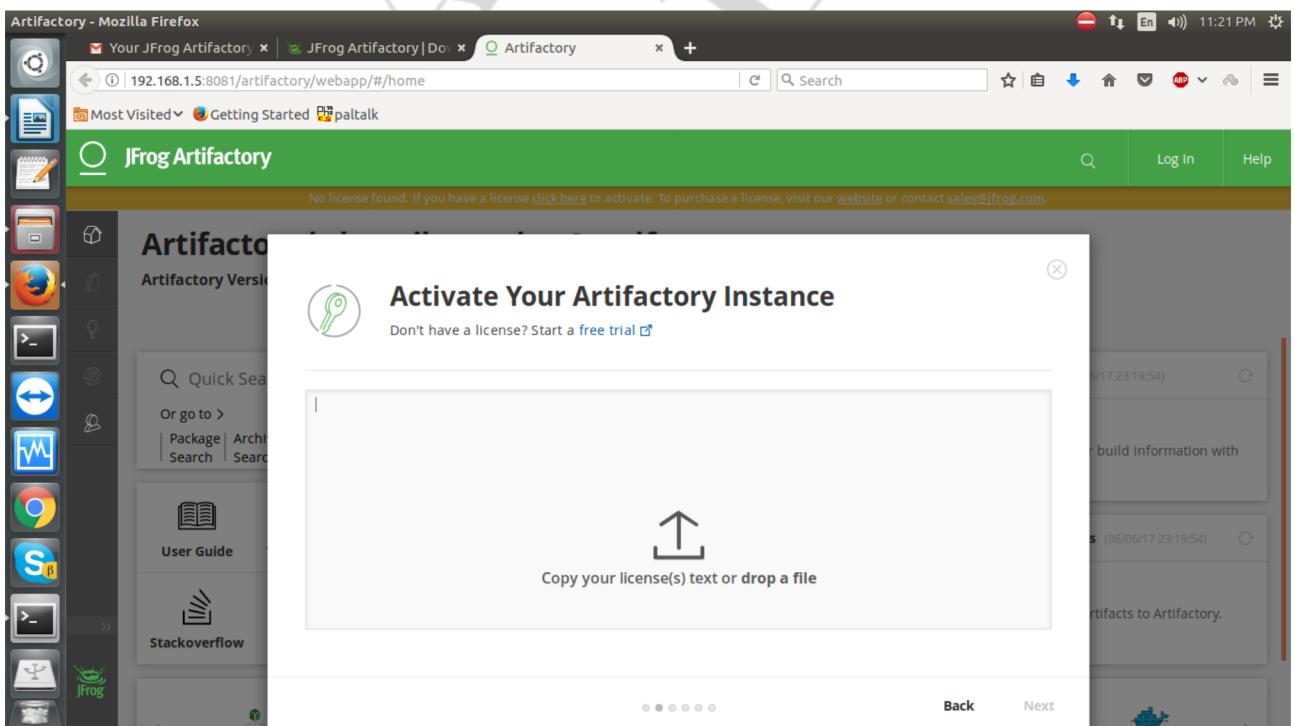


Welcome to JFrog Artifactory!

Now Click on
==> Next

Now the page will ask for the License Key



check your email & Copy and Paste the link in the box and click on next

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

The screenshot shows the JFrog Artifactory web interface. A central modal window titled "Activate Your Artifactory Instance" is displayed. It contains a key icon and the text "Don't have a license? Start a free trial". Below the modal, there is a large amount of encoded license key text. At the bottom of the modal are "Back" and "Next" buttons. The background of the main interface shows a sidebar with various icons and links like "Quick Search", "User Guide", and "Stackoverflow". The top navigation bar includes "Log In" and "Help".

==> Set Admin Password for your choice.
 ==> Skip the proxy settings
 ==> Now select Maven & click on create & then click on Finish.

The screenshot shows the JFrog Artifactory web interface after activation. A modal window titled "Create Repositories" is open. It displays a grid of repository types with their corresponding icons: Bower, Chef, CocoaPods, Conan, Octane, docker, Gems, Generic, Git LFS, Gradle, Ivy, maven (which is highlighted), npm, NuGet, Opcg, and Composer. Below the grid, a note says "Select the package type(s) you want - we'll create the default repositories for you! Need to skip? No worries, create and manage repositories anytime." At the bottom of the modal are "Back", "Skip", and "Create" buttons. The background shows the main Artifactory dashboard with sections for "High Availability", "JFrog Bintray Distribution", "JFrog Xray", "JFrog CLI", and "Build Integration".

