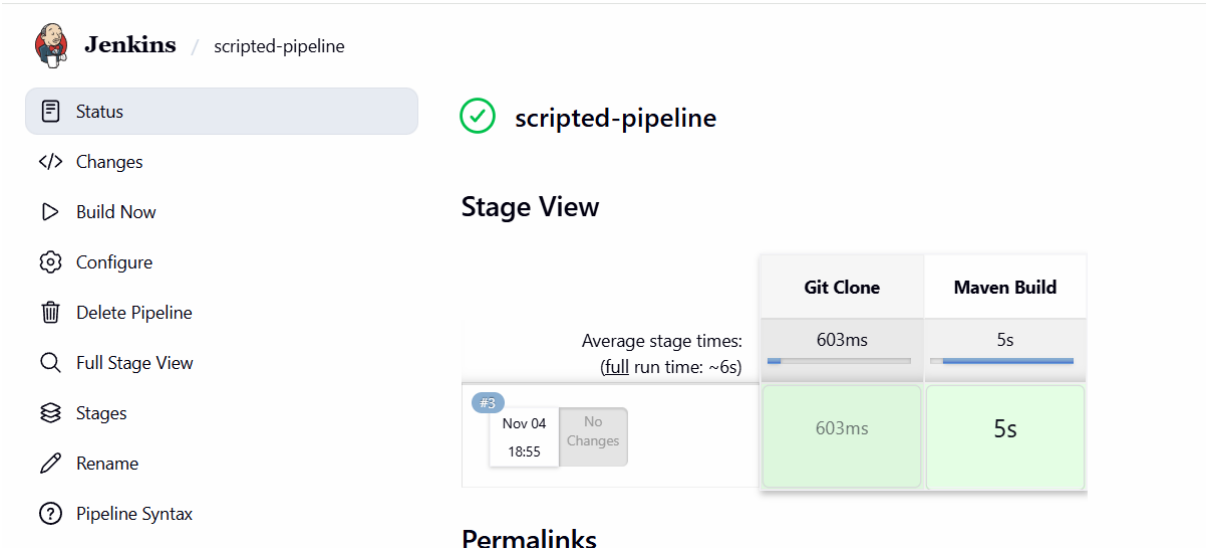




b. maven compailation:

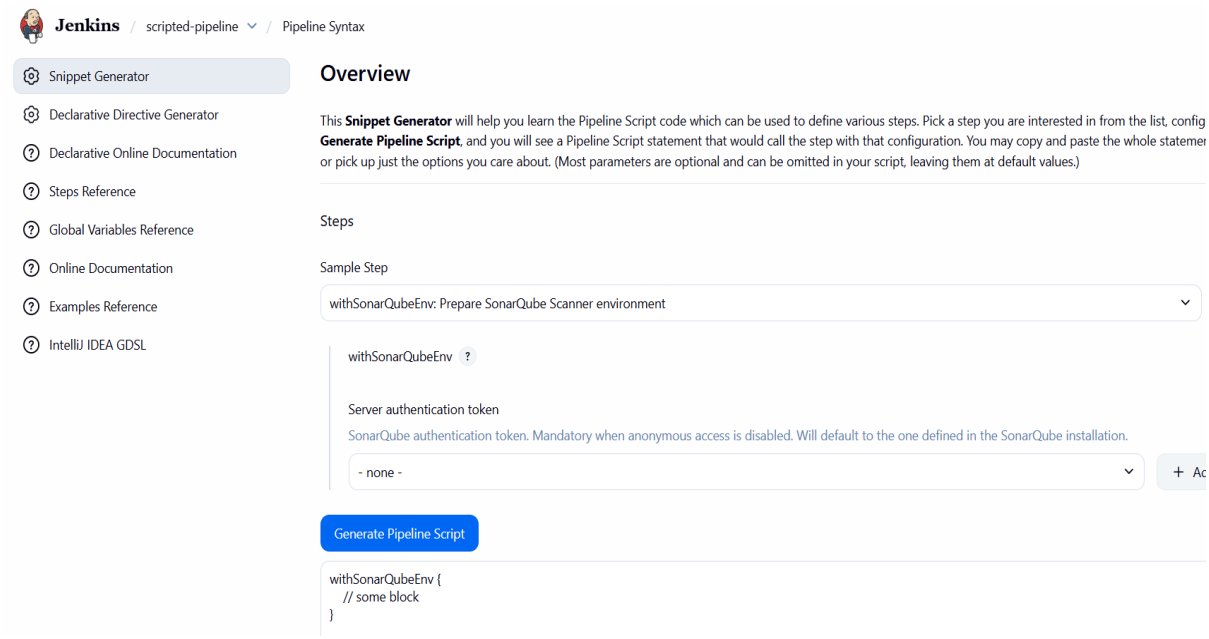


click on build now.

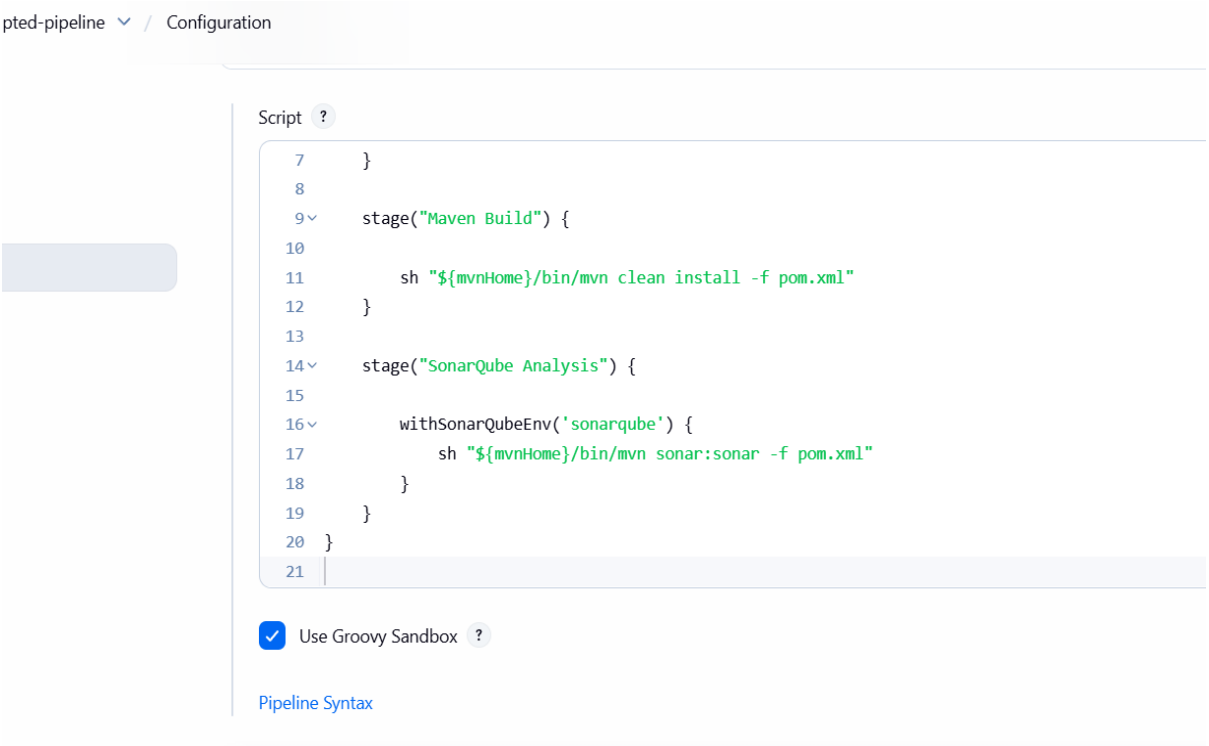


c. sonarqube integration:

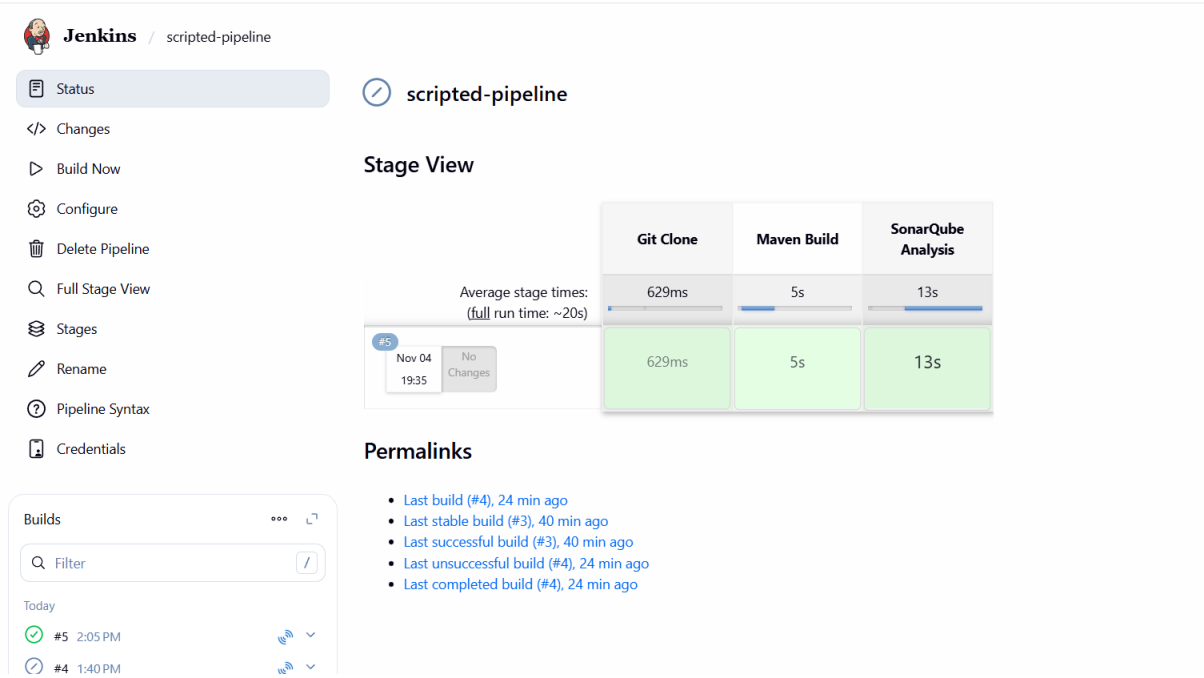
click on pipeline syntax and add with sonarqube env click on generate pipeline syntax.add the syntax in the script.




```
withSonarQubeEnv {
    // some block
}
```

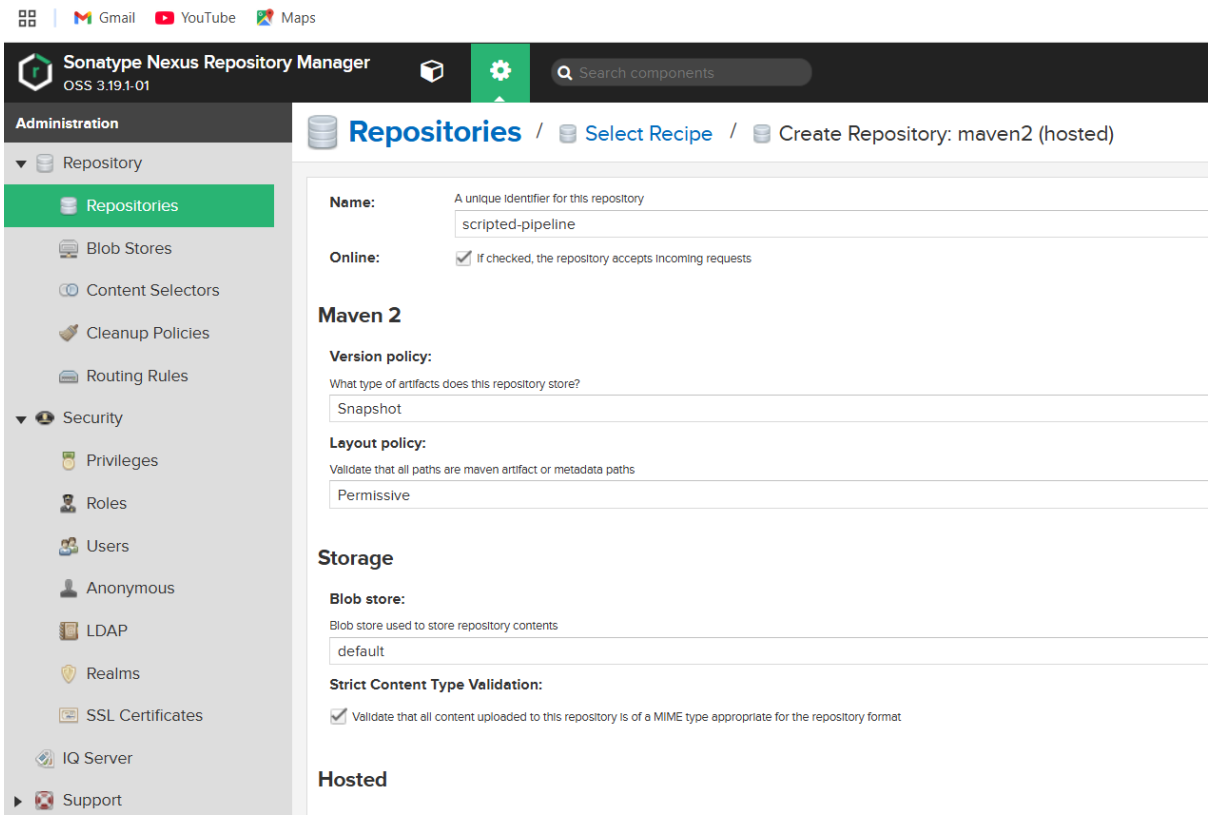


Click on build now.



d. Nexus Artifactory:

create a repository with scripted pipeline name.



