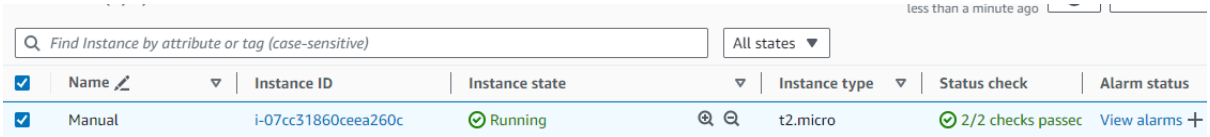
Manual Project Build with Tomcat 9

1) **Launch an EC2 Instance**



2) **Now login and run the below commands to update and upgrade the packages of the OS.**

apt update -y && apt upgrade -y

```
root@ip-172-31-42-17:/home/ubuntu# apt update -y && apt upgrade -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [296 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [69.3 kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [3768 B]
Get:9 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [251 kB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [108 kB]
Get:11 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [8632 B]
Get:12 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [9412 B]
Get:13 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [245 kB]
Get:14 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [47.8 kB]
Get:15 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 c-n-f Metadata [420 B]
```

3) **Now install git and git version using the below command.**

apt install git -y

```
root@ip-172-31-42-17:/home/ubuntu# apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.43.0-1ubuntu7.1).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ip-172-31-42-17:/home/ubuntu#
```

git –version

```
root@ip-172-31-42-17:/home/ubuntu# git --version
git version 2.43.0
```

4) **Now clone the git repository which you want to build the project and check if the repo is cloned properly.**

git clone "https://github.com/ipmdevops/spring-boot-war-example.git"

```
root@ip-172-31-42-17:/home/ubuntu# git clone "https://github.com/ipmdevops/spring-boot-war-example.git"
Cloning into 'spring-boot-war-example'...
remote: Enumerating objects: 411, done.
remote: Counting objects: 100% (177/177), done.
remote: Compressing objects: 100% (71/71), done.
remote: Total 411 (delta 39), reused 100 (delta 4), pack-reused 234 (from 1)
Receiving objects: 100% (411/411), 45.32 KiB | 4.53 MiB/s, done.
Resolving deltas: 100% (77/77), done.
```

ls -ltr

```
root@ip-172-31-42-17:/home/ubuntu# ls -ltr
total 4
drwxr-xr-x 4 root root 4096 Aug 20 15:11 spring-boot-war-example
```

5) **Go inside the cloned project directory and verify the source code and pom.xml file is present so that we can build the project.**

cd spring-boot-war-example

ls -ltr

```
root@ip-172-31-42-17:/home/ubuntu# cd spring-boot-war-example/
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example#
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example#
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example# ls -ltr
total 16
drwxr-xr-x 3 root root 4096 Aug 20 15:11 src
-rw-r--r-- 1 root root 1516 Aug 20 15:11 pom.xml
-rw-r--r-- 1 root root   25 Aug 20 15:11 README.md
-rw-r--r-- 1 root root 1595 Aug 20 15:11 Jenkinsfile
```

6) **To Build the project, we need to install Apache Maven Tool and verify version which helps in testing the source code and builds it as a war or jar file in case of java.**

apt install maven -y

```
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example# apt install maven -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  alsa-topology-conf alsa-ucm-conf ca-certificates-java default-jre-headless fontconfig-config fonts-dejavu-core fonts-dejavu-mono java-common
  libaopalliance-java libapache-pom-java libasound2-data libasound2t64 libatinject-jsr330-api-java libavahi-client3 libavahi-common-data
  libavahi-common3 libcdi-api-java libcommons-cli-java libcommons-io-java libcommons-lang3-java libcommons-parent-java libcups2t64
  liberror-prone-java libfontconfig1 libgeronimo-annotation-1.3-spec-java libgeronimo-interceptor-3.0-spec-java libgraphite2-3 libguava-java
  libguice-java libharfbuzz0b libjansi-java libjpeg-turbo8 libjpeg8 libjsr305-java liblcms2-2 libmaven-parent-java libmaven-resolver-java
  libmaven-shared-utils-java libmaven3-core-java libpcsclite1 libplexus-cipher-java libplexus-classworlds-java
  libplexus-component-annotations-java libplexus-interpolation-java libplexus-sec-dispatcher-java libplexus-utils2-java libsisu-inject-java
  libsisu-plexus-java libslf4j-java libwagon-file-java libwagon-http-shaded-java libwagon-provider-api-java openjdk-21-jre-headless
Suggested packages:
  default-jre alsa-utils libasound2-plugins libatinject-jsr330-api-java-doc libel-api-java libcommons-io-java-doc cups-common libasm-java
  libcglib-java libjsr305-java-doc liblcms2-utils libmaven-shared-utils-java-doc liblogback-java pcscd libplexus-utils2-java-doc junit4 testng
  libcommons-logging-java liblog4j1.2-java libnss-mdns fonts-dejavu-extra fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei
  | fonts-wqy-zenhei fonts-indic
```

