

# Simple Java Maven App

1. Install java in Jenkins Master server from the link below :

wget <https://corretto.aws/downloads/latest/amazon-corretto-8-aarch64-linux-jdk.tar.gz>

2 . Under Jenkins Dashboard, Click on Manage Jenkins and then Click on Global Tool Configuration and update the JDK path as given below



## Global Tool Configuration

### Maven Configuration

Default settings provider

Use default maven settings

Default global settings provider

Use default maven global settings

### JDK

JDK installations

Add JDK

JDK

Name

JDK\_1.8

JAVA\_HOME

/usr/lib/jvm/java-1.8.0-amazon-corretto

☐ Install automatically

3. Install Maven in Jenkins Master by following the below steps:

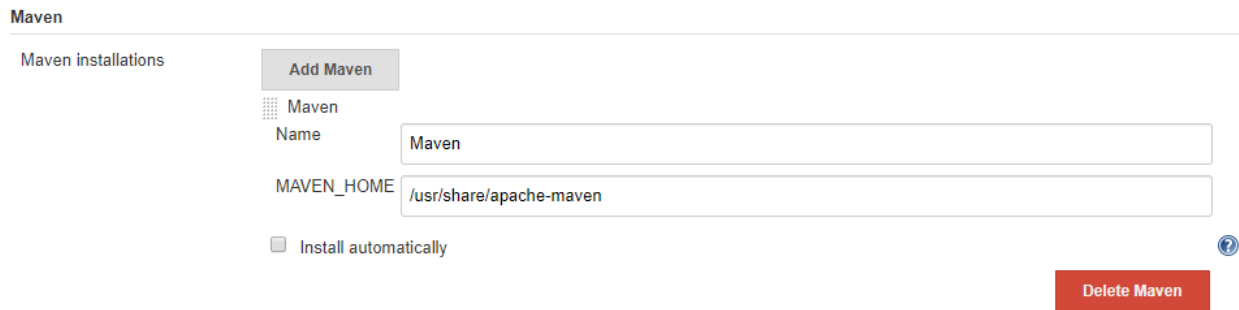
```
sudo wget http://repos.fedorapeople.org/repos/dchen/apache-maven/epel-apache-maven.repo -O /etc/yum.repos.d/epel-apache-maven.repo
```

```
sudo sed -i s/\$releasever/6/g /etc/yum.repos.d/epel-apache-maven.repo
```

```
sudo yum install -y apache-maven
```

Check version using the command -- > mvn -version

4. In the Global Tool Configuration of Jenkins add Maven path as below:



The screenshot shows the 'Maven' configuration page in Jenkins. On the left, under 'Maven installations', there is a list with one entry 'Maven'. To the right of this list is an 'Add Maven' button. Below the list, there are two input fields: 'Name' with the value 'Maven' and 'MAVEN\_HOME' with the value '/usr/share/apache-maven'. Below these fields is a checkbox labeled 'Install automatically' which is unchecked. A red 'Delete Maven' button is located at the bottom right of the configuration area. A help icon (?) is also visible next to the 'Install automatically' checkbox.

5. Create a repo in GitHub and push the below code to the Git repository

```
root@ip-172-31-85-119 JenkinsProject]# pwd
/root/JenkinsProject
[root@ip-172-31-85-119 JenkinsProject]# ls -ld simple-java-maven-app/
drwxr-xr-x 5 root root 4096 May  3 16:11 simple-java-maven-app/
```

6. Create a new pipeline project and configure SCM details as below

Jenkins > mvn\_app >

General Build Triggers Advanced Project Options **Pipeline**

### Pipeline

Definition Pipeline script from SCM

SCM Git

Repositories

Repository URL

Credentials  Add Advanced... Add Repository

Branches to build

Branch Specifier (blank for 'any')  Add Branch

Repository browser (Auto)

Additional Behaviours Add

Script Path

Save Apply

7. Save and Build the Job
8. In the console Output we can see a Jar has been build using maven and displays Hello World

```
-----  
application (which Jenkins built using Maven) to the Jenkins UI.  
+ java -jar target/my-app-1.0-SNAPSHOT.jar  
Hello World!  
[Pipeline] }  
[Pipeline] // stage  
[Pipeline] }  
[Pipeline] // withEnv  
[Pipeline] }  
[Pipeline] // node  
[Pipeline] End of Pipeline  
Finished: SUCCESS
```