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Artifactory Version 5.3.1 (Uptime is 0d 0h 10m 37s)
Latest Release: 5.3.1

Quick Search

Set Me Up

Last Deployed Builds (06/06/17 23:25:23)

No builds to display.
Learn how to Integrate your build information with Artifactory.

Most Downloaded Artifacts (06/06/17 23:25:23)

No artifacts to display.
Learn how to deploy your artifacts to Artifactory.

When you click on Finish the above screen will appear.

Now we must set our own maven repository so that we can upload the artefacts from Jenkins.
Follow the steps to create the repository in JFROG.

Go the the WELCOME ADMIN @ Top Right side of the screen you will find a drop down as soon as you place the cursor over it
Now select ==> Local Repository

Welcome, admin

Create Repositories

- Quick Setup
- Local Repository
- Remote Repository
- Virtual Repository
- Distribution Repository
- Add User
- Add Group
- Add Permission
- Edit Profile
- Log Out

Now the following next page will ask you to select Package Type
Select Maven , New page will open

The screenshot shows the 'New Local Repository' configuration page in JFrog Artifactory. A modal window titled 'Select Package Type' is displayed, listing various package types with their icons: Bower, Chef, CocoaPods, Conan, debian, docker, Gems, Git LFS, Gradle, Ivy, Maven (selected), npm, NuGet, Opm, PHP Composer, Puppet, SBT, RPM, and Generic. The 'Maven' option is highlighted. The background shows fields for 'Basic' settings like 'Repository Key' (mandatory) and 'General' settings like 'Repository Layout' (maven-2-default).

==> New Local Repository page is open where we can enter the name of repository of our choice and then Click on Save & Finish

The screenshot shows the 'New Local Repository' configuration page in JFrog Artifactory. The 'Basic' tab is active, showing the 'Package Type' dropdown set to 'Maven'. The 'Repository Key' field is highlighted with a red border and contains the placeholder text 'You must fill in this field'. On the right side, under 'Maven Settings', the 'Checksum Policy' is set to 'Verify against client checksums'. The 'Save & Finish' button is visible at the bottom right.

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

The screenshot shows the JFrog Artifactory interface. On the left is a sidebar with various icons for repository management. The main area is titled "Local Repositories" and displays a table of three repositories:

Repository Key	Type	Recalculate Index	Replications
jfrog_project1	m Maven	↻	⟳
libs-release-local	m Maven	↻	⟳
libs-snapshot-local	m Maven	↻	⟳

A search bar at the top right says "Search" and shows a user "Welcome, admin".

Now Our Own Repository is Created where we can upload the artifacts.

Now we must setup Jenkins server

1. apt-get update -y
2. apt-get install openjdk-8-jdk -y
3. apt-get install Jenkins -y
4. apt-get install git
5. apt-get install maven -y

now open the url :: <ip>:8080

Jenkins page is open , create a user and login to the Jenkins

Now install few plugins before starting of the project

1. Zentimestamp plugin
2. Artifactory Plugin
3. Maven Invoker plugin
4. GitHub Authentication plugin

Now go to Jenkins --> Manage Jenkins --> Configure System

Enter the details in Global Properties & in Artifactory as show below & then Save.

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Configure System [Jenkins] - Mozilla Firefox

Your JFrog Artifactory | Artifactory | Configure System [Jen...]

192.168.1.5:8080/configure

Most Visited Getting Started paltalk

Jenkins > configuration

Global properties

Restrict project naming

Date pattern for the BUILD_TIMESTAMP (build timestamp) variable

Date and Time Pattern: yyyyMMddHHmm

Environment variables

Tool Locations

Usage Statistics

Help make Jenkins better by sending anonymous usage statistics and crash reports to the Jenkins project.