mvn –version

```
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example# mvn --version
Apache Maven 3.8.7
Maven home: /usr/share/maven
Java version: 21.0.4, vendor: Ubuntu, runtime: /usr/lib/jvm/java-21-openjdk-amd64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "6.8.0-1009-aws", arch: "amd64", family: "unix"
```

7) **Now we will test the code using below Maven command which uses information from pom.xml.**
NOTE: We need to run the below command from the same location where pom.xml is located

mvn test [Below output comes if the source code is without any errors]

```
[INFO] No tests to run.
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  11.357 s
[INFO] Finished at: 2024-08-20T15:26:10Z
[INFO] ------------------------------------------------------------------------
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example#
```

8) **Now once we get the above output, we run the below command to build the project.**

mvn install [Below output shows project build is successful and war file is created]

```
[INFO] Installing /home/ubuntu/spring-boot-war-example/target/hello-world-0.0.1-SNAPSHOT.war to /root/.m2/repository/com/springhow/example/hello-
orld/0.0.1-SNAPSHOT/hello-world-0.0.1-SNAPSHOT.war
[INFO] Installing /home/ubuntu/spring-boot-war-example/pom.xml to /root/.m2/repository/com/springhow/example/hello-world/0.0.1-SNAPSHOT/hello-wor
d-0.0.1-SNAPSHOT.pom
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  10.145 s
[INFO] Finished at: 2024-08-20T15:30:40Z
[INFO] ------------------------------------------------------------------------
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example#
```

9) **Once the build is successful, a folder named “target” is created which has the war file in it.**

ls -ltr

cd target/

war file created by the name hello-world-0.0.1-SNAPSHOT.war

```
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example# ls -ltr
total 20
drwxr-xr-x 3 root root 4096 Aug 20 15:11 src
-rw-r--r-- 1 root root 1516 Aug 20 15:11 pom.xml
-rw-r--r-- 1 root root   25 Aug 20 15:11 README.md
-rw-r--r-- 1 root root 1595 Aug 20 15:11 Jenkinsfile
drwxr-xr-x 7 root root 4096 Aug 20 15:30 target
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example# cd target/
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target#
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target#
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target# ls -ltr
total 26972
drwxr-xr-x 3 root root     4096 Aug 20 15:26 generated-sources
drwxr-xr-x 3 root root     4096 Aug 20 15:26 maven-status
drwxr-xr-x 3 root root     4096 Aug 20 15:26 classes
drwxr-xr-x 4 root root     4096 Aug 20 15:30 hello-world-0.0.1-SNAPSHOT
drwxr-xr-x 2 root root     4096 Aug 20 15:30 maven-archiver
-rw-r--r-- 1 root root 11081467 Aug 20 15:30 hello-world-0.0.1-SNAPSHOT.war.original
-rw-r--r-- 1 root root 16513680 Aug 20 15:30 hello-world-0.0.1-SNAPSHOT.war
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target#
```

10) Now we can rename the war file to example hello.war and move it to an artifact (in our case the artifact is /tmp)

mv hello-world-0.0.1-SNAPSHOT.war hello.war (rename war file)

```
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target# mv hello-world-0.0.1-SNAPSHOT.war hello.war
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target#
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target#
```

