

Setup Jenkins Server

- Setup a Linux EC2 Instance
- Install Java
- Install Jenkins
- Start Jenkins
- Access Web UI on port 8080



Jenkins

So, as a first step, we need

Instal Jenkins :

A screenshot of a web browser window. The address bar shows 'pkg.jenkins.io/redhat-stable/'. The main content area displays the 'Jenkins Redhat Packages' page. It includes instructions for adding the repository via command line (using 'wget' and 'rpm --import') and a note about importing the key. Below that, it shows yum commands to install Jenkins dependencies. A section titled 'The rpm packages were signed using this key:' lists a public RSA key. At the bottom, there is a note about Java runtime environment and a green banner with the text 'If you see the instructions over here, to use'.

To use this repository, run the following command:

```
sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.key
```

If you've previously imported the key from Jenkins, the `rpm --import` will fail because you already have a key. Please ignore that and move on.

```
yum install epel-release # repository that provides 'daemonize'
yum install java-11-openjdk-devel
yum install jenkins
```

The rpm packages were signed using this key:

```
pub rsa4096 2020-03-30 [SC] [expires: 2023-03-30]
62A97568FD7B0C377CF24BA8FCE32E745F2C3D5
uid Jenkins Project
sub rsa4096 2020-03-30 [E] [expires: 2023-03-30]
```

You will need to explicitly install a Java runtime environment, because Oracle's Java RPMs are incorrect and fail to register as providing a java dependency.
Thus, adding an explicit dependency on java-11-openjdk-devel is required.

If you see the instructions over here, to use

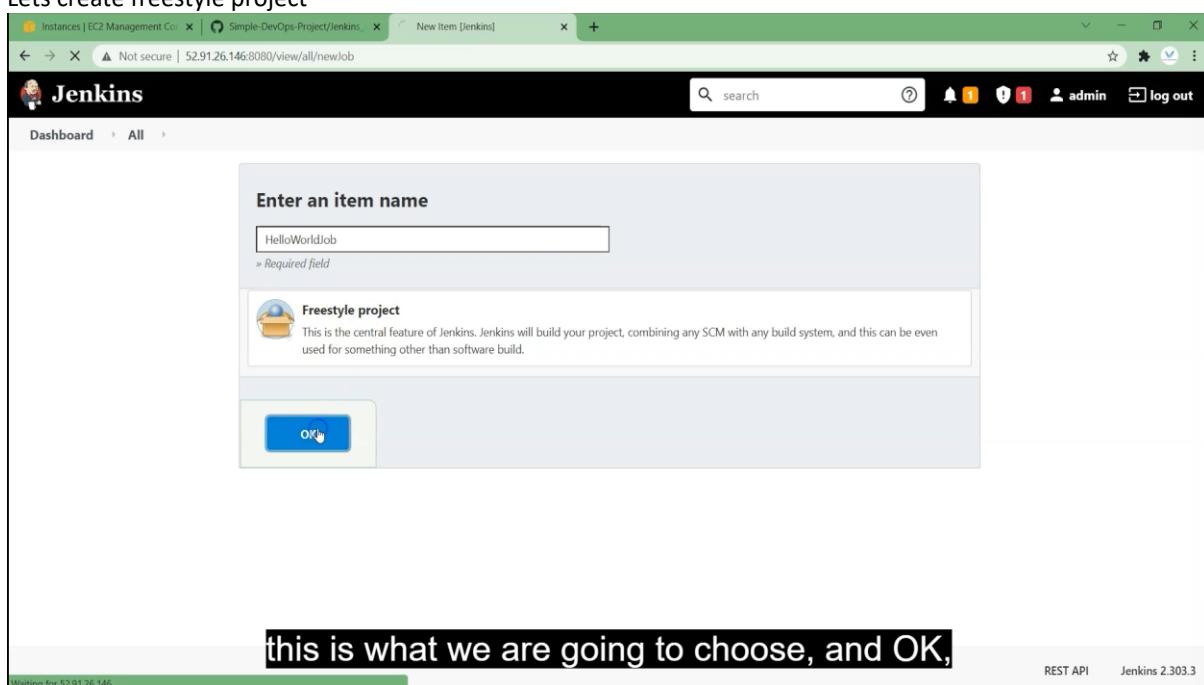
Once epel package is installed Jenkins can be installed .

```
52.91.26.146 (ec2-user)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect... 2. 52.91.26.146 (ec2-user) X server Exit

[root@ip-172-31-85-139 ~]#
[root@ip-172-31-85-139 ~]# service jenkins status
● jenkins.service - LSB: Jenkins Automation Server
  Loaded: loaded (/etc/rc.d/init.d/jenkins; bad; vendor preset: disabled)
  Active: inactive (dead)
    Docs: man:systemd-sysv-generator(8)
[root@ip-172-31-85-139 ~]# service jenkins start
Starting jenkins (via systemctl): [ OK ]
[root@ip-172-31-85-139 ~]# service jenkins status
● jenkins.service - LSB: Jenkins Automation Server
  Loaded: loaded (/etc/rc.d/init.d/jenkins; bad; vendor preset: disabled)
  Active: active (running) since Sun 2021-11-07 04:29:17 UTC; 3s ago
    Docs: man:systemd-sysv-generator(8)
  Process: 3721 ExecStart=/etc/rc.d/init.d/jenkins start (code=exited, status=0/SUCCESS)
  CGroup: /system.slice/jenkins.service
         └─3725 /etc/alternatives/java -Djava.awt.headless=true -DJENKINS_HOME=/var/lib/jenkins -jar /usr/lib/jenkins/jenkins.war --logfile...

Nov 07 04:29:17 ip-172-31-85-139.ec2.internal systemd[1]: Starting LSB: Jenkins Automation Server...
Nov 07 04:29:17 ip-172-31-85-139.ec2.internal jenkins[3721]: Starting Jenkins [ OK ]
Nov 07 04:29:17 ip-172-31-85-139.ec2.internal systemd[1]: Started LSB: Jenkins Automation Server.
[root@ip-172-31-85-139 ~]#
```

Launch our first Jenkins job : Lets create freestyle project



The screenshot shows the Jenkins configuration interface for a project named 'HelloWorldJob'. The 'Post-build Actions' tab is active, indicated by a yellow circle around its tab header. Below the tabs, there's a 'Description' section with a text area and a 'Plain text' preview link. Under 'Source Code Management', the 'None' option is selected. In the 'Build Triggers' section, there are several checkboxes: 'Discard old builds', 'This project is parameterized', 'Disable this project', and 'Execute concurrent builds if necessary'. At the bottom of this section are 'Save' and 'Apply' buttons. The main content area is covered by a large black rectangle, obscuring the details of the post-build actions.

All features comes with plugins .

The first screenshot shows the 'Build' configuration screen with an open 'Add build step' dropdown. The 'Execute shell' option is highlighted and selected. The second screenshot shows the detailed configuration for the 'Execute shell' step. It includes a 'Command' input field containing the text 'echo "Hello World"' on the first line and 'uptime' on the second line. A yellow circle highlights the cursor in the command field. Below the command field is a link 'See the list of available environment variables'.

We created freestyle project that will execute shell command .

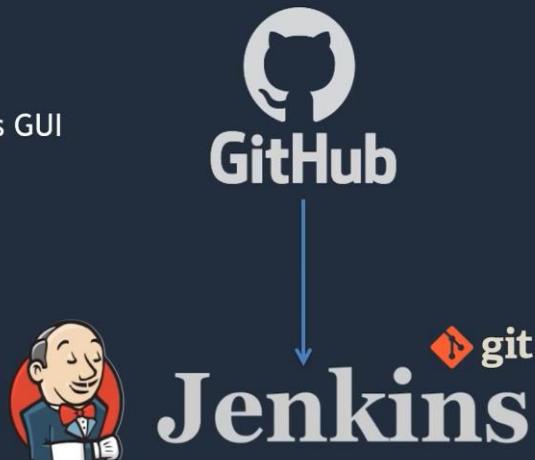
Build Code



So, on Jenkins CLI, first we need to install it.

