

We have deployed java war file using below steps

- 1.Jdk install
- 2.tomcat install
- 3.maven
- 4.jenkins

/mnt/project

git clone <https://github.com/Shantanumajan6/project.git>

Login to jenkins

Step 1:

Create a New Job:

- Click on "**New Item**" on the Jenkins dashboard.
- Enter a name for your job, e.g., Java-Maven-Build.
- Select "**Freestyle project**" and click "**OK**"

Step 2:configure gilt repo.

Source Code Management:

- Select "**Git**".
- Enter the **Repository URL** of your Git project.
- Provide **Credentials** if your repository requires authentication.
- Specify the **Branch** to build, e.g., */main or */master.

The screenshot shows the 'Git' configuration section in Jenkins. It includes a 'Repository URL' field with the value 'https://github.com/shubhangi-zanzane/project1.git', a 'Credentials' dropdown menu set to 'git', and an 'Add' button. There is also an 'Advanced' dropdown menu.

Step 3: provide branch

The screenshot shows the 'Branches to build' configuration section in Jenkins. It includes a 'Branch Specifier (blank for \'any\')' field with the value '*/master'.

4.

Build Steps

≡ Execute shell ?

Command

See [the list of available environment variables](#)

```
rm -rf /root/.m2/repository
```

5. Build Steps:

- Click **"Add build step"**.
- Select **"Invoke top-level Maven targets"**.
- In the **Goals** field, enter the Maven command for building your project, typically clean install.

≡ Invoke top-level Maven targets ?

Maven Version

apache-maven-3.9.7

Goals

clean install

Advanced ▾

6. post build action

≡ Execute shell ?

Command

See [the list of available environment variables](#)

```
mkdir -p /mnt/project  
cp -r /mnt/project/target/LoginWebApp.war /mnt/server/apache-tomcat-9.0.89/webapps
```

Advanced ▾

7. click **"Save"** to apply the configurations

8. go to browser and give public Ip along with port 8080 and war name

