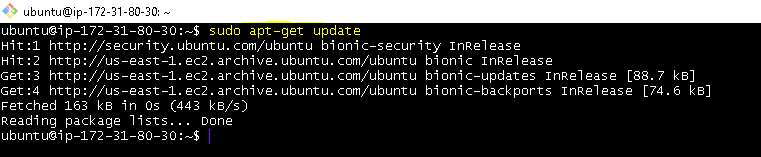
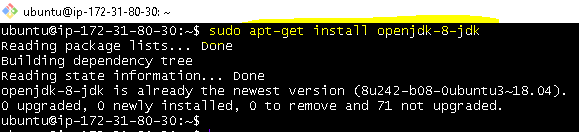
**Jenkins**

**Installing Jenkins:**

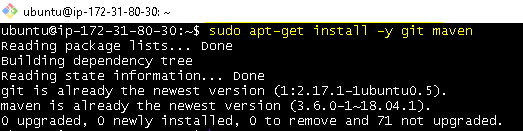
**Update the repository:** sudo apt-get update

****

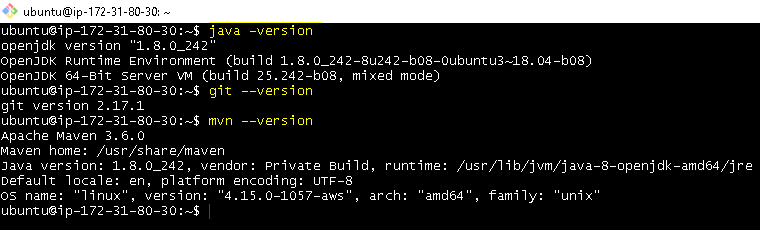
**Install java (version 8) :** sudo apt-get install openjdk-8-jdk

****

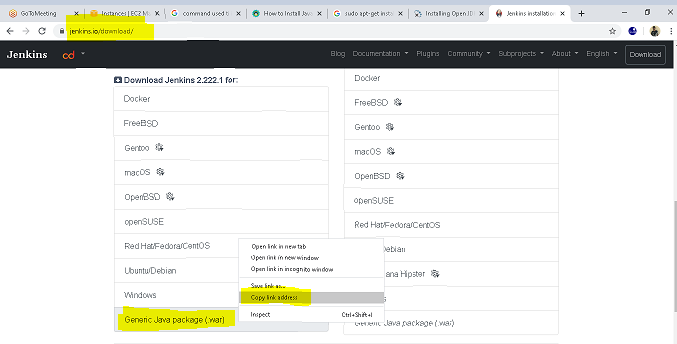
**Install git & maven:** sudo apt-get install -y git maven

****

**Validation:**

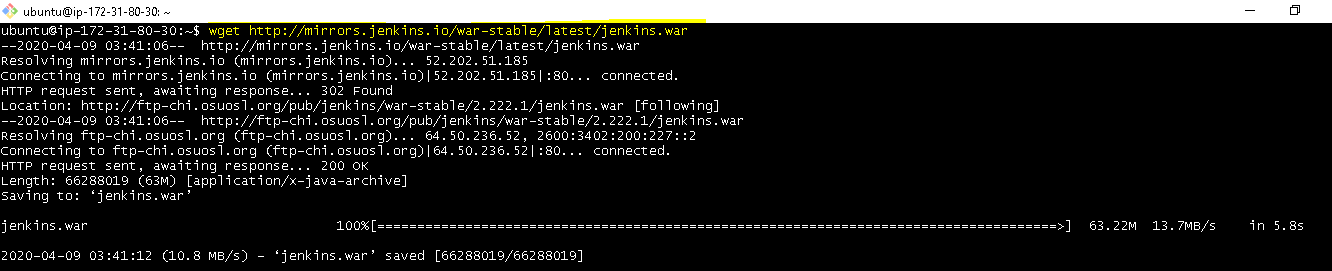
****

**Download Jenkins (copy the below url from chrome )**

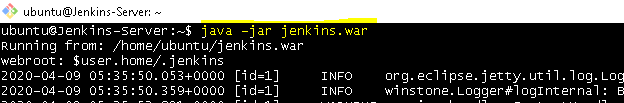
****

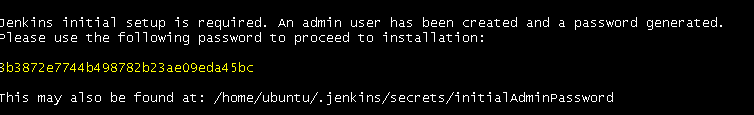
**ubuntu@ip-172-31-80-30:~$**

wget http://mirrors.jenkins.io/war-stable/latest/jenkins.war

****

**To start Jenkins:**

****

****

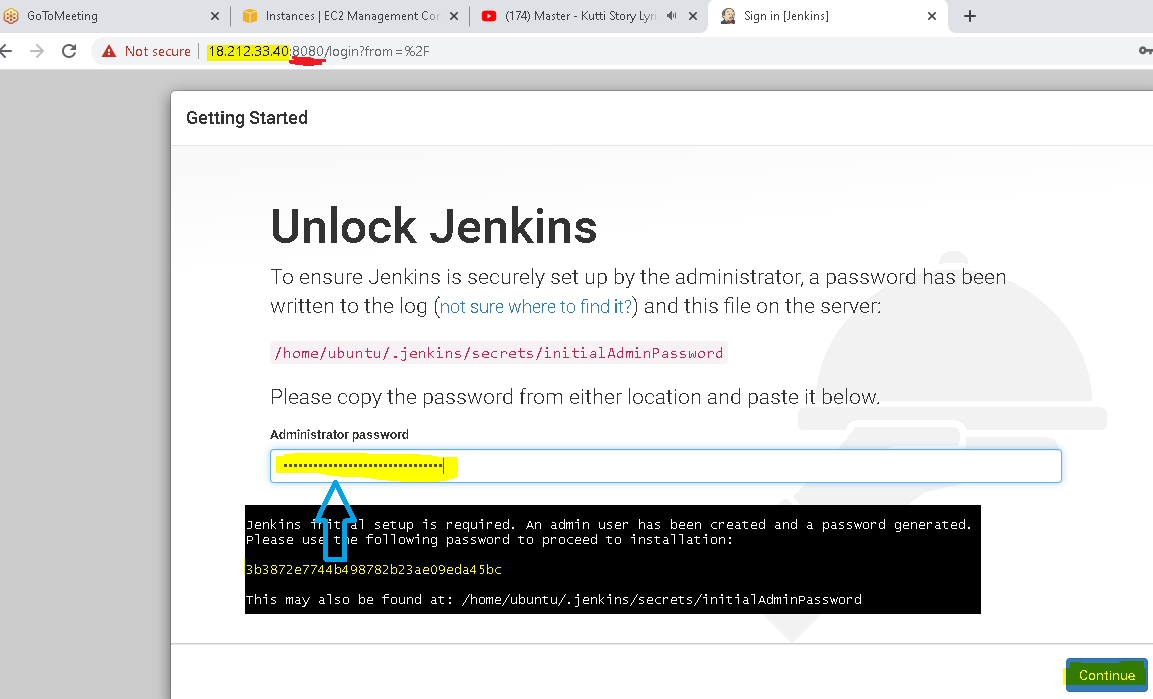
Please use the following password to proceed to installation:

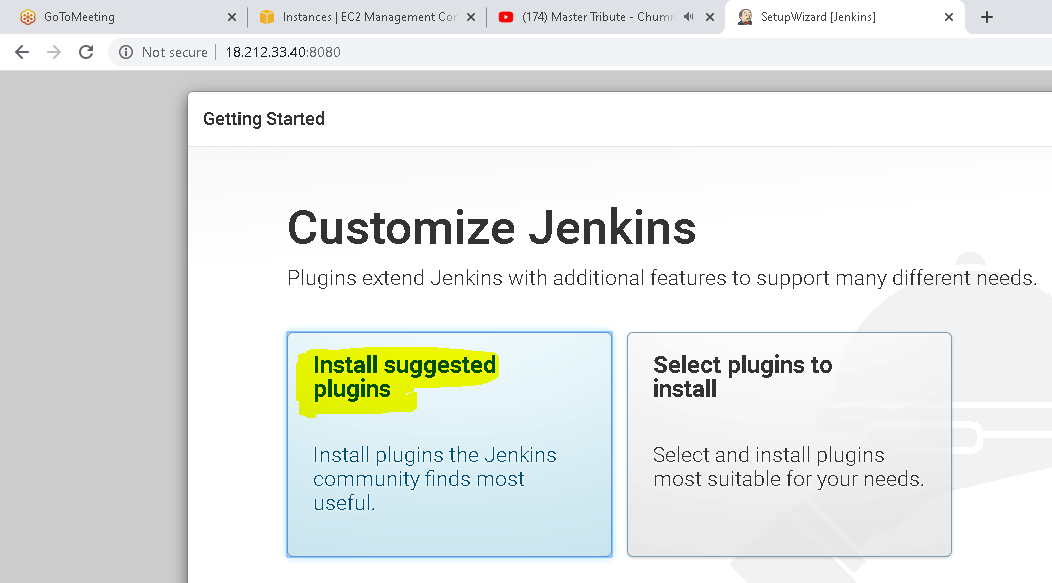
3b3872e7744b498782b23ae09eda45bc

This may also be found at: /home/ubuntu/.jenkins/secrets/initialAdminPassword

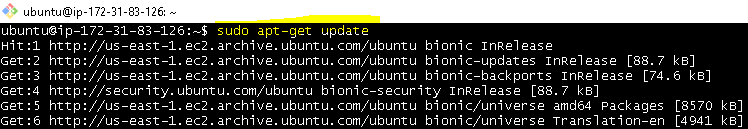
**To access jenkins from browser:**

**IPv4 Public IP 18.212.33.40 Jenkins port no: 8080**

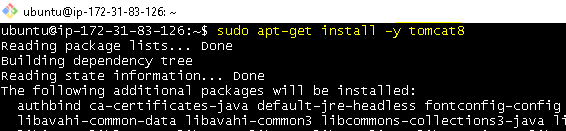
****

****

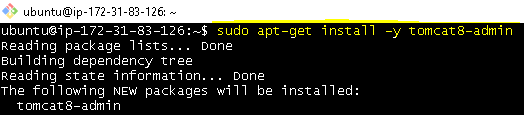
**QA-Server**

****

ubuntu@ip-172-31-83-126:~$ **sudo apt-get install -y tomcat8**

****

ubuntu@ip-172-31-83-126:~$ **sudo apt-get install -y tomcat8-admin**

****

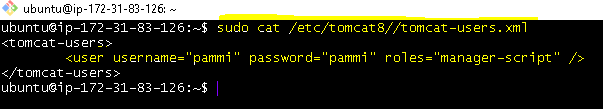
**Edit tomcat-users.xml file:**

ubuntu@ip-172-31-83-126:~$ **sudo vim /etc/tomcat8/tomcat-users.xml**

**<tomcat-users>**

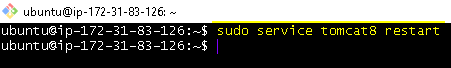
**<user username="pammi" password="pammi" roles="manager-script" />**

**</tomcat-users>**

****

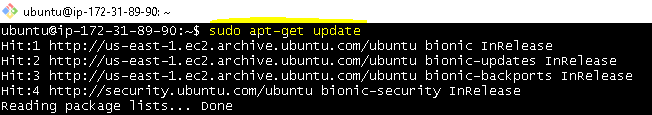
**Restart tomcat8**

ubuntu@ip-172-31-83-126:~$ **sudo service tomcat8 restart**

****

**Prod-server**

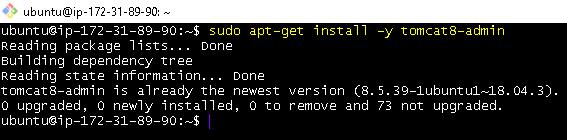
ubuntu@ip-172-31-89-90:~$ **sudo apt-get update**

****

ubuntu@ip-172-31-89-90:~$ **sudo apt-get install -y tomcat8**

****

ubuntu@ip-172-31-89-90:~$ **sudo apt-get install -y tomcat8-admin**

****

**Edit tomcat-users.xml file:**

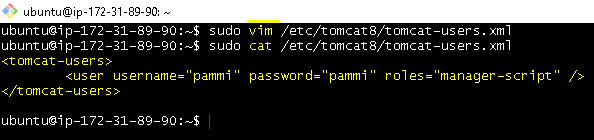
ubuntu@ip-172-31-89-90:~$ **sudo vim /etc/tomcat8/tomcat-users.xml**