Integrate GitHub with Jenkins

- Install Git on Jenkins Instance
- Install GitHub Plugin on Jenkins GUI
- Configure Git on Jenkins GUI



These are the three steps we should follow.

To change hostname

```
[root@jenkins-server ~]# cat /etc/hostname  
jenkins-server  
[root@jenkins-server ~]#
```

Temporarily change hostname :

```
jenkins-server  
[root@jenkins-server ~]# hostname jenkins-server  
[root@jenkins-server ~]#
```

```
yum install git
```

github plugin installation :

The screenshot shows the Jenkins Manage Jenkins page. A prominent yellow warning box at the top states: "Building on the controller node can be a security issue. You should set up distributed builds. See [the documentation](#)". Below this, another box says: "Jenkins URL is empty but is required for the proper operation of many Jenkins features like email notifications, PR status update, and environment variables such as BUILD_URL." A message below asks: "Please provide an accurate value in [Jenkins configuration](#)". In the System Configuration section, there are links for "Configure System", "Global Tool Configuration", and "Manage Plugins". The "Manage Plugins" link is highlighted with a yellow circle. The main content area shows the "Plugin Manager" with a search bar containing "github", tabs for "Updates", "Available" (which is selected), "Installed", and "Advanced", and a list for the "GitHub" plugin.

This screenshot is identical to the one above, showing the Jenkins Manage Jenkins page with its respective warnings and System Configuration section. The "Manage Plugins" link in the System Configuration is again circled in yellow.

The screenshot shows the Jenkins Global Tool Configuration page. At the top, there are three tabs: Instances | EC2 Management Co..., GitHub, and Global Tool Configuration [Jenkins]. The URL in the address bar is Not secure | 52.91.26.146:8080/configureTools/. The main content area has two sections: 'JDK' and 'Git'. Under 'JDK', there is a link to 'JDK installations' and a button to 'Add JDK'. A note says 'List of JDK installations on this system'. Under 'Git', there is a link to 'Git installations'. A section for 'Git' shows a 'Name' field containing 'Default' and a 'Path to Git executable' field containing 'git'. There is also a checkbox for 'Install automatically'.

Global Tool Configuration

Use default maven global settings

JDK

JDK installations

Add JDK

List of JDK installations on this system

Git

Git installations

Git

Name

Default

Path to Git executable ?

git

Install automatically ?

whereis git : executable path of git .

How to pull code from github ?

<https://github.com/ravdy/hello-world>

The screenshot shows the Jenkins configuration interface for a job named "PullCodeFromGitHub". The "Source Code Management" tab is selected. Under "Repositories", "Git" is chosen, and the "Repository URL" is set to "https://github.com/ravdy/hello-world.git". The "Branches to build" section shows a branch specifier of "*/*". A yellow circle highlights the "Add" button under "Credentials". A large black box at the bottom contains the text: "So, no need to provide Credentials, and anch".

The screenshot shows the Jenkins console output for a build step. It displays the command-line logs of a git clone operation from a GitHub repository. The logs show the repository being cloned, upstream changes being fetched, and a commit message being updated. A yellow circle highlights the commit message line: "Commit message: \"Update README.md\"". A large black box at the bottom contains the text: "I mean to say latest Commit message, alright."

```
Started by user admin
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/PullCodeFromGitHub
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/ravdy/hello-world.git
> git init /var/lib/jenkins/workspace/PullCodeFromGitHub # timeout=10
Fetching upstream changes from https://github.com/ravdy/hello-world.git
> git -version # timeout=10
> git -version # 'git version 2.32.0'
> git fetch --tags --force --progress -- https://github.com/ravdy/hello-world.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/ravdy/hello-world.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision e5dec5e8859c5afa286d3231c6ef89bdfa6d0 (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f e5dec5e8859c5afa286d3231c6ef89bdfa6d0 # timeout=10
Commit message: "Update README.md"
First time build. Skipping changelog
Finished: SUCCESS
```

Code is clone in Jenkins workspace .

52.91.26.146 (ec2-user)

```
[root@jenkins-server ~]# cd /var/lib/jenkins/workspace/
[root@jenkins-server ~]# pwd
/root
[root@jenkins-server ~]# cd PullCodeFromGitHub
[root@jenkins-server workspace]# cd PullCodeFromGitHub
[root@jenkins-server PullCodeFromGitHub]# ll
total 16
drwxr-xr-x 2 jenkins jenkins 6 Nov 7 04:41 HelloWorldJob
drwxr-xr-x 5 jenkins jenkins 96 Nov 7 05:21 PullCodeFromGitHub
[root@jenkins-server workspace]# cd PullCodeFromGitHub
[root@jenkins-server PullCodeFromGitHub]# ll
total 16
-rw-r--r-- 1 jenkins jenkins 133 Nov 7 05:21 Dockerfile
-rw-r--r-- 1 jenkins jenkins 5970 Nov 7 05:21 pom.xml
-rw-r--r-- 1 jenkins jenkins 271 Nov 7 05:21 README.md
drwxr-xr-x 3 jenkins jenkins 32 Nov 7 05:21 server
drwxr-xr-x 3 jenkins jenkins 32 Nov 7 05:21 webapp
[root@jenkins-server PullCodeFromGitHub]#
```

default, Jenkins is going to store all the

UNREGISTERED VERSION - Please support Mobaxterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

How to build code ?

Integrate Maven with Jenkins

- Setup Maven on Jenkins Server
- Setup Environment Variables
 - JAVA_HOME, M2, M2_HOME
- Install Maven Plugin
- Configure Maven and Java



Jenkins

Build Code

maven

in the Maven official documentation,

User Centre

Plugin Developer Centre

Maven Central Repository

Maven Developer Centre

Books and Resources

Security

COMMUNITY

Community Overview

Project Roles

How to Contribute

Getting Help

Issue Management

Getting Maven Source

The Maven Team

PROJECT DOCUMENTATION

Project Information

MAVEN PROJECTS

Archetype

Artifact Resolver

Doxia

JXR

Maven

Parent POMs

Plugins

Plugin Testing

Plugin Tools

Resource Bundles

Files

Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the [installation instructions](#). Use a source archive if you intend to build Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to [verify the signature](#) of the release bundles against the public [KEYS](#) used by the Apache Maven developers.

Link	Checksums	Signature
Binary tar.gz archive	apache-maven-3.8.3-bin.tar.gz	apache-maven-3.8.3-bin.tar.gz.sha512
Binary zip archive	apache-maven-3.8.3-bin.zip	apache-maven-3.8.3-bin.zip.sha512
Source tar.gz archive	apache-maven-3.8.3-src.tar.gz	apache-maven-3.8.3-src.tar.gz.sha512
Source zip archive	apache-maven-3.8.3-src.zip	apache-maven-3.8.3-src.zip.sha512

▪ Release Notes
▪ Reference Documentation
▪ Apache Maven Website As Documentation Archive
▪ All current release sources (plugins, shared libraries,...) available at <https://downloads.apache.org/maven/>
▪ latest source code from source repository
▪ Distributed under the Apache License, version 2.0

Previous Releases

It is strongly recommended to use the latest release version of Apache Maven to take advantage of newest features and bug fixes.

If you still want to use an old version you can find more information in the [Maven Releases History](#) and can download files from the [archives](#) for versions 3.0.4+ and legacy archives for earlier releases.

and go back to our Jenkins server.