Timestamper

System clock time format: 'HH:mm:ss '

Elapsed time format: 'HH:mm:ss.S '

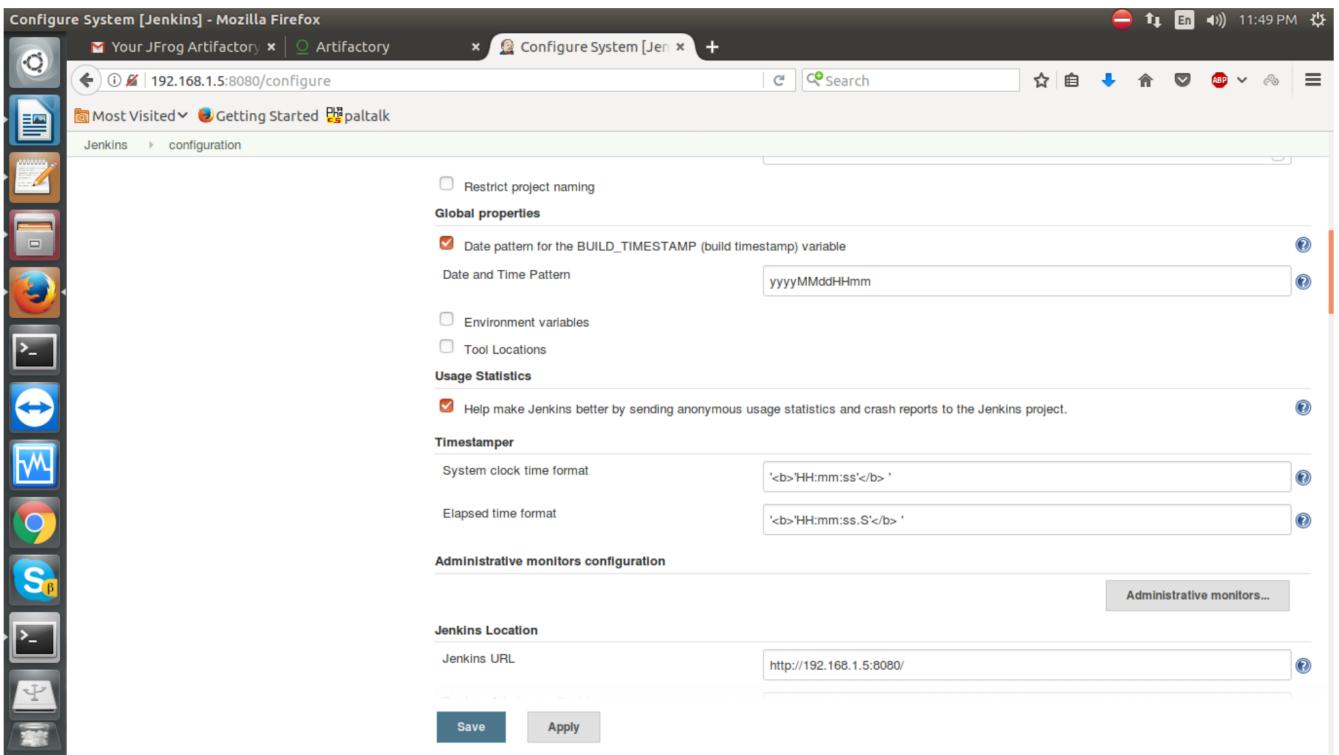
Administrative monitors configuration

Administrative monitors...

Jenkins Location

Jenkins URL: http://192.168.1.5:8080/

Save Apply



Configure System [Jenkins] - Mozilla Firefox

Your JFrog Artifactory | Artifactory | Configure System [Jen...]

192.168.1.5:8080/configure

Most Visited Getting Started paltalk

Jenkins > configuration

Sharable libraries available to any Pipeline jobs running on this system. These libraries will be trusted, meaning they run without "sandbox" restrictions and may use @Grab.