Jenkins / scripted-pipeline / Pipeline Syntax

Online Documentation

Examples Reference

IntelliJ IDEA GDSL

Sample Step

nexusArtifactUploader: Nexus Artifact Uploader

nexusArtifactUploader

Nexus Details

Nexus Version

NEXUS3

Protocol

HTTP

Nexus URL ?

13.221.32.153:8081

Credentials

admin/****** (nexus)

GroupId

Not secure 23.20.86.229:8080/job/scripted-pipeline/pipeline-syntax/

Gmail YouTube Maps

Jenkins / scripted-pipeline / Pipeline Syntax

GroupId

com.javatpoint

Version

${BUILD_NUMBER}-SNAPSHOT

Repository ?

scripted-pipeline

Artifacts

Artifact

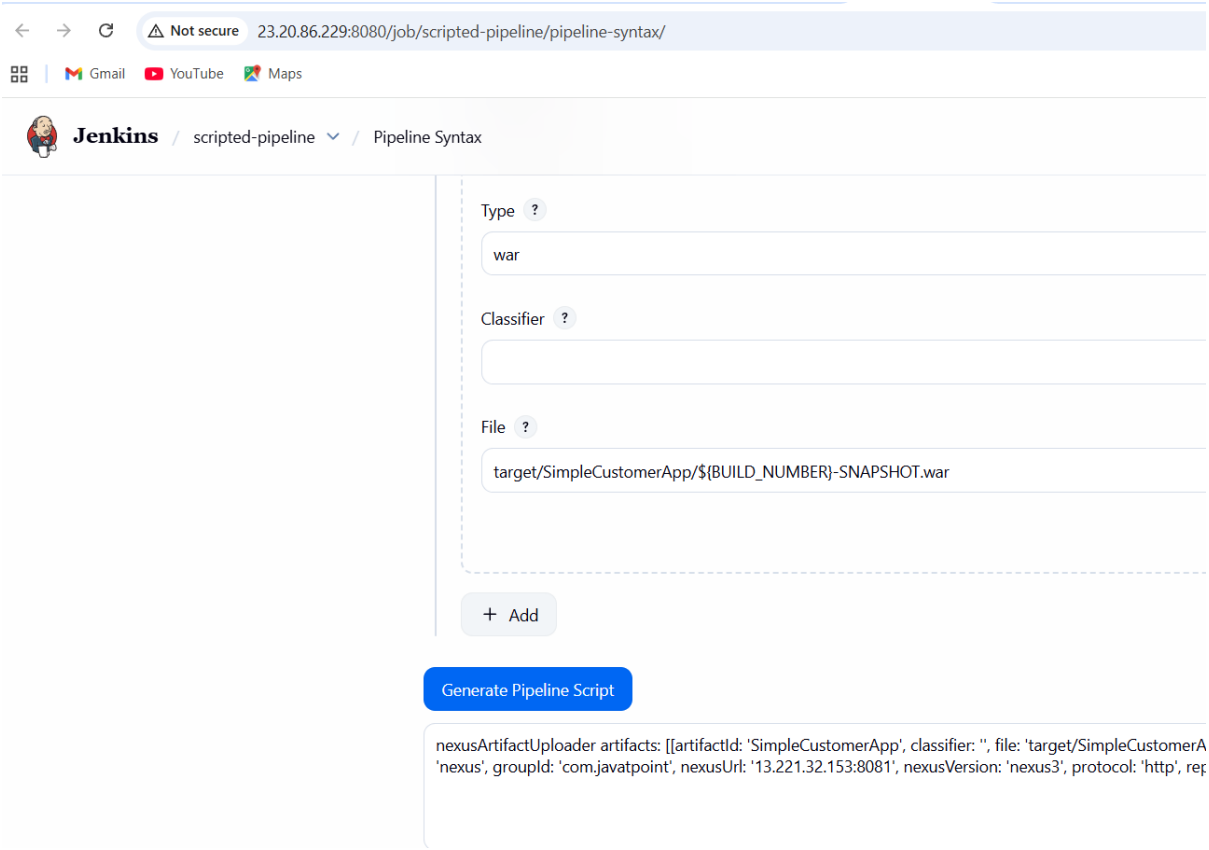
ArtifactId

SimpleCustomerApp

Type ?

war

Classifier ?



nexusArtifactUploader artifacts: [[artifactId: 'SimpleCustomerApp', classifier: '', file: 'target/SimpleCustomerApp/${BUILD_NUMBER}-SNAPSHOT.war', type: 'war']], credentialsId: 'nexus', groupId: 'com.javatpoint', nexusUrl: '13.221.32.153:8081', nexusVersion: 'nexus3', protocol: 'http', repository: 'scripted-pipeline', version: '${BUILD_NUMBER}-SNAPSHOT'

Jenkins / scripted-pipeline / Configuration

ıfigure

General

Triggers

Pipeline

Advanced

Script ?

```
20
21      stage("Upload Artifact to Nexus") {
22
23          script {
24              nexusArtifactUploader(
25                  artifacts: [[
26                      artifactId: 'SimpleCustomerApp',
27                      classifier: '',
28                      file: "target/SimpleCustomerApp-${BUILD_NUMBER}-SNAPSHOT.war",
29                      type: 'war'
30                  ]],
31                  credentialsId: 'nexus',
32                  groupId: 'com.javatpoint',
33                  nexusUrl: '13.221.32.153:8081',
34                  nexusVersion: 'nexus3',
```

Use Groovy Sandbox ?

Pipeline Syntax

Advanced

ıeline / Configuration

Script ?

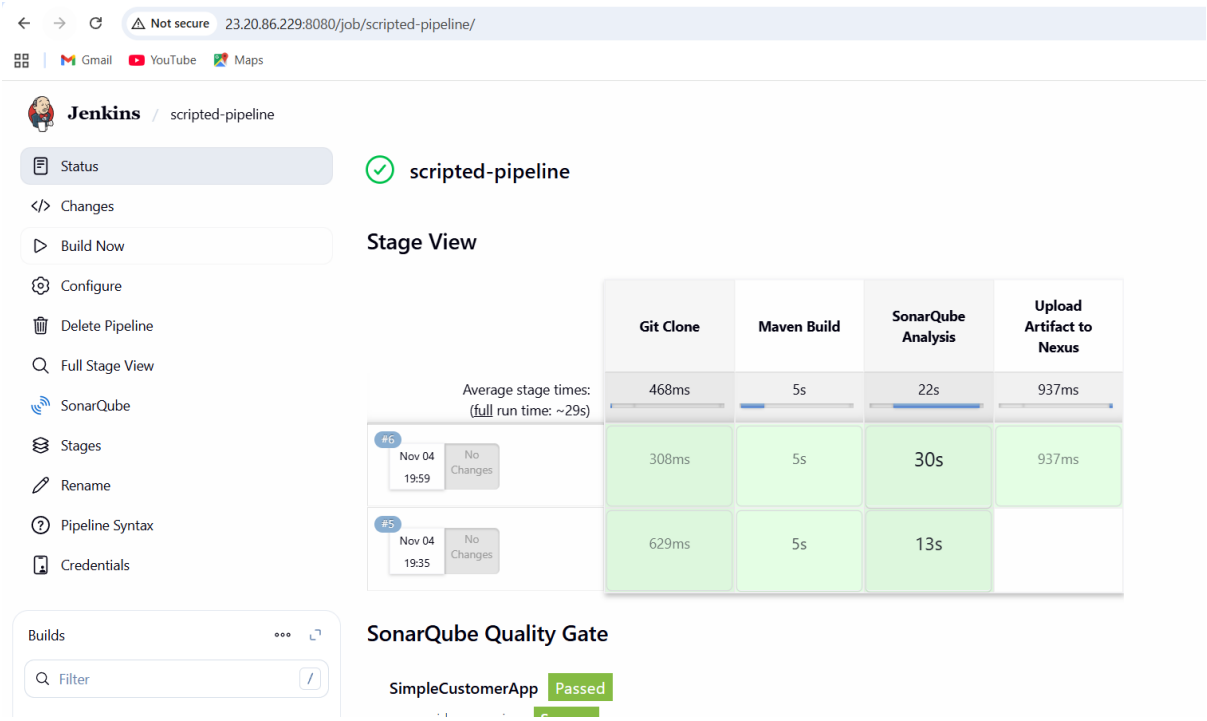
try sample P

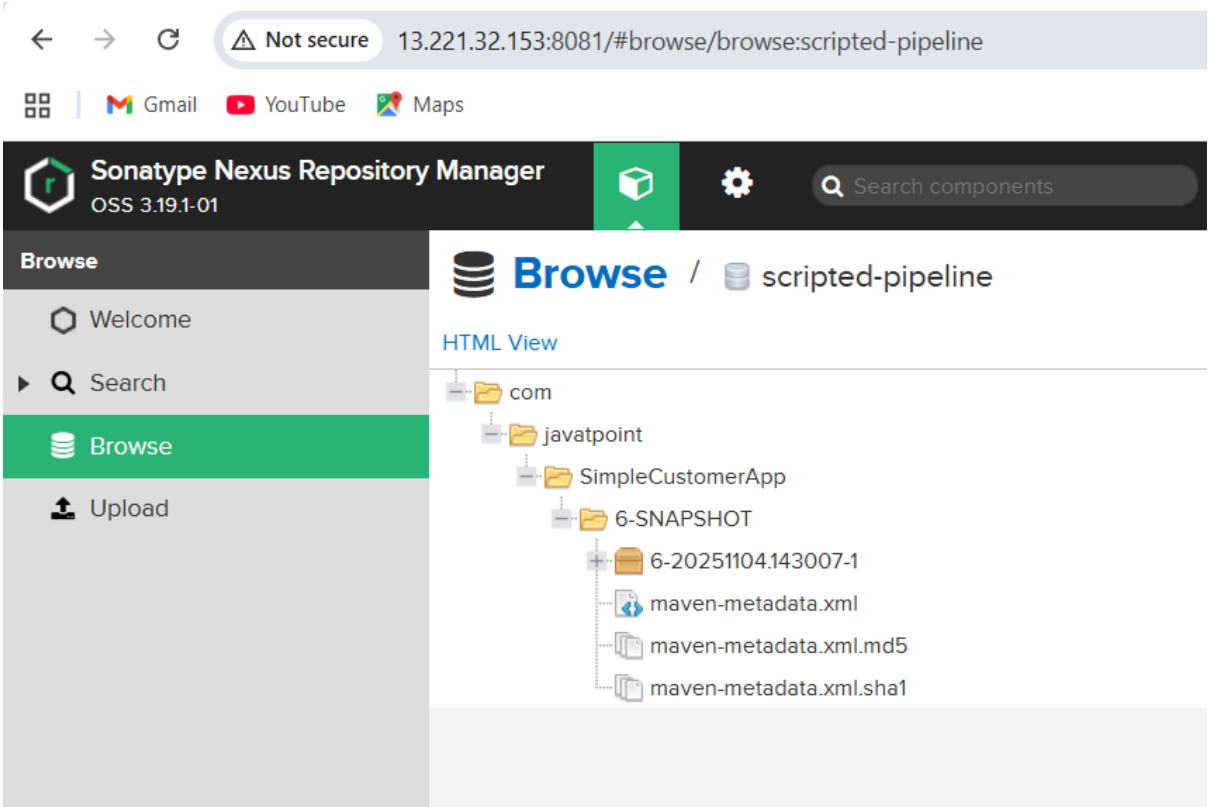
```
28                      file: "target/SimpleCustomerApp-${BUILD_NUMBER}-SNAPSHOT.war",
29                      type: 'war'
30                  ]],
31                  credentialsId: 'nexus',
32                  groupId: 'com.javatpoint',
33                  nexusUrl: '13.221.32.153:8081',
34                  nexusVersion: 'nexus3',
35                  protocol: 'http',
36                  repository: 'scripted-pipeline',
37                  version: "${BUILD_NUMBER}-SNAPSHOT"
38              )
39          }
40      }
41  }
42
```

Use Groovy Sandbox ?

