**CICD SETUP IN SINGLE BOX**

**Prerequisites :**

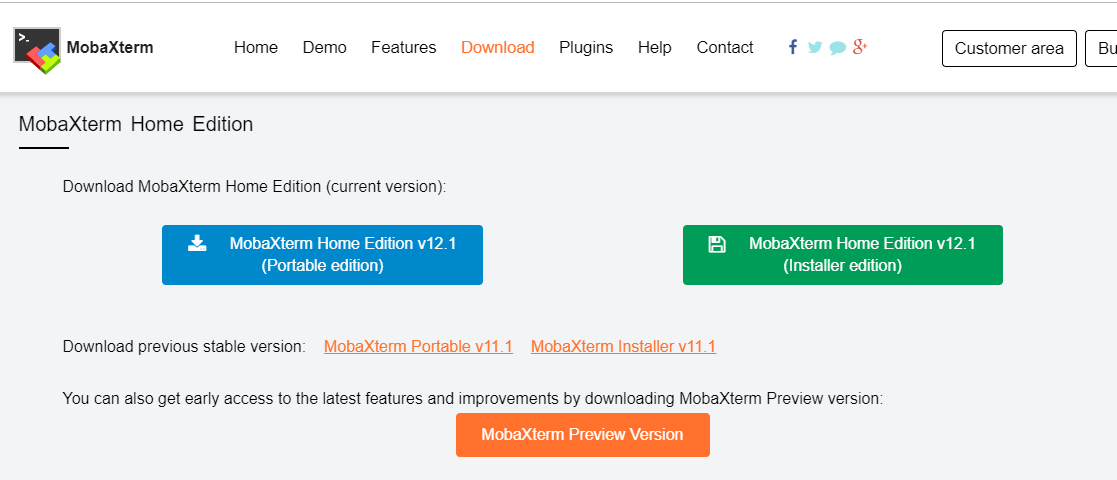
1. Mobaxterm
2. Centos Image
3. GitHub Account
4. Jenkins Server
5. Nexus Server
6. Ansible Server
7. Tomcat Server

**List Of Tasks:**

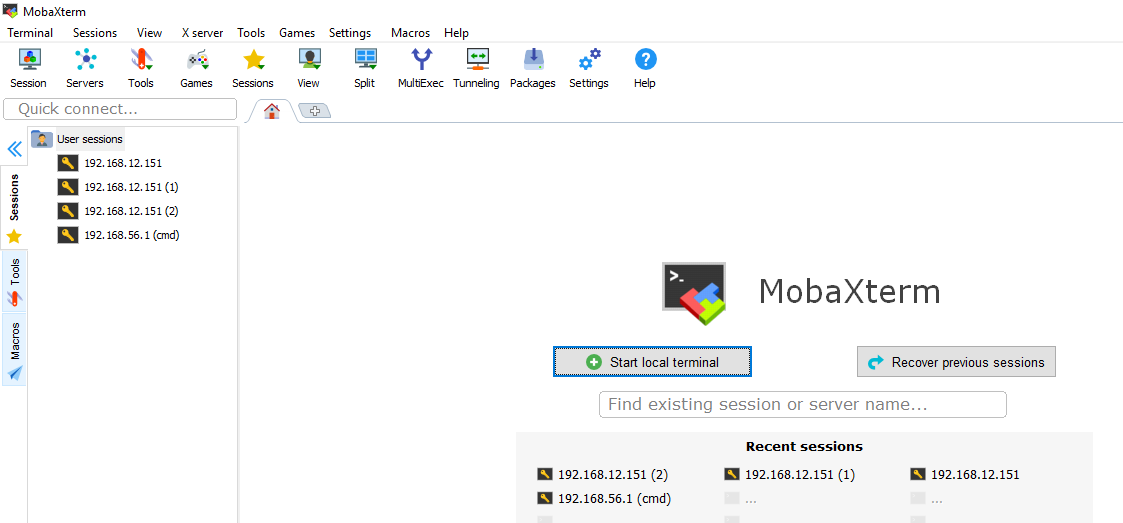
**1) Install Mobaxterm In Your Local Machine(Windows).**  
===================================

* First you need to obtain a copy of the installation package
* Download the mobaxterm package from internet and install it in your local machine