Add

Artifactory

Enable Push to Bintray

Use the Credentials Plugin

Artifactory servers

Server ID: ART

URL: http://192.168.1.5:8081/artifactory/

Default Deployer Credentials

Username: admin

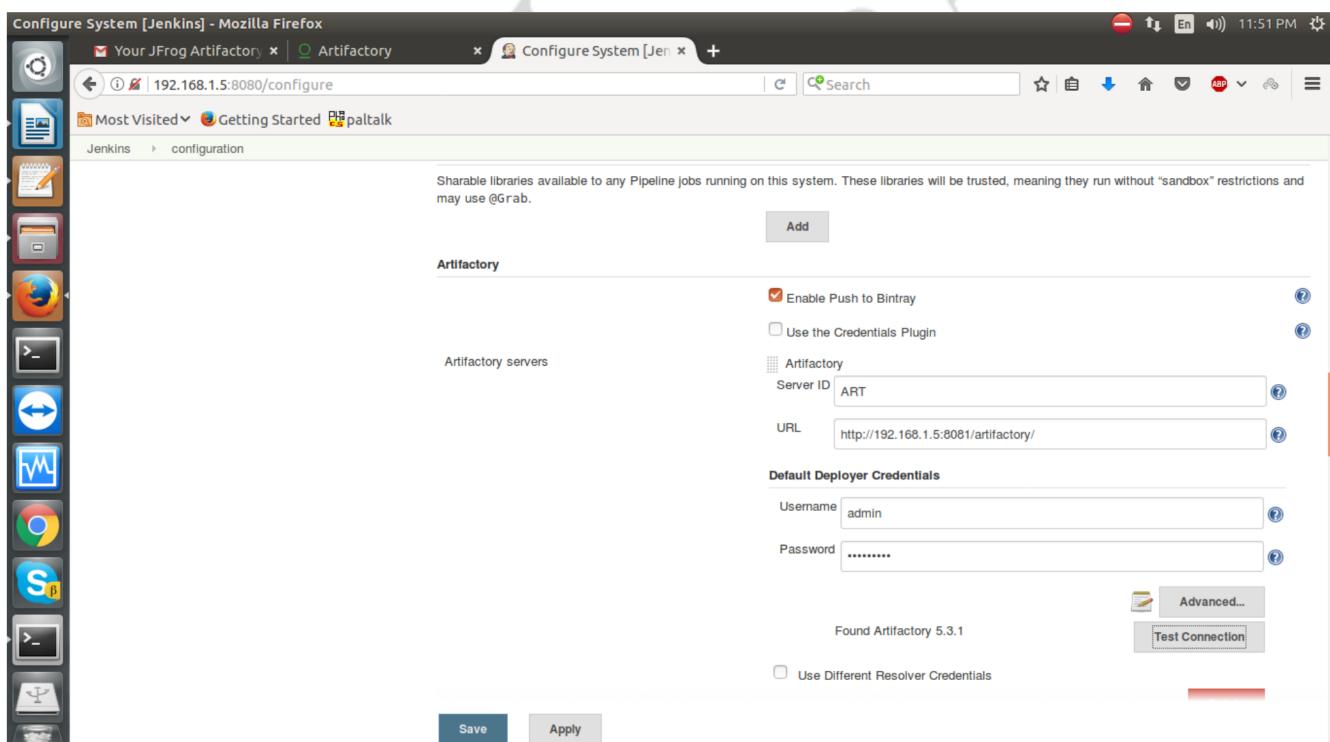
Password: *****

Advanced... Test Connection

Found Artifactory 5.3.1

Use Different Resolver Credentials

Save Apply



Now go to Manage Jenkins --> Global Tool Configuration & do as following

In JDK , type : jdk8

In Maven , type : mvn3

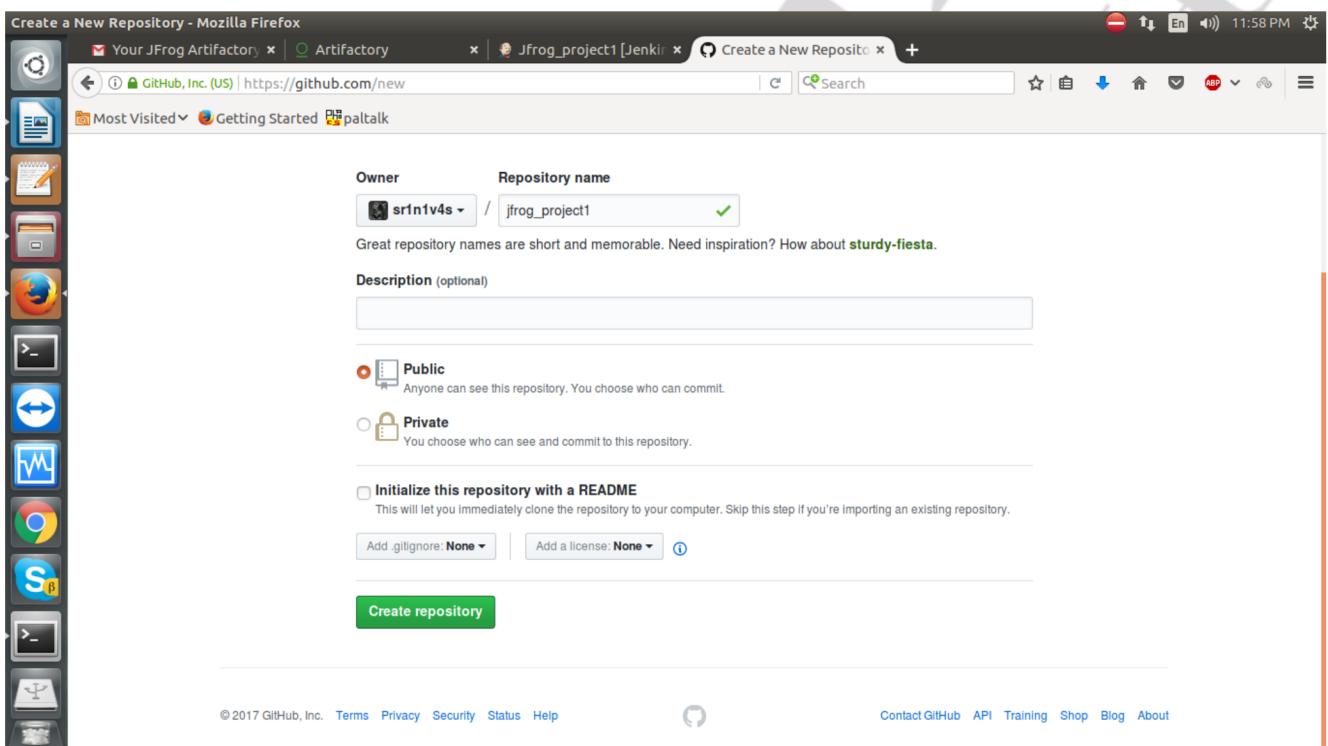
then click on Save

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Jenkins --> New Item --> Enter the Project Name, Select Free Style Project & then Click OK

Now Before doing the configuration of the Project
Let's set over on Git hub for this project
Create a git hub user & password and login to the github
Select a Start a Project & Click On Create Repository



Now go to the Terminal of Jfrog

Now git clone <https://github.com/JFrogDev/project-examples.git>
this url to the jfrog home directory

Now the projects-example directory is created.

Now let's clone the git repo we have created on our github.
Git clone <url>

==>

Now go the path :
/home/project-examples/artifactory-maven-plugin-example
and copy the files & directories to our git directory

==> mv -f * /home/jfrog-project1

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

The files from artifactory-maven-plugin-example is copied to our jfrog-project1

Now we have to add few things to the pom.xml

Go the Jfrog url & select the local repo which we have specified

This screenshot shows the JFrog Artifactory homepage. The top navigation bar includes tabs for 'Your JFrog Artifactory', 'Artifactory', 'Jfrog_project1 [Jenkins]', 'sr1n1v4s/jfrog_project', and 'JFrogDev/project-exa'. The main content area displays the message 'Artifactory is happily serving 0 artifacts' and 'Artifactory Version 5.3.1 (Uptime is 0d 0h 21m 18s)'. Latest Release: 5.3.1. On the left, there's a sidebar with various icons for User Guide, Webinar Signup, Support Portal, Stackoverflow, Blog, and Rest API. The 'Set Me Up' section lists repository keys: libs-release, libs-snapshot, jfrog_project1, libs-release-local, libs-snapshot-local, and jceter. The 'Last Deployed Builds' and 'Most Downloaded Artifacts' sections both show no results.