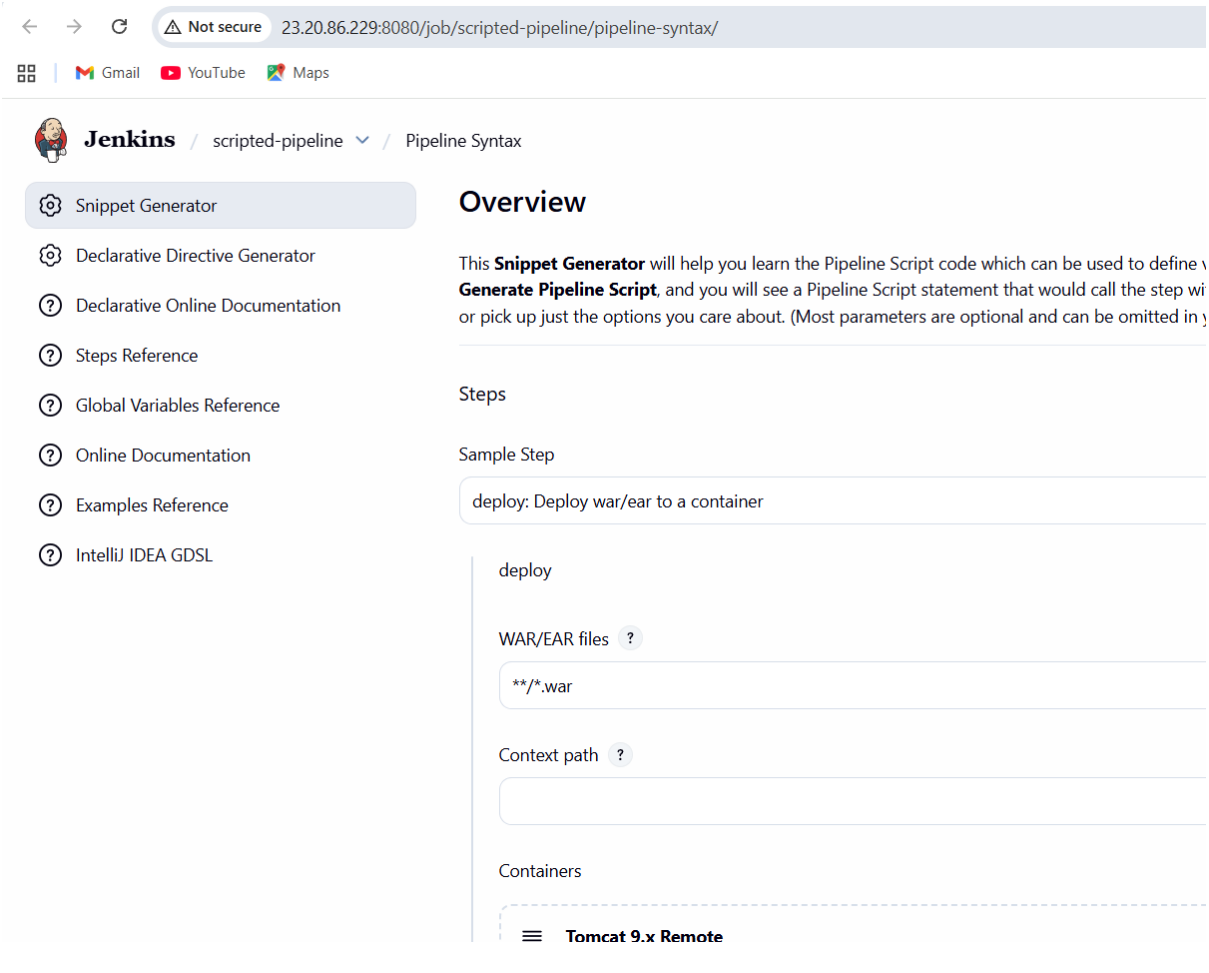
Pipeline Syntax

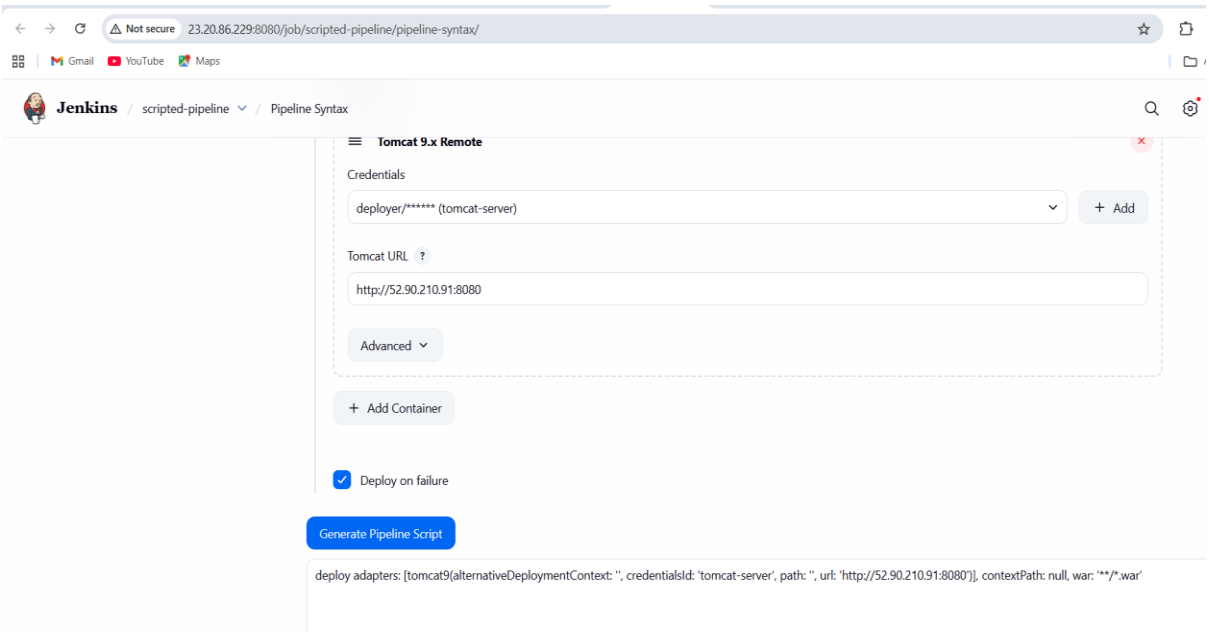
Click on build now.





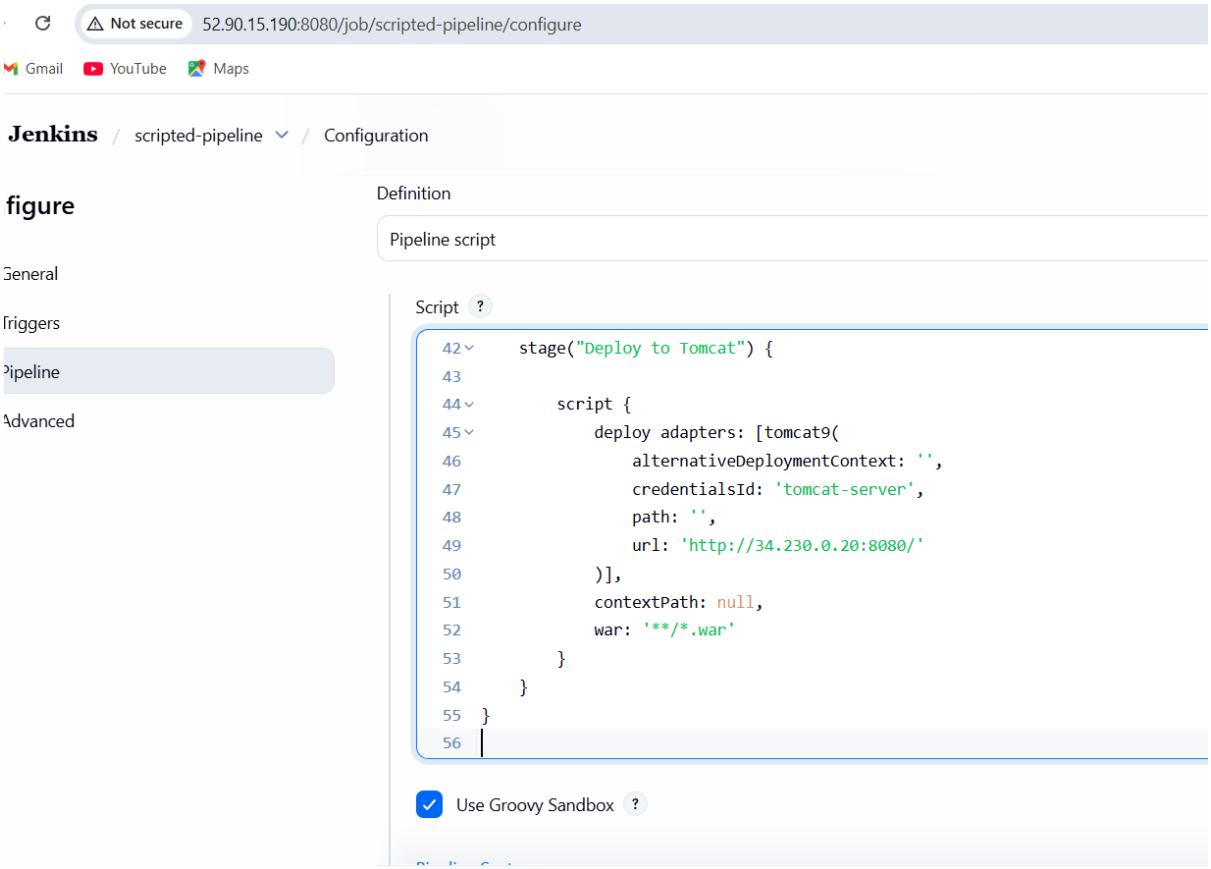
e. deploy to tomcat:



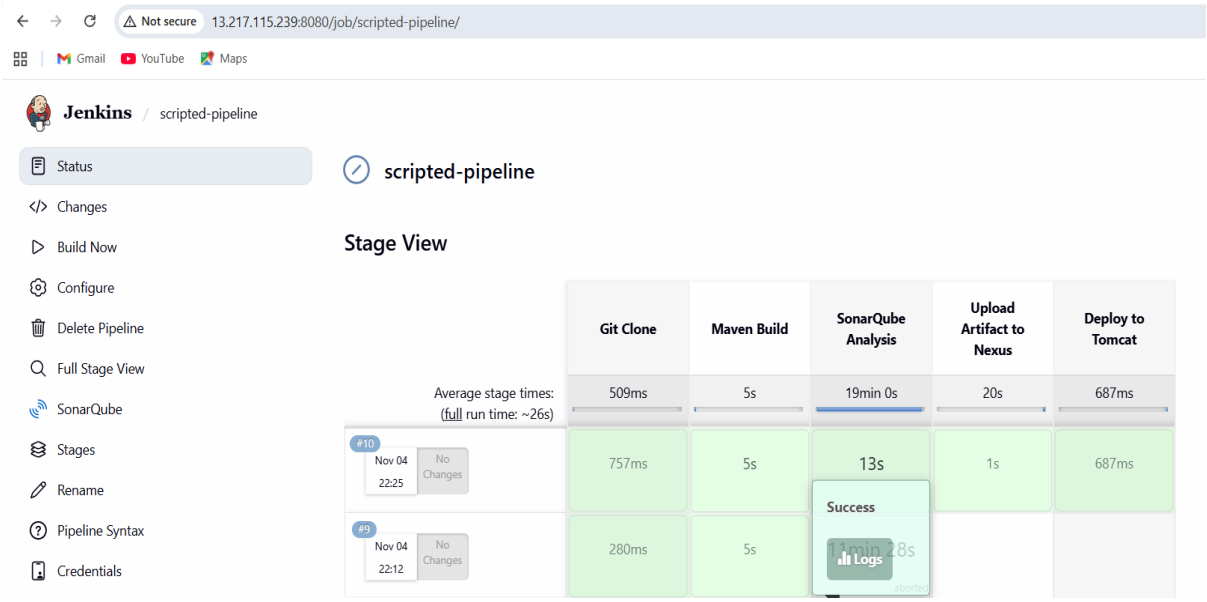


deploy adapters: [tomcat9(alternativeDeploymentContext: '', credentialsId: 'tomcat-server', path: '', url:

'http://52.90.210.91:8080')], contextPath: null, war: '**/*.war'



Click on build now.



f. slack integration:

Jenkins / scripted-pipeline / Pipeline Syntax

- Snippet Generator
- Declarative Directive Generator
- Declarative Online Documentation
- Steps Reference
- Global Variables Reference
- Online Documentation
- Examples Reference
- IntelliJ IDEA GDSL

Overview

This **Snippet Generator** will help you learn the Pipeline Script code which can be used to define various ste
Generate Pipeline Script, and you will see a Pipeline Script statement that would call the step with that cor
or pick up just the options you care about. (Most parameters are optional and can be omitted in your script

Steps

Sample Step

slackSend: Send Slack Message

slackSend ?