mv hello.war /tmp (war file moved to artifact location)

```
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target# mv hello.war /tmp
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target#
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target#
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target#
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target# cd /tmp/
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp# ls -ltr
total 16160
drwx------ 2 root root     4096 Aug 20 14:45 snap-private-tmp
drwx------ 3 root root     4096 Aug 20 14:46 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-systemd-logind.service-GIxqT4
drwx------ 3 root root     4096 Aug 20 14:54 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-chrony.service-5fVTE5
drwx------ 3 root root     4096 Aug 20 14:54 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-systemd-resolved.service-ue3KBZ
drwx------ 3 root root     4096 Aug 20 14:55 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-polkit.service-goQTNz
drwx------ 3 root root     4096 Aug 20 14:55 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-ModemManager.service-wxwgxM
-rw-r--r-- 1 root root 16513680 Aug 20 15:30 hello.war
drwx------ 3 root root     4096 Aug 20 15:31 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-fwupd.service-cy8UX2
drwxr-xr-x 2 root root     4096 Aug 20 15:57 hsperfdata_root
root@ip-172-31-42-17:/tmp#
```

11) Now we install Apache tomcat version 9 so that we can check code is working in WEBGUI.

Pre-requisite for installing Tomcat is to install open-jdk and check java version

apt install openjdk-17-jdk -y

```
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target# apt install openjdk-17-jdk -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  adwaita-icon-theme at-spi2-common at-spi2-core dconf-gsettings-backend dconf-service fontconfig fonts-dejavu-extra gsettings-desktop-schemas
  gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme libatk-bridge2.0-0t64 libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0t64
  libatspi2.0-0t64 libcairo-gobject2 libcairo2 libdatrie1 libdconf1 libdeflate0 libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2 libdrm-radeon1
  libgail-common libgail18t64 libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgif7 libgl1 libgl1-amber-dri libgl1-mesa-dri
  libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libice-dev libice6 libjbig0 liblerc4 libllvm17t64
  libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpciaccess0 libpixman-1-0 libpthread-stubs0-dev librsvg2-2 librsvg2-common libsharpyuv0
  libsm-dev libsm6 libthai-data libthai0 libtiff6 libvulkan1 libwayland-client0 libwebp7 libx11-dev libx11-xcb1 libxau-dev libxaw7 libxcb-dri2-0
  libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0 libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcb1-dev
  libxcomposite1 libxcursor1 libxdamage1 libxdmcp-dev libxfixes3 libxft2 libxi6 libxinerama1 libxkbfile1 libxmu6 libxpm4 libxrandr2 libxrender1
  libxshmfence1 libxt-dev libxt6t64 libxtst6 libxv1 libxxf86dga1 libxxf86vm1 mesa-vulkan-drivers openjdk-17-jdk-headless openjdk-17-jre
  openjdk-17-jre-headless session-migration ubuntu-mono x11-common x11-utils x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  gvfs libice-doc librsvg2-bin libsm-doc libx11-doc libxcb-doc libxt-doc openjdk-17-demo openjdk-17-source visualvm libnss-mdns
  fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei fonts-indic mesa-utils
Recommended packages:
  luit
```

java –version

```
root@ip-172-31-42-17:/home/ubuntu/spring-boot-war-example/target# java --version
openjdk 21.0.4 2024-07-16
OpenJDK Runtime Environment (build 21.0.4+7-Ubuntu-1ubuntu224.04)
OpenJDK 64-Bit Server VM (build 21.0.4+7-Ubuntu-1ubuntu224.04, mixed mode, sharing)
```

12) Since Tomcat version 9 is not available as default package in ubuntu so we have to install Tomcat 9 manually. Below are the steps to download and install tomcat 9.

wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.93/bin/apache-tomcat-9.0.93.tar.gz

ls -ltr to check if the file is downloaded

```
root@ip-172-31-42-17:/tmp# wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.93/bin/apache-tomcat-9.0.93.tar.gz
--2024-08-20 16:09:28--  https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.93/bin/apache-tomcat-9.0.93.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: 12122732 (12M) [application/x-gzip]
Saving to: 'apache-tomcat-9.0.93.tar.gz'

apache-tomcat-9.0.93.tar.gz         100%[================================================================>]  11.56M  --.-KB/s    in 0.08s

2024-08-20 16:09:28 (149 MB/s) - 'apache-tomcat-9.0.93.tar.gz' saved [12122732/12122732]

root@ip-172-31-42-17:/tmp# ls -ltr
total 28000
-rw-r--r-- 1 root root 12122732 Aug  2 22:59 apache-tomcat-9.0.93.tar.gz
drwx------ 2 root root     4096 Aug 20 14:45 snap-private-tmp
drwx------ 3 root root     4096 Aug 20 14:46 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-systemd-logind.service-GIxqT4
drwx------ 3 root root     4096 Aug 20 14:54 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-chrony.service-5fVTE5
drwx------ 3 root root     4096 Aug 20 14:54 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-systemd-resolved.service-ue3KBZ
drwx------ 3 root root     4096 Aug 20 14:55 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-polkit.service-goQTNz
drwx------ 3 root root     4096 Aug 20 14:55 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-ModemManager.service-wxwgxM
-rw-r--r-- 1 root root 16513680 Aug 20 15:30 hello.war
drwx------ 3 root root     4096 Aug 20 15:31 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-fwupd.service-cy8UX2
drwxr-xr-x 2 root root     4096 Aug 20 15:57 hsperfdata_root
root@ip-172-31-42-17:/tmp#
```