ubuntu@ip-172-31-89-90:~$ **sudo cat /etc/tomcat8/tomcat-users.xml**

**<tomcat-users>**

**<user username="pammi" password="pammi" roles="manager-script" />**

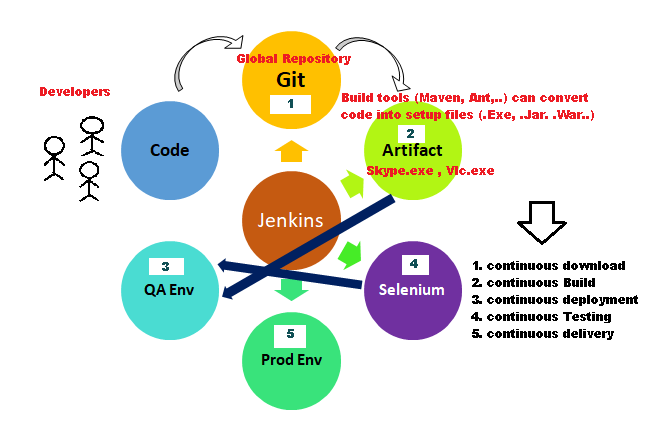
**</tomcat-users>**



**Restart tomcat8**

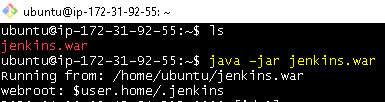


**Jenkins Life cycle**



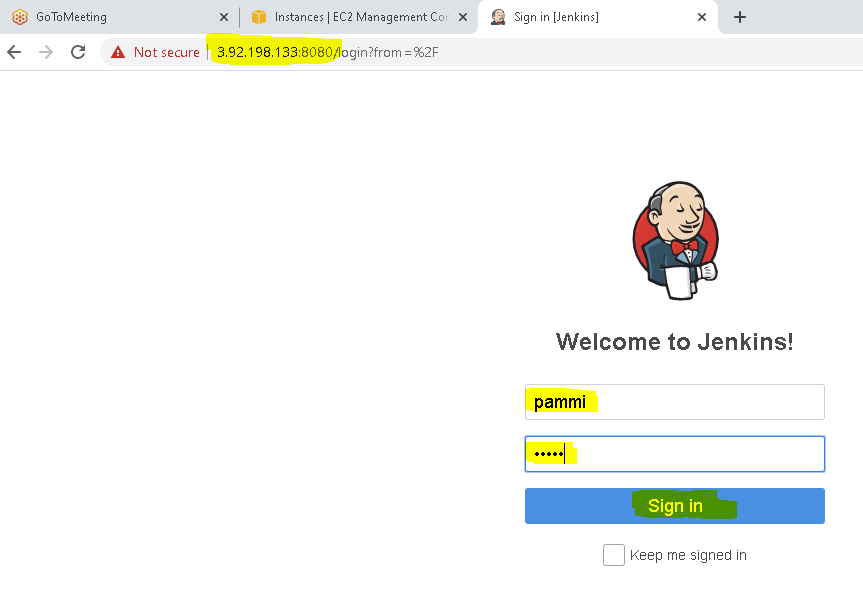
**Project- (Freestyle)**

**To connect Jenkins:** ubuntu@ip-172-31-92-55:~$ **java -jar jenkins.war**

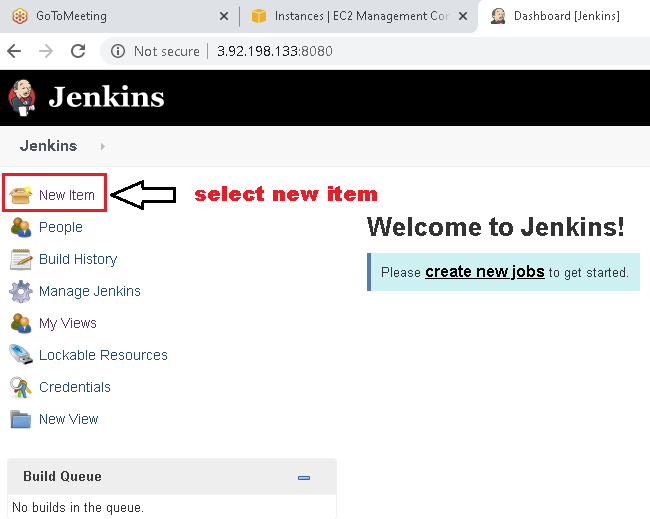


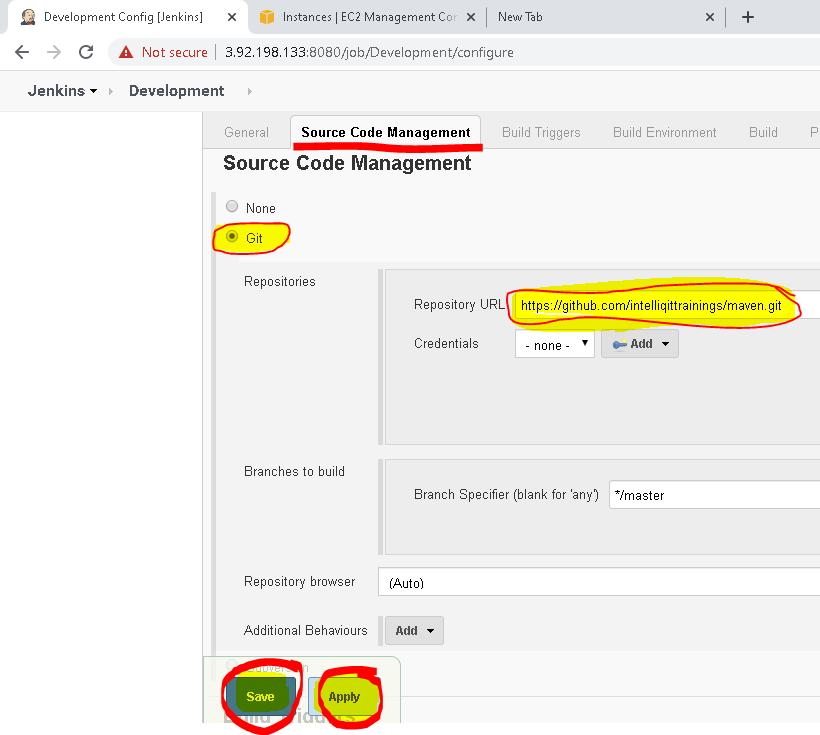
**To access jenkins from browser:**

**IPv4 Public IP 3.92.198.133 Jenkins port no: 8080**

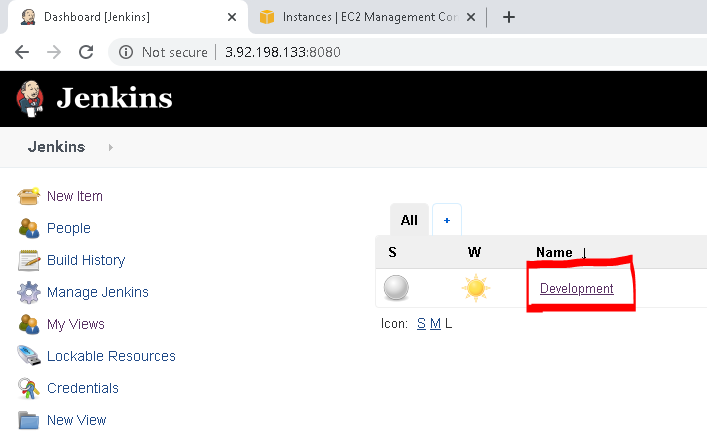
****

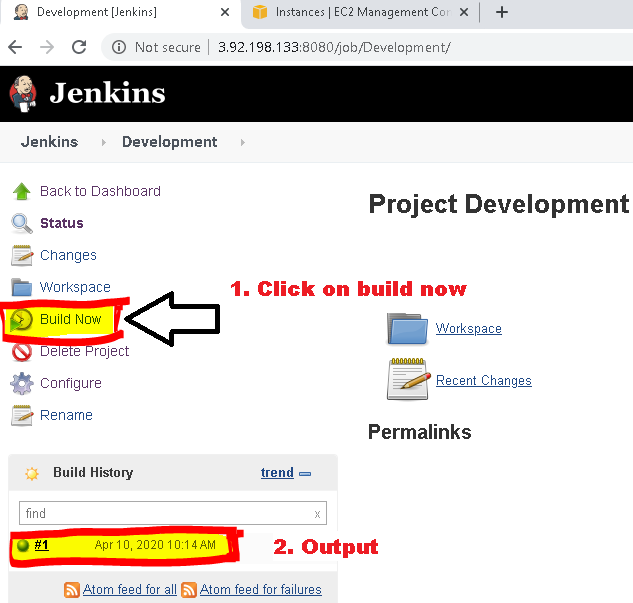
**Stage 1 : Continous Download**

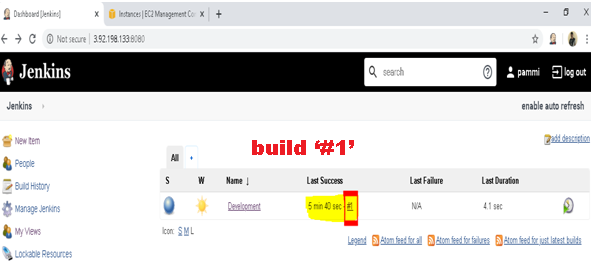
****

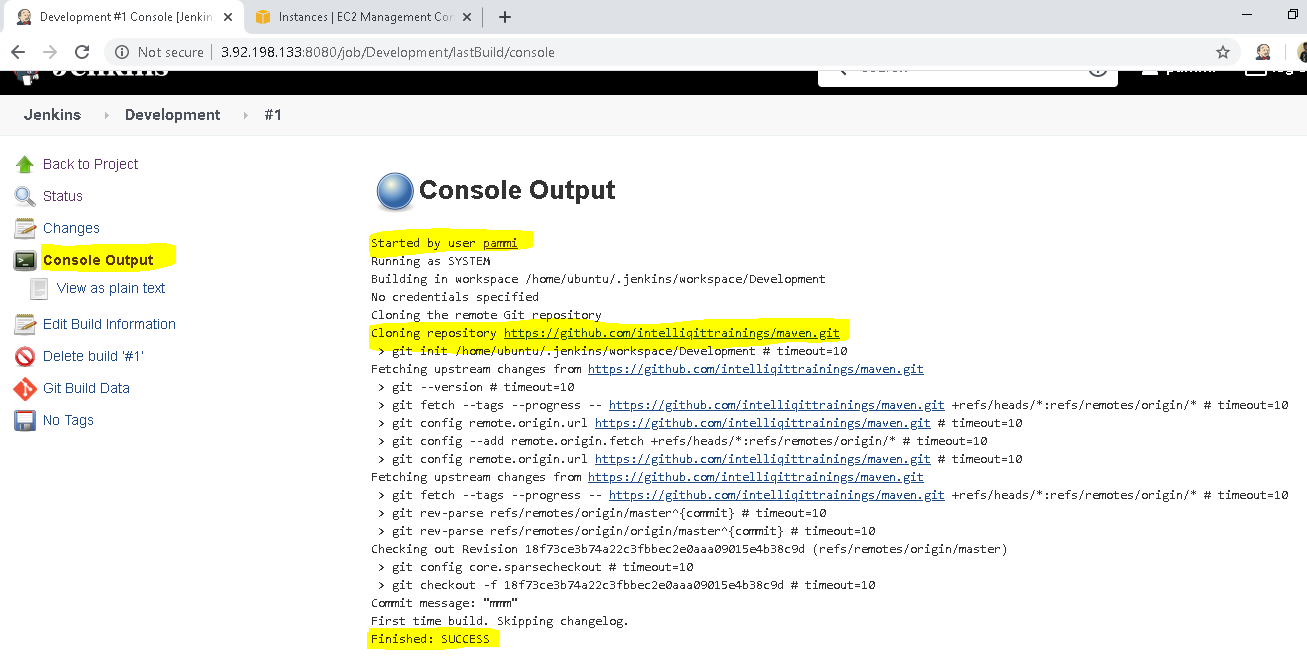
****

**https://github.com/intelliqittrainings/maven.git**

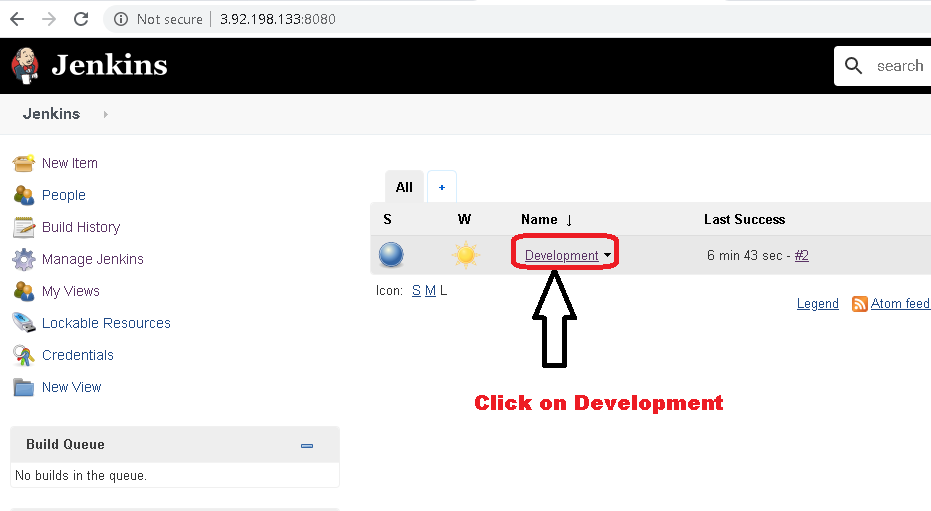
****

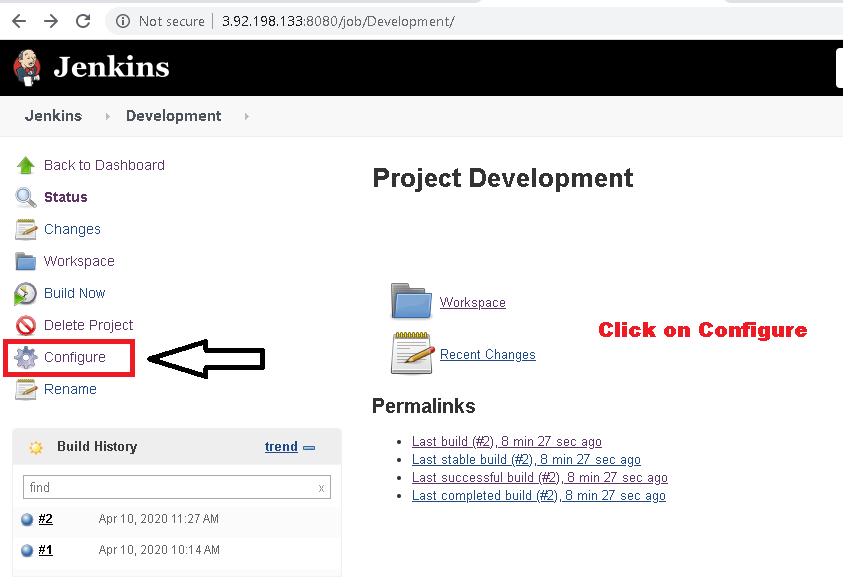
****

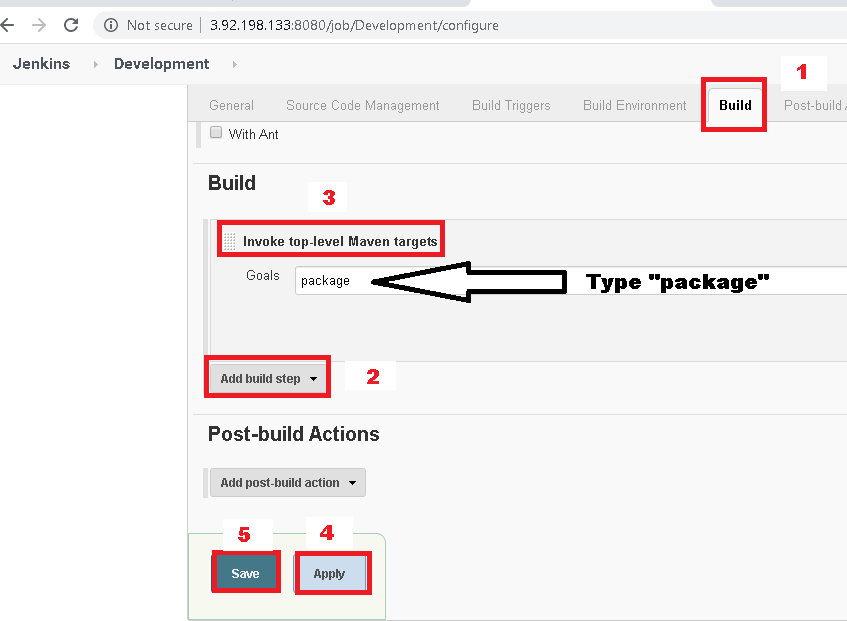
****

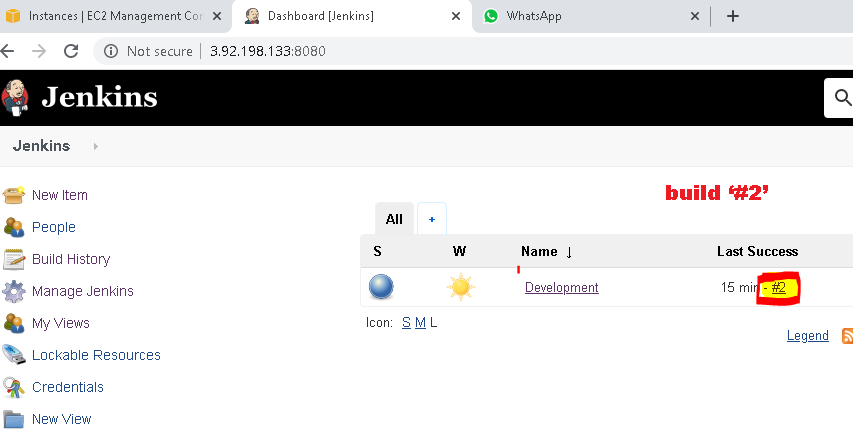
****

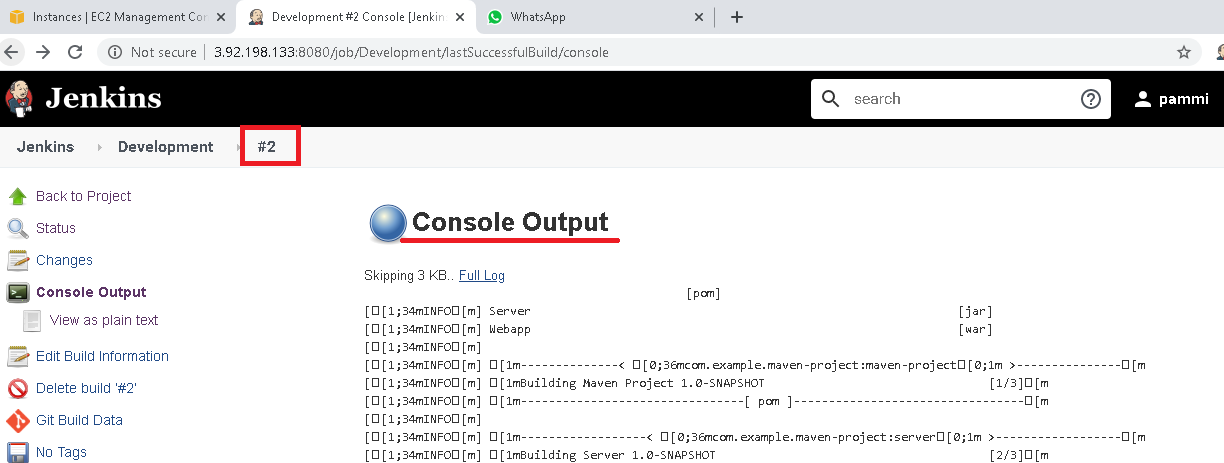
**Stage2. Continious Build**

****

****

****

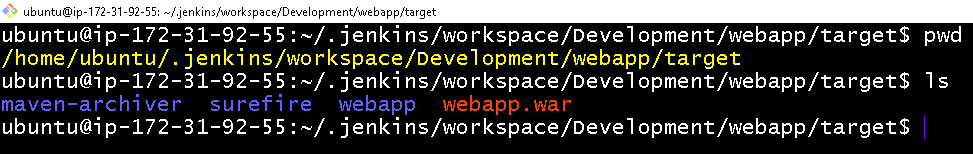
****

****

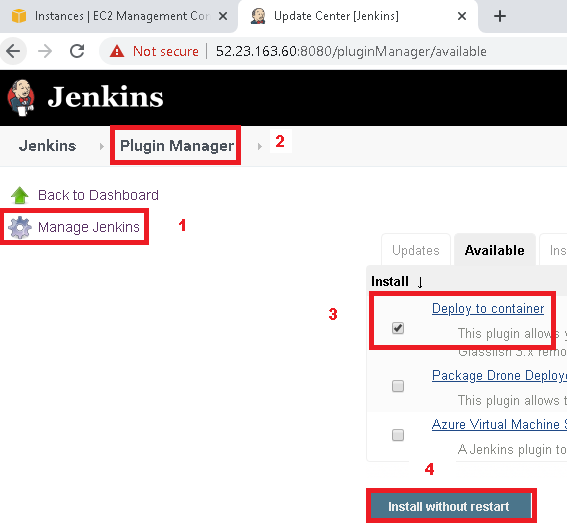
****

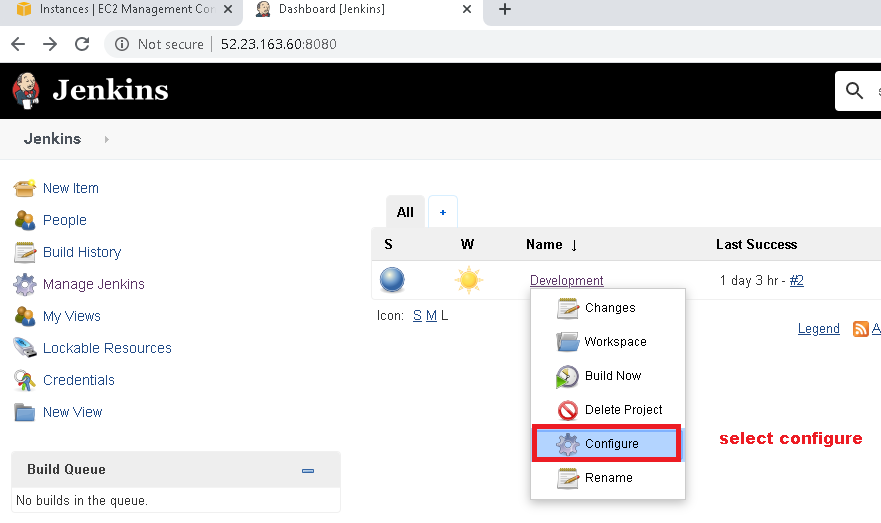
Building war: **/home/ubuntu/.jenkins/workspace/Development/webapp/target/webapp.war**