Channel ?

jenkins-integration

Message ?

scripted pipeline executed

Advanced Edited

Color ?

Not secure 13.217.115.239:8080/job/scripted-pipeline/pipeline-syntax/

Gmail YouTube Maps

Jenkins / scripted-pipeline / Pipeline Syntax

Credential ?

slack_integration

Custom slack app bot user ?

Notify Committers? ?

Icon Emoji ?

Username ?

shaik mohammad mujaheed-15

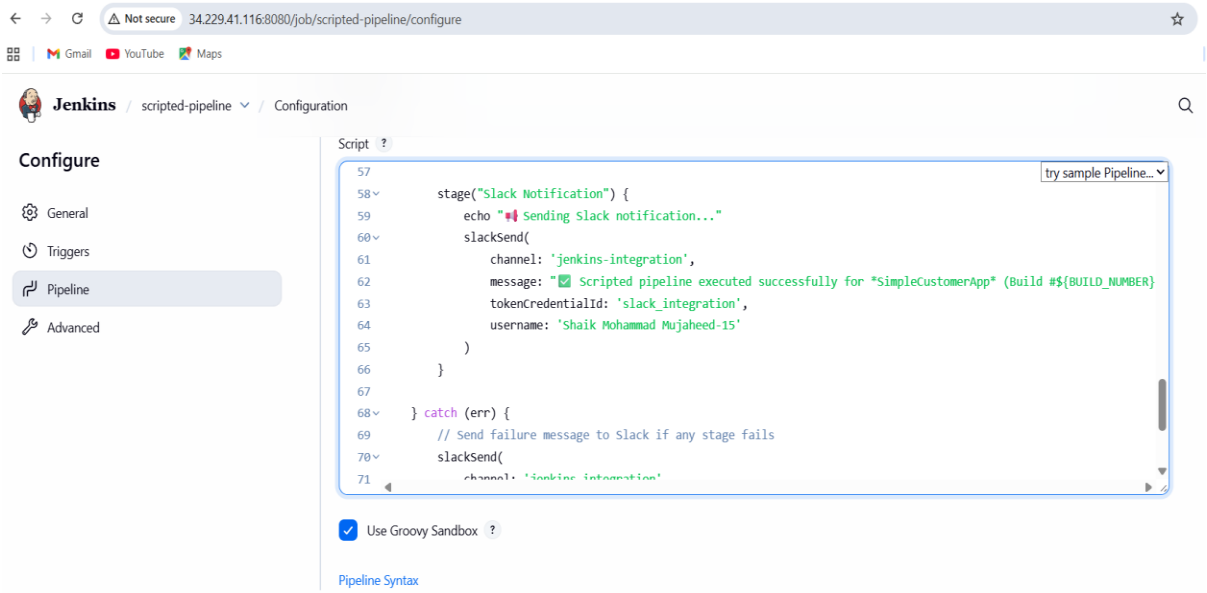
Override url ?

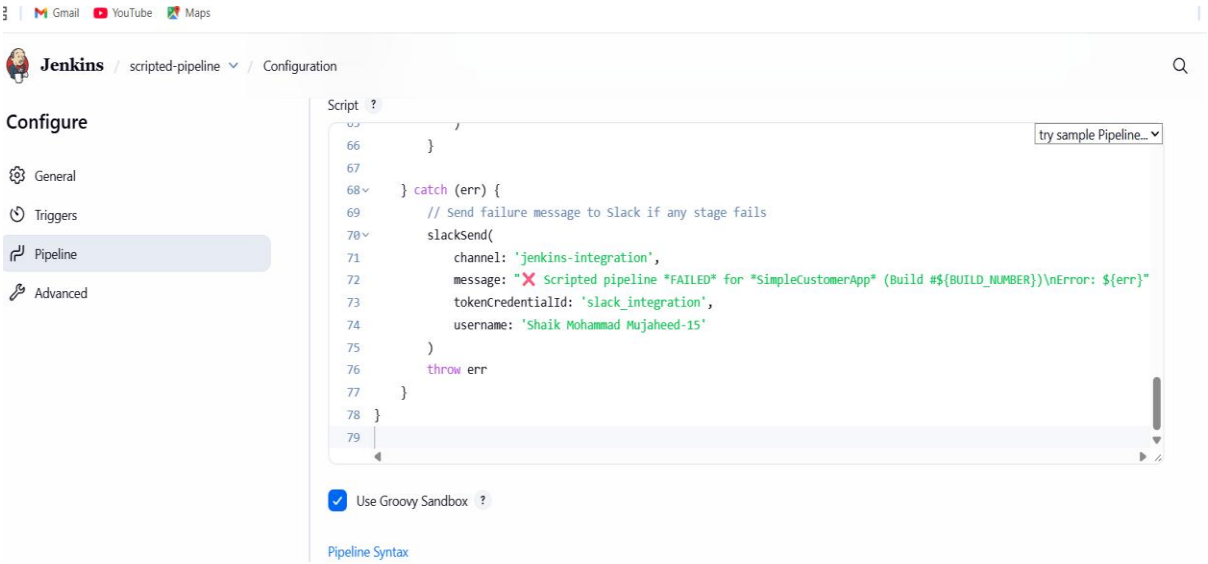
Fail On Error ?

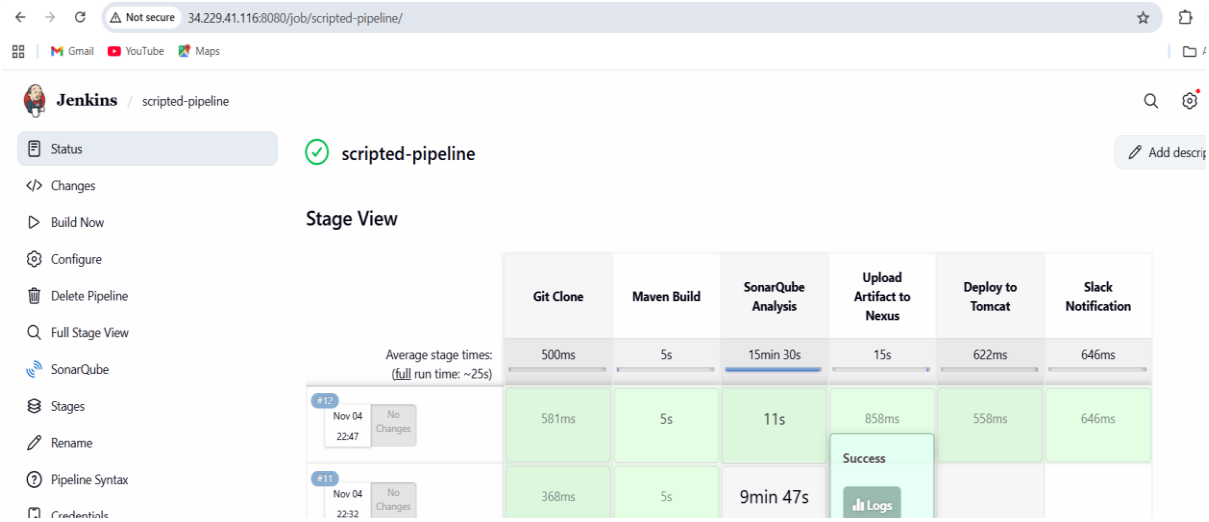
Generate Pipeline Script

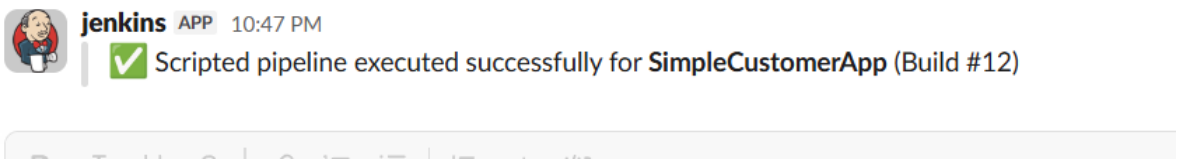
slackSend channel: 'jenkins-integration', message: 'scripted pipeline executed', tokenCredentialId: 'slack_integration', username: 'shaik mohammad mujahee

slackSend channel: 'jenkins-integration', message: 'scripted pipeline executed', tokenCredentialId: 'slack_integration', username: 'shaik mohammad mujaheed-15'









4. Write sample skeleton of pipelines.

Declarative pipeline:

```
pipeline {
    agent any

    tools {
        maven "MVN_HOME"
    }

    stages {

        stage('Checkout') {
```

```
        steps {
            git
'https://github.com/betawins/sabear_simplecutomerapp'
        }
    }

    stage('Build') {
        steps {
            withSonarQubeEnv("sonarqube") {
                sh "mvn clean install -f pom.xml"
            }
        }
    }

    stage('Upload to Nexus') {
        steps {
            script {
                nexusArtifactUploader artifacts: [[
                    artifactId: 'simple-customer-app',
                    classifier: '',
```

```
                file: 'target/SimpleCustomerApp-
${BUILD_NUMBER}-SNAPSHOT.war',
                type: 'war'
            ]],
            credentialsId: 'nexus',
            groupId: 'com.javatpoint',
            nexusUrl: '13.221.32.153:8081',
            nexusVersion: 'nexus3',
            protocol: 'http',
            repository: 'simple-customer-app',
            version: "${BUILD_NUMBER}-SNAPSHOT"
        }
    }
}

stage('Deploy to Tomcat') {
    steps {
        script {
            deploy adapters: [tomcat9(
                alternativeDeploymentContext: '',
                credentialsId: 'tomcat-server',
```

```
                path: '',
                url: 'http://52.90.210.91:8080'
            )],
            contextPath: null,
            war: '**/*.war'
        }
    }
}

    stage('Slack Notification') {
        steps {
            slackSend(
                channel: 'jenkins-integration',
                message: " ✅ Declarative pipeline executed successfully — Build #${BUILD_NUMBER}",
                tokenCredentialId: 'slack_integration',
                username: 'Shaik Mohammad Mujaheed-15'
            )
        }
    }
}
```

```
    post {
        failure {
            slackSend(
                channel: 'jenkins-integration',
                message: "❌ Build #${BUILD_NUMBER} failed.
Please check Jenkins logs.",
                tokenCredentialId: 'slack_integration',
                username: 'Shaik Mohammad Mujaheed-15'
            )
        }
    }
}
```

Scripted pipeline:

```
node {
    def mvnHome = tool "MVN_HOME"

    try {

        stage("Git Clone") {
```

```
        echo " 📥 Cloning source code from GitHub..."
        git branch: 'feature-1.1', url:
'https://github.com/betawins/sabear_simplecutomerapp'
    }

    stage("Maven Build") {
        echo " ⚙️ Building project using Maven..."
        sh "${mvnHome}/bin/mvn clean install -f pom.xml"
    }

    stage("SonarQube Analysis") {
        echo " 🔍 Running SonarQube Analysis..."
        withSonarQubeEnv('sonarqube') {
            sh "${mvnHome}/bin/mvn sonar:sonar -f pom.xml"
        }
    }

    stage("Upload Artifact to Nexus") {
        echo " ⬆️ Uploading artifact to Nexus Repository..."
        script {
            nexusArtifactUploader(
```

