Declarative pipeline code:

pipeline{

agent any

environment{

PATH = "/opt/apache-maven-3.8.6/bin:$PATH"

}

stages{

stage("git clone")

{

steps{

git 'https://github.com/selenium-saikrishna/maven.git'

}

}

stage("build code")

{

steps{

sh 'mvn package'

}

}

stage("code deploy")

{

steps{

sshagent(['deploy\_user']) {

sh 'scp -o StrictHostKeyChecking=no webapp/target/webapp.war ec2-user@3.111.169.238:/opt/apache-tomcat-9.0.64/webapps'

}

}

}

}

}

Adding environments:

Add maven home location in environment stage

PATH = “/opt/apache-tomcat-9.0.64/bin:$PATH”

Deploy war file from Jenkins server to tomcat server:

1. To deploy the war file from Jenkins server to tomcat server through pipeline we need to access the Jenkins server from tomcat server so we have ssh agent plugin
2. Download the ssh agent plugin
3. In pipeline code generator ssh agent then add Jenkins sever username and pem key file
4. Then convert the code copy then paste into the code deploy stage
5. Scp <src \_file > username@ip\_tomcat\_server:<dest\_path>

Scp webapp/target/webapp.war ec2-user@ip\_tomcat\_server: /opt/apache-tomcat-9.0.64/webapps

1. Then give the ec2-user to access the location of webapps it need permission

Chown –R ec2-user:ec2-user /opt

1. We need to add host key permission

-o StrictHostKeyChecking=no (this is available at ssh agent section at code generator)