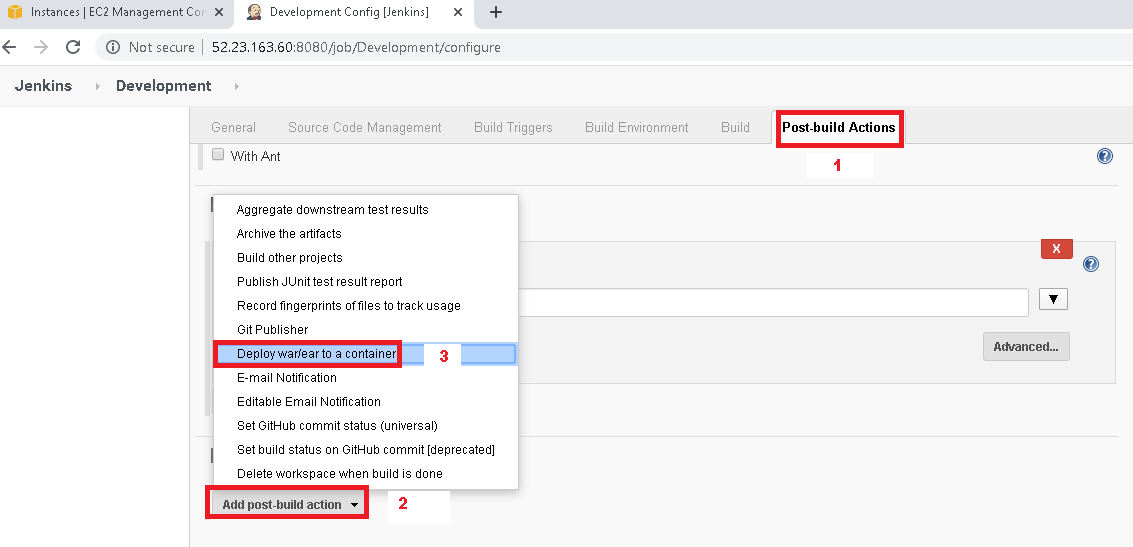
**Output:**

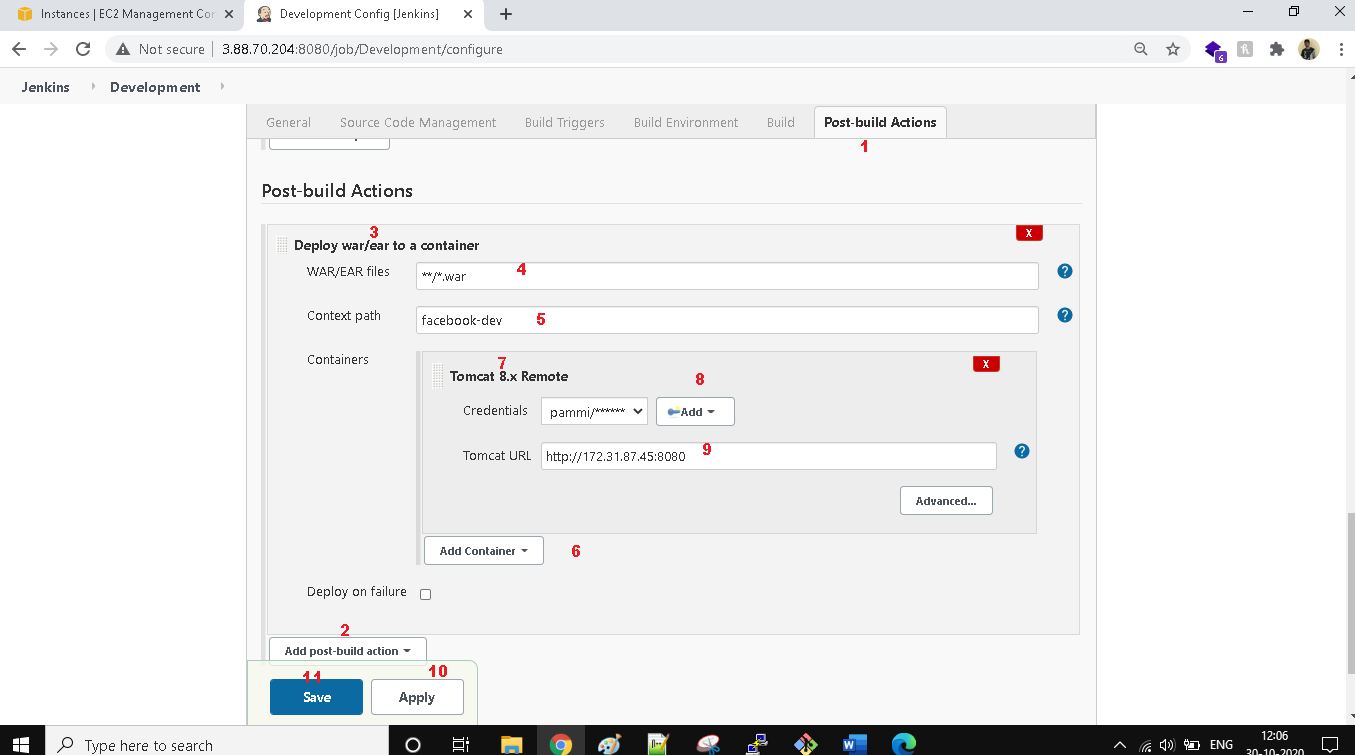
****

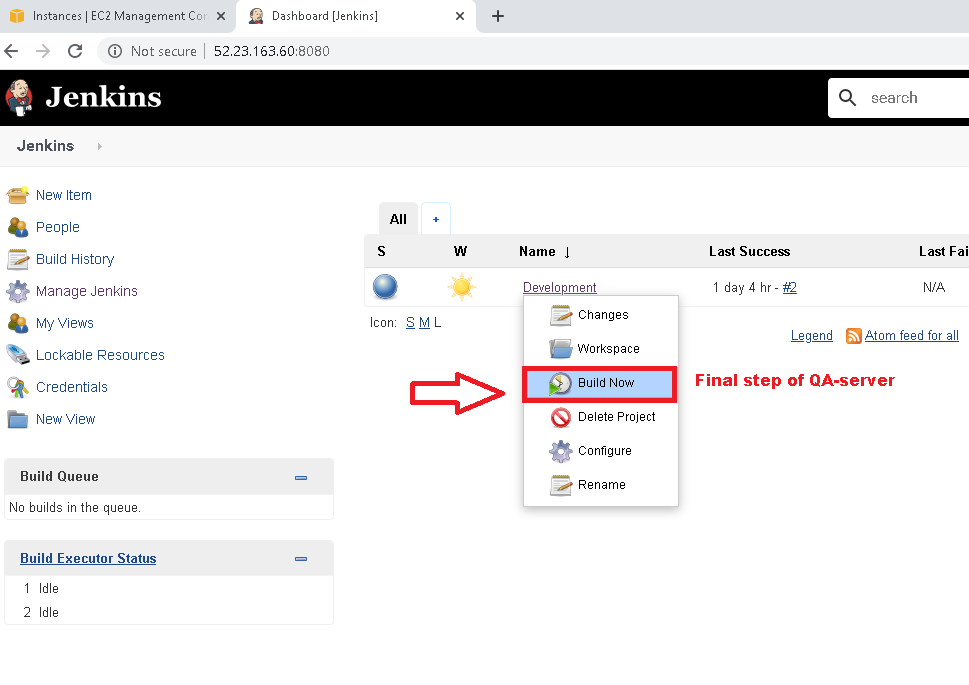
**Stage3. continuous Deploy**

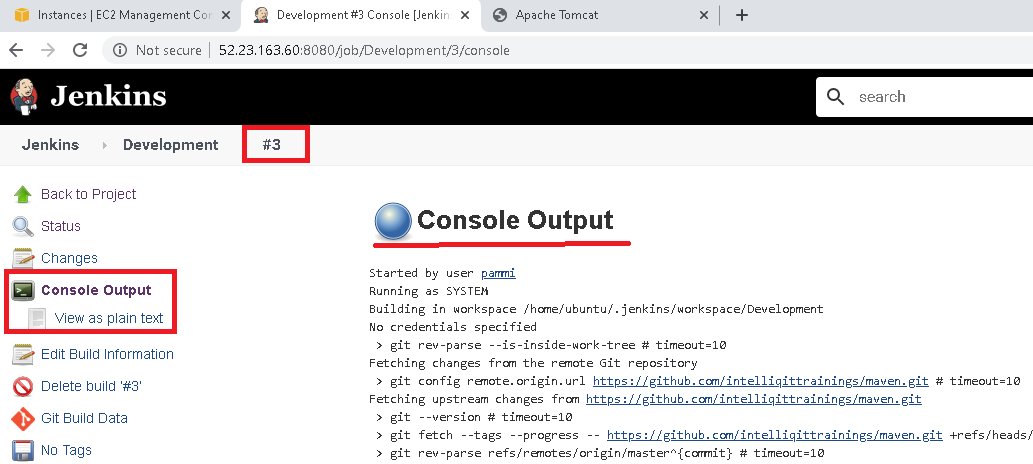
****

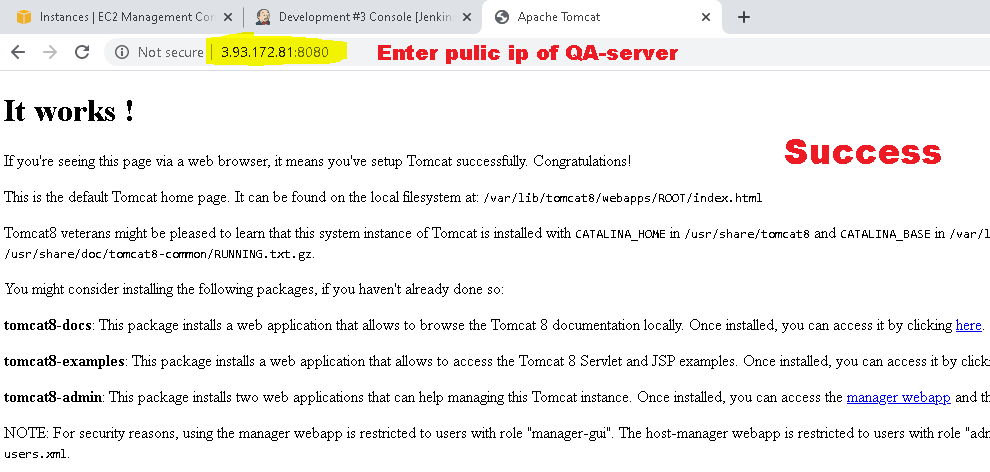
****

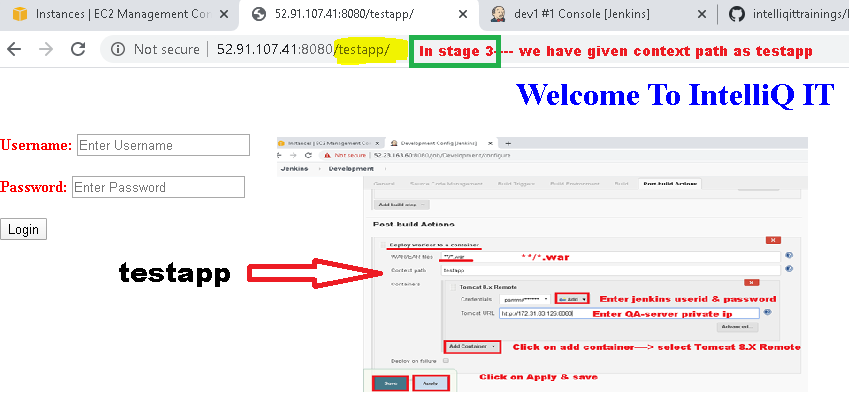
****

****

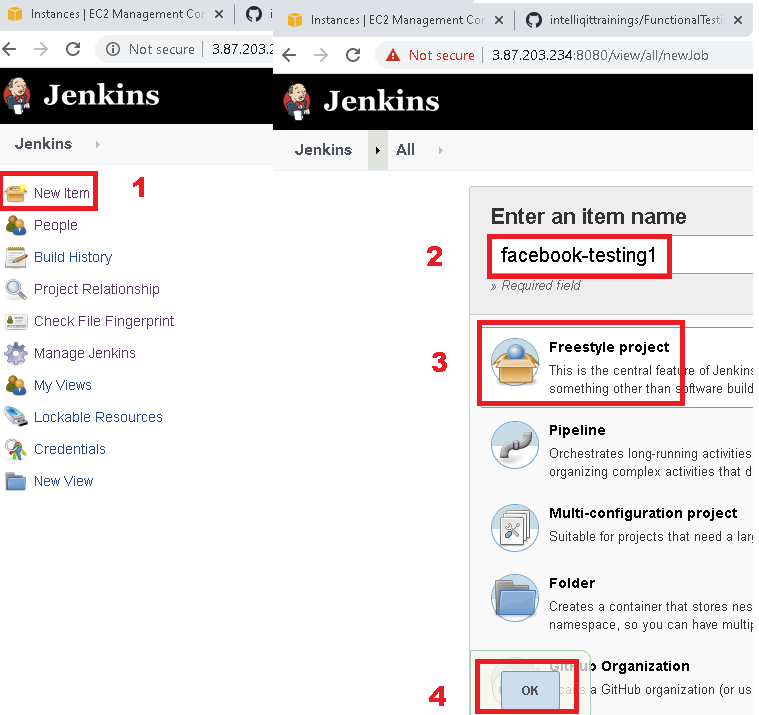
****

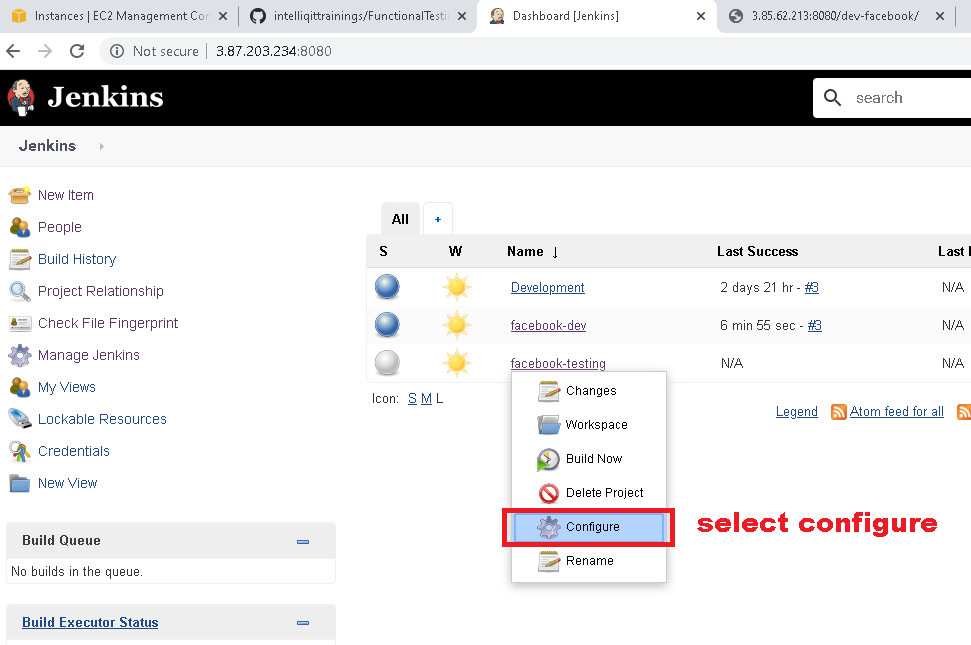
****

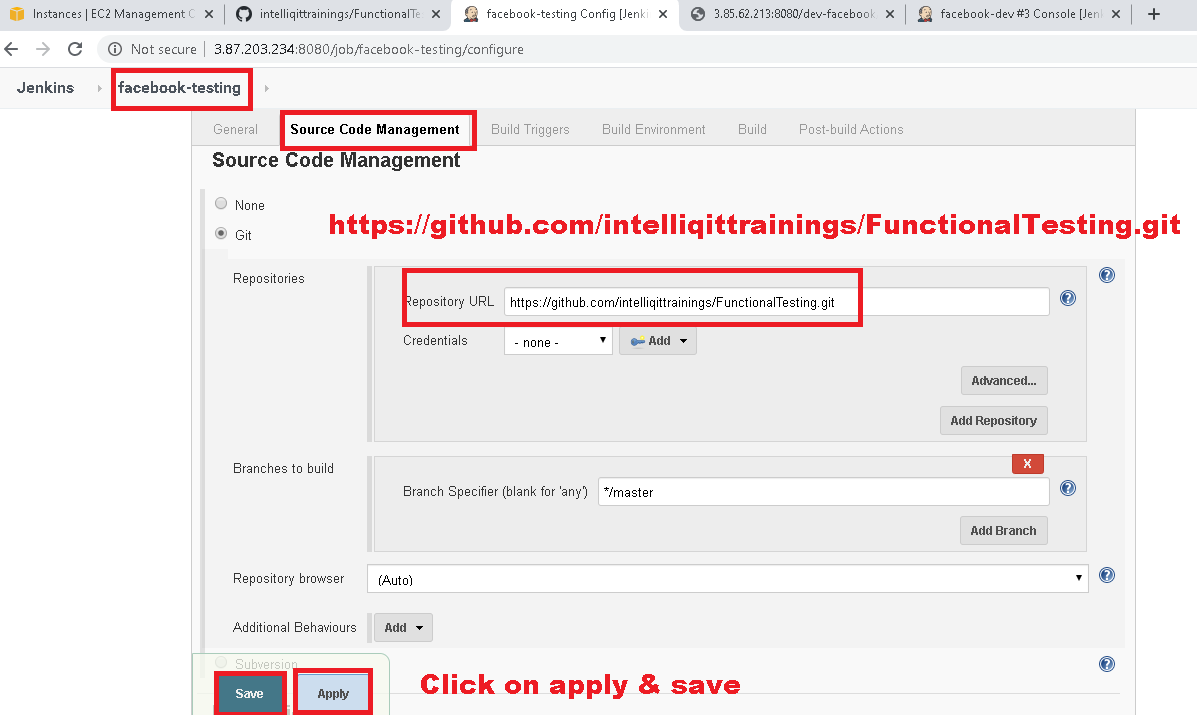
****

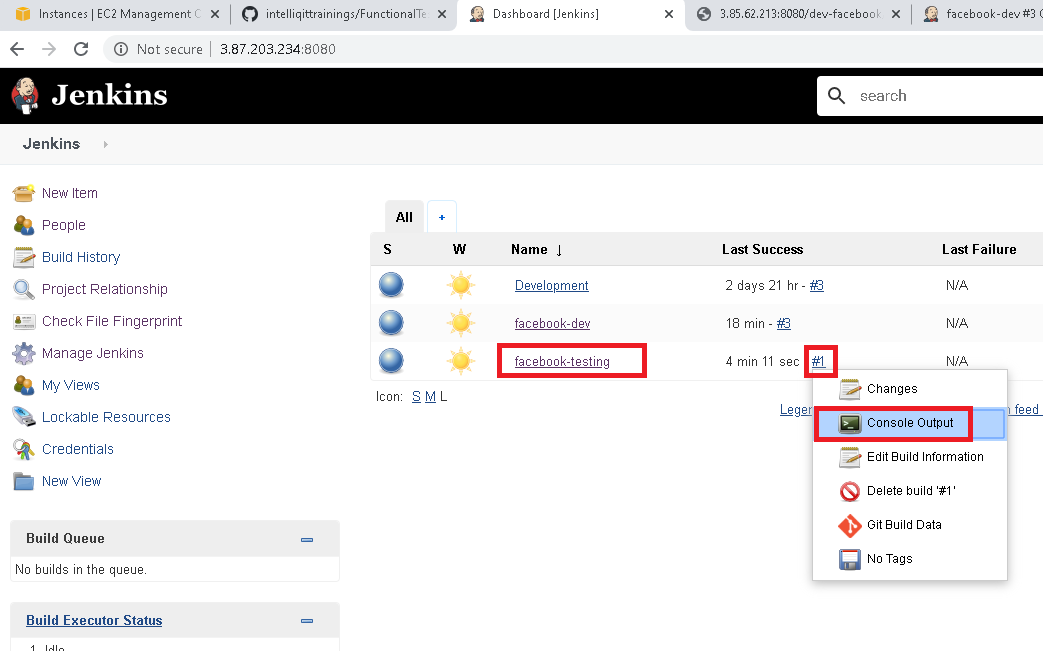


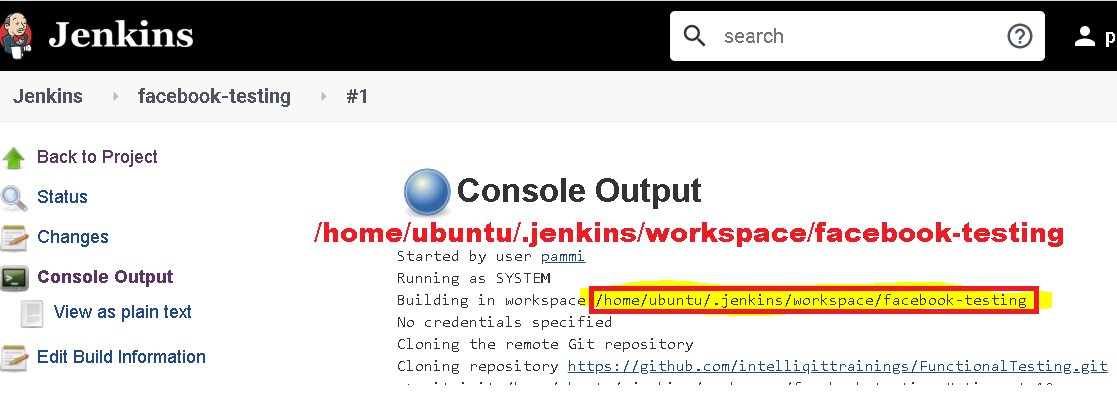
**Stage4. continuous Testing**

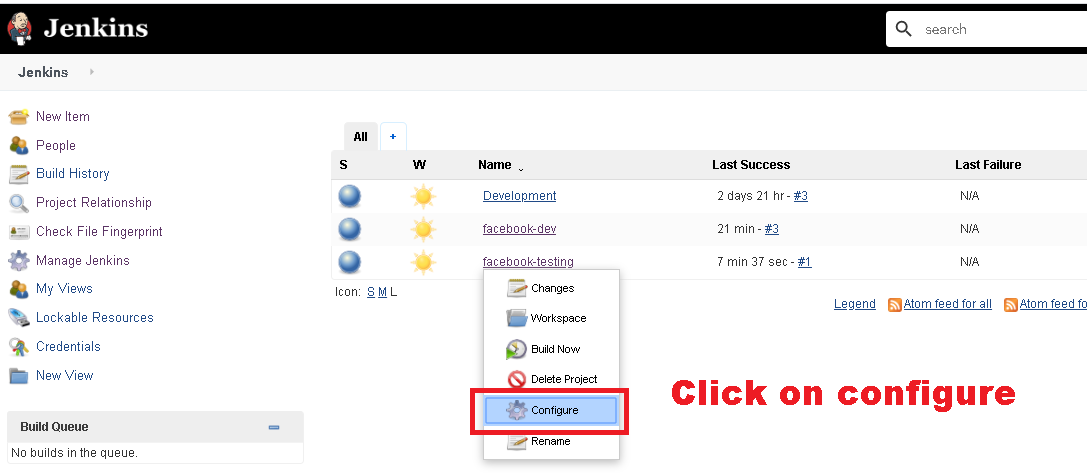
****

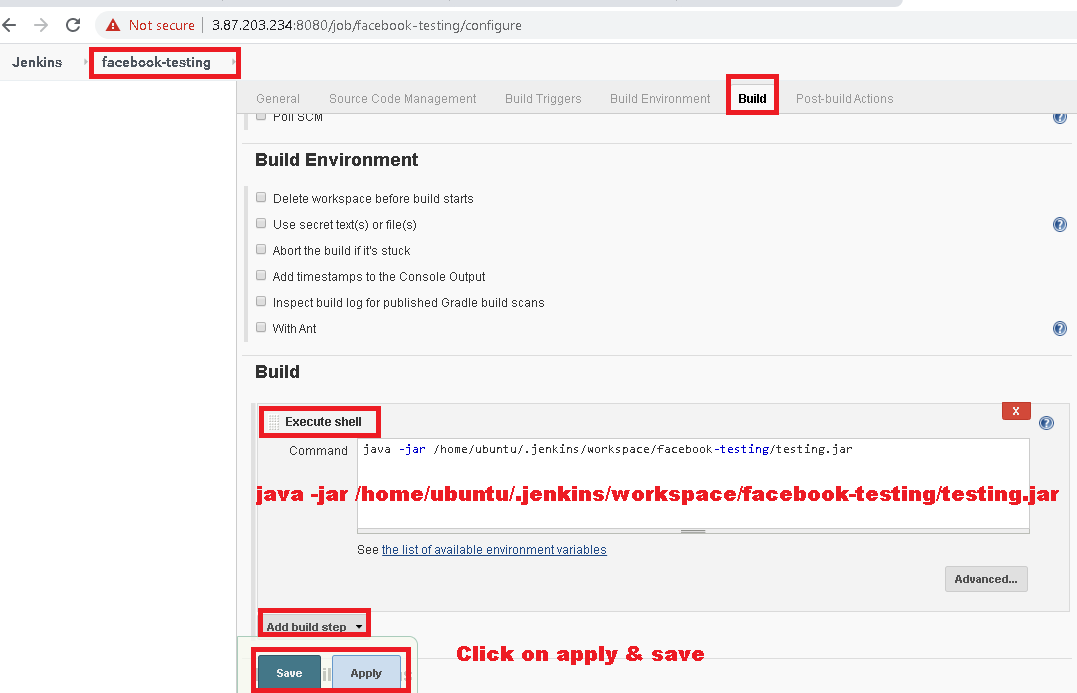
****

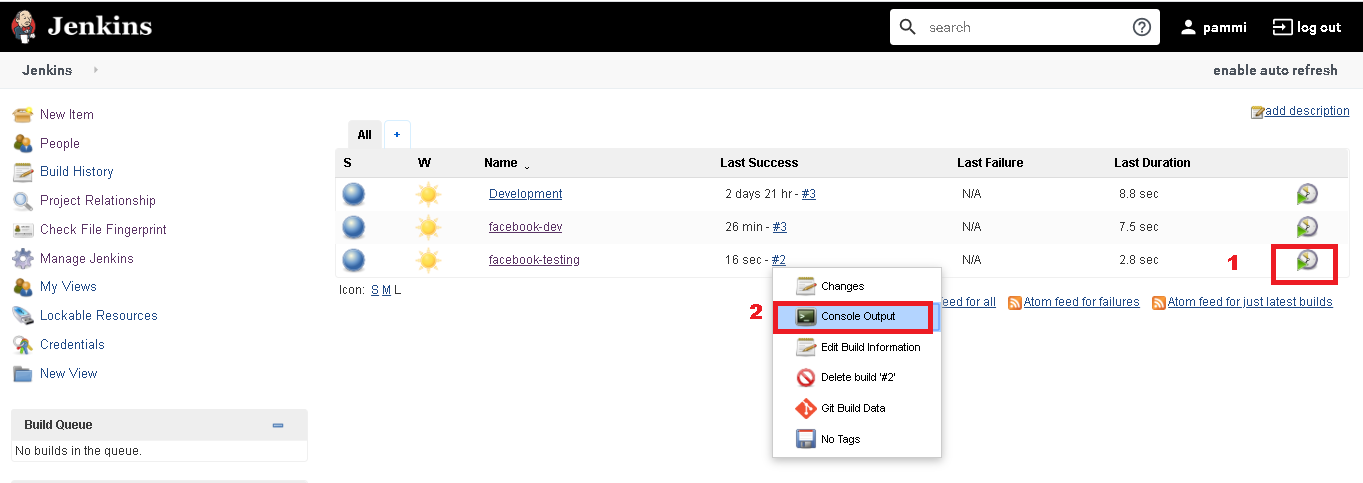
****

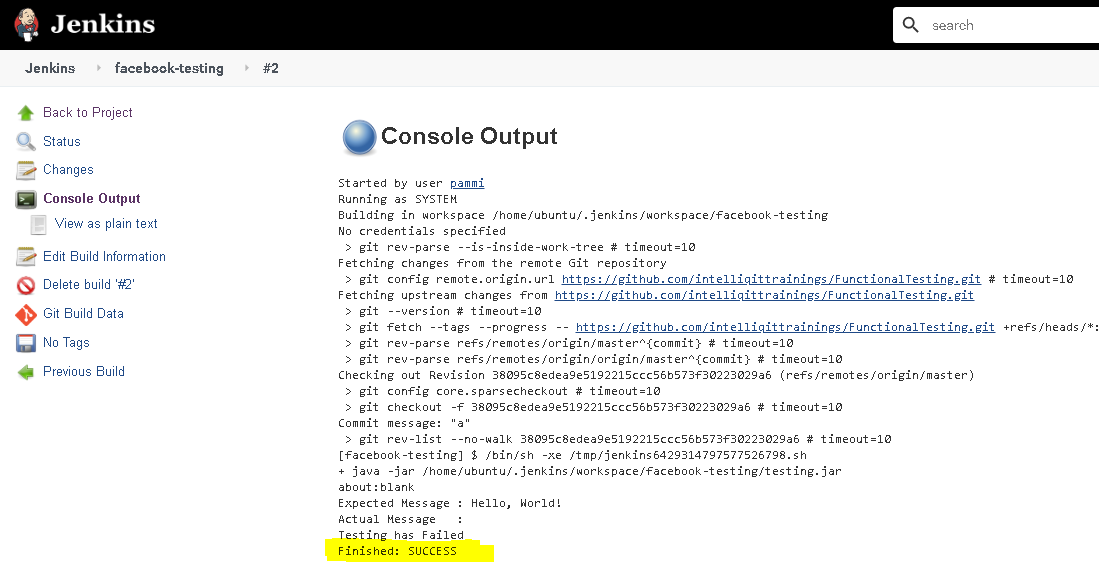
****

****

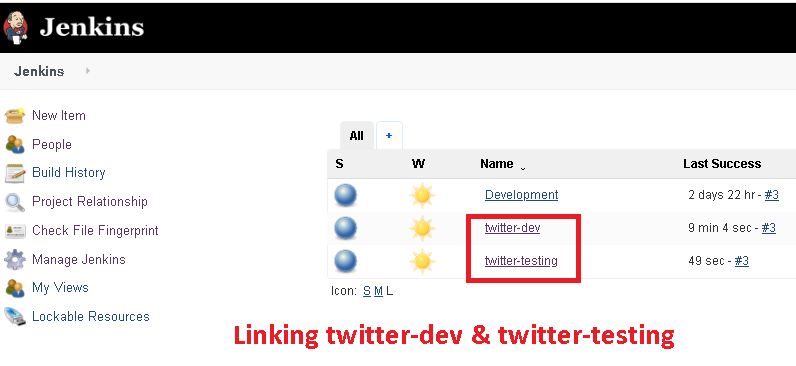
****

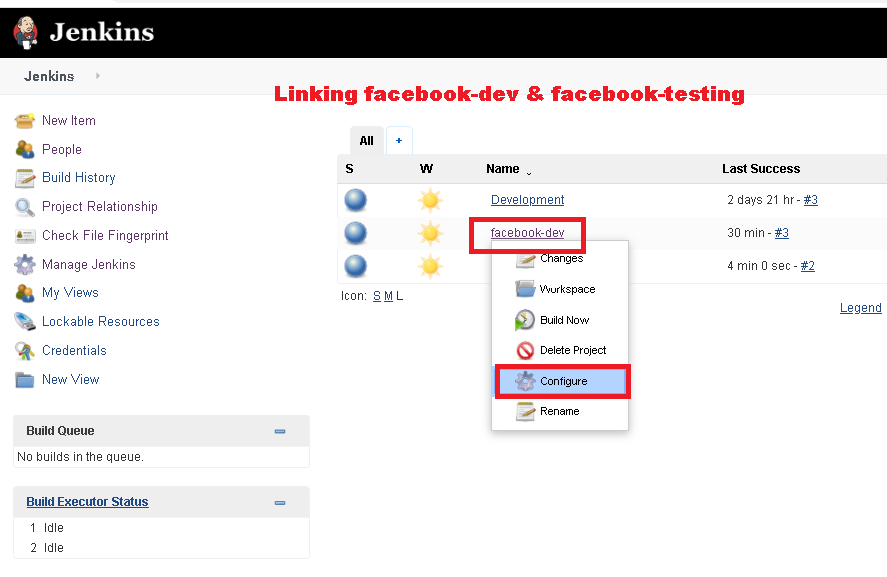
****

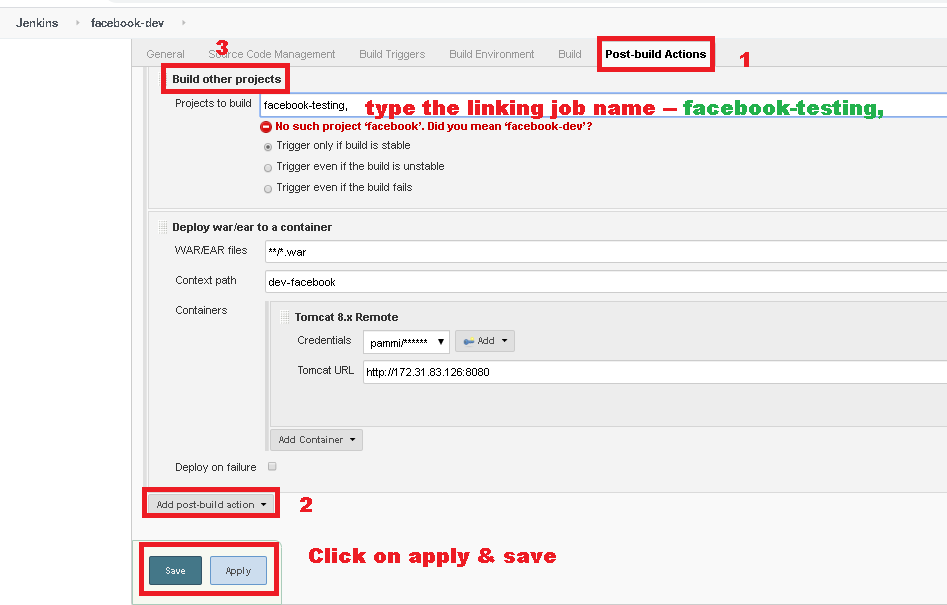
****

****

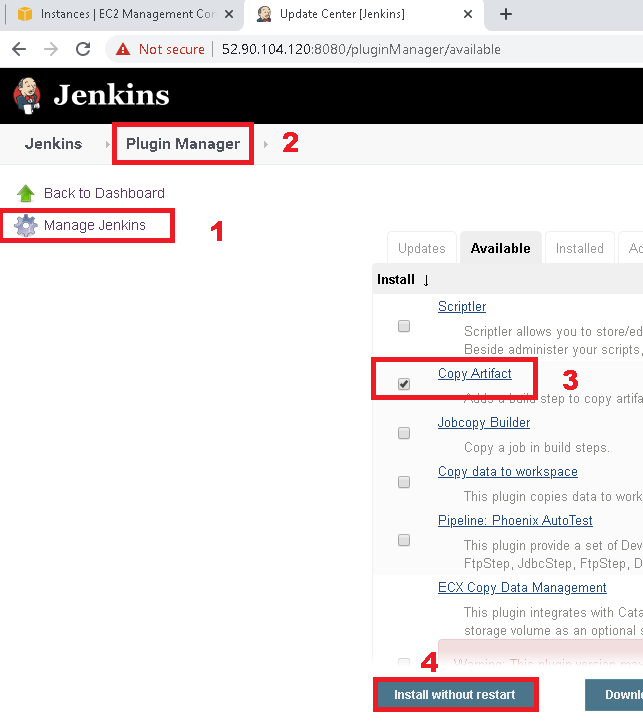