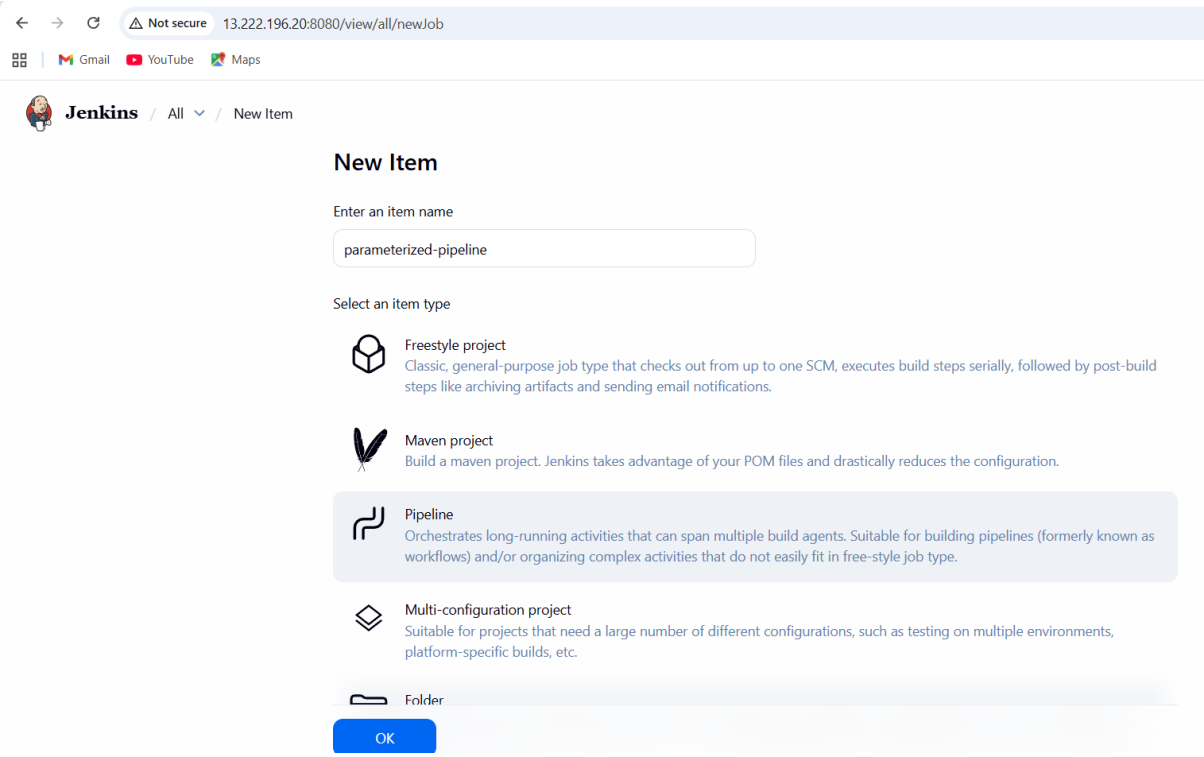
```
                artifacts: [[
                    artifactId: 'SimpleCustomerApp',
                    classifier: '',
                    file: "target/SimpleCustomerApp-
${BUILD_NUMBER}-SNAPSHOT.war",
                    type: 'war'
                ]],
                credentialsId: 'nexus',
                groupId: 'com.javatpoint',
                nexusUrl: '107.21.81.54:8081',
                nexusVersion: 'nexus3',
                protocol: 'http',
                repository: 'scripted-pipeline',
                version: "${BUILD_NUMBER}-SNAPSHOT"
            )
        }
    }

    stage("Deploy to Tomcat") {
        echo " 🚀 Deploying WAR to Tomcat Server..."
        script {
```

```
            deploy adapters: [tomcat9(
                alternativeDeploymentContext: '',
                credentialsId: 'tomcat-server',  // Jenkins
credentials for Tomcat
                path: '',                    // leave blank for ROOT
context
                url: 'http://34.230.0.20:8080'   // Tomcat server
URL
            )],
            contextPath: null,
            war: '**/*.war'
        }
    }

    stage("Slack Notification") {
        echo "📢 Sending Slack notification..."
        slackSend(
            channel: 'jenkins-integration',
            message: "✅ Scripted pipeline executed
successfully for *SimpleCustomerApp* (Build
#${BUILD_NUMBER})",
            tokenCredentialId: 'slack_integration',
```

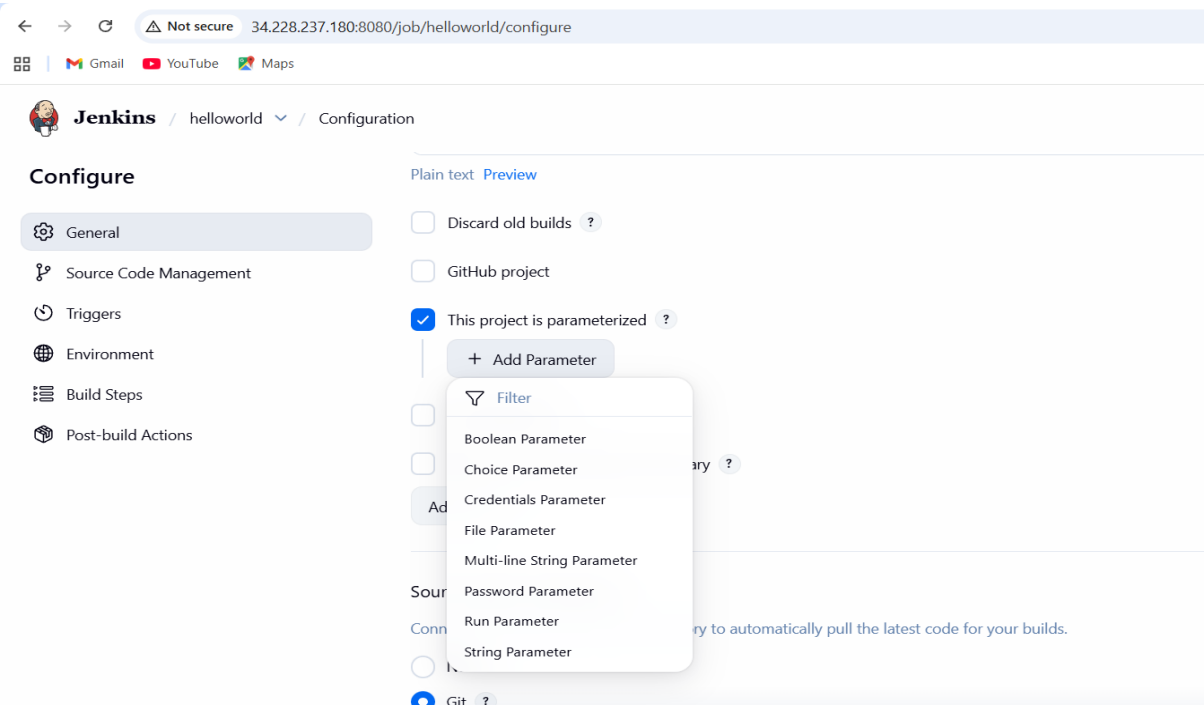
```
            username: 'Shaik Mohammad Mujaheed-15'
        )
    }

  } catch (err) {
    // Send failure message to Slack if any stage fails
    slackSend(
        channel: 'jenkins-integration',
        message: "❌ Scripted pipeline *FAILED* for *SimpleCustomerApp* (Build #${BUILD_NUMBER})\nError: ${err}",
        tokenCredentialId: 'slack_integration',
        username: 'Shaik Mohammad Mujaheed-15'
    )
    throw err
  }
}
```

5.Create a parameterized job in Jenkins. https://github.com/betawins/spring3-mvc-maven-xml-hello-world-1.git

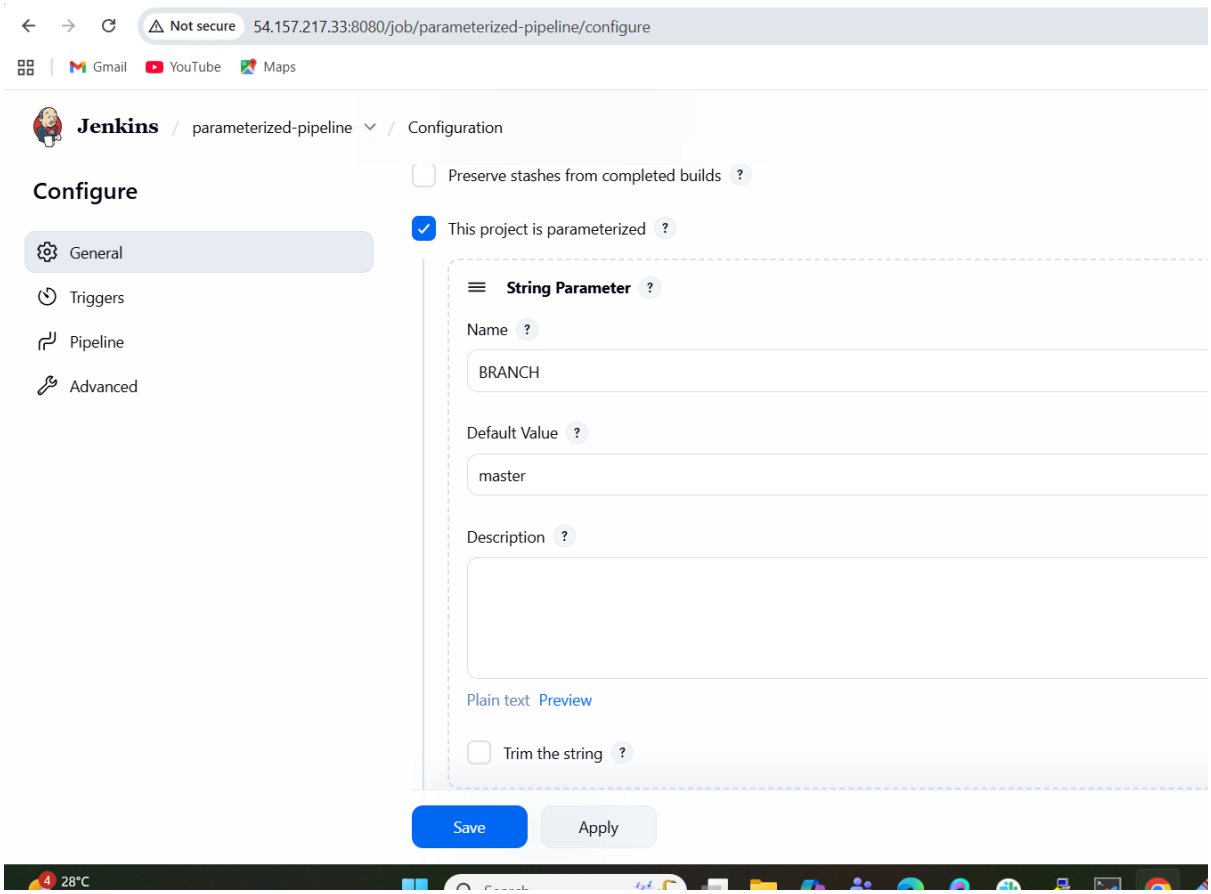
Create a new job with the name parameterized-pipeline and select pipeline type.



Select In the job configuration in general section select this project is parameterized and add relevant parameters.



- Select string parameter give name as branch



a. Git cloning:

And write the code for cloning.

Not secure 54.157.217.33:8080/job/parameterized-pipeline/configure

Gmail YouTube Maps

Jenkins / parameterized-pipeline / Configuration

Configure

- General
- Triggers
- Pipeline
- Advanced

Definition

Pipeline script

Script ?

```
pipeline {
    agent any
 stages {

        stage('Checkout') {
            steps {
                git 'https://github.com/betawins/spring3-mvc-maven-xml-hello-world-1.git'
            }
        }
  }
 }

```

Use Groovy Sandbox ?