13) Untar the tar file using below command and check tomcat folder is created or not.

tar xzvf apache-tomcat-9.0.93.tar.gz

```
root@ip-172-31-42-17:/tmp# tar xzvf apache-tomcat-9.0.93.tar.gz
apache-tomcat-9.0.93/conf/
apache-tomcat-9.0.93/conf/catalina.policy
apache-tomcat-9.0.93/conf/catalina.properties
apache-tomcat-9.0.93/conf/context.xml
apache-tomcat-9.0.93/conf/jaspic-providers.xml
apache-tomcat-9.0.93/conf/jaspic-providers.xsd
apache-tomcat-9.0.93/conf/logging.properties
apache-tomcat-9.0.93/conf/server.xml
apache-tomcat-9.0.93/conf/tomcat-users.xml
apache-tomcat-9.0.93/conf/tomcat-users.xsd
apache-tomcat-9.0.93/conf/web.xml
apache-tomcat-9.0.93/bin/
apache-tomcat-9.0.93/lib/
apache-tomcat-9.0.93/logs/
apache-tomcat-9.0.93/temp/
apache-tomcat-9.0.93/webapps/
apache-tomcat-9.0.93/webapps/ROOT/
apache-tomcat-9.0.93/webapps/ROOT/WEB-INF/
apache-tomcat-9.0.93/webapps/docs/
apache-tomcat-9.0.93/webapps/docs/META-INF/
apache-tomcat-9.0.93/webapps/docs/WEB-INF/
apache-tomcat-9.0.93/webapps/docs/WEB-INF/jsp/
apache-tomcat-9.0.93/webapps/docs/annotationapi/
apache-tomcat-9.0.93/webapps/docs/api/
apache-tomcat-9.0.93/webapps/docs/appdev/
apache-tomcat-9.0.93/webapps/docs/appdev/sample/
apache-tomcat-9.0.93/webapps/docs/appdev/sample/docs/
apache-tomcat-9.0.93/webapps/docs/appdev/sample/src/
```

ls -ltr

```
root@ip-172-31-42-17:/tmp# ls -ltr
total 28004
-rw-r--r-- 1 root root 12122732 Aug  2 22:59 apache-tomcat-9.0.93.tar.gz
drwx------ 2 root root     4096 Aug 20 14:45 snap-private-tmp
drwx------ 3 root root     4096 Aug 20 14:46 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-systemd-logind.service-GIxqT4
drwx------ 3 root root     4096 Aug 20 14:54 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-chrony.service-5fVTE5
drwx------ 3 root root     4096 Aug 20 14:54 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-systemd-resolved.service-ue3KBZ
drwx------ 3 root root     4096 Aug 20 14:55 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-polkit.service-goQTNz
drwx------ 3 root root     4096 Aug 20 14:55 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-ModemManager.service-wxwgxM
-rw-r--r-- 1 root root 16513680 Aug 20 15:30 hello.war
drwx------ 3 root root     4096 Aug 20 15:31 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-fwupd.service-cy8UX2
drwxr-xr-x 2 root root     4096 Aug 20 15:57 hsperfdata_root
drwxr-xr-x 9 root root     4096 Aug 20 16:12 apache-tomcat-9.0.93
root@ip-172-31-42-17:/tmp#
```

14) Now rename the extracted folder tomcat so that its easy to remember and check.