**Linking Development & testing Jobs**

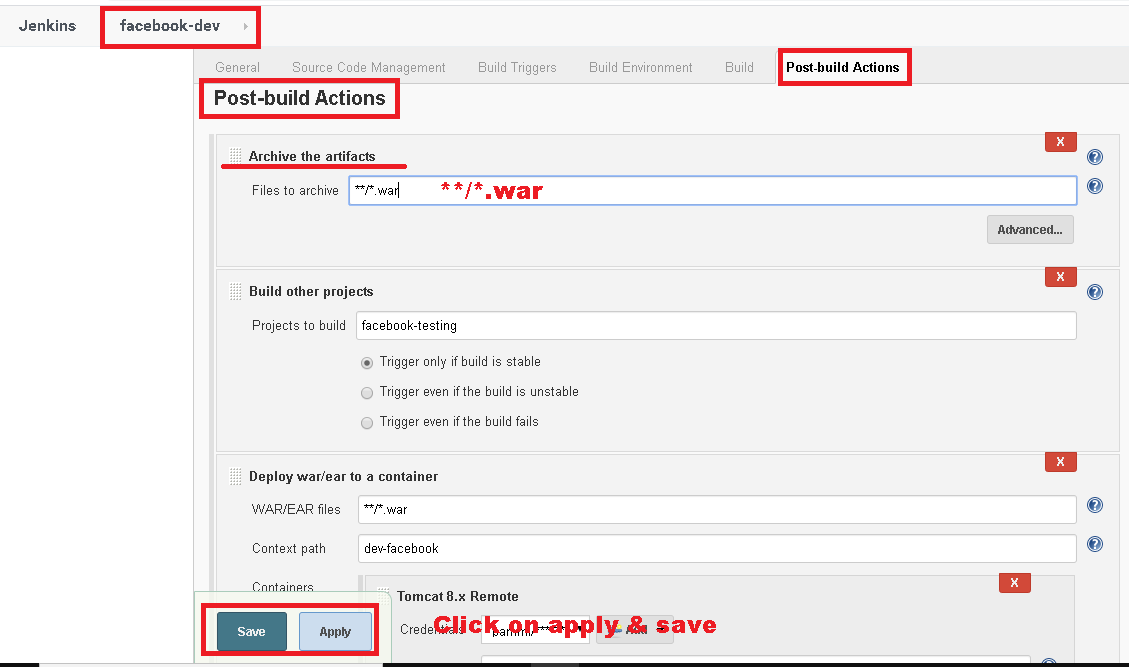
****

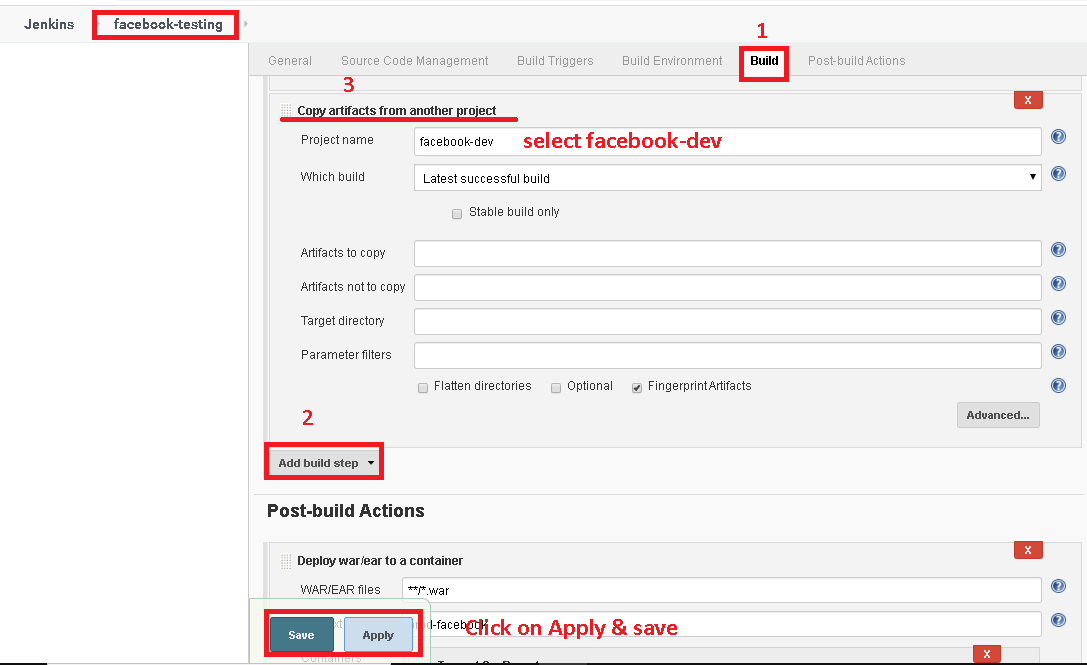
****

****

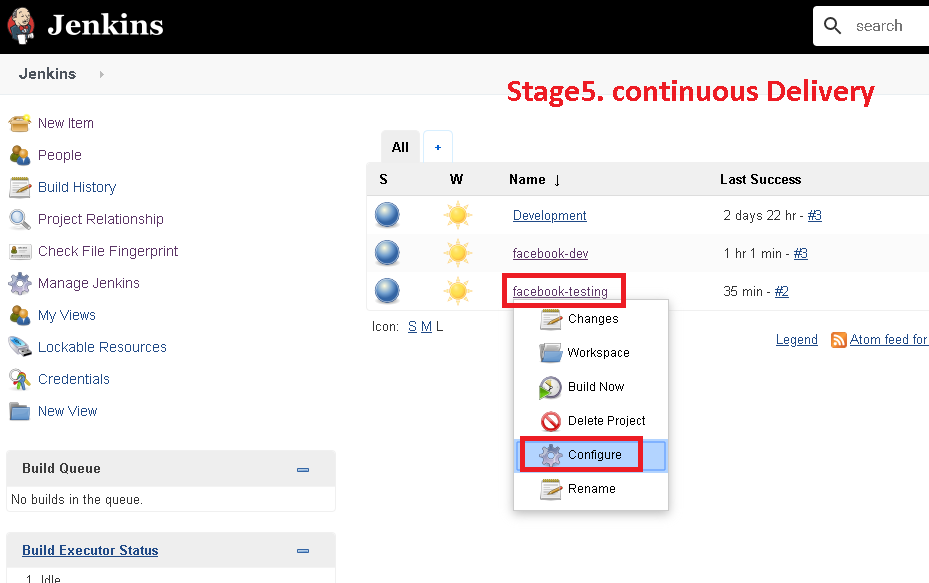
**Copying Artifacts from Development job to Testing job**

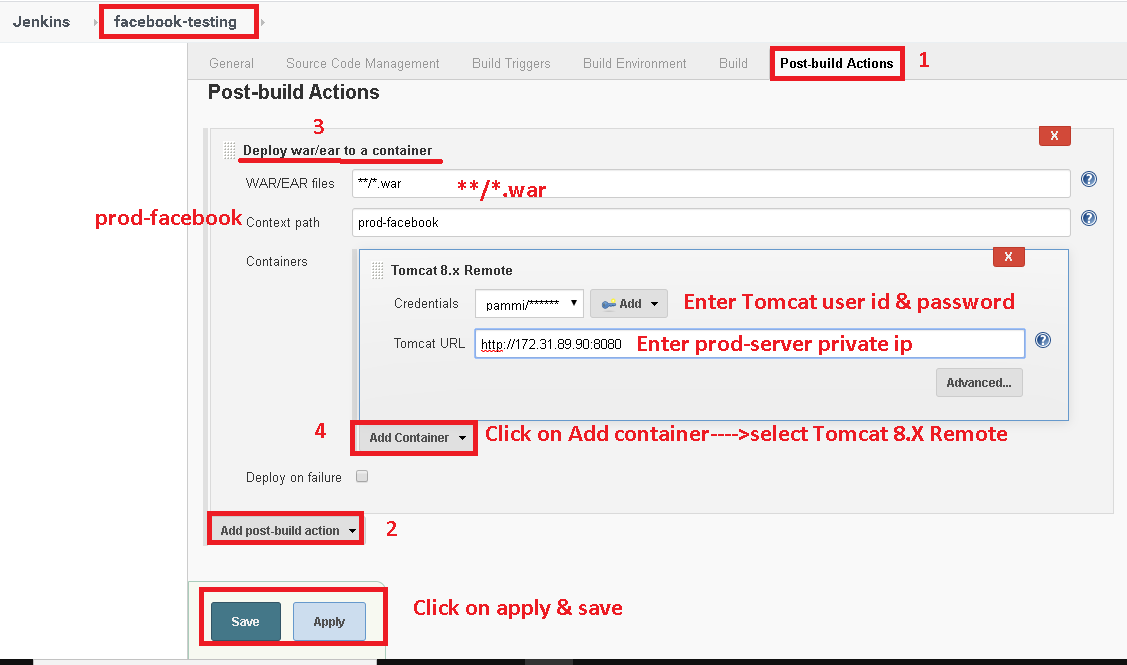
****

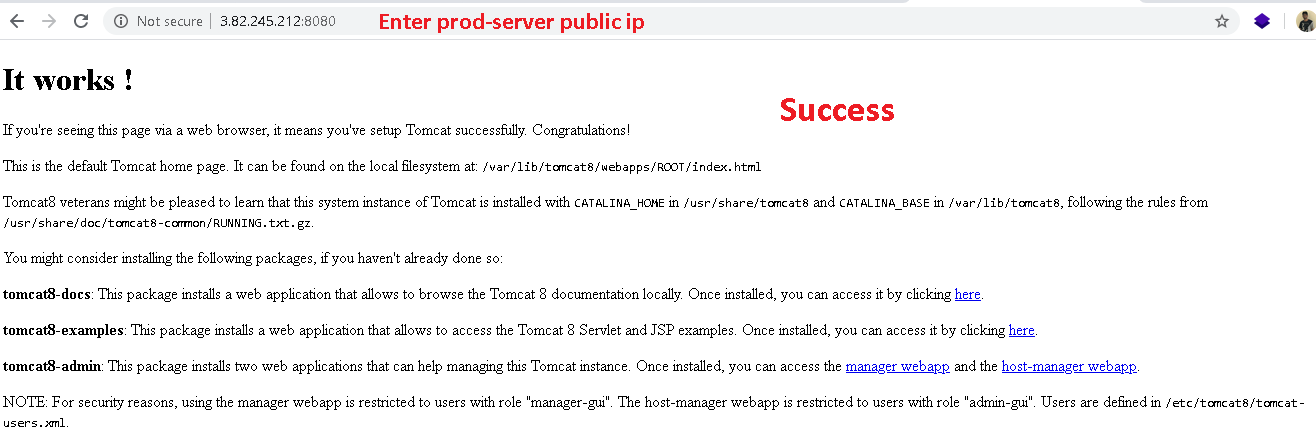
****

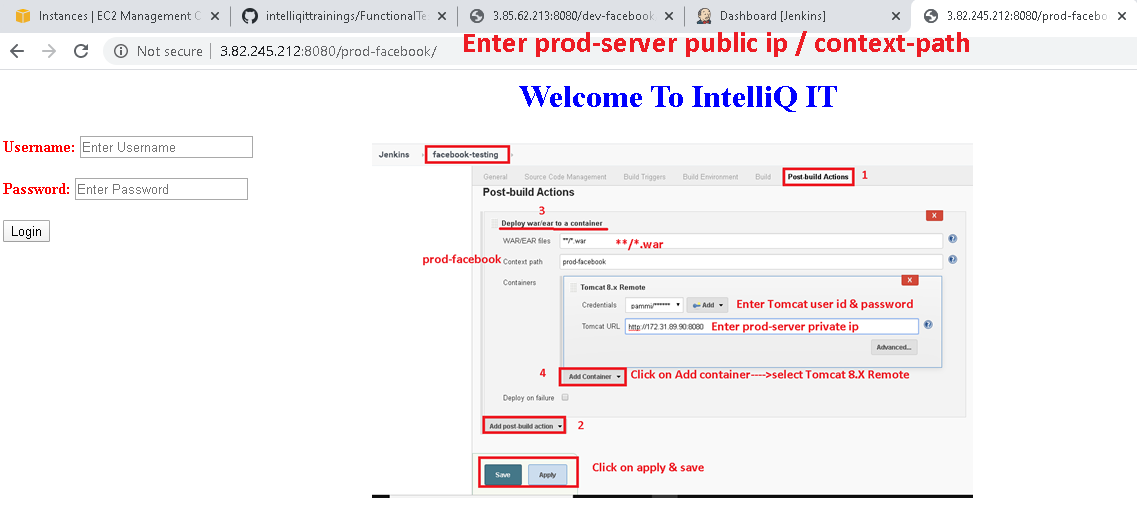
****

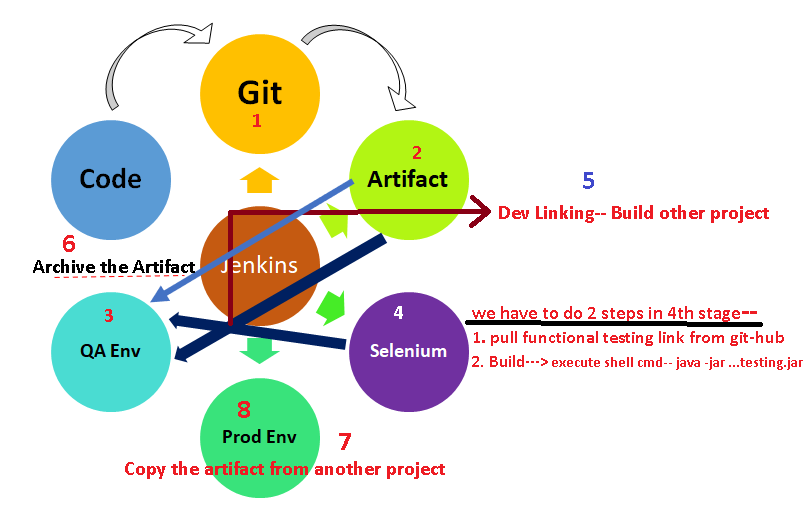
**Stage5. continuous Delivery**





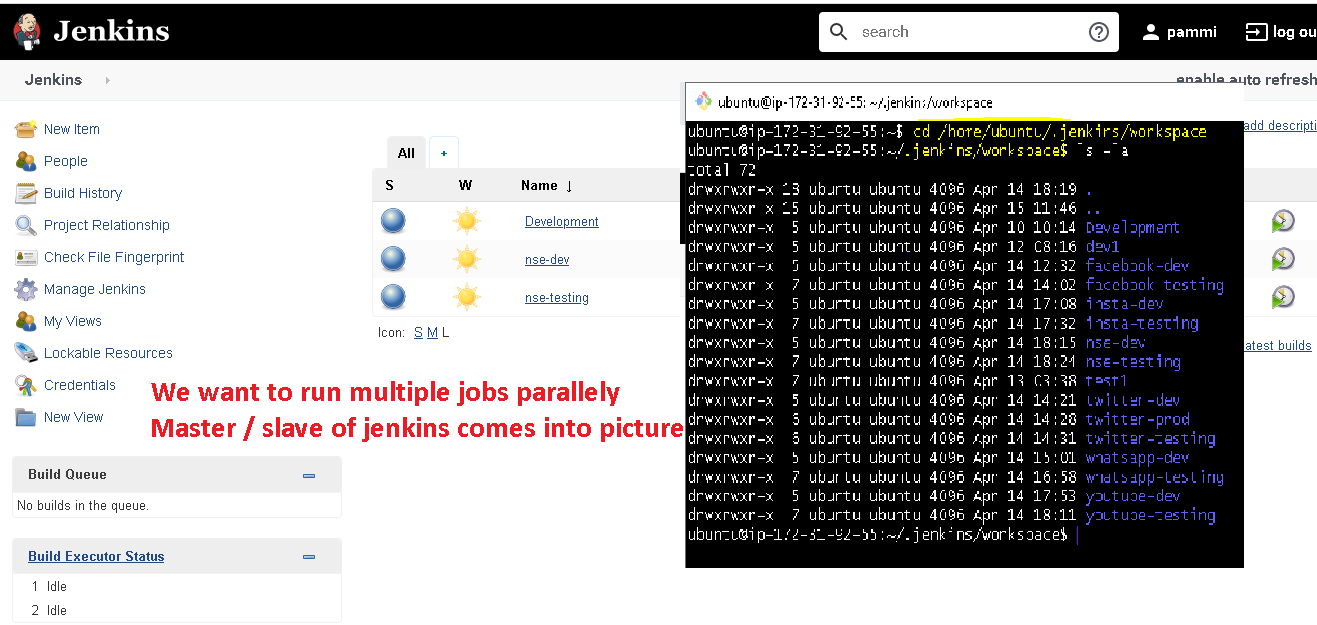




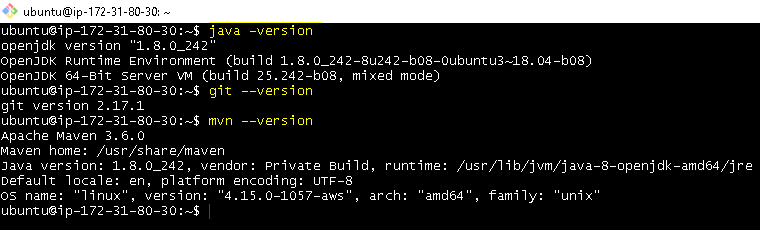


**Master / slave architecture of Jenkins**

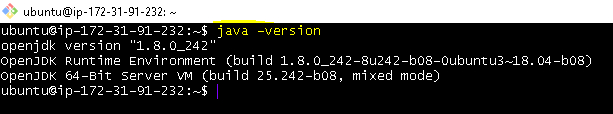
|  |
| --- |
| Master/Slave Setup |
|  | ======================= |
|  | 1 Create a new AWS Ubuntu 18 instance and name it slave |
|  | 2 Connect to the salve machine using git bash |
|  | 3 Set password for default "ubuntu" user |
|  | sudo passwd ubuntu |
|  | 4 Edit the sshd\_config file for allowing ssh access |
|  | sudo vim /etc/ssh/sshd\_config |