Save Apply

Not secure 54.157.217.33:8080/job/parameterized-pipeline/

Gmail YouTube Maps

Jenkins / parameterized-pipeline

- Status
- Changes
- Build with Parameters
- Configure
- Delete Pipeline
- Full Stage View
- Stages
- Rename
- Pipeline Syntax
- Credentials

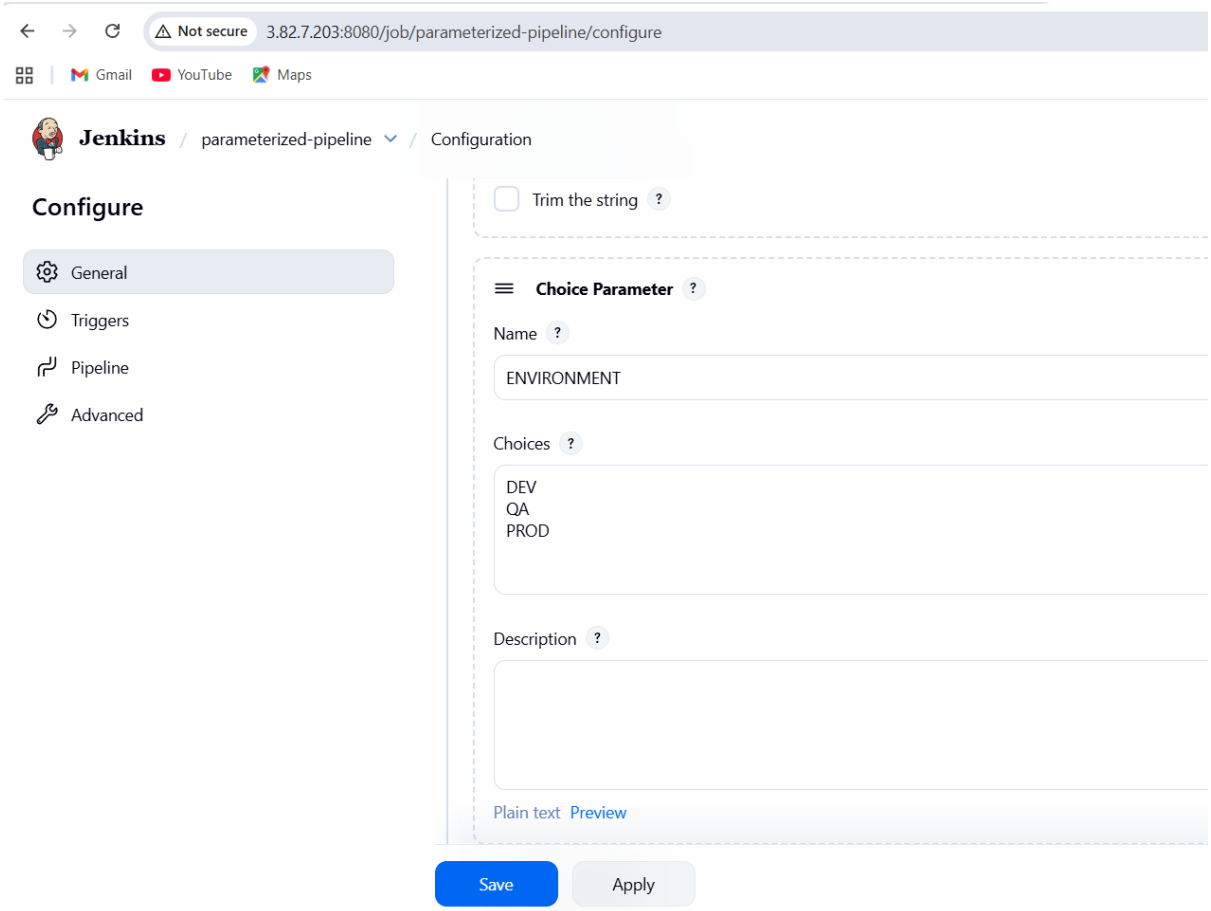
Builds

parameterized-pipeline

Stage View

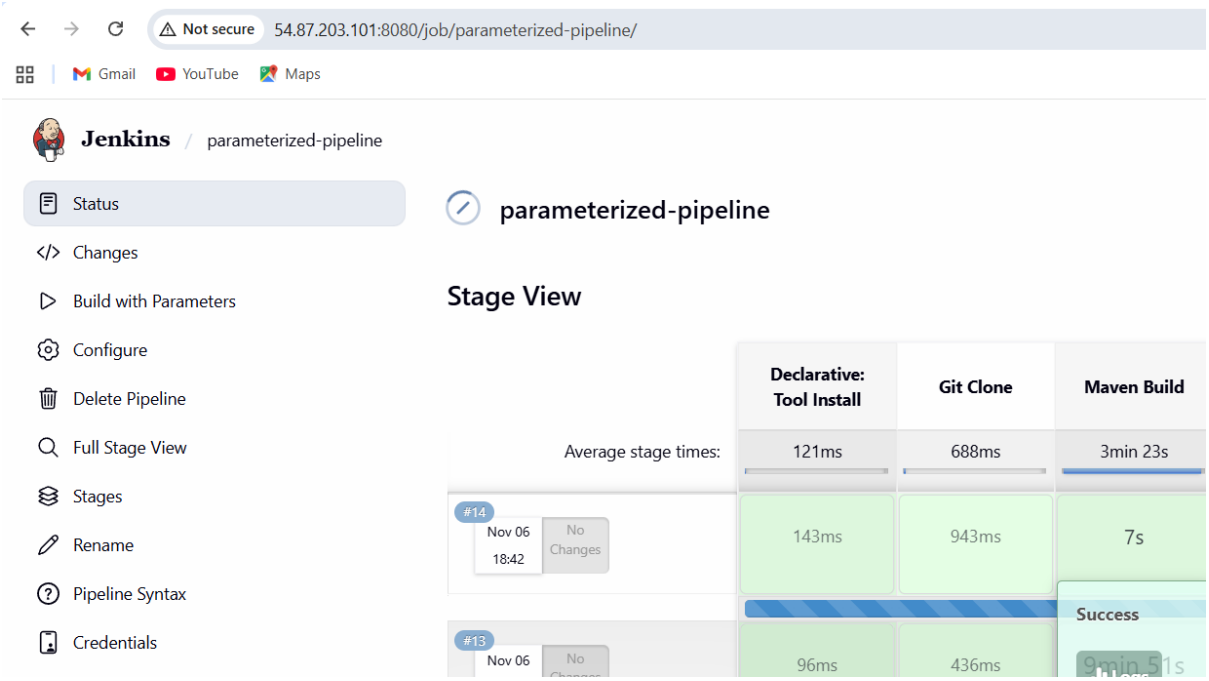
	Checkout
Average stage times: (full run time: ~3s)	883ms
#7 Nov 06 12:55 No Changes	628ms
#6 Nov 06 12:50 No Changes	1s

Permalinks

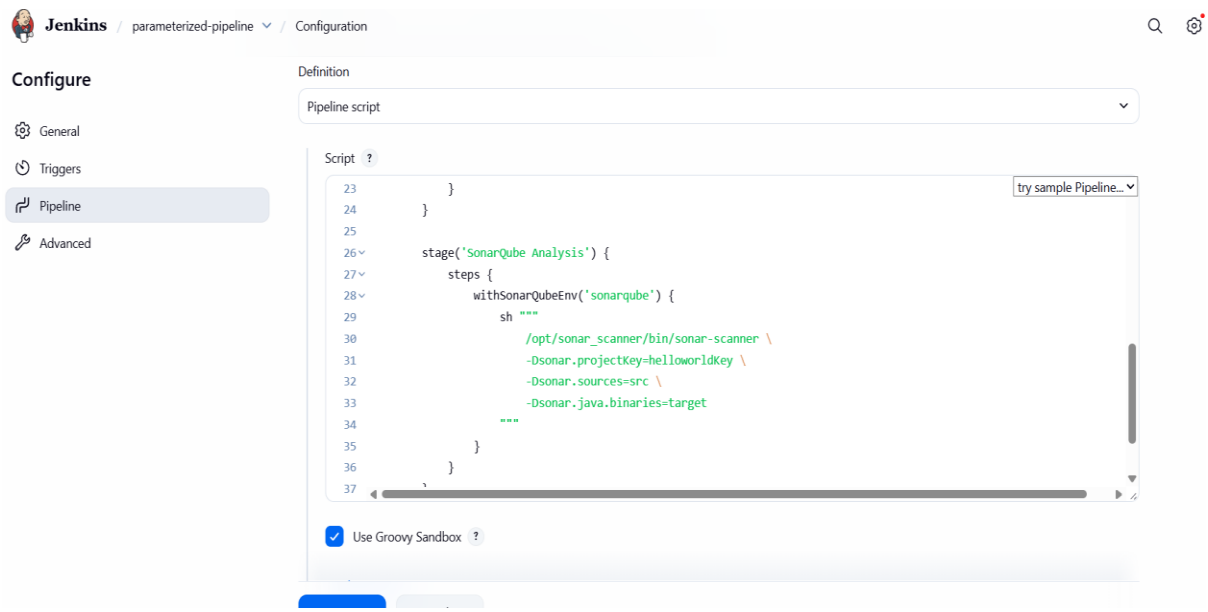

Not secure
3.82.7.203:8080/job/parameterized-pipeline/configure
Gmail
YouTube
Maps
Jenkins / parameterized-pipeline / Configuration
Configure
General
Triggers
Pipeline
Advanced
Trim the string
Choice Parameter
Name
ENVIRONMENT
Choices
DEV
QA
PROD
Description
Plain text Preview
Save
Apply

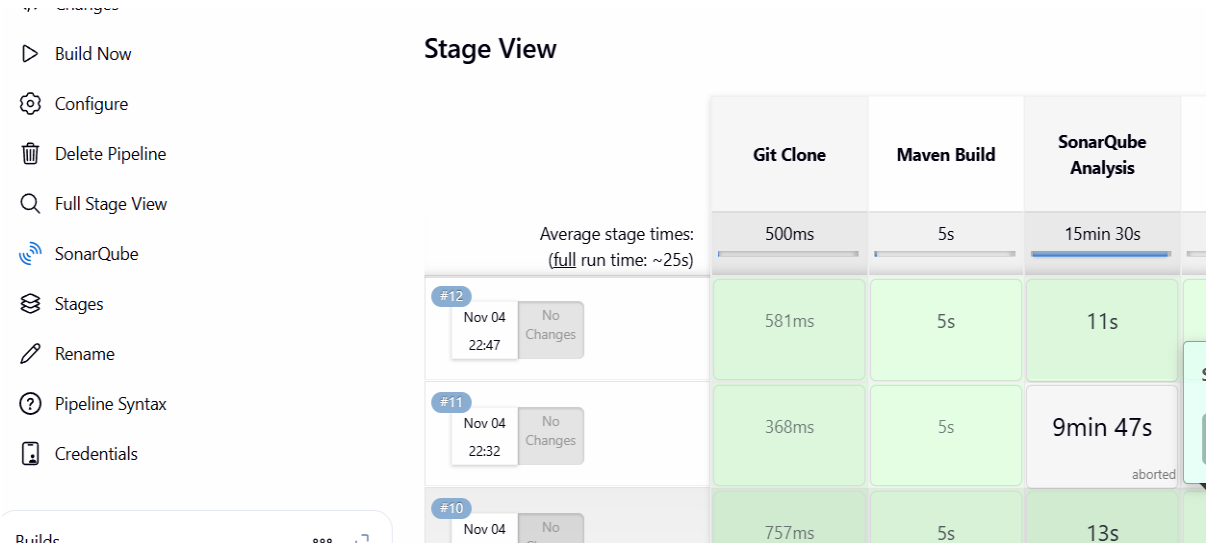
b. maven:

ns / parameterized-pipeline / Configuration
Definition
Pipeline script
Script
parameters {
string(name: 'BRANCH', defaultValue: 'master', description: 'Git branch to build')
choice(name: 'ENVIRONMENT', choices: ['DEV', 'QA', 'PROD'], description: 'Environment for deployment')
}
stages {
stage('Git Clone') {
steps {
git branch: "${BRANCH}", url: "https://github.com/Nagasai31-rgb/spring3-mvc-maven-xml-hello-world-1.
}
}
stage('Maven Build') {
steps {
sh "mvn clean package"
}
}
Use Groovy Sandbox