We will install maven in Jenkins server .

```
[root@jenkins-server ~]# 
[root@jenkins-server ~]# 
[root@jenkins-server ~]# pwd
/root
[root@jenkins-server ~]# cd /opt
[root@jenkins-server opt]# ll
total 0
drwxr-xr-x 4 root root 33 Oct 11 21:54 aws
drwxr-xr-x 2 root root 6 Aug 16 2018 rh
[root@jenkins-server opt]# wget https://dlcdn.apache.org/maven/maven-3/3.8.3/binaries/apache-maven-3.8.3-bin.tar.gz
--2021-11-07 05:41:07-- https://dlcdn.apache.org/maven/maven-3/3.8.3/binaries/apache-maven-3.8.3-bin.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9042049 (8.6M) [application/x-gzip]
Saving to: 'apache-maven-3.8.3-bin.tar.gz'

100%[=====] 9,042,049  --.-K/s   in 0.08s
2021-11-07 05:41:07 (103 MB/s) - 'apache-maven-3.8.3-bin.tar.gz' saved [9042049/9042049]

[root@jenkins-server opt]# 
```

I hope wget is installed,

UNREGISTERED VERSION - Please support Mobaxterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

52.91.26.146 (ec2-user)

Terminal Sessions View X server Tools Games Settings Macros Help

Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help

Quick connect... 2 52.91.26.146 (ec2-user)

Sessions 2 52.91.26.146 (ec2-user)

Macros Tools Help

```
drwxr-xr-x 4 root root 33 Oct 11 21:54 aws
drwxr-xr-x 2 root root 6 Aug 16 2018 rh
[root@jenkins-server opt]# mv apache-maven-3.8.3 maven
[root@jenkins-server opt]# ll
total 8832
-rw-r--r-- 1 root root 9042049 Sep 27 18:48 apache-maven-3.8.3-bin.tar.gz
drwxr-xr-x 4 root root 33 Oct 11 21:54 aws
drwxr-xr-x 6 root root 99 Nov 7 05:41 maven
drwxr-xr-x 2 root root 6 Aug 16 2018 rh
[root@jenkins-server opt]# cd maven
[root@jenkins-server maven]# ll
total 36
drwxr-xr-x 2 root root 97 Nov 7 05:41 bin
drwxr-xr-x 2 root root 76 Nov 7 05:41 boot
drwxr-xr-x 3 root root 63 Sep 27 18:25 conf
drwxr-xr-x 4 root root 4096 Nov 7 05:41 lib
-rw-r--r-- 1 root root 17800 Sep 27 18:25 LICENSE
-rw-r--r-- 1 root root 5141 Sep 27 18:25 NOTICE
-rw-r--r-- 1 root root 2612 Sep 27 18:25 README.txt
[root@jenkins-server maven]# cd bin
[root@jenkins-server bin]# ll
total 32
-rw-r--r-- 1 root root 228 Sep 27 18:25 m2.conf
-rwxr-xr-x 1 root root 5761 Sep 27 18:25 mvn
-rw-r--r-- 1 root root 6223 Sep 27 18:25 mvn.cmd
-rwxr-xr-x 1 root root 1601 Sep 27 18:25 mvnDebug
-rw-r--r-- 1 root root 2082 Sep 27 18:25 mvnDebug.cmd
-rwxr-xr-x 1 root root 1532 S
[root@jenkins-server bin]# █ command of maven, nothing but mvn -v
```

UNREGISTERED VERSION - Please support MobatecTerm by subscribing to the professional edition here: <https://mobatexterm.mobatek.net>

```
-bash: mvn: command not found
[root@jenkins-server bin]# ./mvn -v
Apache Maven 3.8.3 (ff8e977a158738155dc465c6a97ffaf31982d739)
Maven home: /opt/maven
Java version: 11.0.12, vendor: Red Hat, Inc., runtime: /usr/lib/jvm/java-11-openjdk-11.0.12.0.7-0.amzn2.0.2.x86_64
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "4.14.248-189.473.amzn2.x86_64", arch: "amd64", family: "unix"
```

Add maven path to bash profile

Find where our jvm is installed .

```
[root@jenkins-server ~]# find / -name jvm  
/etc/jvm  
/usr/lib/jvm  
/usr/share/jvm  
[root@jenkins-server ~]# cd /usr/lib/jvm  
[root@jenkins-server jvm]# █
```

```
[root@jenkins-server jvm]# find / -name java-11
[root@jenkins-server jvm]# find / -name java-11*
/usr/lib/jvm/java-11-openjdk-11.0.12.0.7-0.amzn2.0.2.x86_64
```

The screenshot shows a terminal window titled '52.91.26.146 (ec2-user)'. The terminal displays the contents of the .bash_profile file. It includes environment variable assignments for M2_HOME, JAVA_HOME, and PATH, along with aliases and functions. A message at the bottom of the terminal says 'Okay, let's save this file.'

```

# .bash_profile
# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi
M2_HOME=/opt/maven
M2=/opt/maven/bin
JAVA_HOME=/usr/lib/jvm/java-11-openjdk-11.0.12.0.7-0.amzn2.0.2.x86_64
# User specific environment and startup programs

PATH=$PATH:$HOME/bin:$JAVA_HOME:$M2_HOME:$M2

export PATH

```

-- INSERT --

UNREGISTERED VERSION - Please support Mobaxterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Path will not take new value unless server reloaded so we used source command .

The screenshot shows a terminal window on a Jenkins server. It runs the command 'echo \$PATH' to show the current path, then 'source .bash_profile' to reload the profile, and finally 'mvn -v' to print the Maven version. The output shows Apache Maven 3.8.3 and Java version 11.0.12.

```

[root@jenkins-server ~]# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin
[root@jenkins-server ~]# source .bash_profile
[root@jenkins-server ~]# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin:/usr/lib/jvm/java-11-openjdk-11.0.12.0.7-0.amz
6_64:/opt/maven:/opt/maven/bin
[root@jenkins-server ~]# mvn -v
Apache Maven 3.8.3 (ff8e977a158738155dc465c6a97ffaf31982d739)
Maven home: /opt/maven
Java version: 11.0.12, vendor: Red Hat, Inc., runtime: /usr/lib/jvm/java-11-openjdk-11.0.12.0.7-0.amzn2.0.2.x86_64
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "4.14.248-189.473.amzn2.x86_64", arch: "amd64", family: "unix"

```

The screenshot shows the Jenkins Plugin Manager interface. A search bar at the top contains 'maven'. Below it, tabs for 'Updates', 'Available', 'Installed', and 'Advanced' are visible, with 'Available' selected. A table lists several Jenkins plugins, with the 'Maven Integration' plugin highlighted. This plugin is described as providing deep integration between Jenkins and Maven. It has two versions listed: 3.15.1 (3 days 10 hr ago) and 3.8.1 (4 mo 2 days ago). A note indicates the plugin is up for adoption. Other listed plugins include 'Jira' and 'Pipeline Maven Integration'.

Name	Version	Released
Maven Integration	3.15.1	3 days 10 hr ago
Config File Provider	3.8.1	4 mo 2 days ago
Jira	3.6	1 mo 28 days ago
Pipeline Maven Integration	3.10.0	8 mo 26 days ago

Select this one, and Install without restart,

Use above plugin .

Add jdk path :

The screenshot shows the Jenkins Global Tool Configuration page for JDK. A yellow circle highlights the 'JAVA_HOME' field, which contains the value '/usr/lib/jvm/java-11-openjdk-11.0.12.0.7-0.amzn2.0.2.x86_64'. Below this field is a checkbox labeled 'Install automatically'.

JDK

JDK installations

Add JDK

JDK

Name

java-11

JAVA_HOME

/usr/lib/jvm/java-11-openjdk-11.0.12.0.7-0.amzn2.0.2.x86_64

Install automatically

? **Delete JDK**

Add JDK

List of JDK installations on this system

Git

Git installations

Git

and go back to our Jenkins home directory.

Save **Apply**

Add maven path :

The screenshot shows the Jenkins Global Tool Configuration page for Maven. A yellow circle highlights the 'MAVEN_HOME' field, which contains the value '/opt/maven'. Below this field is a checkbox labeled 'Install automatically'.

