

By: Sudarshan\_DevOps\_Engineer

=====

## Integrate Jenkins with Nexus Repository

---

### Step:1

=====

Take 2 servers (t2.medium)

- ➔ On first Sever install java, maven, Jenkins.
- ➔ On second server install docker.
- ➔ Install Nexus using docker container.

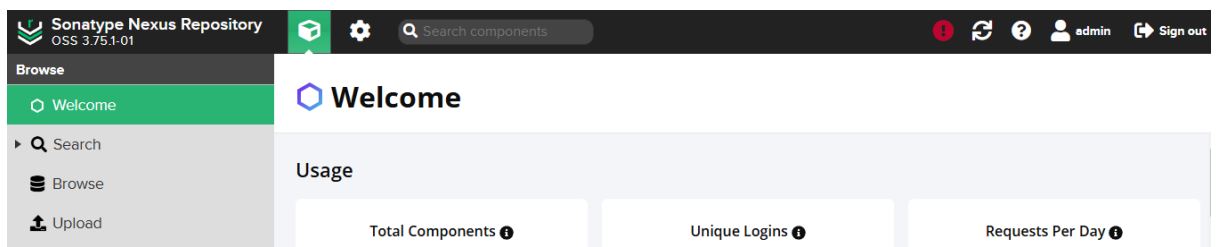
```
docker run --name nexus -d -p 8081:8081 sonatype/nexus3
```

- ➔ Now take pub-ip-nexus-server:8081 access the nexus application from the browser.

### Step:2

=====

- ➔ Now login into Nexus repository, it looks like as bellow



- ➔ Now create Repository in Nexus as bellow  
1. Click on gear icon ➔ click on Repositories ➔ click

on create Repository→select maven2(hosted)  
these is for uploading the artifact, and  
maven2(proxy) is for downloading the  
artifact,maven2(group) is for uploading and  
downloading the artifact.



➔ Now write any anyname as bellow

A screenshot of a form for creating a repository. The form has the following sections:

- Name:** A unique identifier for this repository. The input field contains 'mymaven'.
- Online:** A checkbox labeled 'If checked, the repository accepts incoming requests' is checked.
- Maven 2**
  - Version policy:** What type of artifacts does this repository store? The dropdown menu is set to 'Release'.
  - Layout policy:** Validate that all paths are maven artifact or metadata paths. The dropdown menu is set to 'Strict'.
  - Content Disposition:** Add Content-Disposition header as 'Attachment' to disable some content from being inline in a browser. The dropdown menu is set to 'Inline'.
- Storage**

Here Select mixed (it means you can upload war,jar,ear), then  
scroll down and create repository.

### Step:3

---

→ Now install Nexus Artifact uploader Pluggin in Jenkins

Now go to Jenkins dashboard and create free style job as bellow

→

None

☒ Git ?

Repositories ?

Repository URL ?

`https://github.com/sudarshansw7/mymaven.git`

Credentials ?

- none -

+ Add

Advanced

Add Repository

Save Apply

Here give Url of the project from Github

→ Next build step

#### Build Steps

≡ Invoke top-level Maven targets ?

Goals

package

Advanced

➔ Now build the project .and open the console copy the artifact path.

```
[1;34mINFO[1;34m] Assembling webapp [webapp] in [/var/lib/jenkins/workspace/dev-o1/webapp/target/webapp]
[1;34mINFO[1;34m] Processing war project
[1;34mINFO[1;34m] Copying webapp resources [/var/lib/jenkins/workspace/dev-o1/webapp/src/main/webapp]
[1;34mINFO[1;34m] Building war: /var/lib/jenkins/workspace/dev-o1/webapp/target/webapp.war
[1;34mINFO[1;34m] -----[1;34m]
[1;34mINFO[1;34m] [1mReactor Summary for Maven Project 1.0-SNAPSHOT:[1;34m]
[1;34mINFO[1;34m]
[1;34mINFO[1;34m] Maven Project ..... [1;32mSUCCESS[1;34m] [ 0.006 s]
[1;34mINFO[1;34m] Server ..... [1;32mSUCCESS[1;34m] [ 2.300 s]
[1;34mINFO[1;34m] Webapp ..... [1;32mSUCCESS[1;34m] [ 1.116 s]
```

Copy that path and use it in the file field of nexus configuration

➔ Next Configure Nexus as bellow

**Nexus artifact uploader**

**Nexus Details**

Nexus Version

NEXUS3

Protocol

HTTP

Nexus URL ?

98.84.119.137:8081

Credentials

admin/\*\*\*\*\*

Here paste the public ip:8081 of nexus.