|  | Go into insert mode by pressing i |
|  | Search for "PasswordAuthentication" and change it from no to yes |
|  | Save and quit Esc :wq Enter |
|  | 5 Restart ssh |
|  | sudo service ssh restart |
|  | 6 Connect to master machine using git bash |
|  | 7 Generate the ssh keys |
|  | ssh-keygen |
|  | This will generate 2 keys called public and private keys in .ssh folder |
|  | 8 Copy the public keys into slave machine |
|  | ssh-copy-id ubuntu@private\_ip\_of\_slave |
|  | This command will copy the content of public keys into a file called "authorised\_keys" |
|  | on the remote server |
|  |  |
|  | Note: Once the above setup is done the master will be able to do ssh to slave without password |
|  | ssh ubuntu@private\_ip\_of\_slave |

****

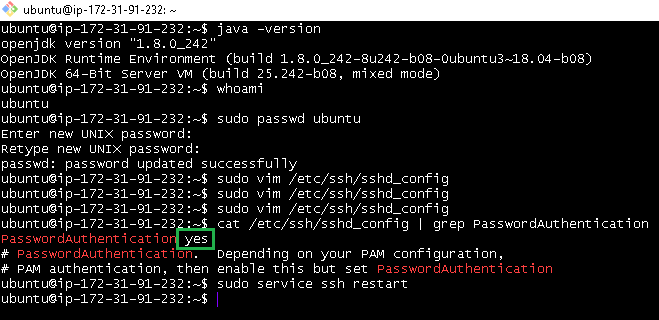
**Master : we need to install below s/w**

****

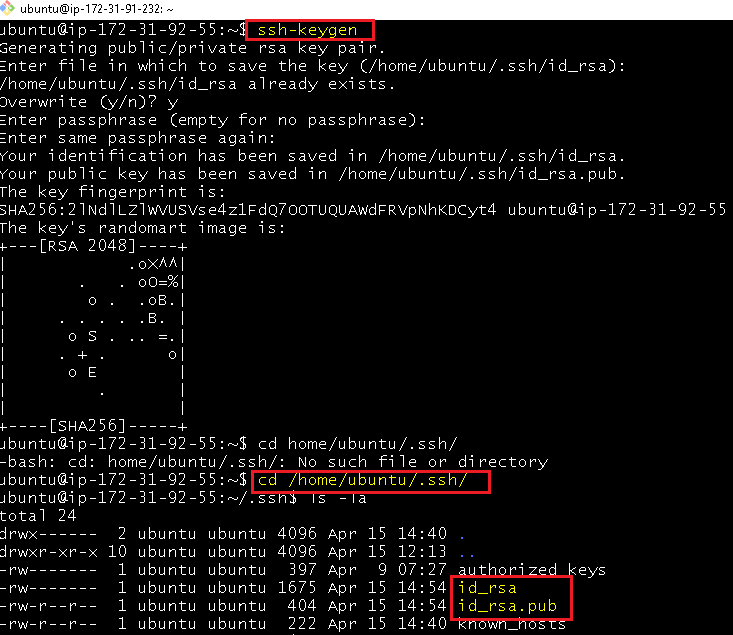
**Slave : we need to install only Java = Jenkins java version**