Maven

Maven installations

Add Maven

Maven

Name

maven-3.8.3

MAVEN_HOME

/opt/maven

Install automatically

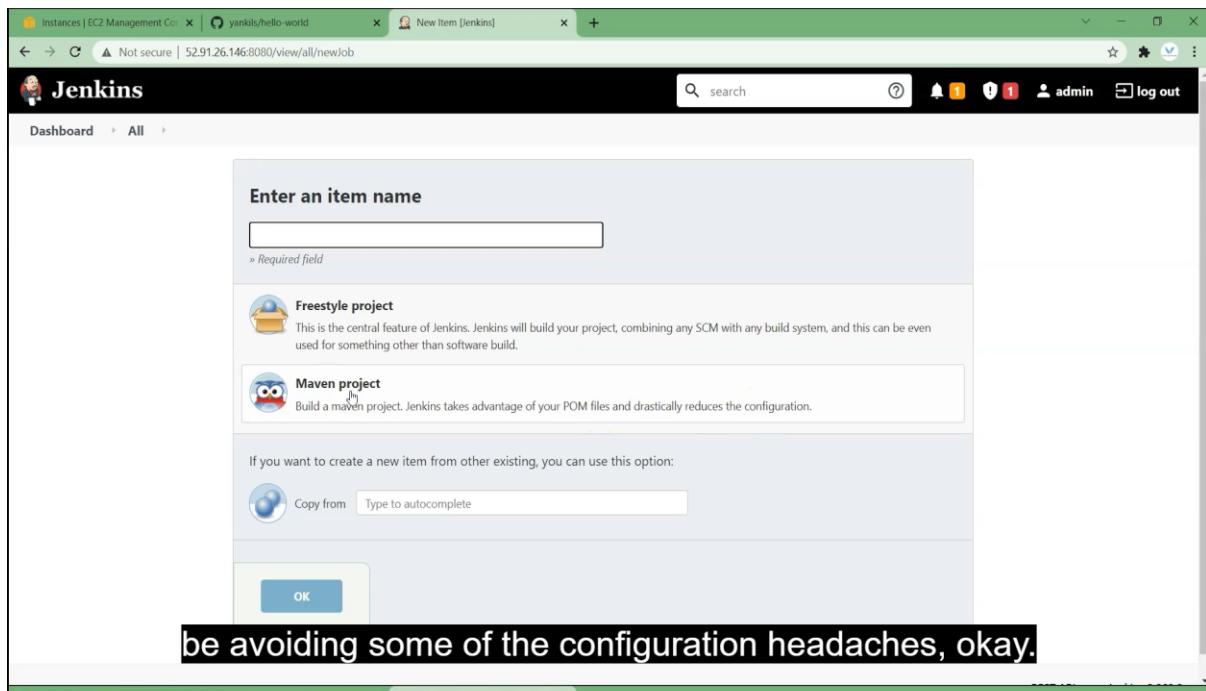
? **Delete Maven**

Add Maven

List of Maven installations on this system

Save **Apply** **That is what even we have**

Lets create new maven project .



The screenshot shows the 'Source Code Management' configuration page for a 'FirstMavenProject'. The 'Source Code Management' tab is selected. Under 'Repositories', the 'Git' option is chosen, and the 'Repository URL' is set to 'https://github.com/ravdy/hello-world.git'. A red error message 'Please enter Git repository.' is displayed next to the URL input field. The 'Credentials' section shows a dropdown menu with 'none' selected and an 'Add' button. A yellow circle highlights the 'Add' button. At the bottom, there are 'Save' and 'Apply' buttons, and a note 'Branches to build Branch Specification (blank for "main")'.

in any repository, and credentials are not required.

Build triggers :

The screenshot shows the Jenkins configuration interface for the 'FirstMavenProject'. The 'Build Triggers' tab is selected. Under 'Post-build Actions', there is a 'Repository browser' section with '(Auto)' selected. Below it, under 'Additional Behaviours', there is an 'Add' button. A yellow circle highlights this 'Add' button. The 'Build Triggers' section contains several options: 'Build whenever a SNAPSHOT dependency is built' (checked), 'Schedule build when some upstream has no successful builds' (unchecked), 'Trigger builds remotely (e.g., from scripts)' (unchecked), 'Build after other projects are built' (unchecked), 'Build periodically' (unchecked), 'GitHub hook trigger for GITScm polling' (unchecked), and 'Poll SCM' (unchecked). A yellow circle highlights the first option, 'Build whenever a SNAPSHOT dependency is built'. The 'Build Environment' and 'Pre Steps' sections are also visible. At the bottom, there are 'Save' and 'Apply' buttons.

The screenshot shows the Jenkins configuration interface for the 'FirstMavenProject'. The 'Build Triggers' tab is selected. Under 'Post-build Actions', there are two options: 'GitHub hook trigger for GITScm polling' (unchecked) and 'Poll SCM' (unchecked). A yellow circle highlights the 'GitHub hook trigger for GITScm polling' option. The 'Build Environment' and 'Pre Steps' sections are also visible. In the 'Build' section, the 'Root POM' field contains 'pom.xml'. The 'Goals and options' field is empty. A yellow circle highlights the 'pom.xml' field. At the bottom, there are 'Save' and 'Apply' buttons, and a note: 'Post S that is how the Maven projects does work.' with three radio button options: 'Run only if build succeeds' (unchecked), 'Run only if build succeeds or is unstable' (unchecked), and 'Run regardless of build result' (checked).

Pipeline will look for pom.xml file .

We have different maven goal .

A Build Lifecycle is Made Up of Phases

Each of these build lifecycles is defined by a different list of build phases, wherein a build phase represents a stage in the lifecycle.

For example, the default lifecycle comprises of the following phases (for a complete list of the lifecycle phases, refer to the [Lifecycle Reference](#)):

- `validate` - validate the project is correct and all necessary information is available
- `compile` - compile the source code of the project
- `test` - test the compiled source code using a suitable unit testing framework. These tests should not require the code be packaged or deployed
- `package` - take the compiled code and package it in its distributable format, such as a JAR
- `verify` - run any checks on results of integration tests to ensure quality criteria are met
- `install` - install the package into the local repository, for use as a dependency in other projects locally
- `deploy` - done in the build environment, copies the final package to the remote repository for sharing with other developers and projects.

Build Triggers

- GitHub hook trigger for GITScm polling
- Poll SCM

Build Environment

- Use secret text(s) or file(s)

Pre Steps

Add pre-build step ▾

Build

Root POM

pom.xml

Goals and options

clean install

If we run the job it will build the code . Maven will pull all dependency .

```
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/com/thoughtworks/xstream/xstream/1.3.1/xstream-1.3.1.jar (431 kB at 11 MB/s)
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.thoughtworks.xstream.core.util.Fields (file:/var/lib/jenkins/.m2/repository/com/thoughtworks/xstream/xstream/1.3.1/xstream-1.3.1.jar) to field java.util.Properties.defaults
WARNING: Please consider reporting this to the maintainers of com.thoughtworks.xstream.core.util.Fields
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Packaging webapp
[INFO] Assembling webapp [webapp] in [/var/lib/jenkins/workspace/FirstMavenProject/webapp/target/webapp]
[INFO] Processing war project
[INFO] Copying webapp resources [/var/lib/jenkins/workspace/FirstMavenProject/webapp/src/main/webapp]
[INFO] Webapp assembled in [37 ms]
[INFO] Building war: /var/lib/jenkins/workspace/FirstMavenProject/webapp/target/webapp.war
[INFO] WEB-INF/web.xml already added, skipping
[INFO]
[INFO] --- maven-install-plugin:2.4:install (default-install) @ webapp ---
[INFO] Installing /var/lib/jenkins/workspace/FirstMavenProject/webapp/target/webapp.war to /var/lib/jenkins/.m2/repository/com/example/maven-project/webapp/1.0-SNAPSHOT/webapp-1.0-SNAPSHOT.war
[INFO] Installing /var/lib/jenkins/workspace/FirstMavenProject/webapp/pom.xml to /var/lib/jenkins/.m2/repository/com/example/maven-project/webapp/1.0-SNAPSHOT/webapp-1.0-SNAPSHOT.pom
[INFO] -----
[INFO] Reactor Summary for Maven Project 1.0-SNAPSHOT:
[INFO]
[INFO] Maven Project ..... SUCCESS [ 4.120 s]
[INFO] Server ..... SUCCESS [ 8.064 s]
[INFO] Webapp ..... SUCCESS [ 1.681 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 16.777 s
[INFO] Finished at: 2021-11-07T06:03:14Z
[INFO] -----
[JENKINS] Archiving /var/lib/jenkins/workspace/FirstMavenProject/webapp/pom.xml to com.example.maven-project/webapp/1.0-SNAPSHOT/webapp-1.0-SNAPSHOT.pom
[JENKINS] Archiving /var/lib/jenkins/workspace/FirstMavenProject/webapp/target/webapp.war to com.example.maven-project/webapp/1.0-SNAPSHOT/webapp-1.0-SNAPSHOT.war
[JENKINS] Archiving /var/lib/jenkins/workspace/FirstMavenProject/server/pom.xml to com.example.maven-project/server/1.0-SNAPSHOT/server-1.0-SNAPSHOT.pom
[JENKINS] Archiving /var/lib/jenkins/workspace/FirstMavenProject/server/target/maven-project/1.0-SNAPSHOT/server-1.0-SNAPSHOT.jar to com.example.maven-project/server/1.0-SNAPSHOT/server-1.0-SNAPSHOT.jar
[JENKINS] Archiving /var/lib/jenkins/workspace/FirstMavenProject/maven-project/pom.xml to com.example.maven-project/maven-project/1.0-SNAPSHOT/maven-project-1.0-SNAPSHOT.pom
```

where it has been executed.