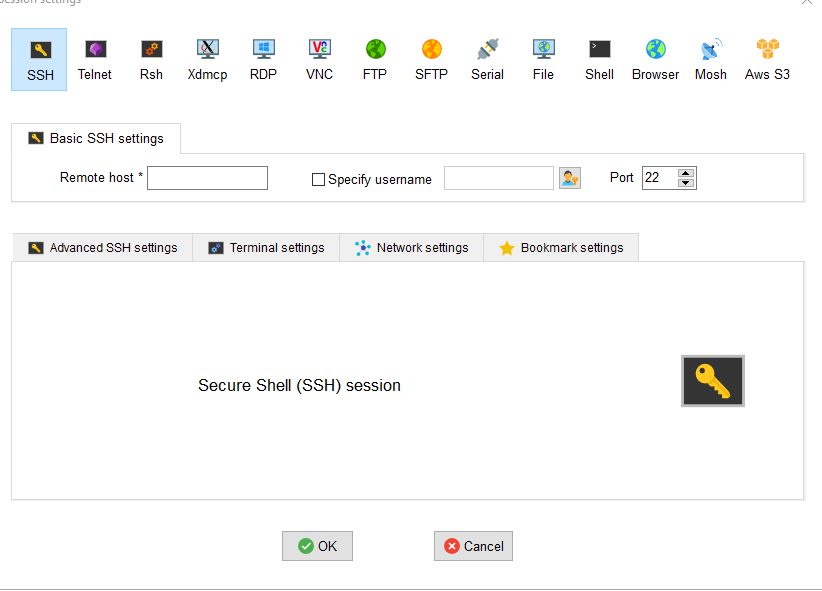
<https://mobaxterm.mobatek.net/download-home-edition.html>



* After downloading the file double click on installer file and run as administrator.
* After the installing mobaxterm in your local machine.
* Use mobzxterm to connect to server.

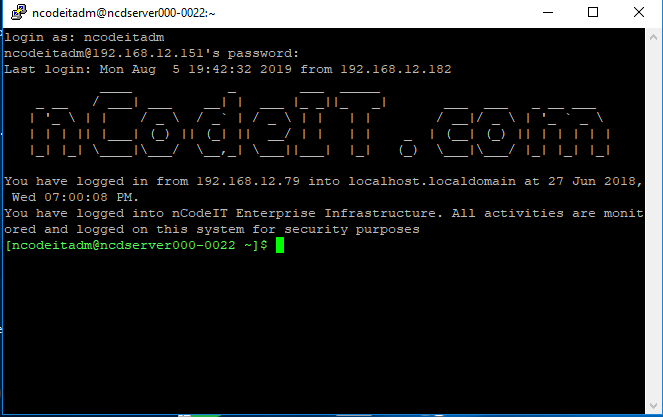


* Now Open mobaxterm and enter Remote host or ip adress of server and click on ok.



* Enter the username and password.  
  Username:ncodeitadm

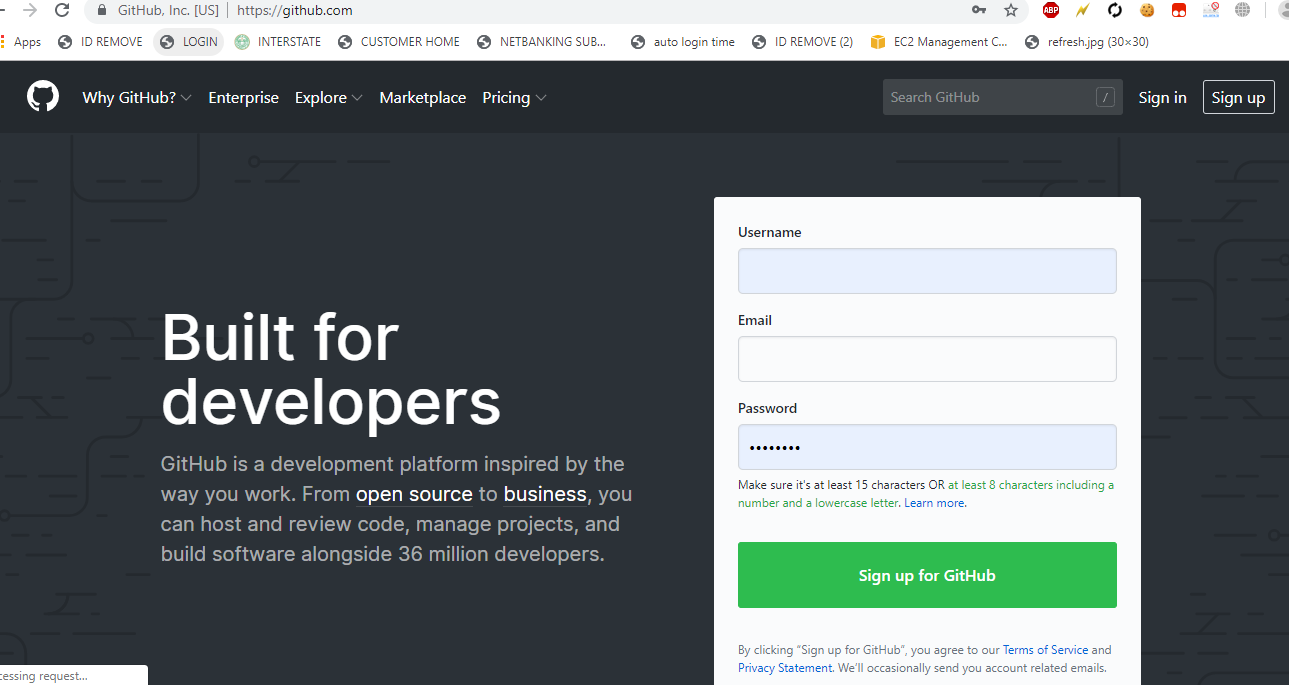
Password:ncodeit123



1. **Create one github account**.