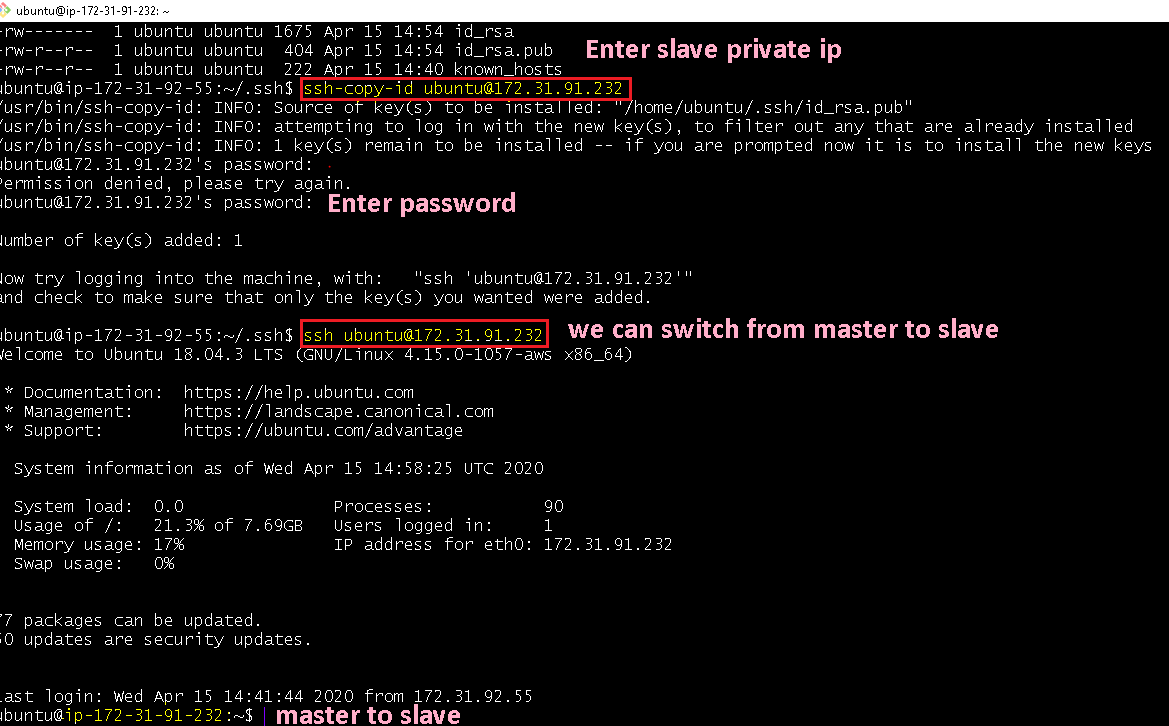
****

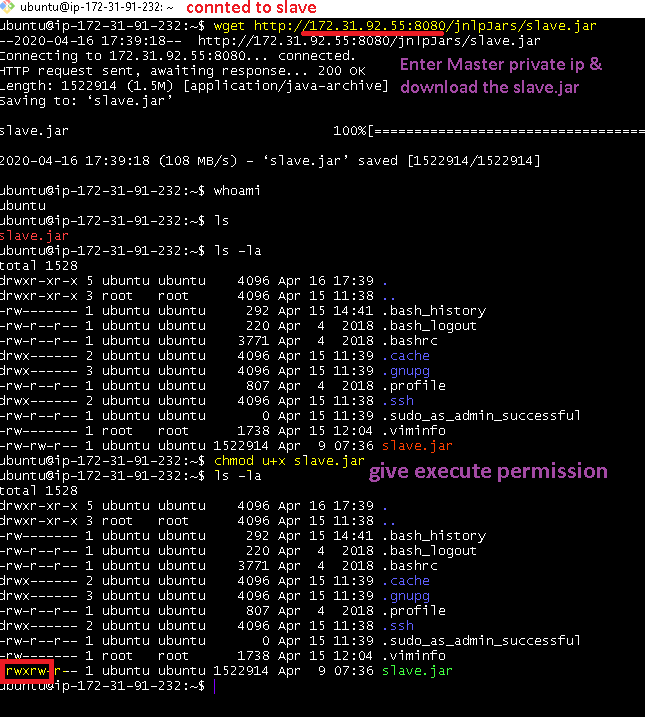
**Slave**

****

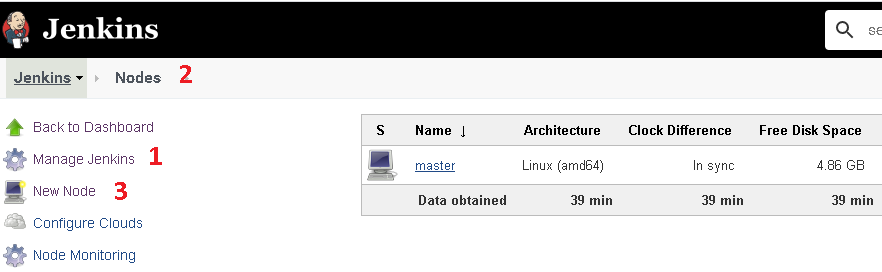
**Master**

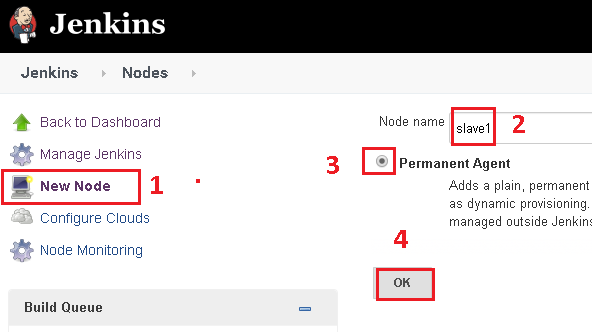
****

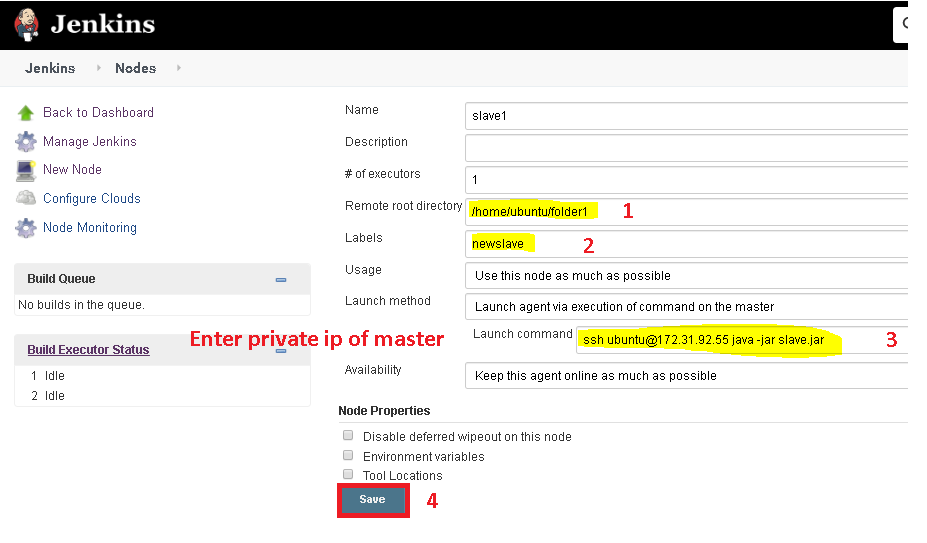
****

****

**Node—Creation**

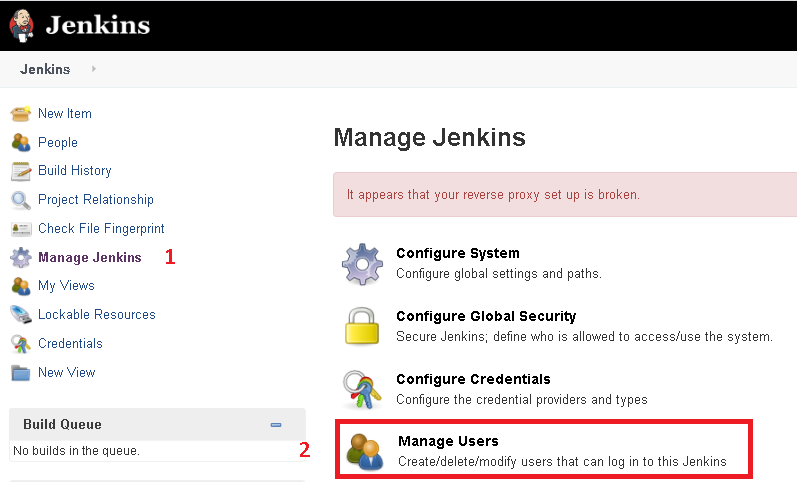
****

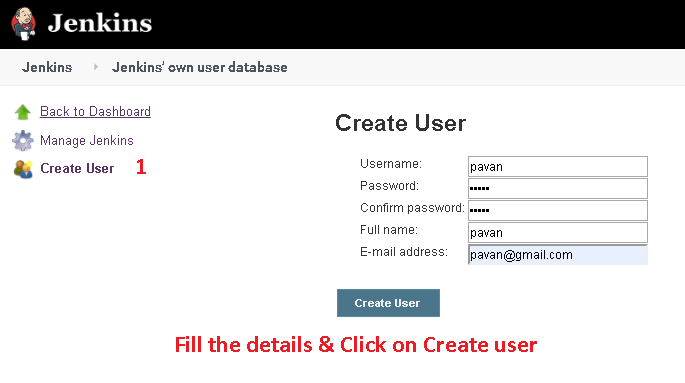
****

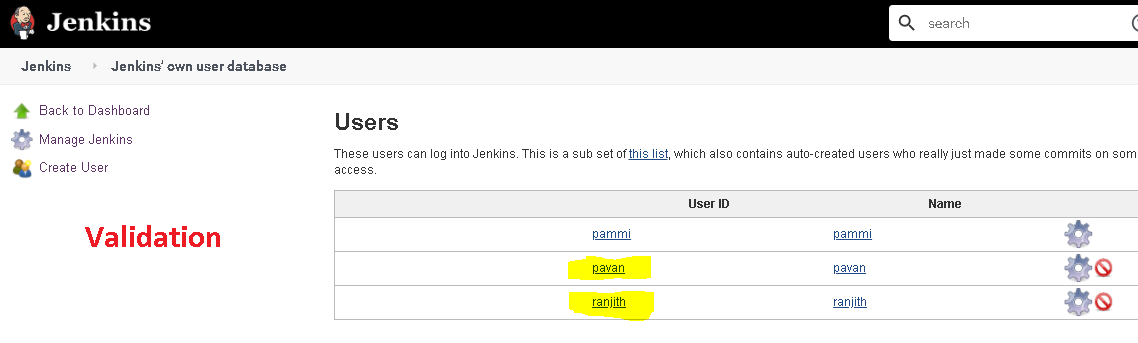
****

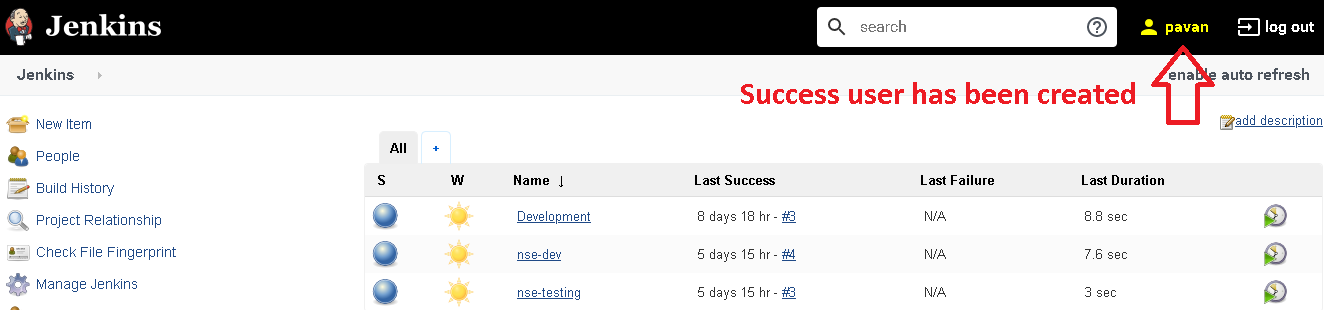
**User Administration**

**Users**

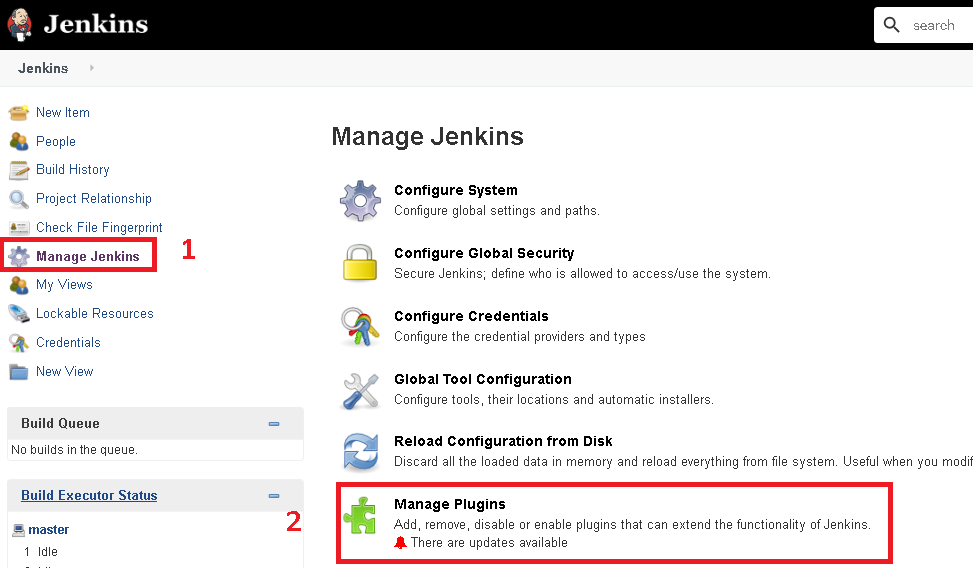
****

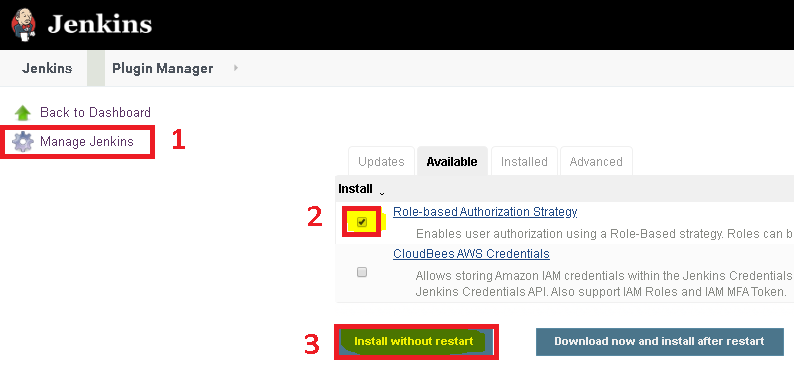
****

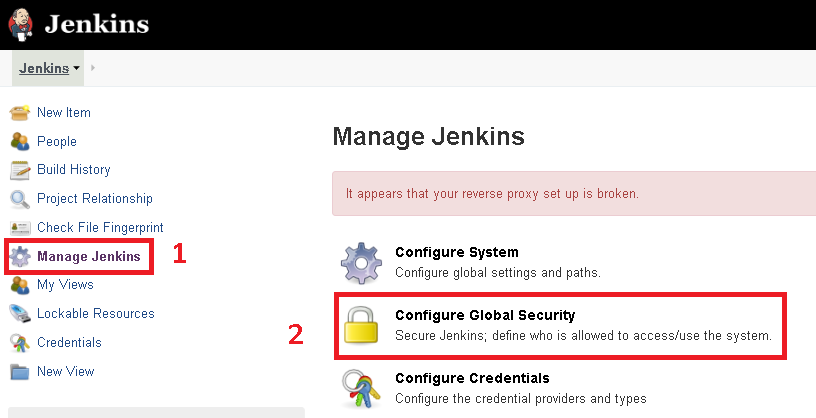
****

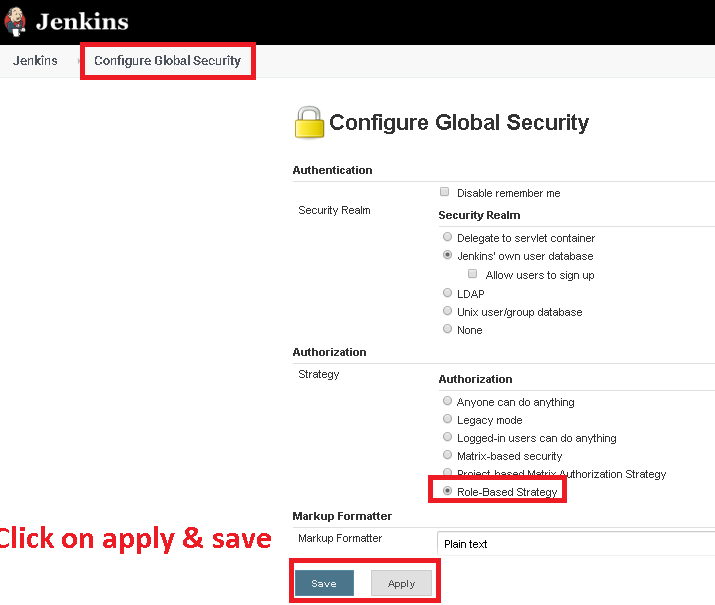
****

**Roles & Assignments**

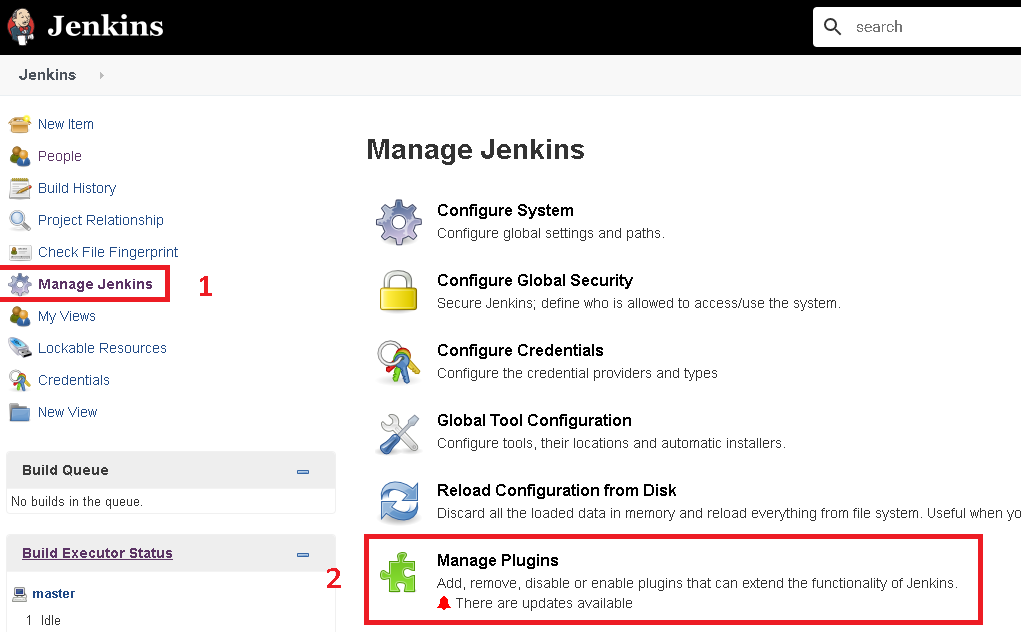
****

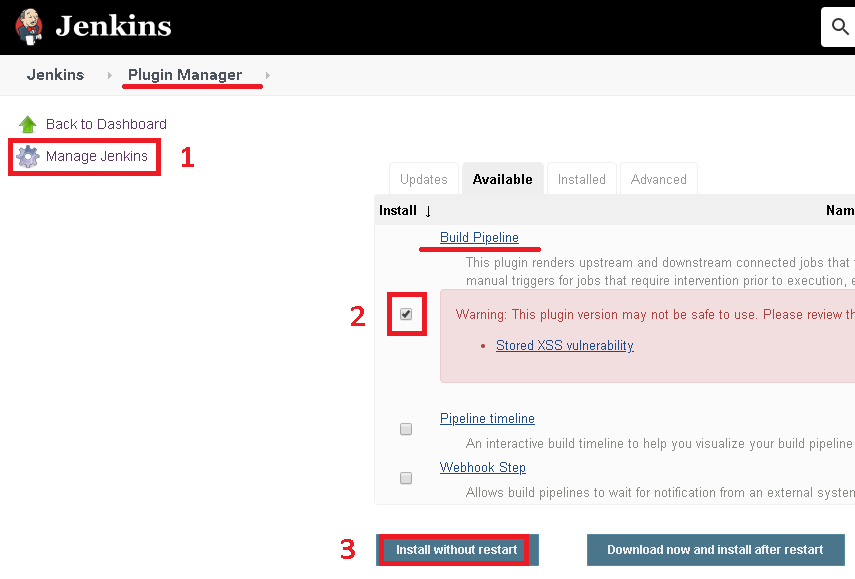
****

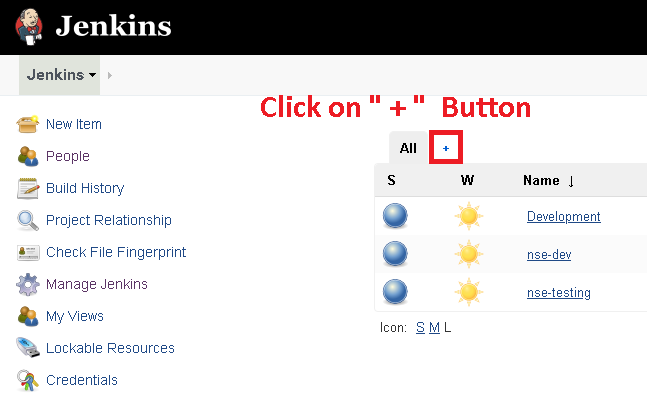
****

****

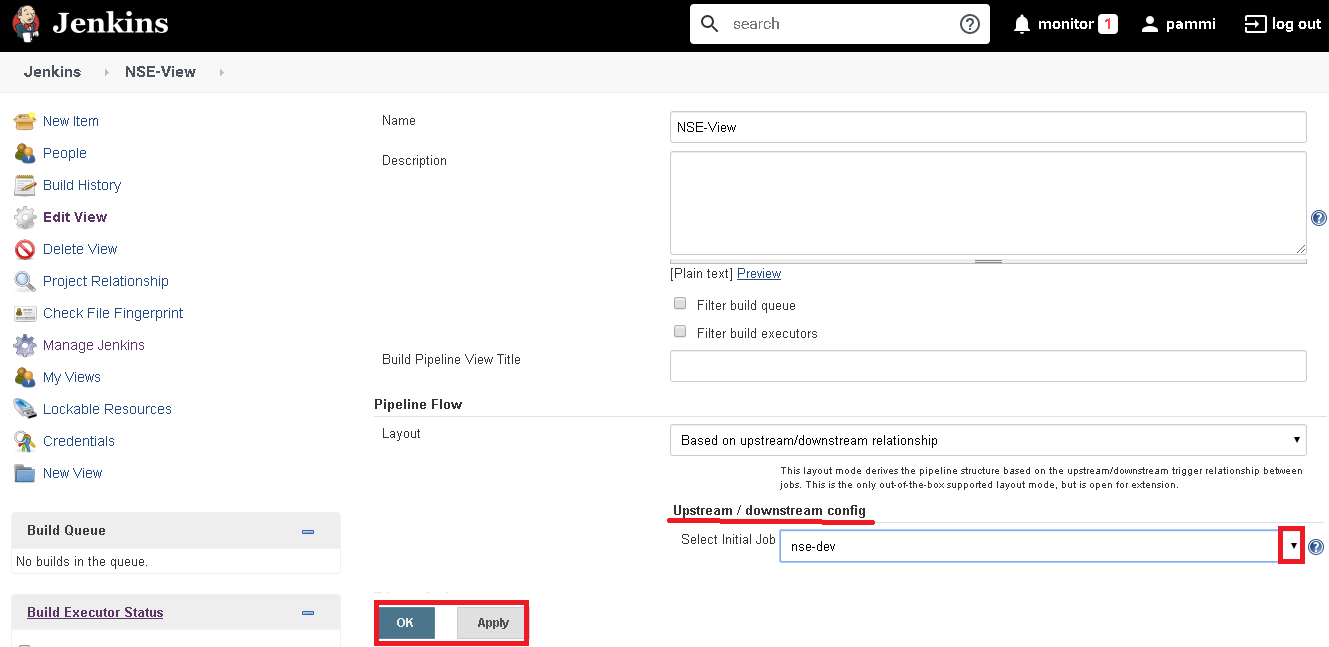
**Build Pipeline plugin**

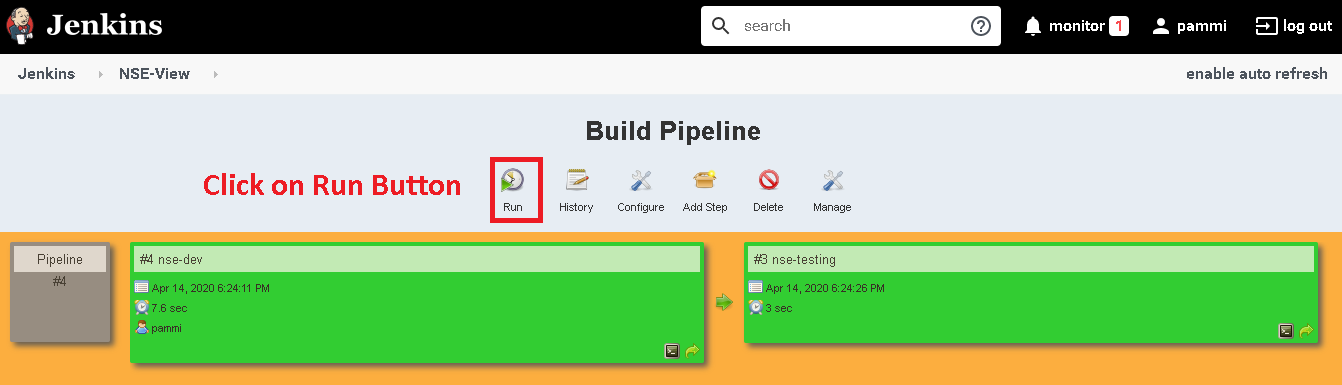
****

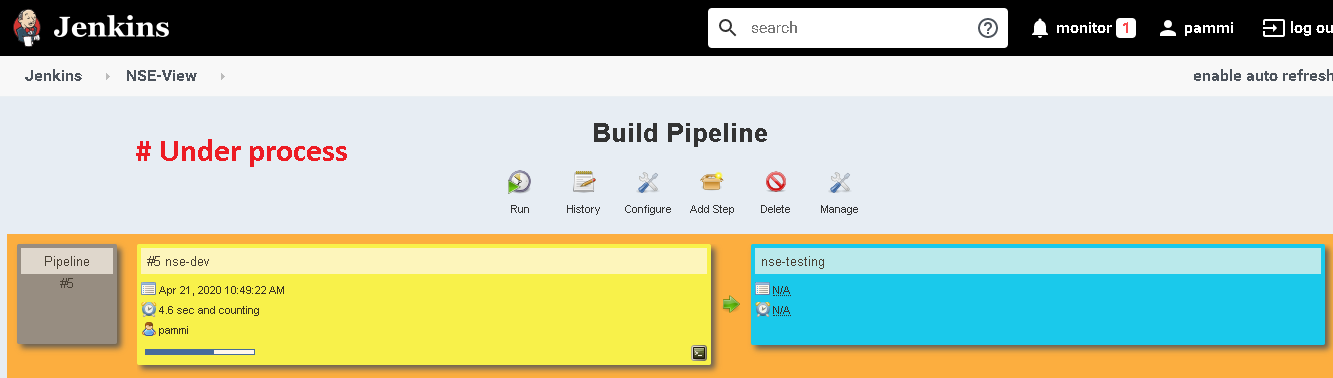
****

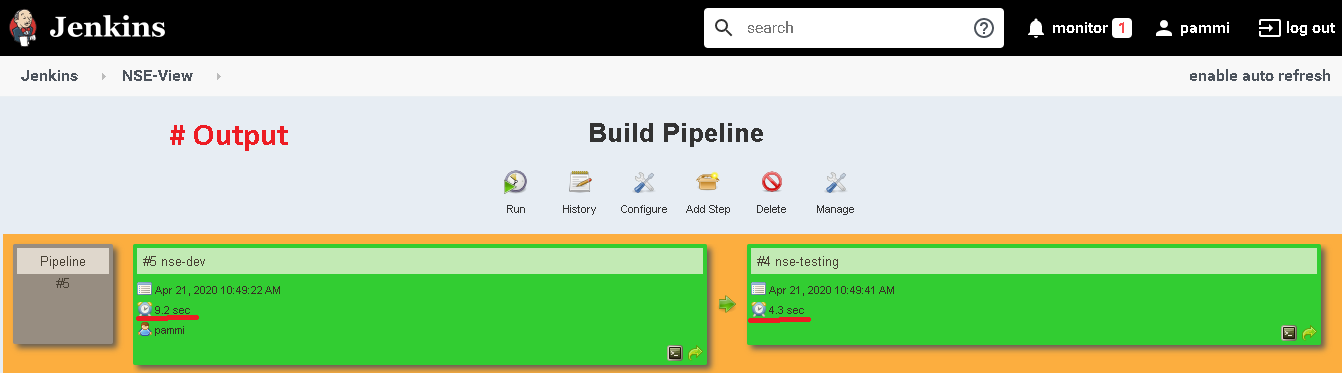
****

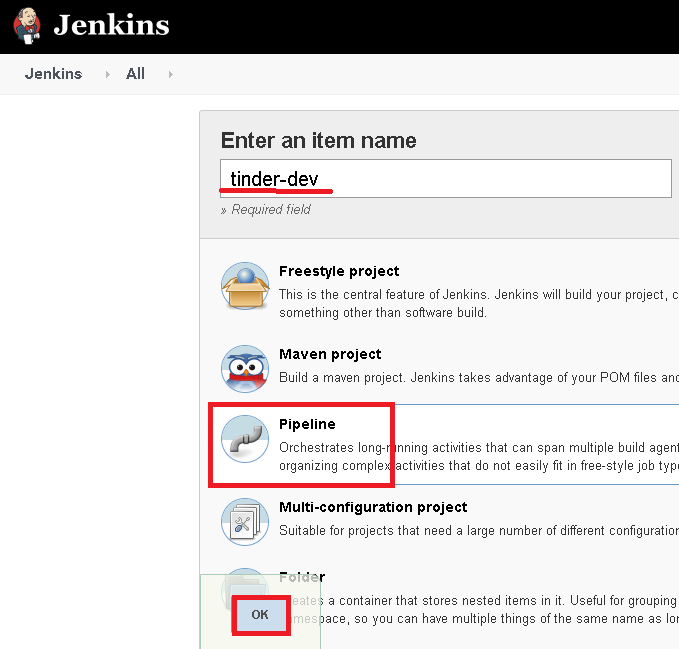
****

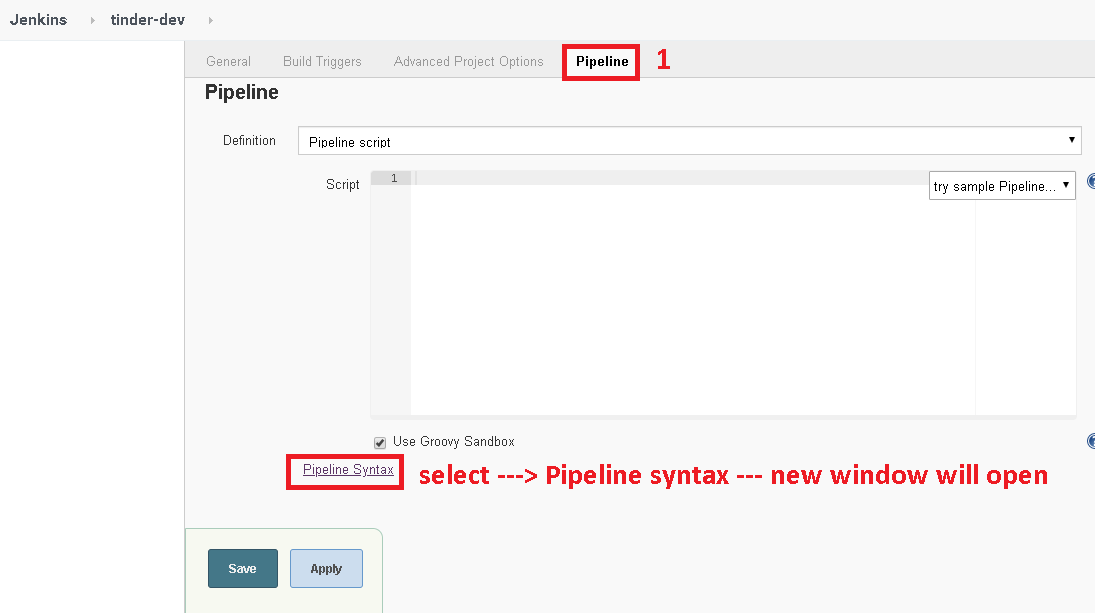
****

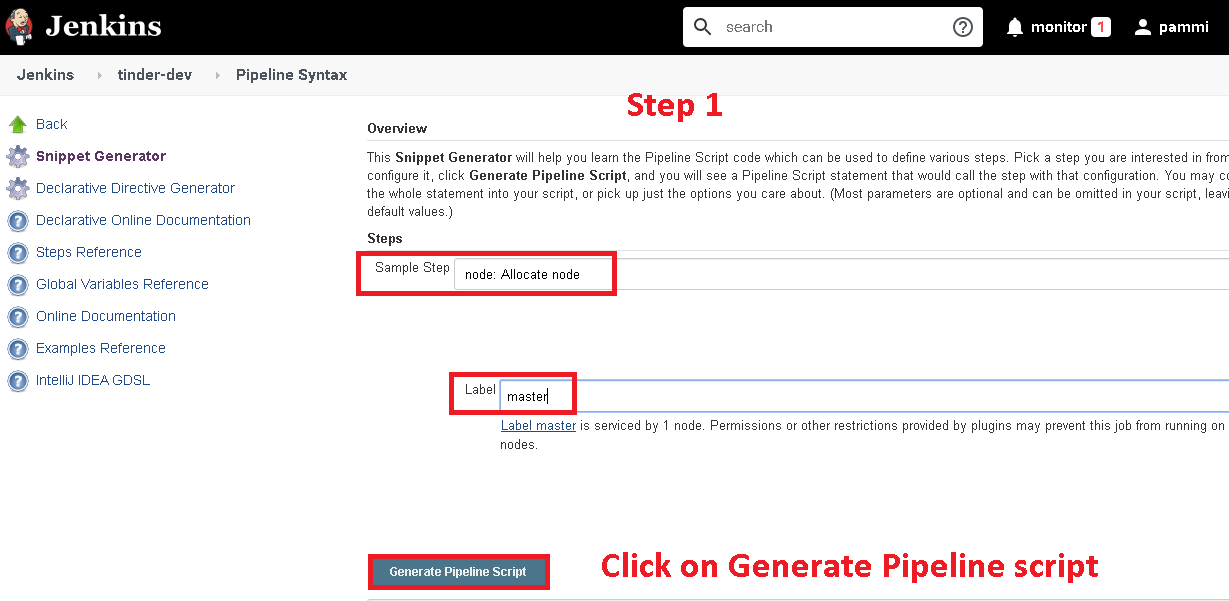
****

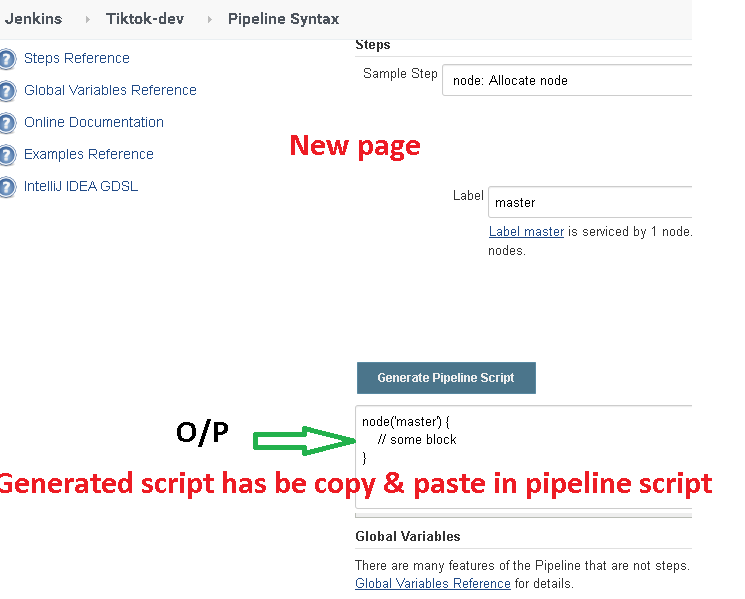
****

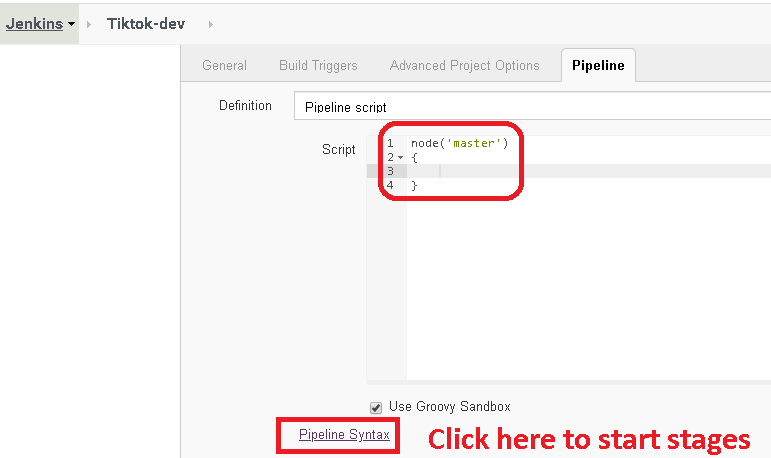
****

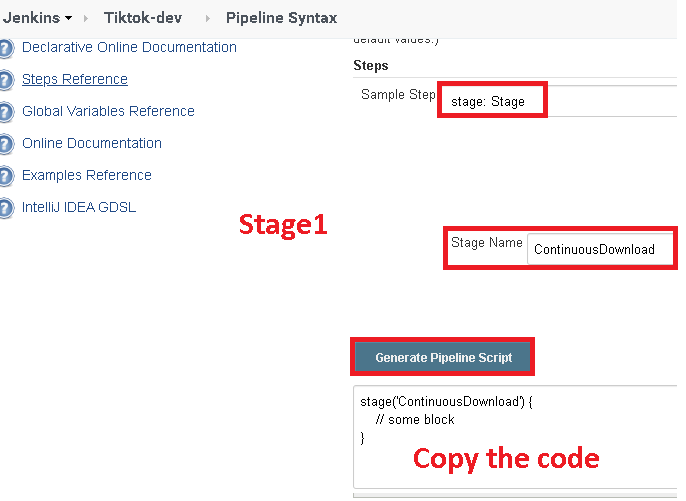
****

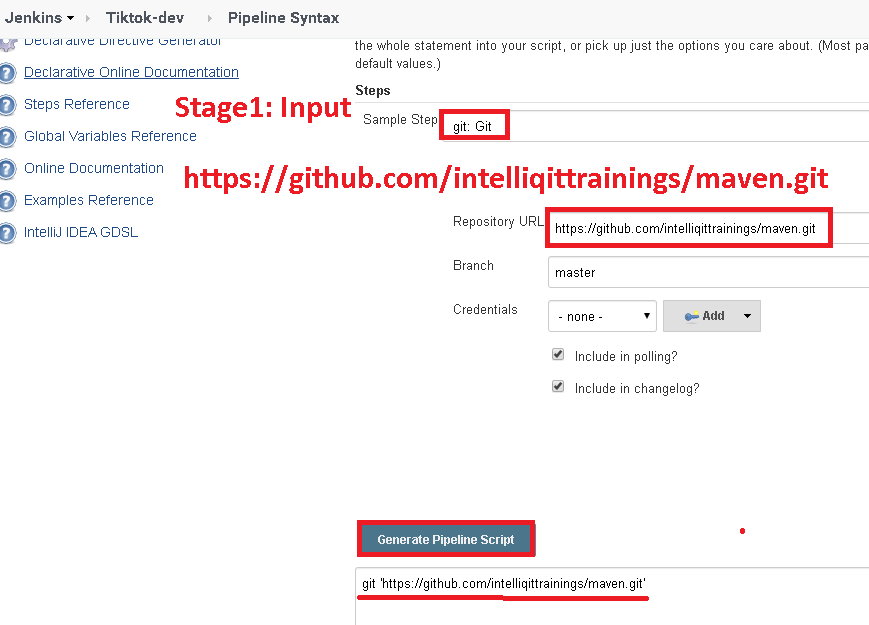
****

****

****

****

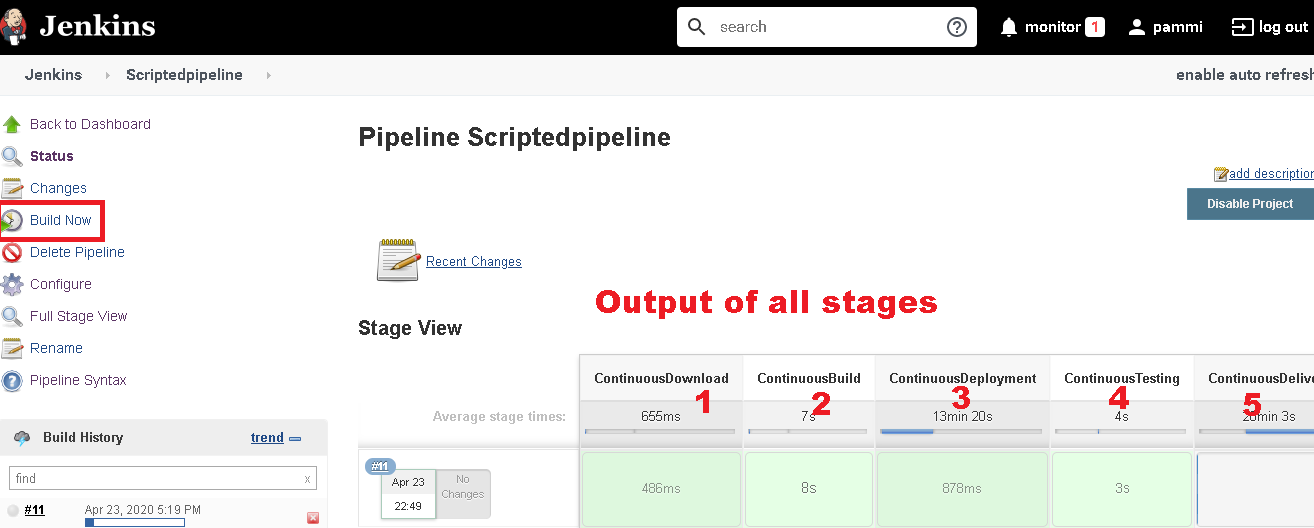
****

****

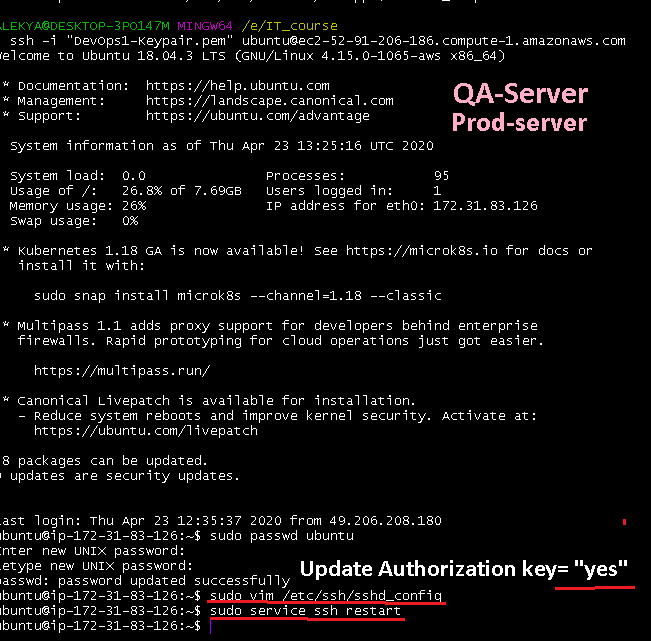
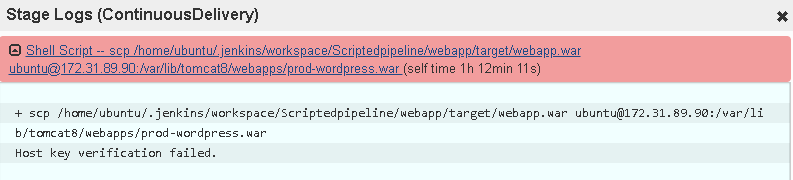
**Pipeline as Code**

**Pipeline can be implemented in 2 ways**

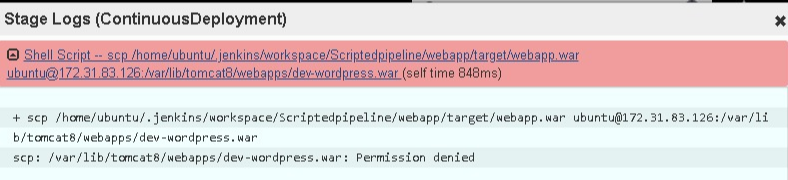
**Proof**

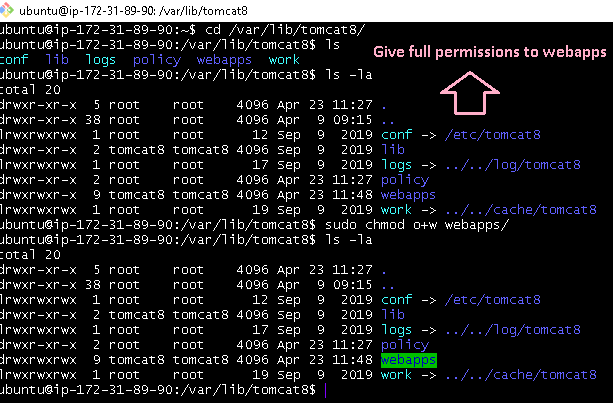
****

**Note: Validate QA/Prod-Server Authorization key= “Yes”**

****

**Note: Validate all the permission are available to Webapp dir**

****

****

**Scripted Pipeline**

**Example1:**

**node('master')**

**{**

**stage('ContinuousDownload')**

**{**

**git 'https://github.com/intelliqittrainings/maven.git'**

**}**

**stage('ContinuousBuild')**

**{**

**sh label: '', script: 'mvn package'**

**}**

**stage('ContinuousDeployment')**

**{**

**sh label: '', script: '''**

**scp /home/ubuntu/.jenkins/workspace/ScriptedPipeline/webapp/target/webapp.war ubuntu@172.31.6.65:/var/lib/tomcat8/webapps/testapp.war'''**

**}**

**stage('ContinuousTesting')**

**{**

**git 'https://github.com/intelliqittrainings/FunctionalTesting.git'**

**sh label: '', script: 'java -jar /home/ubuntu/.jenkins/workspace/ScriptedPipeline/testing.jar'**

**}**

**stage('ContinuousDelivery')**

**{**

**sh label: '', script: '''**

**scp /home/ubuntu/.jenkins/workspace/ScriptedPipeline/webapp/target/webapp.war ubuntu@172.31.2.227:/var/lib/tomcat8/webapps/prodapp.war'''**

**}**

**}**

**Example2--**

**node('master')**

**{**

**stage('ContinuousDownload')**

**{**

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