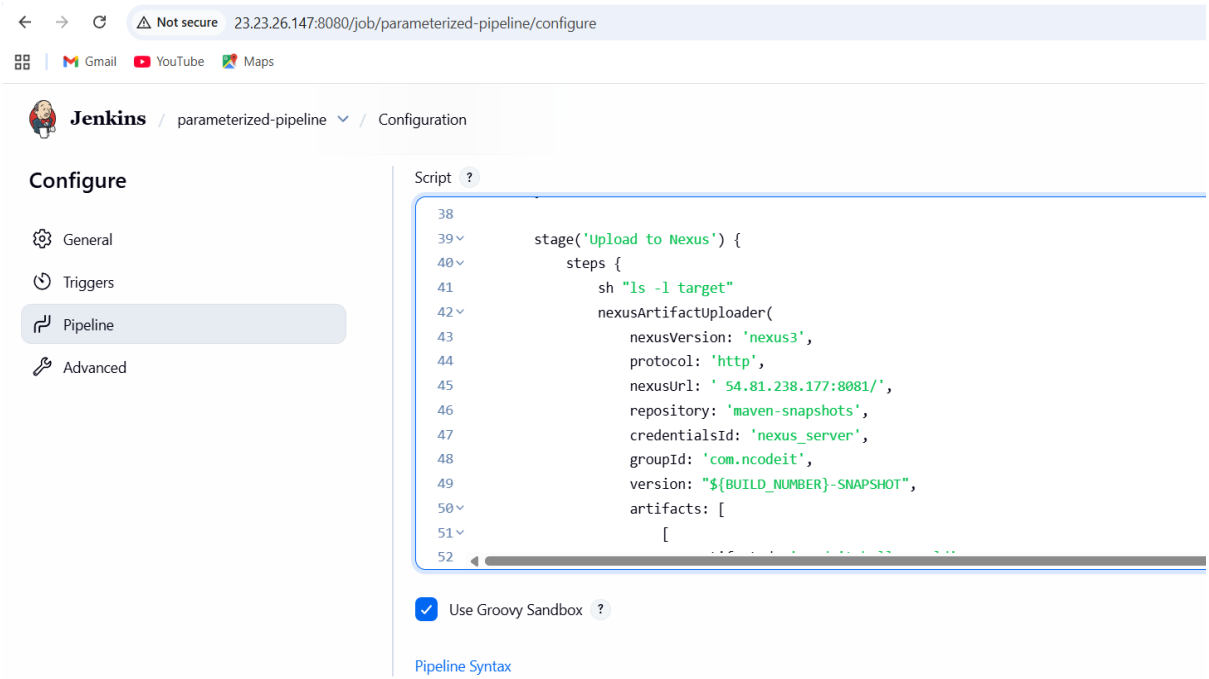

Not secure
54.87.203.101:8080/job/parameterized-pipeline/
Gmail
YouTube
Maps
Jenkins
parameterized-pipeline
Status
Changes
Build with Parameters
Configure
Delete Pipeline
Full Stage View
Stages
Rename
Pipeline Syntax
Credentials
parameterized-pipeline
Stage View
Declarative: Tool Install
Git Clone
Maven Build
Average stage times:
121ms
688ms
3min 23s
#14
Nov 06
18:42
No Changes
143ms
943ms
7s
Success
#13
Nov 06
96ms
436ms

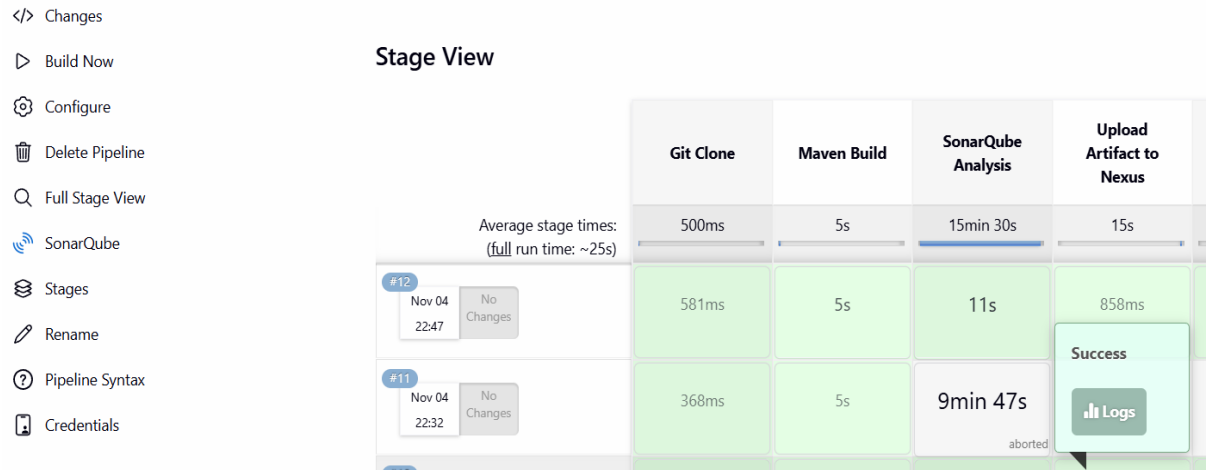
c. sonarqube:


Jenkins
parameterized-pipeline
Configuration
Configure
General
Triggers
Pipeline
Advanced
Definition
Pipeline script
Script
try sample Pipeline...
}
}
stage('SonarQube Analysis') {
steps {
withSonarQubeEnv('sonarqube') {
sh """
/opt/sonar_scanner/bin/sonar-scanner \
-Dsonar.projectKey=helloworldKey \
-Dsonar.sources=src \
-Dsonar.java.binaries=target
"""
}
}
Use Groovy Sandbox

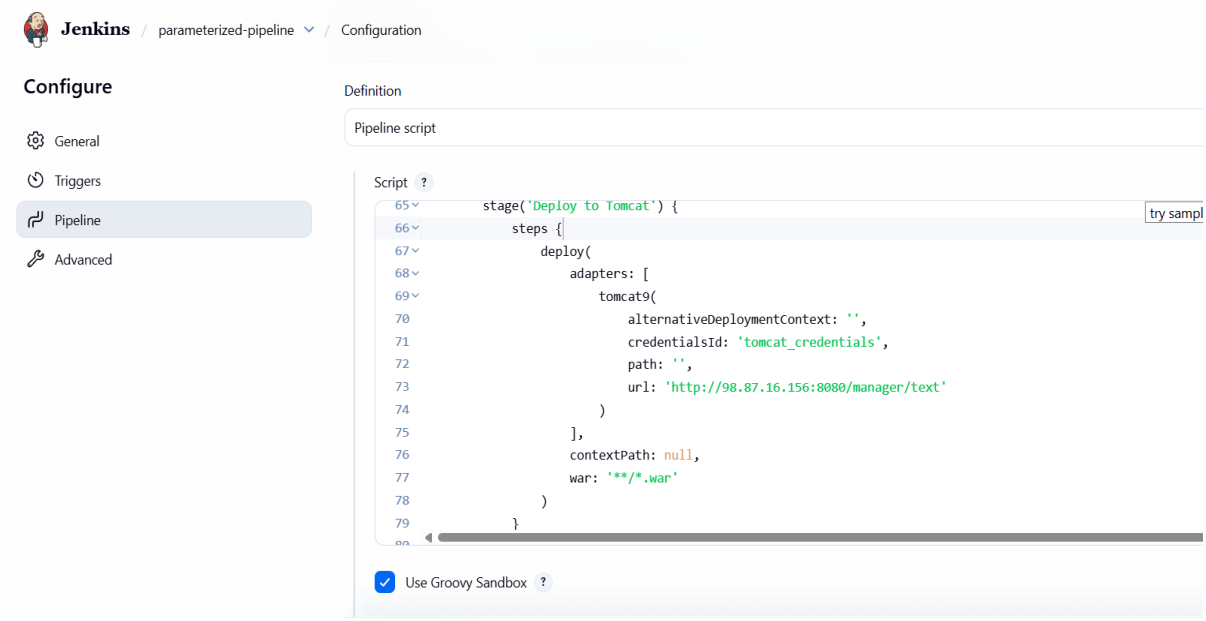


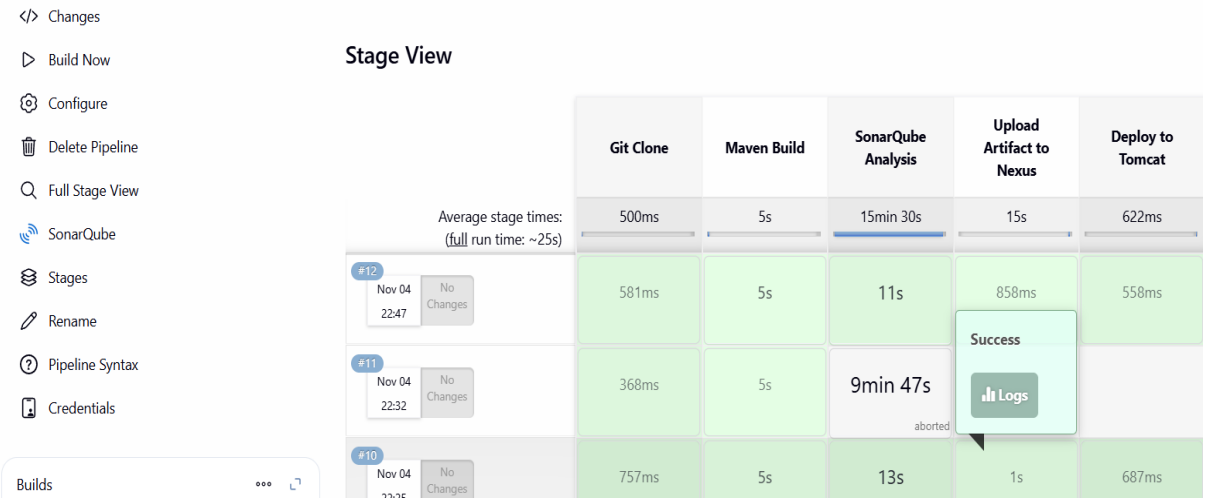
d. nexus artifactory:





e.tomcat deployment:






```
pipeline {
  agent any
  tools {
    maven 'MVN_HOME'
  }
```

```
    parameters {
        string(name: 'BRANCH', defaultValue: 'master',
description: 'Git branch to build')
        choice(name: 'ENVIRONMENT', choices: ['DEV', 'QA',
'PROD'], description: 'Environment for deployment')
    }

    stages {

        stage('Git Clone') {
            steps {
                git branch: "${BRANCH}", url:
"https://github.com/Nagasai31-rgb/spring3-mvc-maven-xml-
hello-world-1.git"
            }
        }

        stage('Maven Build') {
            steps {
                sh "mvn clean package"
            }
        }
```

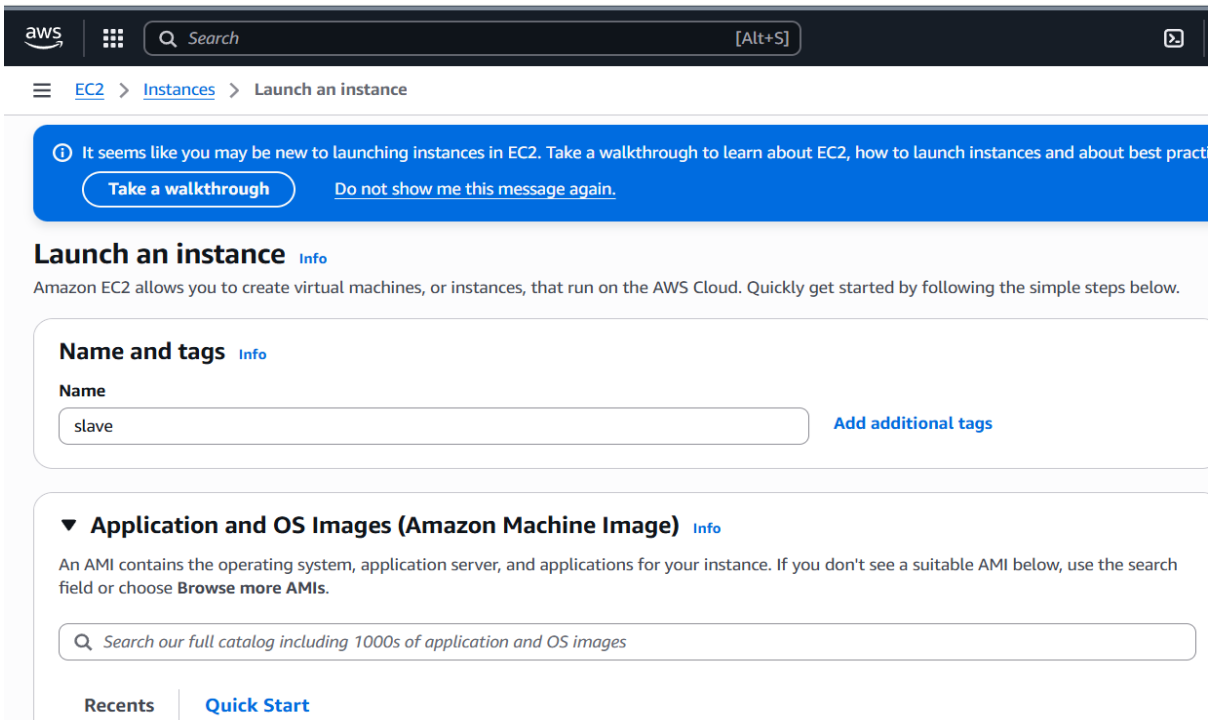
```
stage('SonarQube Analysis') {
    steps {
        withSonarQubeEnv('sonarqube') {
            sh """
                /opt/sonar_scanner/bin/sonar-scanner \
                -Dsonar.projectKey=helloworldKey \
                -Dsonar.sources=src \
                -Dsonar.java.binaries=target
            """
        }
    }
}

stage('Upload to Nexus') {
    steps {
        sh "ls -l target"
        nexusArtifactUploader(
            nexusVersion: 'nexus3',
            protocol: 'http',
            nexusUrl: ' 54.81.238.177:8081/',
```

```
            repository: 'maven-snapshots',
            credentialsId: 'nexus_server',
            groupId: 'com.ncodeit',
            version: "${BUILD_NUMBER}-SNAPSHOT",
            artifacts: [
                [
                    artifactId: 'ncodeit-hello-world',
                    classifier: '',
                    type: 'war',
                    file: "target/ncodeit-hello-world-3.0.war"
                ]
            ]
        )
    }
}

stage('Deploy to Tomcat') {
    steps {
        deploy(
            adapters: [
                tomcat9(
```

```
                alternativeDeploymentContext: '',
                credentialsId: 'tomcat_credentials',
                path: '',
                url: 'http:// 98.87.16.156:8080/'
            )
        ],
        contextPath: null,
        war: '**/*.war'
    )
  }
 }
}
```