So this has created a war file for project .

All build artifact are stored in target .

Dashboard > FirstMavenProject >

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

Workspace of FirstMavenProject on master

webapp / target / [\[\]](#)

maven-archiver
surefire
webapp
webapp.war

Nov 7, 2021, 6:03:13 AM 2.47 KB [view](#)

[\(all files in zip\)](#)

```
graph LR; Git[git] -- "Commit Code" --> GitHub[GitHub]; GitHub -- "Pull Code" --> Jenkins[Jenkins]; Jenkins -- "Build Code" --> Maven[maven]
```

Deploy code on tomcat server .

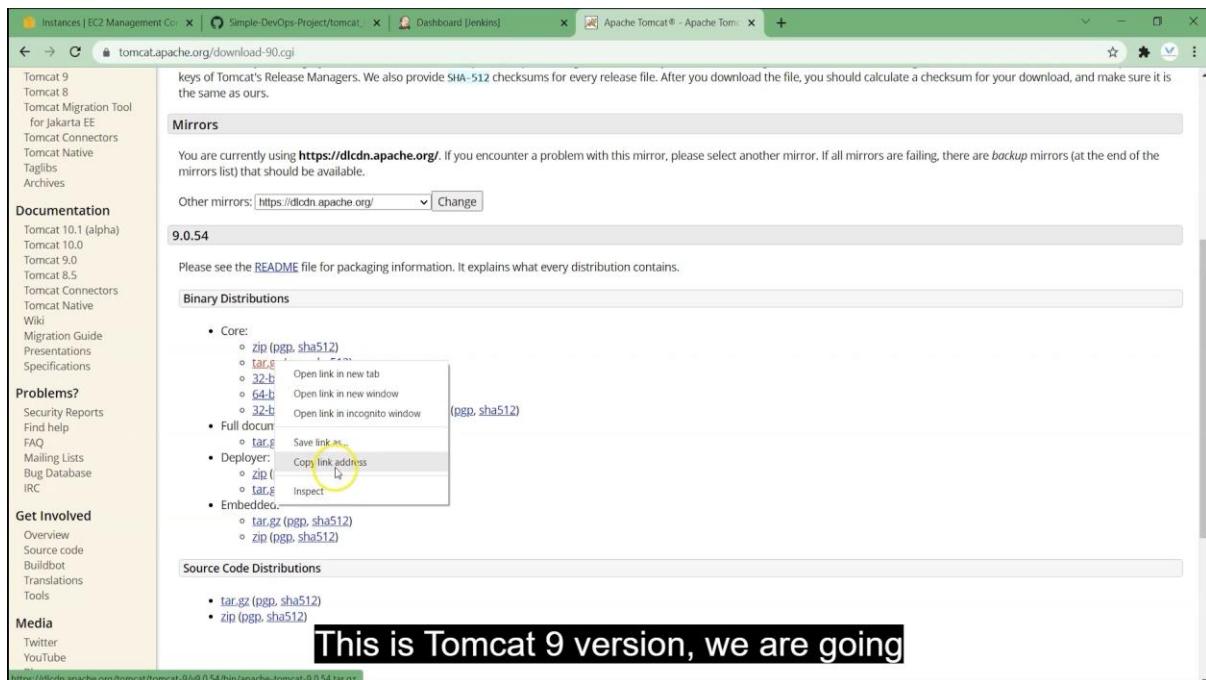
Setup Tomcat Server

- Setup a Linux EC2 Instance
- Install Java
- Configure Tomcat
- Start Tomcat Server
- Access Web UI on port 8080



steps in my GitHub repository,

https://github.com/yankils/Simple-DevOps-Project/blob/master/Tomcat/tomcat_installation.MD



Copy tomcat package .

Install java .

```
61 dnsmasq2.85           available [ =stable ]
[root@ip-172-31-81-112 ~]# java -version
openjdk version "11.0.12" 2021-07-20 LTS
OpenJDK Runtime Environment 18.9 (build 11.0.12+7-LTS)
OpenJDK 64-Bit Server VM 18.9 (build 11.0.12+7-LTS, mixed mode, sharing)
[root@ip-172-31-81-112 ~]# pwd
```

```
drwxr-xr-X 2 root root 6 Aug 16 2018 rn
[root@ip-172-31-81-112 opt]# wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.54/bin/apache-tomcat-9.0.54.tar.gz
--2021-11-07 08:17:57-- https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.54/bin/apache-tomcat-9.0.54.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11576317 (11M) [application/x-gzip]
Saving to: 'apache-tomcat-9.0.54.tar.gz'

100%[=====] 11,576,317 --.
2021-11-07 08:17:58 (244 MB/s) - 'apache-tomcat-9.0.54.tar.gz' saved [11576317/11576317]
```

Extract the file .

To start tomcat services :

```
-rwxr-x--- 1 root root 1908 Sep 28 13:51 startup.sh
[root@ip-172-31-81-112 bin]# ./startup.sh
Using CATALINA_BASE: /opt/tomcat
Using CATALINA_HOME: /opt/tomcat
Using CATALINA_TMPDIR: /opt/tomcat/temp
Using JRE_HOME: /usr
Using CLASSPATH: /opt/tomcat/bin/bootstrap.jar:/opt/tomcat/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
```

Find context.xml file path :

```
[root@ip-172-31-81-112 opt]# cd tomcat/
[root@ip-172-31-81-112 tomcat]# find / -name context.xml
/opt/tomcat/conf/context.xml
/opt/tomcat/webapps/examples/META-INF/context.xml
/opt/tomcat/webapps/host-manager/META-INF/context.xml
```

```
52.90.65.63 (ec2-user)
Terminal Sessions View Xserver Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect... 2.52.91.26.146 (ec2-user) 4 52.90.65.63 (ec2-user)
<?xml version="1.0" encoding="UTF-8"?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" >
    <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
        sameSiteCookies="strict" />
<!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127.\d{1,3}.\d{1,3}.\d{1,3}:[0-9]{2,5}" /> -->
    <Manager sessionAttributeValueClassNameFilter="java.lang.(?:Boolean|Integer|Long|Number|String)|org\\.apache\\.catalina\\.filters\\.CsrfPreventionFilter\\$LruCache(?:\$1)?|java\\.util\\.\\?(?:Linked)?HashMap"/>
</Context>
~
~
~
~
~
:~
```

Okay. So now, save this file, and same thing

Okay. So now, save this file, and same thing

UNREGISTERED VERSION - Please support Mobaxterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

52.90.65.63 (ec2-user)

Terminal Sessions View Xserver Tools Games Settings Macros Help

Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help

Quick connect... [2. 52.91.26.140 (ec2-user)] [3. 4. 52.90.65.63 (ec2-user)]