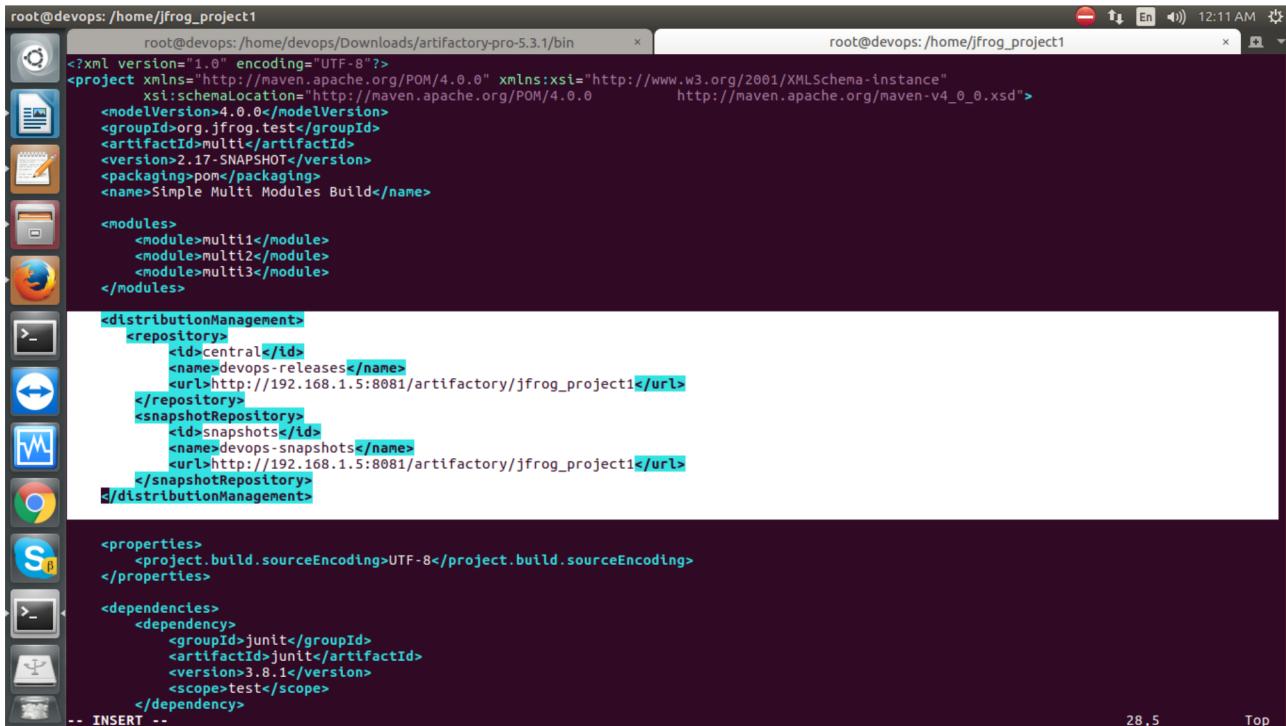
This screenshot shows the same JFrog Artifactory homepage as above, but with a modal window titled 'Set Me Up' overlaid. The modal contains a 'Deploy' section with instructions: 'To deploy build artifacts through Artifactory you need to add a deployment element with the URL of a target local repository to which you want to deploy your artifacts. For example:' followed by a code snippet:

```
1  <distributionManagement>
2    <repository>
3      <id>central</id>
4      <name>devops-releases</name>
5      <url>http://192.168.1.5:8081/artifactory/jfrog_project1</url>
6    </repository>
7    <snapshotRepository>
8      <id>snapshots</id>
9      <name>devops-snapshots</name>
10     <url>http://192.168.1.5:8081/artifactory/jfrog_project1</url>
11   </snapshotRepository>
12 </distributionManagement>
```

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: - +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

We will get the deploy code we have to copy this 12 line to the pom.xml



```
<?xml version="1.0" encoding="UTF-8"?>
<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://www.w3.org/2001/XMLSchema-instance">
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.jfrog.test</groupId>
  <artifactId>multi</artifactId>
  <version>2.17-SNAPSHOT</version>
  <packaging>pom</packaging>
  <name>Simple Multi Modules Build</name>

  <modules>
    <module>multi1</module>
    <module>multi2</module>
    <module>multi3</module>
  </modules>

  <distributionManagement>
    <repository>
      <id>central</id>
      <name>devops-releases</name>
      <url>http://192.168.1.5:8081/artifactory/jfrog_project1</url>
    </repository>
    <snapshotRepository>
      <id>snapshots</id>
      <name>devops-snapshots</name>
      <url>http://192.168.1.5:8081/artifactory/jfrog_project1</url>
    </snapshotRepository>
  </distributionManagement>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

```

In the same pom.xml replace the content as below

```
<contextUrl>http://localhost:8081/artifactory</contextUrl>
<username>admin</username> ==< enter username
<password>iamdevops</password> ==< enter password
<excludePatterns>*-tests.jar</excludePatterns> ==< no change
<repoKey>jfrog_project1</repoKey> ==< enter the repo name
<snapshotRepoKey>jfrog_project1</snapshotRepoKey> ==< enter the repo name
```