6. Setup one slave machine for Jenkins.

Create a instance with the name slave.



Steps to do in slave machine:

- Login to slave machine
- Switch to root user
- Install java
- Create ssh-keygen
- Cat id_rsa.pub > authorized_keys
- Chmod 700 authorized_key

```
[root@ip-172-31-13-246 ~]# yum install java-21 -y
Last metadata expiration check: 0:03:07 ago on Wed Nov  5 10:10:51 2025.
Dependencies resolved.
================================================================================
 Package                                    Architecture            Version
================================================================================
Installing:
 java-21-amazon-corretto                    x86_64                  1:21.0.9+10-1.amzn2
Installing dependencies:
 alsa-lib                                   x86_64                  1.2.7.2-1.amzn2023.
 cairo                                      x86_64                  1.18.0-4.amzn2023.0
 dejavu-sans-fonts                          noarch                  2.37-16.amzn2023.0.
 dejavu-sans-mono-fonts                     noarch                  2.37-16.amzn2023.0.
 dejavu-serif-fonts                         noarch                  2.37-16.amzn2023.0.
 fontconfig                                 x86_64                  2.13.94-2.amzn2023.
 fonts-filesystem                           noarch                  1:2.0.5-12.amzn2023
 freetype                                   x86_64                  2.13.2-5.amzn2023.0
 giflib                                     x86_64                  5.2.1-9.amzn2023.0.
 google-noto-fonts-common                   noarch                  20240401-1.amzn2023
 google-noto-sans-vf-fonts                  noarch                  20240401-1.amzn2023
 graphite2                                  x86_64                  1.3.14-7.amzn2023.0
```

Create a key-gen by using ssh-keygen.

```
[root@ip-172-31-13-246 ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:vy2iJTAtQx+ozJVfWdNn3ox8n5hy0rZlIzcGcl6ORDg root@ip-172-31-13-246.ec2.internal
The key's randomart image is:
+---[RSA 3072]----+
|          o..    |
|     o   o E..o  |
|    = . o   o= + |
| o + + o   . ++.+|
|  + = + S   * B.o|
|     =   . o X X.|
|      . . . = B o|
|       o. .o .   |
|      .. ....    |
+----[SHA256]-----+
[root@ip-172-31-13-246 ~]#
```

- Change the directory cd .ssh
- Cat id_rsa.pub > authorized_keys
- Chmod 700 authorized_key

```
[root@ip-172-31-13-246 .ssh]# ll
total 12
-rw-------. 1 root root  551 Nov  5 10:08 authorized_keys
-rw-------. 1 root root 2622 Nov  5 10:17 id_rsa
-rw-r--r--. 1 root root  588 Nov  5 10:17 id_rsa.pub
[root@ip-172-31-13-246 .ssh]# cat id_rsa.pub > authorized_keys
[root@ip-172-31-13-246 .ssh]# ll
total 12
-rw-------. 1 root root  588 Nov  5 10:20 authorized_keys
-rw-------. 1 root root 2622 Nov  5 10:17 id_rsa
-rw-r--r--. 1 root root  588 Nov  5 10:17 id_rsa.pub
[root@ip-172-31-13-246 .ssh]# chmod 700 authorized_keys
[root@ip-172-31-13-246 .ssh]# ll
total 12
-rwx------. 1 root root  588 Nov  5 10:20 authorized_keys
-rw-------. 1 root root 2622 Nov  5 10:17 id_rsa
-rw-r--r--. 1 root root  588 Nov  5 10:17 id_rsa.pub
[root@ip-172-31-13-246 .ssh]#
```

Go to master account and follow this steps:

- Switch to root user
- mkdir -p/var/lib/Jenkins/.ssh

- ssh-keyscan -H ip address of slave >> /var/lib/Jenkins/.ssh/known_hosts
- chown Jenkins:jenkins known_hosts
- chmod 700 known_hosts

Login into master account :

Create a directory by using

- Go to cd /var/lib/Jenkins
- Mkdir .ssh change to that ssh directory

```
[root@ip-172-31-50-64 ~]# cd /var/lib/jenkins
[root@ip-172-31-50-64 jenkins]# mkdir .ssh
[root@ip-172-31-50-64 jenkins]# cd .ssh
[root@ip-172-31-50-64 .ssh]# ll
total 0
[root@ip-172-31-50-64 .ssh]#
```

Execute ssh-keyscan -H ip of slave >> /var/lib/jenkins/.ssh/known_hosts

```
[root@ip-172-31-50-64 .ssh]# ssh-keyscan -H 54.224.78.244 >> /var/lib/jenkins/.ssh/known_hosts
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
[root@ip-172-31-50-64 .ssh]#
```

Execute command to change ownership:

- Chown Jenkins:jenkins known_hosts

```
[root@ip-172-31-50-64 .ssh]# ssh-keyscan -H 54.224.78.244 >> /var/lib/jenkins/.ssh/known_hosts
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
# 54.224.78.244:22 SSH-2.0-OpenSSH_8.7
[root@ip-172-31-50-64 .ssh]# ls
known_hosts
[root@ip-172-31-50-64 .ssh]# chown jenkins:jenkins known_hosts
[root@ip-172-31-50-64 .ssh]#
```

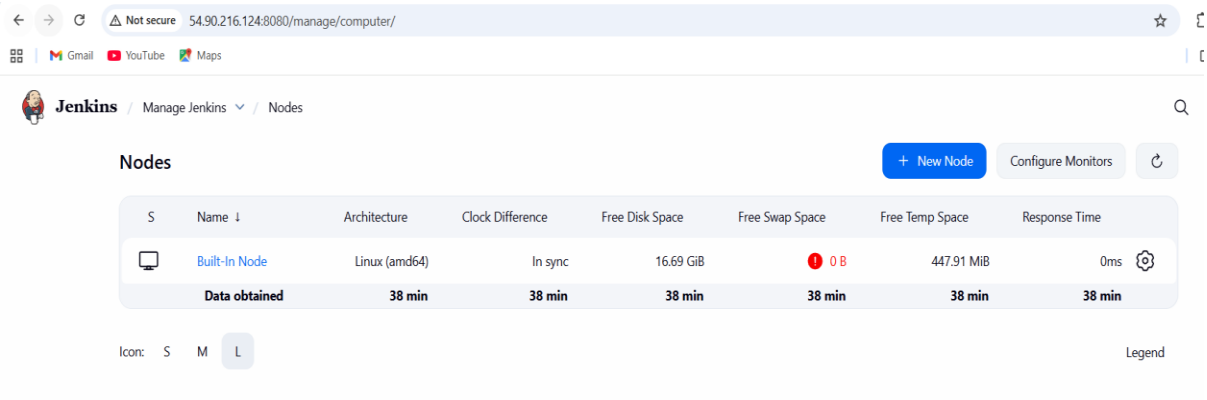
Change permissions

```
[root@ip-172-31-50-64 .ssh]# ll
total 4
-rw-r--r--. 1 jenkins jenkins 364 Nov  5 10:30 known_hosts
[root@ip-172-31-50-64 .ssh]# chmod 700 known_hosts
[root@ip-172-31-50-64 .ssh]# ll
total 4
-rwx------. 1 jenkins jenkins 364 Nov  5 10:30 known_hosts
[root@ip-172-31-50-64 .ssh]#
```

Steps to do in jenkins gui:

- Open manage Jenkins
- Under launch method select “launch via ssh”
- Host public ip of slave
- Credentials:select user with private key
- Enter ec2-user as name and paste the pem key
- Host key verification strategy to be selected
- save

Go to Jenkins gui and click on managed Jenkins and click on node create node.



Not secure 54.90.216.124:8080/manage/computer/createItem

Gmail YouTube Maps

Jenkins / Manage Jenkins / Nodes

Description ?

slave-01

Plain text Preview

Number of executors ?

2

Remote root directory ?

root/slave/workspace

Are you sure you want to use a relative path for the FS root? Note that relative paths require that you can assure that the selected launcher provides a consi
absolute path is highly recommended.

Labels ?

java

Usage ?

Save

kins / Nodes / slave / Configure

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 ?

slave-01-key

Description ?

Jenkins / Nodes / slave / Configure

Jenkins Credentials Provider: Jenkins

Treat username as secret ?

Private Key

Enter directly

Key

Enter New Secret Below


```
XOdi1JFgGn+6BLjtYi2WH2uwObq6hqzifH5yhOvJ1zROeZLXDwO56UREnhzpfklw
LFwIQGt3PPIUxaL7aYzxFNr0sIciL0YEAeUPDQkCgYEAi5S4c5RMPdN3V7axqzIR
A+w4jfsvWZiqtqLR2pZX22kk31A7flKGDpFfgfUbFoO7ge1/OQMEyOxqh0BfX8l6
gGq8D/XioQX29XabVWuoS5YpShEhyec+mQ407Y6mog7KMu1Yy8lAV6rOAdM3UpG1
ks/qv/Fg6+QpuWMYsj34HjUCgYBUPN6xPK7dZ8LCYxS1IBMBu/eiZG9XPwJa0xCb
srOhVj3VtZGi7GTCOF7ghriRF0+mpR7+BUb2ZDuB3NUBjN+OM2keErXCK8swnOEk
/1Ez8LY8tbwIg+ElTVmZaDt5ixpeS3AAiFNCG36YQpQC/rhe/7tQLkluuTkDNS0b
NEaEKQKBgQCWJC+rNnrZRwPyCFAB5AhamNoCFLbGl32cW5vwLmqwsTGJnG5uu6Ta
6d9j4wsP3gPjEi8l/O8FLGpWODMlxqeV+xhLAqpSZW0PNWkd9CoInwqhFn+T65AL
L7NsTswJUmSWY5Y2L2g2RVpC6pOVbQy8znbDmx3GQ4MjwuyrgkQWiw==
-----END RSA PRIVATE KEY-----
```


Passphrase

Not secure 35.172.115.185:8080/manage/computer/

Gmail YouTube Maps All Bookmarks

Jenkins / Manage Jenkins / Nodes

Nodes

+ New Node | Configure Monitors

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Linux (amd64)	In sync	5.63 GiB	! 0 B	447.91 MiB	0ms
	slave-01		N/A	N/A	N/A	N/A	N/A
	Data obtained	**9 min 12 sec**	**9 min 12 sec**	**9 min 12 sec**	**9 min 12 sec**	**9 min 12 sec**	**9 min 12 sec**

Icon: S M L

Legend