**}**

**stage('ContinuousBuild')**

**{**

**sh 'mvn package'**

**}**

**stage('ContinuousDeployment')**

**{**

**sh 'scp /var/lib/jenkins/workspace/Pipeline/webapp/target/webapp.war vagrant@10.10.10.32:/var/lib/tomcat7/webapps/qaenv.war'**

**}**

**stage('ContinuousTesting')**

**{**

**git 'https://github.com/selenium-saikrishna/TestingOnLinux.git'**

**sh 'java -jar /var/lib/jenkins/workspace/Pipeline/testing.jar'**

**}**

**stage('ContinuousDelivery')**

**{**

**input message: 'Waiting for approval from DM', submitter: 'Srinivas'**

**sh 'scp /var/lib/jenkins/workspace/Pipeline/webapp/target/webapp.war vagrant@10.10.10.33:/var/lib/tomcat7/webapps/prodenv.war'**

**}**

**}**

**Declarative Pipeline**

**pipeline**

**{**

**agent any**

**stages**

**{**

**stage('ContinousDownload')**

**{**

**steps**

**{**

**git 'https://github.com/intelliqittrainings/maven.git'**

**}**

**}**

**stage('ContinousBuild')**

**{**

**steps**

**{**

**sh label: '', script: 'mvn package'**

**}**

**}**

**stage('ContinousDeployment')**

**{**

**steps**

**{**

**sh label: '', script: 'scp /home/ubuntu/.jenkins/workspace/DeclarativePipeline/webapp/target/webapp.war ubuntu@172.31.18.176:/var/lib/tomcat8/webapps/testapp.war'**

**}**

**}**

**stage('ContinousTesting')**

**{**

**steps**

**{**

**git 'https://github.com/ intelliqittrainings /FunctionalTesting.git'**

**sh label: '', script: 'java -jar /home/ubuntu/.jenkins/workspace/DeclarativePipeline/testing.jar'**

**}**

**}**

**stage('ContinousDelivery')**

**{**

**steps**

**{**

**sh label: '', script: 'scp /home/ubuntu/.jenkins/workspace/DeclarativePipeline/webapp/target/webapp.war ubuntu@172.31.21.163:/var/lib/tomcat8/webapps/prodapp.war'**

**}**

**}**

**}**

**}**

**Mail Notification via Declarative pipeline script**

**pipeline**

**{**

**agent any**

**stages**

**{**

**stage('ContinousDownload')**

**{**

**steps**

**{**

**script**

**{**

**try**

**{**

**git 'https://github.com/intelliqittrainings/maven.git'**

**}**

**catch(Exception e1)**

**{**

**mail bcc: '', body: 'Jenkins is not able to download from the remote git repo', cc: '', from: '', replyTo: '', subject: 'Unable to Download from github', to: 'git\_admin@gmail.com'**

**exit(1);**

**}**

**}**

**}**

**}**

**stage('ContinousBuild')**

**{**

**steps**

**{**

**script**

**{**

**try**

**{**

**sh label: '', script: 'mvn package'**

**}**

**catch(Exception e2)**

**{**

**mail bcc: '', body: 'Jenkins is not able to create an artifact using maven', cc: '', from: '', replyTo: '', subject: 'Build failed', to: 'developers@gmail.com'**

**exit(1);**