mv apache-tomcat-9.0.93 tomcat
ls -ltr

```
root@ip-172-31-42-17:/tmp# mv apache-tomcat-9.0.93 tomcat
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp# ls -ltr
total 28004
-rw-r--r-- 1 root root 12122732 Aug  2 22:59 apache-tomcat-9.0.93.tar.gz
drwx------ 2 root root     4096 Aug 20 14:45 snap-private-tmp
drwx------ 3 root root     4096 Aug 20 14:46 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-systemd-logind.service-GIxqT4
drwx------ 3 root root     4096 Aug 20 14:54 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-chrony.service-5fVTE5
drwx------ 3 root root     4096 Aug 20 14:54 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-systemd-resolved.service-ue3KBZ
drwx------ 3 root root     4096 Aug 20 14:55 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-polkit.service-goQTNz
drwx------ 3 root root     4096 Aug 20 14:55 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-ModemManager.service-wxwgxM
-rw-r--r-- 1 root root 16513680 Aug 20 15:30 hello.war
drwx------ 3 root root     4096 Aug 20 15:31 systemd-private-61e3c7adb5f54724a28669bf7f5b1f08-fwupd.service-cy8UX2
drwxr-xr-x 2 root root     4096 Aug 20 15:57 hsperfdata_root
drwxr-xr-x 9 root root     4096 Aug 20 16:12 tomcat
root@ip-172-31-42-17:/tmp#
```

15) Now we have to create the tomcat service manually in the system location mentioned below and add the contents in the service file.

vi /etc/systemd/system/tomcat.service

Contents of tomcat.service file

[Unit]
Description=Tomcat 9 servlet container
After=network.target

[Service]
Type=forking

Environment="JAVA_HOME=/usr/lib/jvm/java-17-openjdk-amd64"
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom -Djava.awt.headless=true"

Environment="CATALINA_BASE=/tmp/tomcat/"
Environment="CATALINA_HOME=/tmp/tomcat/"
Environment="CATALINA_PID=/tmp/tomcat/temp/tomcat.pid"
Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"

ExecStart=/tmp/tomcat/bin/startup.sh
ExecStop=/tmp/tomcat/bin/shutdown.sh

[Install]

WantedBy=multi-user.target

```
root@ip-172-31-42-17:/tmp# vi /etc/systemd/system/tomcat.service
```

```
root@ip-172-31-42-17: /tmp
[Unit]
Description=Tomcat 9 servlet container
After=network.target

[Service]
Type=forking

Environment="JAVA_HOME=/usr/lib/jvm/java-17-openjdk-amd64"
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom -Djava.awt.headless=true"

Environment="CATALINA_BASE=/tmp/tomcat/"
Environment="CATALINA_HOME=/tmp/tomcat/"
Environment="CATALINA_PID=/tmp/tomcat/temp/tomcat.pid"
Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"

ExecStart=/tmp/tomcat/bin/startup.sh
ExecStop=/tmp/tomcat/bin/shutdown.sh

[Install]
WantedBy=multi-user.target
```

16) Now we will reload the daemon and start tomcat service and also verify that the service is running.