Here give the nexus credentials.

→ Next

A screenshot of a web form for configuring a Maven repository. It contains three input fields: 'GroupId' with the value 'com.example.maven-project', 'Version' with the value '1.0-SNAPSHOT', and 'Repository' with the value 'mymaven'. Each field has a blue arrow pointing from it to an explanatory text block below.

Here give the groupid from your pom.xml file.

Here give version for your artifact as above or any number for your version.

Here give the repository name which you have created in Nexus.

→ Next

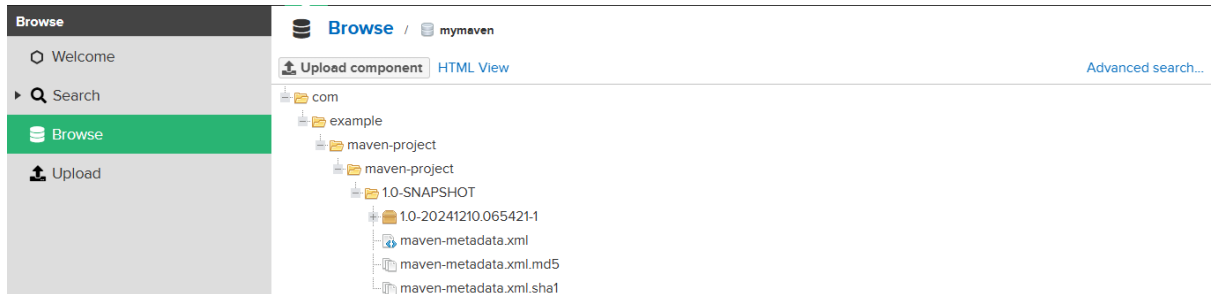
A screenshot of the 'Artifact' configuration form in Nexus. It contains four input fields: 'ArtifactId' with the value 'maven-project', 'Type' with the value 'war', 'Classifier' (empty), and 'File' with the value '/var/lib/jenkins/workspace/dev-o1/webapp/target/webapp.war'. Each field has a blue arrow pointing from it to an explanatory text block below.

Here give artifactid from pom.xml file.

Here specify whether it is war or jar or ear.

Here paste the path of the artifact from your Jenkins console.

➔ Next your artifacts are uploaded in Nexus when you build the job, Go to browse section and click on your repository then you can find the artifact.



=====Download Artifact from Nexus to Jenkins =====

Step: 1

=====

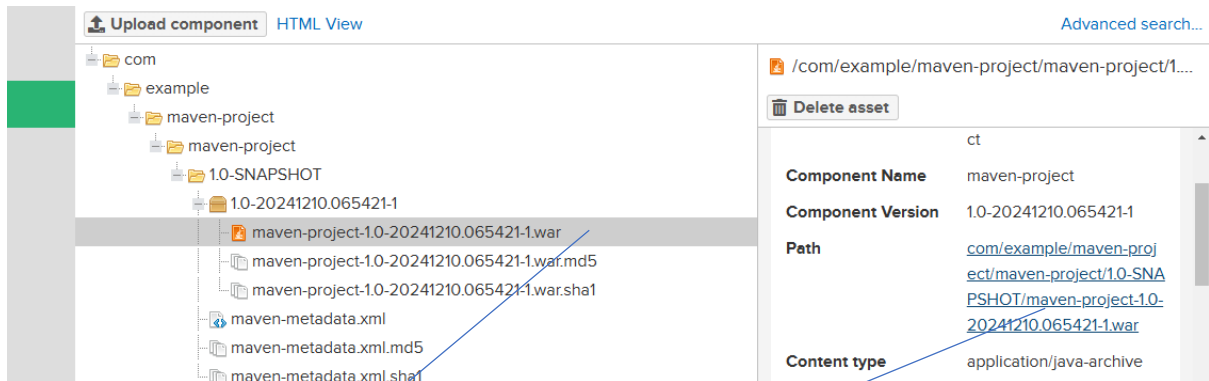
➔ In build steps click on Add build ➔ click on execute shell and write as bellow



Here use nexus user and password and paste the path of the artifact in nexus which present on the right hand side of artifact

Here I am renaming the artifact because the name is too long .

→ The path is copied from here:



Here click on the artifact.

Here click on the link and copy link address.

Step:2

=====

→ And deploy to tomcat container using post build actions .

→ Now deploy to tomcat container in Prod environment



☰ Tomcat 9.x Remote

×

Credentials

admin/\*\*\*\*\*

▼

+ Add

Tomcat URL ?

http://52.204.231.90:8080

Advanced ▼

===== End =====