

## **Step - 1 : Create Linux VM in AWS**

**Create Ubuntu VM using AWS EC2 (t2.medium, because jenkins require minimum 4gb RAM)**

## **Step-2 : Install Java using below commands**

```
sudo apt update
sudo apt install fontconfig openjdk-17-jre
java -version
```

## **Step-3 : To Install Jenkins use the below commands**

```
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
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
sudo apt-get install jenkins
```

## **Step-4 : Start Jenkins services using below commands**

```
sudo systemctl enable jenkins
sudo systemctl start jenkins
```

## **Step-5 : Verify Jenkins setup**

```
sudo systemctl status jenkins
```

## **Step-6 : Open jenkins server in browser using VM public ip of linux vm**

**`http://public-ip:8080/`**

## **Step-7 : Copy jenkins admin pwd**

**`sudo cat /var/lib/jenkins/secrets/initialAdminPassword` ---> it is in git bash**

**paste it and add all pulgins and get url paste in console**

**`http://43.204.32.252:8080/`**

## **Step-8 : Create Admin Account & Install Required Plugins in Jenkins**

## Setup Docker in Jenkins

#####

```
curl -fsSL get.docker.com | /bin/bash
sudo usermod -aG docker jenkins
sudo usermod -aG docker ubuntu
sudo systemctl restart jenkins
sudo docker version
```

---

## Configure Maven as Global Tool in Jenkins

#####

Manage Jenkins -> Tools -> Maven Installation -> Add maven

-----

create a job

name : pipeline

choose pipeline

if you went back to dash board choose name pipeline and configure --> pipe line

choose hello world and choose pipeline syntax and git : git --> add url plus change to master to main and paste the code in pipeline

then apply and save ,build now

and then config again and pipe line now paste maven code ,make user maven is added ,manage jenkins ,tools ,maven installation ,add maven

```
pipeline {
    agent any

    tools {
        maven 'maven-3.9.9'
    }
}
```

```

stages {
    stage('Git Checkout') {
        steps {
            git branch: 'main', url: 'https://github.com/sasank2323/docker-test.git'
        }
    }

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

```

docker process :

First Jenkins JOB

---

```

pipeline {
    agent any

    tools {
        maven 'x-3.9.9'
    }

    stages {
        stage('git clone') {
            steps {
                git branch: 'main', url: 'https://github.com/pankajmutha14/docker-test.git'
            }
        }
        stage('mvn') {
            steps{
                sh 'mvn clean test package'
            }
        }
        stage('build image') {
            steps{
                sh 'docker build -t psait/test1 .'
            }
        }
    }
}

```

```

    stage('deployment'){
      steps{
        sh 'docker run -d -p 9090:8080 --name psait/test1'
      }
    }
  }
}

```

check it name according to tools name that you configure in manage tools , tools -> maven you would have set some name same name you have to keep

```

pipeline {
  agent any

  tools{
    maven 'maven-3.9.9'
  }

  stages {
    stage('Git Clone') {
      steps {
        git branch: 'main', url: 'https://github.com/sasank2323/docker-test.git'
      }
    }

    stage('Maven Execute Commands') {
      steps {
        sh 'mvn clean test package'
      }
    }

    stage('docker image') {
      steps{
        sh 'docker build -t psait/test1 .'
      }
    }

    stage('deployment'){
      steps{
        sh 'docker run -d -p 9090:8080 --name sasank psait/test1'
      }
    }
  }
}

```

```
}
```

these will give error because we cant run one docker multiple tims

so we have to stop it and remove this by using name sasank

```
sh "docker run -d -p --name [sasank] sasank2323/test1"
```

CODE :

```
pipeline {
  agent any

  tools {
    maven 'maven-3.9.9'
  }

  stages {
    stage('Git Checkout') {
      steps {
        git branch: 'main', url: 'https://github.com/sasank2323/docker-test.git'
      }
    }

    stage('Maven Build') {
      steps {
        sh 'mvn clean package'
      }
    }
    stage('docker image') {
      steps{
        sh 'docker build -t sasank2323/test1 .'
      }
    }
    stage('deployment'){
      steps{
        sh 'docker stop sasank'
        sh 'docker rm sasank'
        sh 'docker run -d -p 9090:8080 --name sasank sasank2323/test1'
      }
    }
  }
}
```

```
}  
}
```

<http://3.110.195.13:8080/>