systemctl daemon-reload

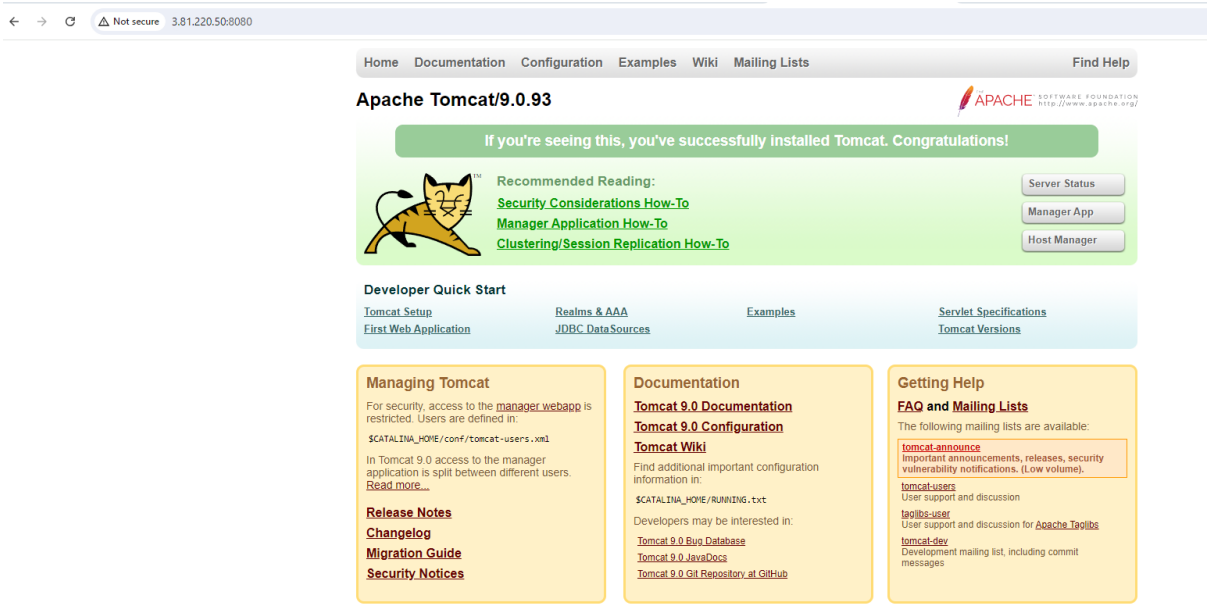
systemctl start tomcat

systemctl status tomcat

```
root@ip-172-31-42-17:/tmp# vi /etc/systemd/system/tomcat.service
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp# systemctl daemon-reload
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp# systemctl start tomcat
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp# systemctl status tomcat
● tomcat.service - Tomcat 9 servlet container
     Loaded: loaded (/etc/systemd/system/tomcat.service; disabled; preset: enabled)
     Active: active (running) since Tue 2024-08-20 16:21:04 UTC; 8s ago
    Process: 16290 ExecStart=/tmp/tomcat/bin/startup.sh (code=exited, status=0/SUCCESS)
   Main PID: 16297 (java)
      Tasks: 31 (limit: 1130)
     Memory: 185.2M (peak: 185.5M)
        CPU: 3.878s
     CGroup: /system.slice/tomcat.service
             └─16297 /usr/lib/jvm/java-17-openjdk-amd64/bin/java -Djava.util.logging.config.file=/tmp/tomcat//conf/logging.properties -Djava.util

Aug 20 16:21:04 ip-172-31-42-17 systemd[1]: Starting tomcat.service - Tomcat 9 servlet container...
Aug 20 16:21:04 ip-172-31-42-17 startup.sh[16290]: Tomcat started.
Aug 20 16:21:04 ip-172-31-42-17 systemd[1]: Started tomcat.service - Tomcat 9 servlet container.
lines 1-14/14 (END)
```

17) We will verify that tomcat is running or not on the WEG UI by accessing tomcat URL – PublicIP of EC2 instance:8080 (Tomcat works on port 8080)



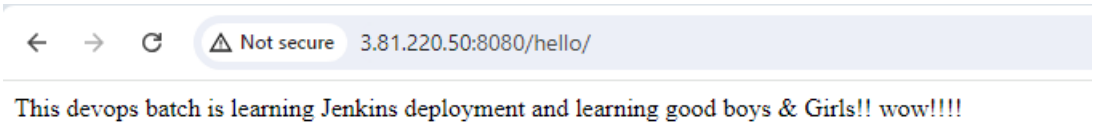
18) Now we will copy the hello.war file from /tmp to tomcat/webapps. Once the war file is copied in webapps folder we see that hello folder is also created in it.

cp hello.war tomcat/webapps

ls -ltr

```
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp# cp hello.war tomcat/webapps/
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp#
root@ip-172-31-42-17:/tmp# cd tomcat/webapps/
root@ip-172-31-42-17:/tmp/tomcat/webapps#
root@ip-172-31-42-17:/tmp/tomcat/webapps#
root@ip-172-31-42-17:/tmp/tomcat/webapps# ls -ltr
total 16152
drwxr-x---  3 root root     4096 Aug 20 16:12 ROOT
drwxr-x--- 16 root root     4096 Aug 20 16:12 docs
drwxr-x---  7 root root     4096 Aug 20 16:12 examples
drwxr-x---  6 root root     4096 Aug 20 16:12 host-manager
drwxr-x---  6 root root     4096 Aug 20 16:12 manager
-rw-r--r--  1 root root 16513680 Aug 20 16:28 hello.war
drwxr-x---  5 root root     4096 Aug 20 16:28 hello
```

19) Now we will verify that code is accessible on tomcat GUI by accessing tomcat URL:8080/hello



20) Once we get the above output, we are sure that our manual build is successful visible and tested.