======================

* Go to the GitHub sign up page(https://github.com)
* Enter a username, valid email address, and password and click on signup for github



* Choose a plan. You have two choice: Free and paid,Click on free version.
* You finished! Your Github account created!

1. **Installation of Git on centos7 and Configuring Git.**

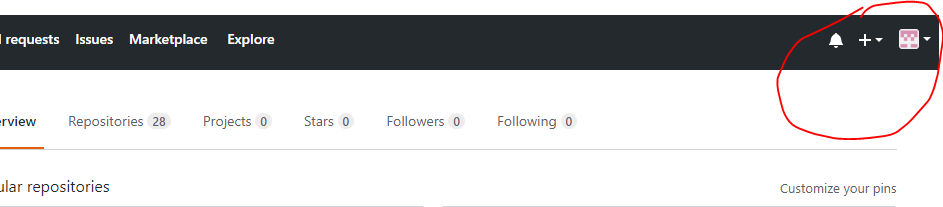
**========================================**

* sudo yum install git
* If the command completes without error, you will have git downloaded and installed. To double-check that it is working correctly,
* try running Git's built-in version check
* git --version
* Git has been successfully installed.
* Now you need to configure git with email and name globally.
* git config --global user.name "Your Name"
* git config --global user.email "you@example.com"
* To cofirm the configurations
* git config --list

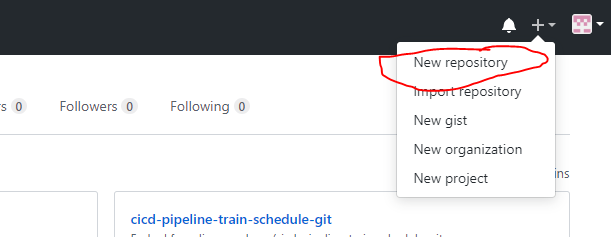
1. **Create a Repository in Github.**

**==========================**

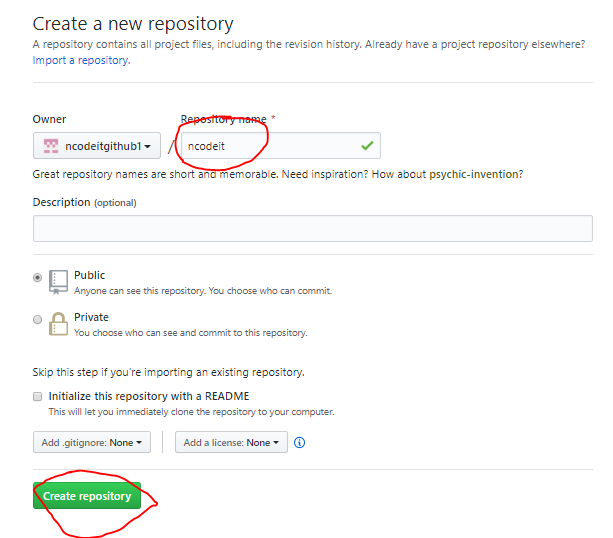
* Login to your Github account(https://github.com).
* At the top right of any Github page, you should see a '+' icon. Click on it.



* Then select 'New Repository'.



* Give your repository a name.



* Click 'Create Repository'.
* Done!!

1. **Pull the code to local machine**

**=========================**

* Pull the code to your local machine(Same directory which you created)from github.
* git pull <https://github.com/ncodeitgithub1/spring3-mvc-maven-xml-hello-world>

1. **Pushing the code to your github account.**

**=============================**

* Add the files to staging area
* git add .
* Commit the added files.
* git commit -m “enter your message”
* Add remote origin to your git
* git remote add origin <your github account repo url>
* Push the Commited file to Git hub.
* git push -u origin master
* Now you can go to github and check in your repo
* Done !!

**7) Installing Jenkins.**

**================**

* Jenkins is a Java application, so the first step is to install Java. Run the following command to install the OpenJDK 8 package.
* sudo yum install java-1.8.0-openjdk-devel
* The next step is to enable the Jenkins repository. To do that, import the GPG key using the following curl command.
* curl --silent --location <http://pkg.jenkins-ci.org/redhat-stable/jenkins.repo> | sudo tee /etc/yum.repos.d/jenkins.repo

And add the repository to your system with:

-> sudo rpm --import <https://jenkins-ci.org/redhat/jenkins-ci.org.key>

Once the repository is enabled, install the latest stable version of Jenkins by typing:

-> sudo yum install jenkins

After the installation process is completed, start the Jenkins service with:

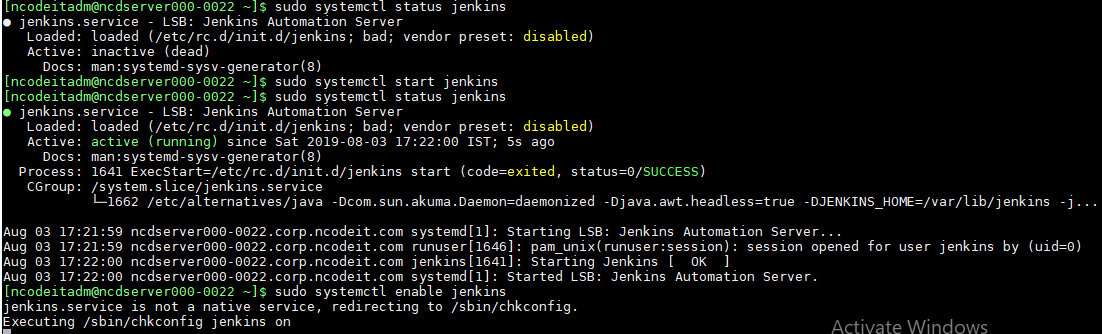
-> sudo systemctl start jenkins

To check whether it started successfully run:

-> systemctl status jenkins

Finally enable the Jenkins service to start on system boot.

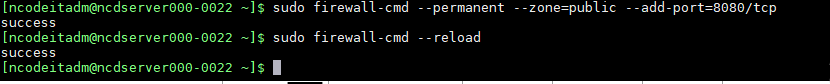
-> sudo systemctl enable jenkins



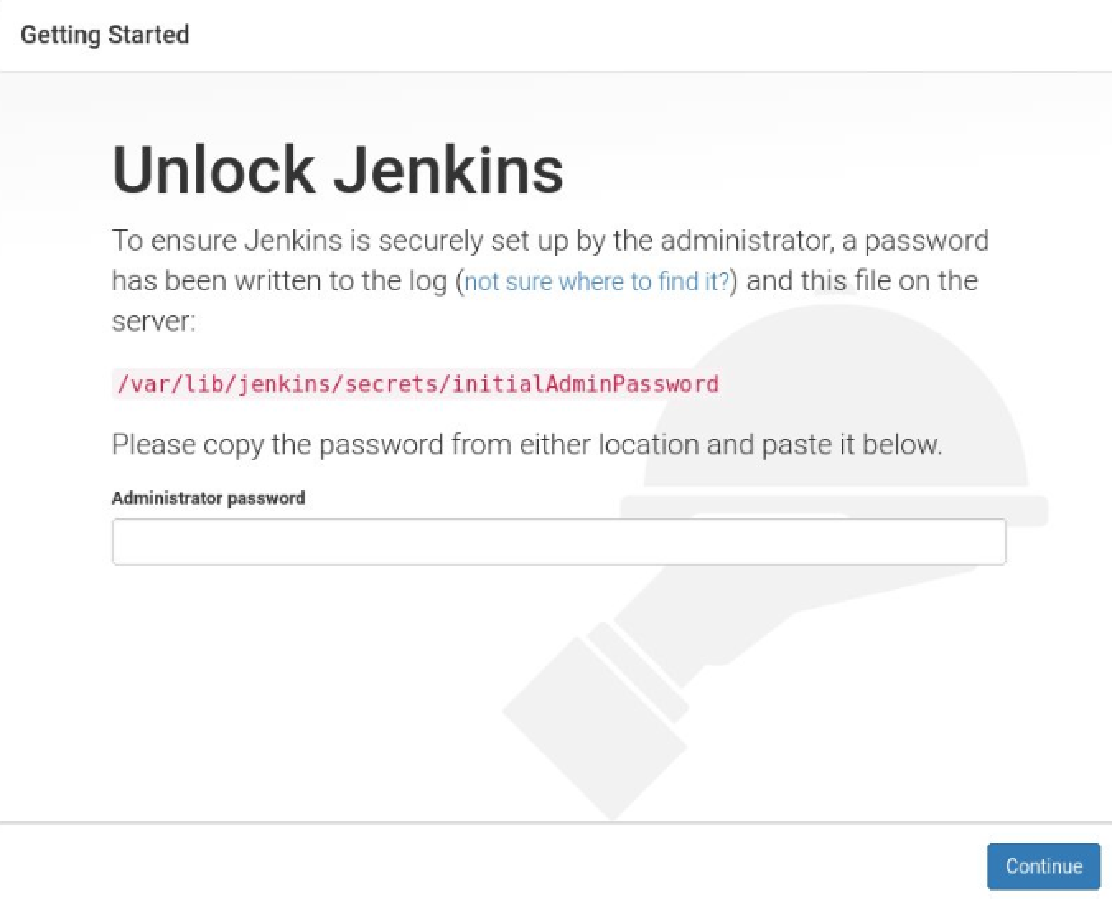
Use the following commands to open the necessary port:

-> sudo firewall-cmd --permanent --zone=public --add-port=8080/tcpsudo firewall-cmd --reload

