

## Jenkins Assignment 2

Owner	Sanket Chaudhari
Tags	Guides and Processes

### CI / CD Pipeline Practical 21 Feb 2024 : Assignment 2 :

- Create Build Tool Folder to download maven in that -

```

aws | Services | Search [Alt+S]
EFS VPC EC2 Simple Notification Service CloudTrail CloudWatch IAM
[root@ip-172-31-8-221 apache-maven-3.9.6]#
[root@ip-172-31-8-221 apache-maven-3.9.6]#
[root@ip-172-31-8-221 apache-maven-3.9.6]#
[root@ip-172-31-8-221 apache-maven-3.9.6]# pwd
/mnt/build-tool/apache-maven-3.9.6
[root@ip-172-31-8-221 apache-maven-3.9.6]#
  
```

- Downloaded particular version of maven -

	Link	Checksums	Signature
Binary tar.gz archive	<a href="#">apache-maven-3.9.6-bin.tar.gz</a>	<a href="#">apache-maven-3.9.6-bin.tar.gz.sha512</a>	<a href="#">apache-maven-3.9.6-bin.tar.gz.asc</a>
Binary zip archive	<a href="#">apache-maven-3.9.6-bin.zip</a>	<a href="#">apache-maven-3.9.6-bin.zip.sha512</a>	<a href="#">apache-maven-3.9.6-bin.zip.asc</a>
Source tar.gz archive	<a href="#">apache-maven-3.9.6-src.tar.gz</a>	<a href="#">apache-maven-3.9.6-src.tar.gz.sha512</a>	<a href="#">apache-maven-3.9.6-src.tar.gz.asc</a>
Source zip archive	<a href="#">apache-maven-3.9.6-src.zip</a>	<a href="#">apache-maven-3.9.6-src.zip.sha512</a>	<a href="#">apache-maven-3.9.6-src.zip.asc</a>

- [3.9.6 Release Notes](#) and [Release Reference Documentation](#)
- latest source code from source repository
- Distributed under the [Apache License](#), version 2.0
- other:

## How to create environment variable ?

- go to **jenkins home** and edit profile file which is **/root/.bash\_profile**
- Edit This File **Vi .bash\_profile**
- Add the following Command And maven home path

```
export MAVEN_HOME=/mnt/build-tools/apache-maven-3.9.6
export PATH=$MAVEN_HOME/bin:$PATH
```

```
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/bin

export MAVEN_HOME=/mnt/build-tools/apache-maven-3.9.6
export PATH=$MAVEN_HOME/bin:$PATH
export PATH
~
```

- Now **Login & Logout** Your ec2 instance.
- **Now You can run mavn from any folder**

```
aws | Services | Search [Alt+S]
EFS VPC EC2 Simple Notification Service CloudTrail CloudWatch IAM

[ec2-user@ip-172-31-8-221 ~]$
[ec2-user@ip-172-31-8-221 ~]$
[ec2-user@ip-172-31-8-221 ~]$ sudo su -
Last login: Tue Feb 27 07:13:19 UTC 2024 on pts/0
[root@ip-172-31-8-221 ~]# mvn
[INFO] Scanning for projects...
[INFO] -----
[INFO] BUILD FAILURE
[INFO] -----
[INFO] Total time: 0.251 s
[INFO] Finished at: 2024-02-27T07:20:47Z
[INFO] -----
[ERROR] No goals have been specified for this build. You must specify a valid lifecycle phase or a goal in the
maven-artifact-id[:<plugin-version>]:<goal>. Available lifecycle phases are: pre-clean, clean, post-clean, validate,
generate-resources, process-resources, compile, process-classes, generate-test-sources, process-test-sources, generate-test-classes, test, prepare-package, package, pre-integration-test, integration-test, post-integration-test, site, site-deploy. -> [Help 1]
[ERROR]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]
[ERROR] For more information about the errors and possible solutions, please read the following articles:
[ERROR] [Help 1] http://cwiki.apache.org/confluence/display/MAVEN/NoGoalSpecifiedException
[root@ip-172-31-8-221 ~]# clear
```

- Now Restart Jenkins for Safer side

```
root@ip-172-31-8-221 ~]# cd /mnt/server/
root@ip-172-31-8-221 server]# cd apache-tomcat-9.0.85/
root@ip-172-31-8-221 apache-tomcat-9.0.85]# cd bin
root@ip-172-31-8-221 bin]# ./shutdown.sh
Using CATALINA_BASE:   /mnt/server/apache-tomcat-9.0.85
Using CATALINA_HOME:   /mnt/server/apache-tomcat-9.0.85
Using CATALINA_TMPDIR: /mnt/server/apache-tomcat-9.0.85/temp
Using JRE_HOME:        /usr
Using CLASSPATH:        /mnt/server/apache-tomcat-9.0.85/bin/bootstrap.jar:/mnt/server/apache-tomcat-9.0.85/bin/tomcat-juli.jar
NOTE: Picked up JDK_JAVA_OPTIONS:  --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.util.concurrent=ALL-UNNAMED --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED
root@ip-172-31-8-221 bin]# ./startup.sh
Using CATALINA_BASE:   /mnt/server/apache-tomcat-9.0.85
Using CATALINA_HOME:   /mnt/server/apache-tomcat-9.0.85
Using CATALINA_TMPDIR: /mnt/server/apache-tomcat-9.0.85/temp
Using JRE_HOME:        /usr
Using CLASSPATH:        /mnt/server/apache-tomcat-9.0.85/bin/bootstrap.jar:/mnt/server/apache-tomcat-9.0.85/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
root@ip-172-31-8-221 bin]#
```

i-Ofab297cba1ba2ba8 (Jenkins)