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->
<Context antiResourceLocking="false" privileged="true" >
    <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
        sameSiteCookies="strict" />
    <!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"
        allow="127\\.\\d{1,2}\\.\\d{1,2}\\.\\d{1,2}\\.\\d{1,2}|0:0:0:0:0:1" /> -->
    <Manager sessionAttributeValueClassNameFilter="java.lang.(?>Boolean|Integer|Long|Number|String)|org.apache\\.catalina\\.filters\\.CsrfPreventionFilter\$LruCache(?:\$1)?|java\\.util\\.\\(?>Linked)?HashMap"/>
</Context>
~
```

If you want to know how do I know?

If you want to know how do I know?

~~Comment out the part where it is allowing only internal address for communication~~

```
Comment out the part where it is allowing only internal address for communication .  
/opt/tomcat/webapps/manager/META-INF/context.xml  
[root@ip-172-31-81-112 tomcat]# vi /opt/tomcat/webapps/host-manager/META-INF/context.xml  
[root@ip-172-31-81-112 tomcat]# vi /opt/tomcat/webapps/manager/META-INF/context.xml
```

```

-rwxr-x--- 1 root root 1908 Sep 28 13:51 version.sh
[root@ip-172-31-81-112 bin]# ./shutdown.sh
Using CATALINA_BASE: /opt/tomcat
Using CATALINA_HOME: /opt/tomcat
Using CATALINA_TMPDIR: /opt/tomcat/temp
Using JRE_HOME: /usr
Using CLASSPATH: /opt/tomcat/bin/bootstrap.jar:/opt/tomcat/lib/*
Using CATALINA_OPTS:
NOTE: Picked up JDK_JAVA_OPTIONS: --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED
ED
[root@ip-172-31-81-112 bin]# ./startup.sh
Using CATALINA_BASE: /opt/tomcat
Using CATALINA_HOME: /opt/tomcat
Using CATALINA_TMPDIR: /opt/tomcat/temp
Using JRE_HOME: /usr

```

Instances | EC2 Management Consoles | Simple-DevOps-Project/tomcat | Dashboard [jenkins] | Apache Tomcat® - Apache Tomcat | Apache Tomcat/9.0.54

Not secure | 52.90.65.63:8080

Home Documentation Configuration Examples Wiki Mailing Lists Find Help

Apache Tomcat/9.0.54

If you're seeing this, you've successfully installed Tomcat. Congratulations!

Recommended Reading:

- [Security Considerations How-To](#)
- [Manager Application How-To](#)
- [Clustering/Session Replication How-To](#)

Developer Quick Start

- [Tomcat Setup](#)
- [First Web Application](#)
- [Realms & AAA](#)
- [JDBC DataSources](#)
- [Examples](#)
- [Servlet Specifications](#)
- [Tomcat Versions](#)

Managing Tomcat

For security, access to the manager webapp is restricted. Users are defined in \${CATALINA_HOME}/conf/tomcat-users.xml

In Tomcat 9.0 access to the manager application is split between different users. [Read more...](#)

Release Notes

- [Changelog](#)
- [Migration Guide](#)
- [Security Notices](#)

Documentation

- [Tomcat 9.0 Documentation](#)
- [Tomcat 9.0 Configuration](#)
- [Tomcat Wiki](#)

Find additional important configuration information in \${CATALINA_HOME}/RUNNING.txt

Developers may be interested in:

Tomcat 9.0 Git Repository at GitHub

Getting Help

[FAQ and Mailing Lists](#)

The following mailing lists are available:

- [tomcat-announce](#) Important announcements, releases, security vulnerability notifications. (Low volume).
- [tomcat-users](#) User support and discussion.
- [taglibs-user](#) User support and discussion for Apache Taglibs.

App, it should ask for the Credentials.

For login to manager app it asks for creds . So we have to add users in users.xml .

3. Update users information in the tomcat-users.xml file goto tomcat home directory and Add below users to conf/tomcat-users.xml

```

<role rolename="manager-gui"/>
<role rolename="manager-script"/>
<role rolename="manager-jmx"/>
<role rolename="manager-status"/>
<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>
<user username="deployer" password="deployer" roles="manager-script"/>
<user username="tomcat" password="s3cret" roles="manager-gui"/>

```

We are creating 3 users .

```

MacOS [root@ip-172-31-81-112 conf]# ln -s /opt/tomcat/bin/shutdown.sh /usr/local/bin/tomcatdown
[MacOS SFTP]# tomcatdown
Using CATALINA_BASE:   /opt/tomcat
Using CATALINA_HOME:   /opt/tomcat
Using CATALINA_TMPDIR: /opt/tomcat/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/tomcat/bin/bootstrap.jar:/opt/tomcat/bin/tomcat-juli.jar
Using CATALINA_OPTS:
NOTE: Picked up JDK_JAVA_OPTIONS: --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.util.concurrent=ALL-UNNAMED --add-opens=java.util.concurrent.forkjoin=ALL-UNNAMED
[root@ip-172-31-81-112 conf]# to
toe      tomcatdown tomcatup    top      touch
[root@ip-172-31-81-112 conf]# to
toe      tomcatdown tomcatup    top      touch
[root@ip-172-31-81-112 conf]# tomcatup
Using CATALINA_BASE:   /opt/tomcat
Using CATALINA_HOME:   /opt/tomcat
Using CATALINA_TMPDIR: /opt/tomcat/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/tomcat/bin/bootstrap.jar:/opt/tomcat/bin/tomcat-juli.jar
Using CATALINA_OPTS:

```

Lets deploy our application to tomcat server.

Integrate Tomcat with Jenkins

- Install “Deploy to container”
- Configure tomcat server with Credentials

Change password for admin user in Jenkins :

The view selected by default when navigating to the user's private views

Notification URL

Default

Password

Password:

Confirm Password:

SSH Public Keys

The screenshot shows the Jenkins Update Center interface. On the left sidebar, there are links: 'Back to Dashboard', 'Manage Jenkins', and 'Update Center'. The main area has a search bar with the text 'deploy to c'. Below it, tabs are labeled 'Updates', 'Available' (which is selected), 'Installed', and 'Advanced'. A sub-menu 'Install' is open under 'Available', showing a list item 'Deploy to container'. This item has a yellow circle around it. The description below says: 'This plugin allows you to deploy a war to a container after a successful build. Glassfish 3.x remote deployment'. At the bottom, there are two buttons: 'Install without restart' and 'Download now and install after restart', followed by a link 'Update information'.

Install plugin .

Lets add creds to deploy war file to tomcat server .

We will use deployer user of tomcat server .

The screenshot shows the Jenkins 'Global credentials (unrestricted)' configuration page. The navigation path is 'Dashboard > Credentials > System > Global credentials (unrestricted)'. On the left, there are links: 'Back to credential domains' and 'Add Credentials'. The main form is for adding a new credential. It has sections for 'Kind' (set to 'Username with password'), 'Scope' (set to 'Global (Jenkins, nodes, items, all child items, etc)'), 'Username' (set to 'deployer'), and 'Password' (represented by a series of dots). There is also a checkbox 'Treat username as secret' which is unchecked. Below these are sections for 'ID' (set to 'to') and 'Description' (an empty text area). At the bottom right is a blue 'OK' button.

Create maven job .

Enter an item name

BuildAndDeployJob

» Required field



Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining a used for something other than software build.



Maven project

Build a maven project. Jenkins takes advantage of your POM files and drastically

If you want to create a new item from other existing, you can use this option:



Copy from

Type to autocomplete



and let it be Maven p

<https://github.com/yankils/hello-world>

Pre Steps

Add pre-build step ▾

Build

Root POM

pom.xml

Goals and options

clean **install**

Clean and install goals will clean package build test the code .

In post build actions .

The screenshot shows the Jenkins 'Post Steps' configuration screen. At the top, there are three radio button options: 'Run only if build succeeds' (unchecked), 'Run only if build succeeds or is unstable' (unchecked), and 'Run regardless of build result' (checked). Below these options is a list of available actions. One action, 'Deploy war/ear to a container', is highlighted with a yellow circle. The 'Add post-build action ▾' button is visible at the bottom left of the list.

The screenshot shows the configuration for the 'Deploy war/ear to a container' action. It includes fields for 'WAR/EAR files' (containing '**/*.war) and 'Tomcat URL' (containing http://52.90.65.63:8080). A note below states: 'The path states that where ever it will find .war file it will deploy to container .'.

The screenshot shows the expanded configuration for the 'Deploy war/ear to a container' action. It includes sections for 'Containers' (with 'Tomcat 8.x Remote' selected), 'Credentials' (with 'deployer/***** (tomcat deployer)'), 'Tomcat URL' (containing http://52.90.65.63:8080), and an 'Add Container' button. There is also a checkbox for 'Deploy on failure' and an 'Add post-build action ▾' button at the bottom.