-> sudo firewall-cmd --reload

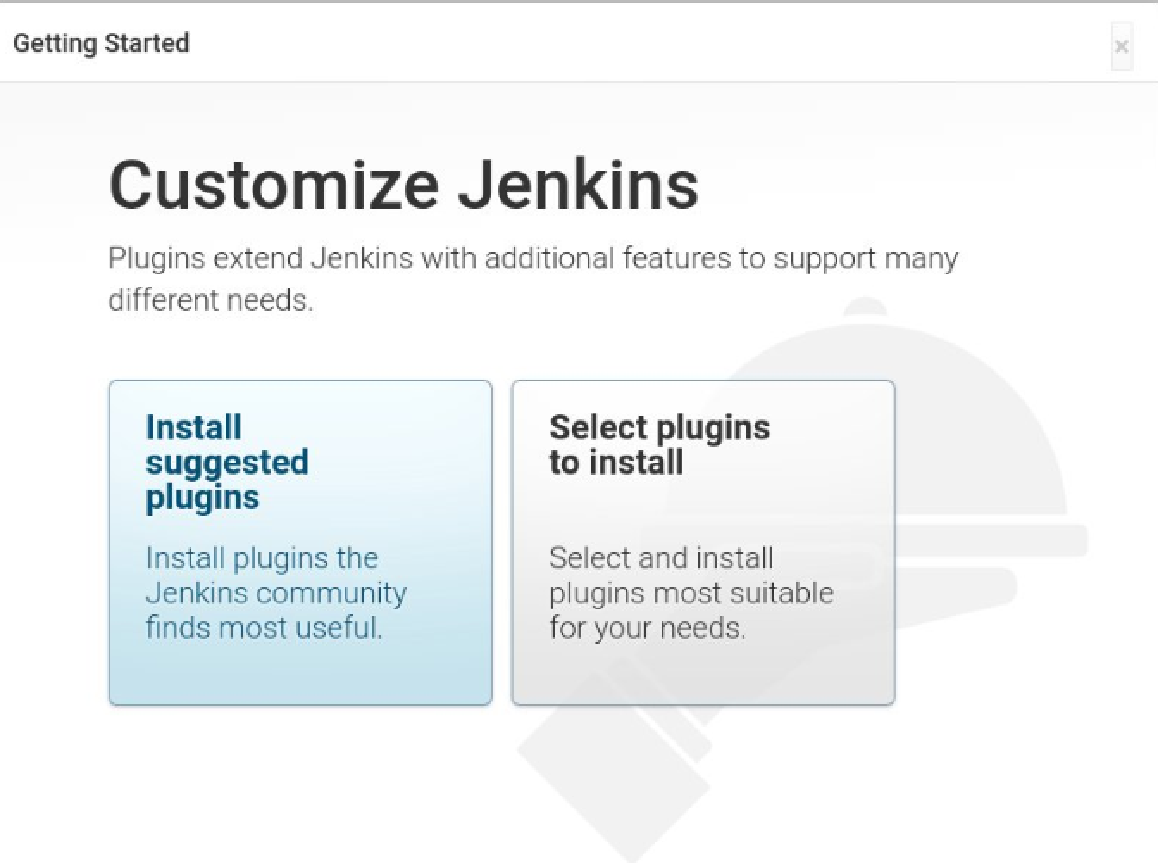


-> <http://your_ip_or_domain:8080>

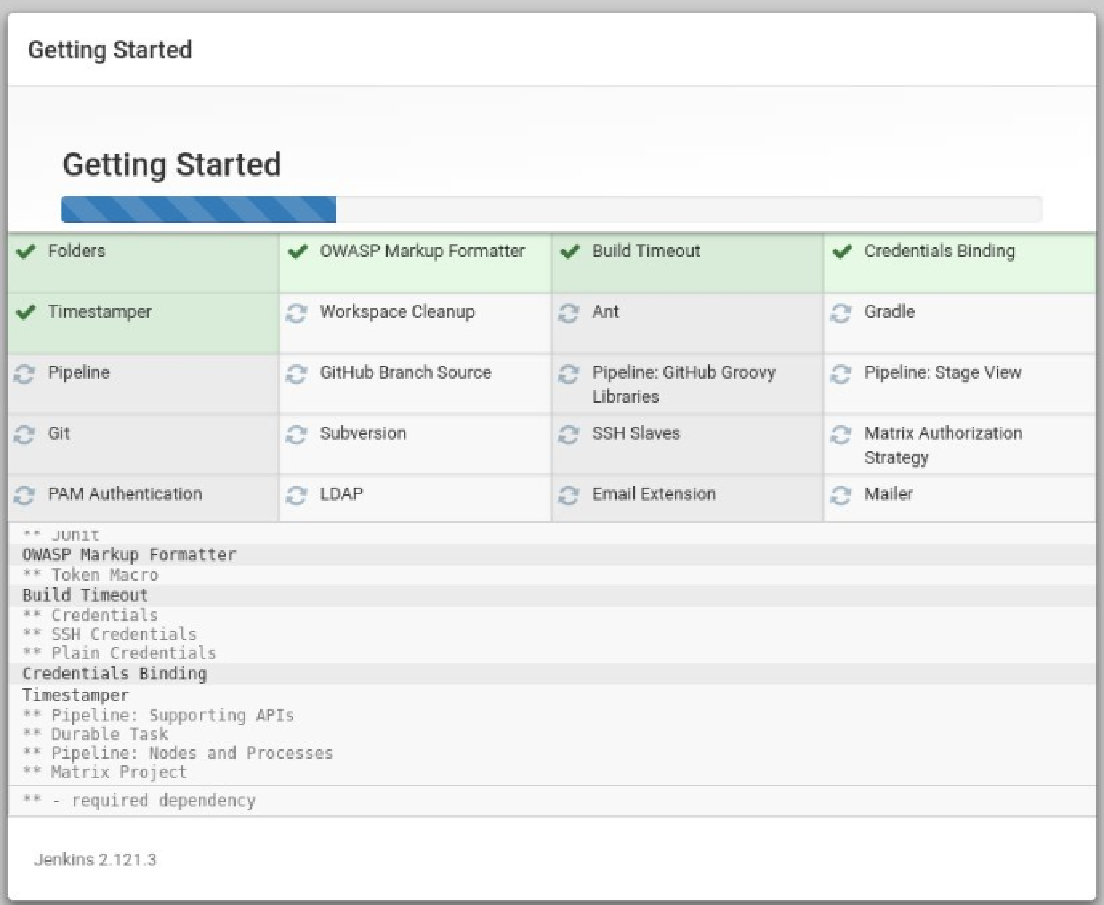


sudo cat /var/lib/jenkins/secrets/initialAdminPassword

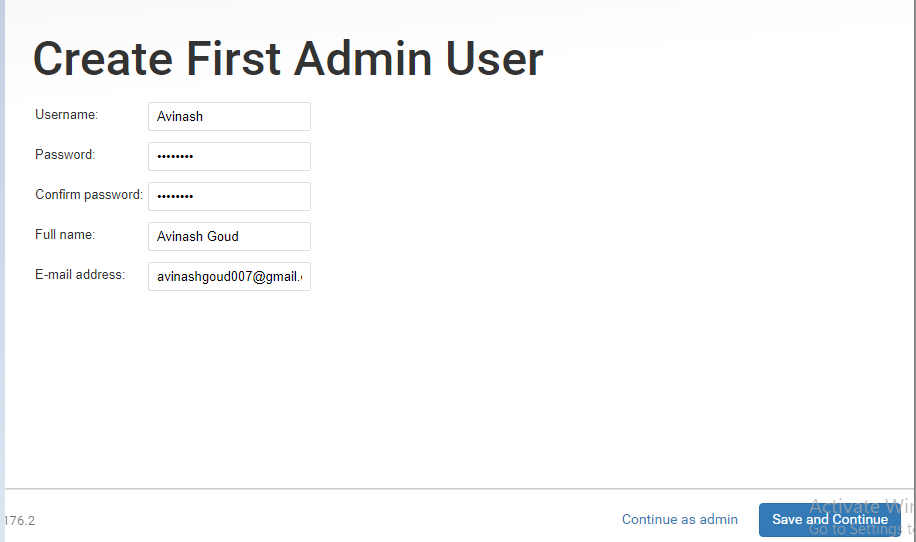
On the next screen, you will be asked whether you want to install the suggested plugins or to select specific plugins. Click on the Install suggested plugins box, and the installation process will start immediately.