PublicIPs: 13.201.193.108 PrivateIPs: 172.31.8.221

Restart Means- ./shutdown.sh & ./starup.sh

## How to set maven tool after manual installation in jenkins :

- you can add by going **Dashboard >maven jenkins > tools -scroll down > Maven intstallation.**

## Maven installations

Add Maven

≡ Maven

Name

apache-maven-3.9.6

MAVEN\_HOME

/mnt/build-tool/apache-maven-3.9.6

☐ Install automatically ?

Add Maven

Save

Apply

- You can add **multiple version** of maven install in jenkins by Clicking **Add Maven**.

## Now Everything is done now you can create job for CI/CD Pipeline :

- Create job (Item name) - **LoginWebApp**

Add description

All +

S	W	Name ↓	Last Success	Last Failure	Last Duration	
✓	☁	git-clone	12 hr #11	12 hr #10	1.1 sec	▶
✓	☁	Loginwebapp	8 min 13 sec #3	9 min 19 sec #2	15 sec	▶
✓	☀	sanket default workspace	13 hr #4	N/A	29 ms	▶

Icon: S M L

Icon legend

Atom feed for all

Atom feed for failures

Atom feed for just latest builds


- Select Project Type - **Freestyle Project**
- General** → **discard old builds**

## General

Enabled 

Description

Plain text [Preview](#)

☒ Discard old builds 

Strategy

Log Rotation 

Days to keep builds

if not empty, build records are only kept up to this number of days


2

Max # of builds to keep

Save


Apply

- **Advanced → Use custom workspace**

☒ Use custom workspace 

Directory

/mnt/myprojects

Display Name 

☐ Keep the build logs of dependencies 

- **Select source code management → Git → Repository URL + Credentials**

## Source Code Management

☐ None

☒ Git ?

Repositories ?

Repository URL ?

Credentials ?

- Build Steps : **Execute Shell** → **Invoke top-level Maven targets**(In goal write down the maven command) → **Execute shell**

≡ **Execute shell** ?

Command

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

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

≡ **Invoke top-level Maven targets** ?

Maven Version

Goals

≡ Execute shell ?



Command

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

```
rm -rf /mnt/apache-tomcat-9.0.86/webapps/LoginWebApp.war
cp -r target/LoginWebApp.war /mnt/server/apache-tomcat-9.0.85/webapps
```

## • After Build Now : Console output

Dashboard > Loginwebapp > #3 > Console Output

Status

Changes

Console Output

View as plain text

Edit Build Information

Delete build '#3'

Git Build Data

Previous Build

### ✓ Console Output

```
Started by user sanket chaudhari
Running as SYSTEM
Building in workspace /mnt/myprojects
[WS-CLEANUP] Deleting project workspace...
[WS-CLEANUP] Deferred wipeout is used...
[WS-CLEANUP] Done
The recommended git tool is: NONE
using credential gitusername
Cloning the remote Git repository
Cloning repository https://github.com/sanketchaudhari45/project-maven.git
> git init /mnt/myprojects # timeout=10
Fetching upstream changes from https://github.com/sanketchaudhari45/project-maven.git
> git --version # timeout=10
> git --version # 'git version 2.40.1'
using GIT_ASKPASS to set credentials git clone
> git fetch --tags --force --progress -- https://github.com/sanketchaudhari45/project-maven.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/sanketchaudhari45/project-maven.git # timeout=10
```

## • Now Copy Public Ip and Check Your Website Page :

Instance: i-0fab297cba1ba2ba8 (Jenkins)

Details | Status and alarms **New** | Monitoring | Security | Networking | Storage | Tags

### ▼ Instance summary [Info](#)

Instance ID

i-0fab297cba1ba2ba8 (Jenkins)

IPv6 address

—

✓ Public IPv4 address copied

13.201.193.108 [open address](#)

Instance state

✓ Running

Private IPv4 addresses

172.31.8.221

Public IPv4 DNS

ec2-13-201-193-108.ap-south-1.compute.amazonaws.com [open address](#)

## • Its Perfectly working this automation successfully done

Not secure 13.201.193.108:8080/LoginWebApp/

**Login Page**

Username

Password

New User [Register Here](#)

- Always delete local repository dependencies
- In this way in one job you can deploy Website
- In CI-CD daily build and deployment going on daily basis so this one time automation is use.
- For multiple server we have to use or create slave .