```
[JENKINS] Archiving /var/lib/jenkins/workspace/BuildAndDeployJob/pom.xml to com.example.maven-project/maven-project/1.0-SNAPSHOT/maven-project-1.0-SNAPSHOT.jnlp
channel stopped
[DeployPublisher][INFO] Attempting to deploy 1 war file(s)
[DeployPublisher][INFO] Deploying [/var/lib/jenkins/workspace/BuildAndDeployJob/webapp/target/webapp.war] to container Tomcat 8.x Remote with context null
[/var/lib/jenkins/workspace/BuildAndDeployJob/webapp/target/webapp.war] is not deployed. Doing a fresh deployment.
Deploying [/var/lib/jenkins/workspace/BuildAndDeployJob/webapp/target/webapp.war]
Finished: SUCCESS
```

When we build pipeline it will deploy war file to tomcat server .
In tomcat our webapp is created .

 <i>/webapp</i>	<i>None specified</i>
---	-----------------------

← → ⌂ ▲ Not secure | 52.90.65.63:8080/webapp/

Hello, Welcome to Simple DevOps Project !!

Deploying on a kubernetes using ansible for Valaxy Technologies

Glad to see you here



```

graph LR
    Git[git icon] -- Commit Code --> GitHub[GitHub icon]
    GitHub -- Pull Code --> Jenkins[Jenkins icon]
    Jenkins -- Deploy Code --> Tomcat[Tomcat Server icon]
    Jenkins -- Build Code --> Maven[maven icon]
  
```

The diagram illustrates a continuous integration and deployment pipeline. It starts with a developer committing code to a GitHub repository. Jenkins then pulls the code from GitHub. Finally, Jenkins deploys the code to a Tomcat server. Additionally, Jenkins triggers a build process using Maven.

Build trigger :

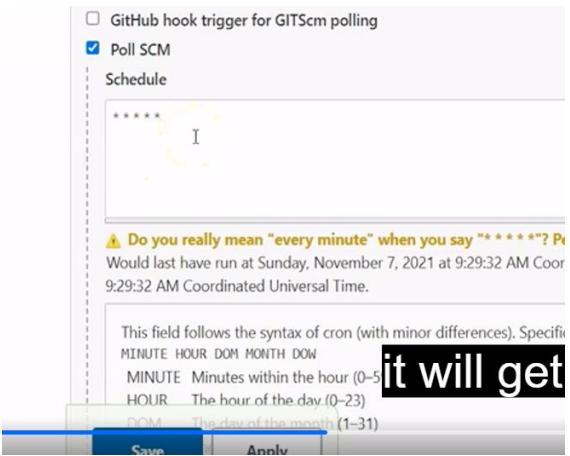
Build Triggers

- Build whenever a SNAPSHOT dependency is built
 - Schedule build when some upstream has no successful builds
- Trigger builds remotely (e.g., from scripts)
- Build after other projects are built
- Build periodically
- GitHub hook trigger for GITScm polling
- Poll SCM

Trigger build remotely – when we want to build in some remote server .

Build periodically – cron jobs .

Poll scm – we will schedule time in which it will check whether any change in code and based on that build will happen .



It will check every minute for changes .

Integrating docker :



Deploy to docker container .

Setup Docker Host

- Setup a Linux EC2 Instance
- Install docker
- Start docker services
- Basic docker commands



my simple DevOps project, GitHub repository.

https://github.com/yankils/Simple-DevOps-Project/blob/master/Docker/Docker_Installation_Steps.MD

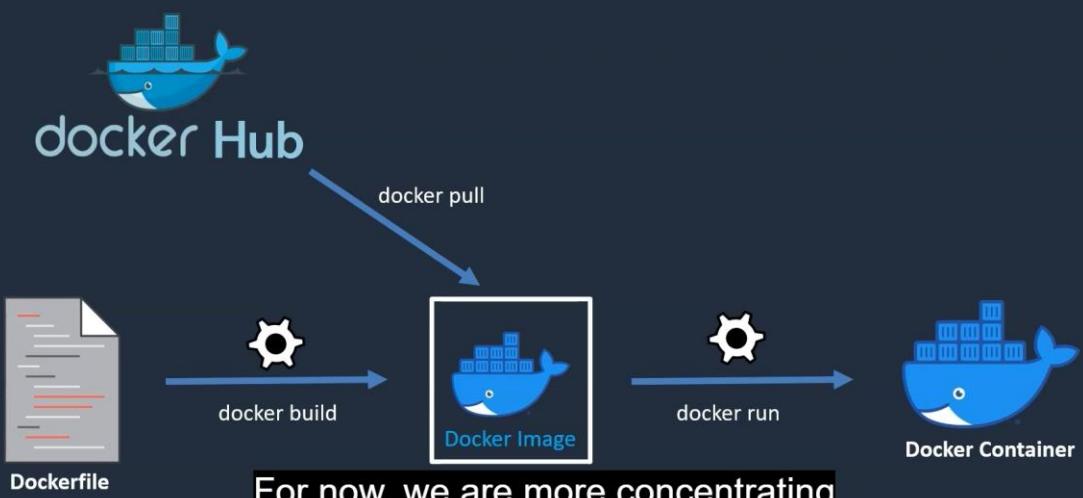
```

Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-21-252 ~]$ sudo su -
[root@ip-172-31-21-252 ~]# yum install docker -y
  
```

```
[root@ip-172-31-21-252 ~]# service docker status
Redirecting to /bin/systemctl status docker.service
● docker.service - Docker Application Container Engine
  Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; vendor preset: disabled)
  Active: inactive (dead)

[root@ip-172-31-21-252 ~]# service docker start
Redirecting to /bin/systemctl start docker.service
[root@ip-172-31-21-252 ~]# service docker status
Redirecting to /bin/systemctl status docker.service
● docker.service - Docker Application Container Engine
  Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; vendor preset: disabled)
  Active: active (running) since Mon 2021-11-08 05:53:52 UTC; 2s ago
    Docs: https://docs.docker.com
   Process: 3597 ExecStartPre=/usr/libexec/docker/docker-setup-runtimes
   Process: 3587 ExecStartPre=/bin/mkdir -p /run/docker (code=exited, s
Hint: Some lines were ellipsized, use -l to see them all.
[root@ip-172-31-21-252 ~]# docker images
REPOSITORY      TAG          IMAGE ID      CREATED      SIZE
[root@ip-172-31-21-252 ~]# docker ps
CONTAINER ID    IMAGE        COMMAND      CREATED      STATUS
[root@ip-172-31-21-252 ~]# docker ps -a
```

How to create docker container



```

[root@dockerhost ~]# service docker start
Redirecting to /bin/systemctl start docker.service
[root@dockerhost ~]# docker pull tomcat
Using default tag: latest
latest: Pulling from library/tomcat
bb7d5a84853b: Extracting [=====]
f02b617c6a8c: Download complete
d32e17419b7e: Download complete
c9d2d81226a4: Download complete
fab4960f9cd2: Download complete
da1c1e7baf6d: Download complete
1d2ade66c57e: Downloading [=====]
ea2ad3f7cb7c: Download complete
d75cb8d0a5ae: Download complete
[6cRer!f6f~f~nParv}~nlicm~lates
[root@dockerhost ~]# docker images
REPOSITORY      TAG          IMAGE ID      CREATED        SIZE
tomcat          latest        b0e0b0a92cf9   2 weeks ago   680MB

```