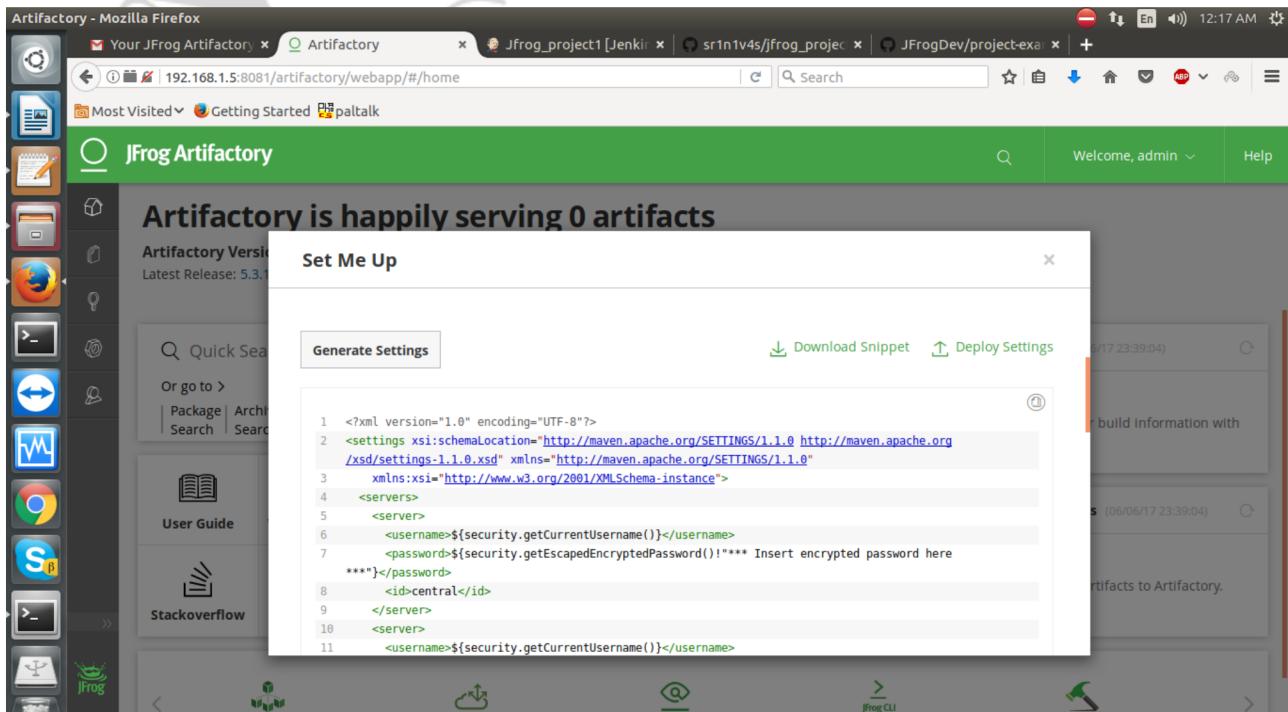
then save & quit the file

Now go to /home

create a directory .m2

==> mkdir .m2 & open a file settings.xml

From the Jfrog , select our repo & click on Generate Settings and copy the settings & paste in the



```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <settings xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.1.0 http://maven.apache.org/xsd/settings-1.1.0.xsd" xmlns="http://maven.apache.org/SETTINGS/1.1.0"
3    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
4    <servers>
5      <server>
6        <username>${security.getCurrentUsername()}</username>
7        <password>${security.getEscapedEncryptedPassword()}!*** Insert encrypted password here
8        ***</password>
9      <id>central</id>
10     </server>
11     <server>
12       <username>${security.getCurrentUsername()}</username>
```

settings.xml file.

Now all the changes have been made to the pom.xml file & maven default file

Now push the directory to our git hub repository

==> git add .

==> git commit -m "first push"

=> git push origin master

all the files have been updated to out git hub

Check the Github Now

The screenshot shows a GitHub repository page for 'sr1n1v4s/jfrog_project1'. The repository has 1 commit, 1 branch, 0 releases, and 1 contributor. The commit list shows files like README.md and pom.xml being pushed by user 'sr1n1v4s'.

File	Pushed By	Message	Time Ago
README.md	sr1n1v4s	first push	39 seconds ago
pom.xml	sr1n1v4s	first push	39 seconds ago
multi3	sr1n1v4s	first push	39 seconds ago
multi2	sr1n1v4s	first push	39 seconds ago
multi1	sr1n1v4s	first push	39 seconds ago
sr1n1v4s first push	sr1n1v4s		Latest commit 6b9940d 40 seconds ago

Now copy this url to the Jenkins Source Code Management (GIT)