On the next screen, you will be asked whether you want to install the suggested plugins or to select specific plugins. Click on the Install suggested plugins box, and the installation process will start immediately.



Once the installation is complete, you will be prompted to set up the first administrative user. Fill out all required information and click Save and Continue.



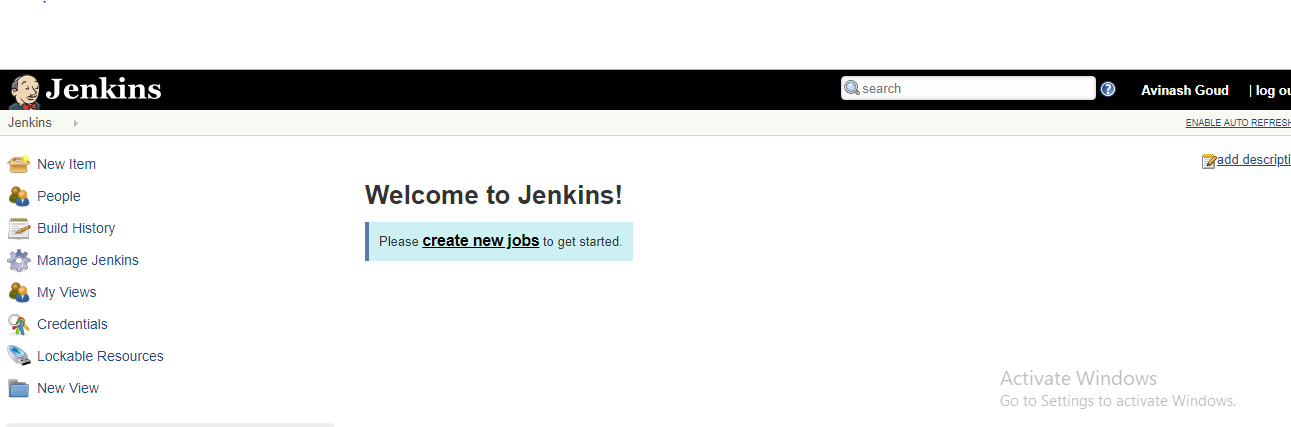
Now click save and continue

Now the jenkins is ready

To login to jenkins use the below username and password:

Ip : 192.168.12.151:8080

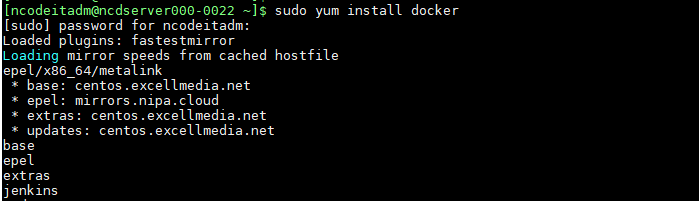
Username : ncodeit

Password: ncodeit123 

1. **Install Docker**

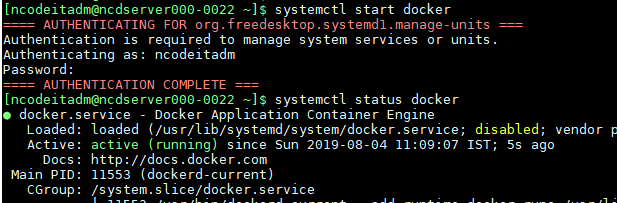
**=============**

-> Sudo yum install docker



-> sudo systemctl start docker

-> sudo systemctl status docker



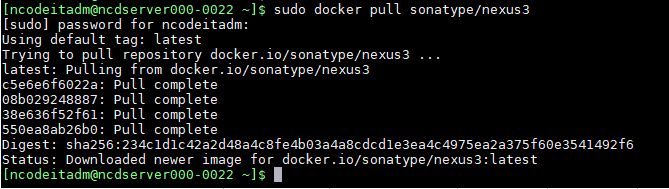
-> sudo systemctl enable docker

d3

1. **Install Nexus**

**============**

-> docker pull sonatype/nexus3



-> sudo docker run -d -p 8081:8081 --name nexus sonatype/nexus3

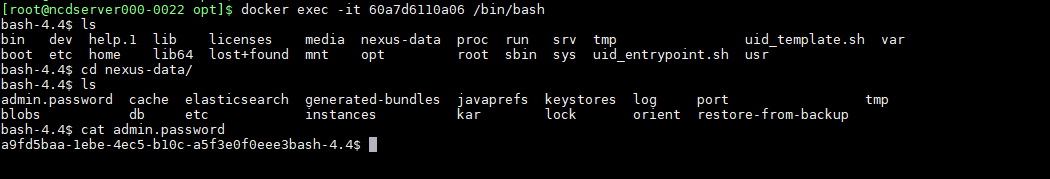
n2

To now nexus login password followed the below steps

-> docker exec -it imageid /bin/bash

cd /nexus-data