```

[root@dockerhost ~]# docker images
REPOSITORY      TAG          IMAGE ID      CREATED        SIZE
tomcat          latest        b0e0b0a92cf9   2 weeks ago   680MB
[root@dockerhost ~]# docker run -d --name tomcat-container -p 8081:8080 tomcat
[root@dockerhost ~]# docker ps -a
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
[root@dockerhost ~]# docker run -d --name tomcat-container -p 8081:8080 tomcat
723062882fbf9e1f80ca885efa5b41058648304248908b5c0dab501cdab1b5bd
[root@dockerhost ~]# docker ps -a
-bash: dokcer: command not found
[root@dockerhost ~]# docker ps -a
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS
ES
723062882fbf    tomcat    "catalina.sh run"  15 seconds ago  Up 14 seconds  0.0.0.0:8081->8080/tcp,
tomcat-container

```

Inbound rules Info					
Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	
sgr-0463f507448993145	SSH	TCP	22	Custom ▾	<input type="text" value="0.0.0.0"/> X
sgr-0bfbb5d215b261ce0	Custom TCP	TCP	8080	Custom ▾	<input type="text" value="0.0.0.0"/> X
-	Custom TCP	TCP	8081 - 9001	Anywh... ▾	<input type="text" value="0.0.0.0"/> X

Open port .

Our tomcat server is giving 400 error .

Login inside container .

```

root@dockerhost ~]# docker exec -it tomcat-container /bin/bash
root@723062882fbf:/usr/local/tomcat# ls
LICENSE.txt      LICENSE      README.md      RUNNING.txt      conf      logs      temp      webapps.dist
NOTICE          NOTICE      RELEASE-NOTES      bin      lib      native-jni-lib      webapps      work

```

All webapp related folder are present in different directory so copy those to webapp .

```

root@723062882fbf:/usr/local/tomcat/webapps.dist# ls
ROOT  docs  examples  host-manager  manager

```

```

root@docs examples host-manager manager
root@723062882fbf:/usr/local/tomcat/webapps.dist# cp -R * ../webapps/
root@723062882fbf:/usr/local/tomcat/webapps.dist# cd ../webapps
root@723062882fbf:/usr/local/tomcat/webapps# ll
bash: ll: command not found
root@723062882fbf:/usr/local/tomcat/webapps# ls
ROOT docs examples host-manager manager
root@723062882fbf:/usr/local/tomcat/webapps# exit

```

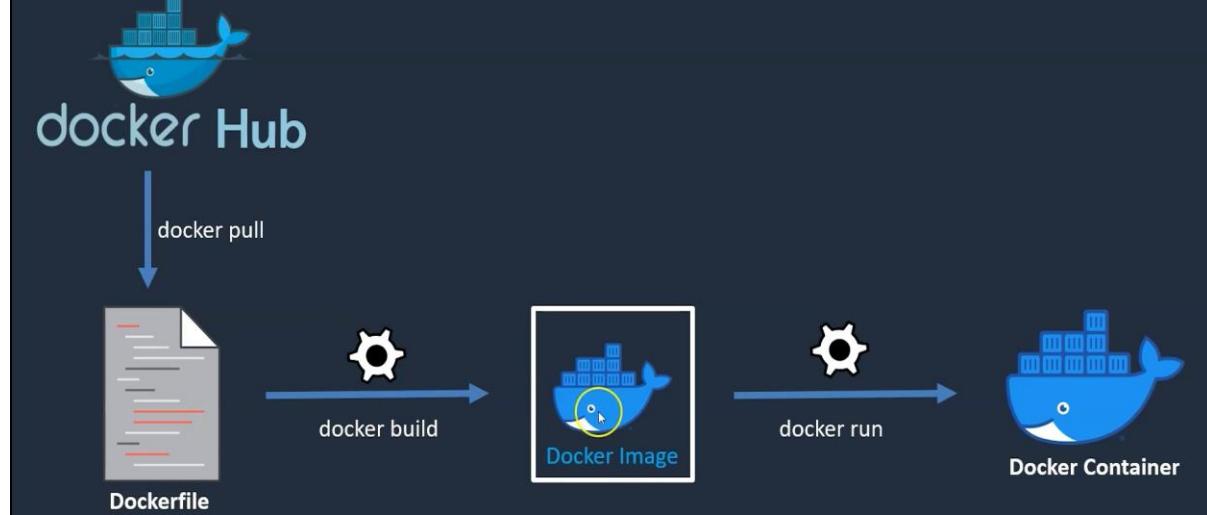
Now tomcat will be accessible .

```

[root@dockerhost ~]# docker ps -a
CONTAINER ID   IMAGE      COMMAND          CREATED
ES
723062882fbf  tomcat     "catalina.sh run"  47 minutes ago
cat-container
[root@dockerhost ~]# docker stop tomcat-container
tomcat-container
[root@dockerhost ~]# docker ps -a
CONTAINER ID   IMAGE      COMMAND          CREATED
723062882fbf  tomcat     "catalina.sh run"  48 minutes ago
[root@dockerhost ~]# docker ps
CONTAINER ID   IMAGE      COMMAND          CREATED          STATUS    PORTS
NAMES
56fb48cc3135  tomcat:latest "catalina.sh run"  4 seconds ago  Up 3 seconds  0.0.0.0:8082->

```

Lets run 1 more container similar steps we have to do by moving files .



Write dockerfile

Write Your 1st Docker File



Dockerfile

- **FROM:** To pull the base image
- **RUN:** To execute commands
- **CMD:** To provide defaults for an executing container
- **ENTRYPOINT:** To configure a container that will run as an executable
- **WORKDIR:** To sets the working directory
- **COPY:** To copy a directory from your local machine to the docker container
- **ADD:** To copy files and folders from your local machine to docker containers
- **EXPOSE:** Informs Docker that the container listens on the specified network ports at runtime
- **ENV:** To set environment variables

Install tomcat on Centos

- | | |
|--|-------------------|
| • Pull centos from dockerhub | - FROM |
| • Install java | - RUN |
| • Create /opt/tomcat directory | - RUN |
| • Change work directory to /opt/tomcat | - WORKDIR |
| • Download tomcat packages | - ADD /RUN |
| • Extract tar.gz file | - RUN |
| • Rename to tomcat directory | - RUN |
| • Tell to docker that it runs on port 8080 | - EXPOSE |
| • Start tomcat services | - CMD |

Starting Tomcat: The script can be used to start the Tomcat server by running `./catalina.sh start` or `./catalina.sh run`. The latter option starts Tomcat in the current terminal session, allowing you to see the console output.

Stopping Tomcat: To stop the Tomcat server, you can use `./catalina.sh stop`. This command initiates a graceful shutdown of Tomcat, allowing it to finish processing current requests before shutting down.

```
FROM centos:latest
RUN yum install java -y
RUN mkdir /opt/tomcat
WORKDIR /opt/tomcat
ADD https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.54/bin/apache-tomcat-9.0.54.tar.gz .
RUN tar -xvzf apache-tomcat-9.0.54.tar.gz
RUN mv apache-tomcat-9.0.54/* /opt/tomact
EXPOSE 8080
CMD ["/opt/tomcat/bin/catalina.sh", "run"]
~ [root@dockerhost ~]# docker build -t mytomcat .
Sending build context to Docker daemon 10.24kB
Step 1/9 : FROM centos:latest
latest: Pulling from library/centos
a1d0c7532777: Extracting [=====]

```

Lets create container using this image .

```
[root@dockerhost ~]# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
mytomcat        latest   351cad176bf1  46 seconds ago  535MB
tomcat          latest   b0e0b0a92cf9  2 weeks ago    680MB
centos          latest   5d0da3dc9764  7 weeks ago    231MB
[root@dockerhost ~]# docker run -d --name mytomcat-server -p 8083:8080 mytomcat
e67b013390e405732dbf2233184b665f38e97a23c9a2f5d6496184ea32e1a416
[root@dockerhost ~]# docker ps -a
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS
NAMES
e67b013390e4      mytomcat      "/opt/tomcat/bin/cat..."  7 seconds ago  Up 6 sec
:8083 -> 8080/tcp  mytomcat server
```