**}**

**}**

**}**

**}**

**stage('ContinousDeployment')**

**{**

**steps**

**{**

**script**

**{**

**try**

**{**

**sh label: '', script: 'scp /home/ubuntu/.jenkins/workspace/DeclarativePipeline/webapp/target/webapp.war ubuntu@172.31.18.176:/var/lib/tomcat8/webapps/testapp.war'**

**}**

**catch(Exception e3)**

**{**

**mail bcc: '', body: 'Jenkins is not able to deploy into tomcat on the QAServers', cc: '', from: '', replyTo: '', subject: 'Deployment Failed', to: 'middleware@gmail.com'**

**exit(1);**

**}**

**}**

**}**

**}**

**stage('ContinousTesting')**

**{**

**steps**

**{**

**script**

**{**

**try**

**{**

**git 'https://github.com/selenium-saikrishna/FunctionalTesting.git'**

**sh label: '', script: 'echo "Testing Passed"'**

**}**

**catch(Exception e4)**

**{**

**mail bcc: '', body: 'Functional Testing has failed', cc: '', from: '', replyTo: '', subject: 'TestingFailed', to: 'testers@gmail.com'**

**exit(1);**

**}**

**}**

**}**

**}**

**stage('ContinousDelivery')**

**{**

**steps**

**{**

**script**

**{**

**try**

**{**

**sh label: '', script: 'scp /home/ubuntu/.jenkins/workspace/DeclarativePipeline/webapp/target/webapp.war ubuntu@172.31.21.163:/var/lib/tomcat8/webapps/prodapp.war'**

**}**

**catch(Exception e5)**

**{**

**mail bcc: '', body: 'Jenkins is not able to deploy into production servers', cc: '', from: '', replyTo: '', subject: 'Delivery Failed', to: 'deliveryteam@gmail.com'**

**exit(1);**

**}**

**}**

**}**

**}**

**}**

**}**

**Post- Declarative**

**pipeline**

**{**

**agent any**

**stages**

**{**

**stage('ContinousDownload')**

**{**

**steps**

**{**

**git 'https://github.com/intelliqittrainings/maven.git'**

**}**

**}**

**stage('ContinousBuild')**

**{**

**steps**

**{**

**sh label: '', script: 'mvn package'**

**}**

**}**

**stage('ContinousDeployment')**

**{**

**steps**

**{**

**sh label: '', script:**

**'scp /home/ubuntu/.jenkins/workspace/Post-DeclarativePipeline/webapp/target/webapp.war ubuntu@172.31.83.126:/var/lib/tomcat8/webapps/dev-khazana.war'**

**}**

**}**

**stage('ContinousTesting')**

**{**

**steps**

**{**

**git 'https://github.com/intelliqittrainings/FunctionalTesting.git'**

**sh label: '', script: 'echo "Testing Passed"'**

**}**

**}**

**}**

**post**

**{**

**success**

**{**

**sh label: '', script:**

**'scp /home/ubuntu/.jenkins/workspace/Post-DeclarativePipeline/webapp/target/webapp.war ubuntu@172.31.89.90:/var/lib/tomcat8/webapps/prod-khazana.war'**

**}**

**failure**

**{**

**mail bcc: '', body: 'Jenkins has failed to perform ci-cd',**

**cc: 'gitadmin@gmail.com,testers@gmail.com,middleware@gmail.com',**

**from: '', replyTo: '', subject: 'Jenkins job failed',**

**to: 'developers@gmail.com'**

**}**

**}**

**}**