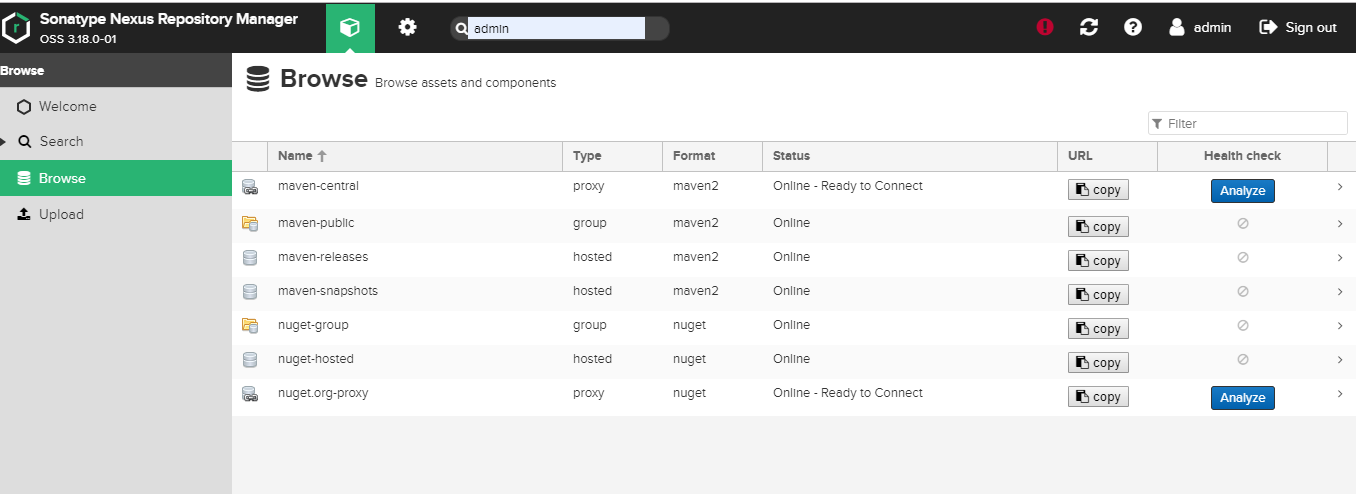
cat admin..password



Now nexus is ready

Username: ncodeit

Password : ncodeit123



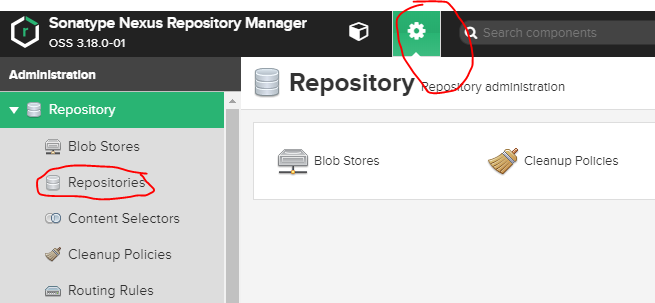
1. **Create a Repository in Nexus**

=========================

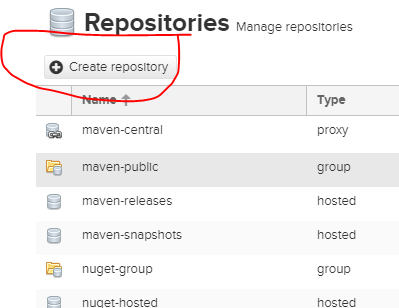
Goto Nexus server <yourip:portno>

Sign in to nexus server

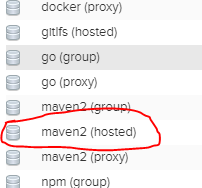
Click on settings and click on repositories



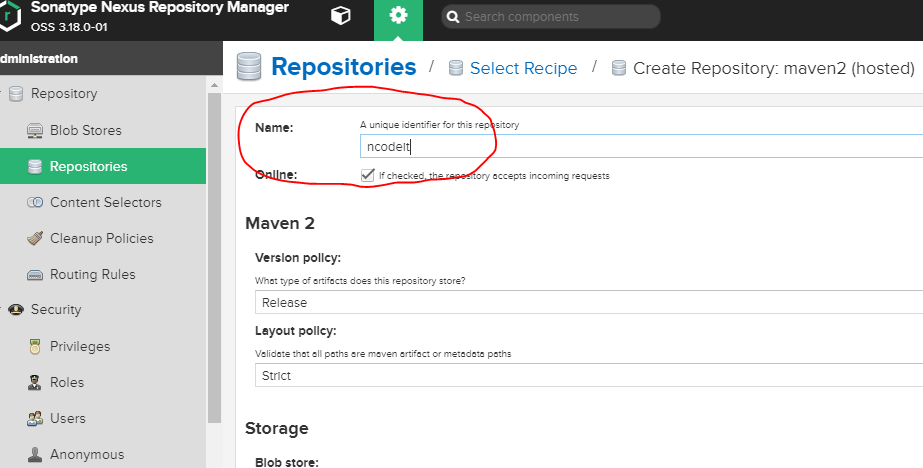
Click on create repository



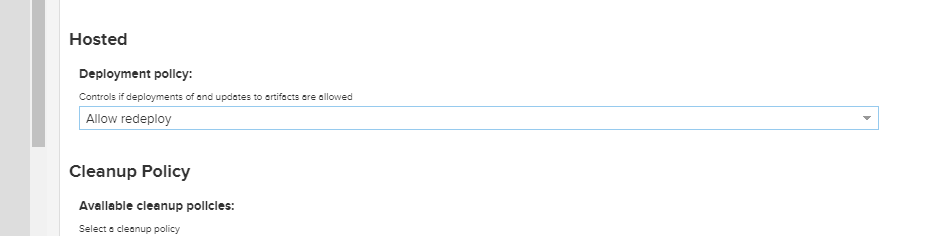
Click on maven2(Hosted)



Enter Name for Repositroy



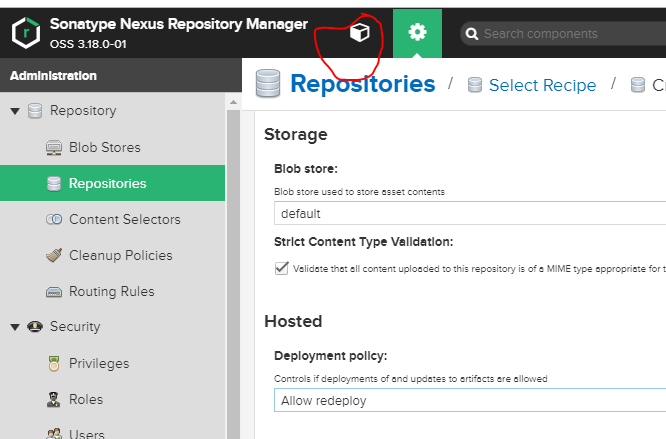
Select Deployment policy as “Allow Redeploy”



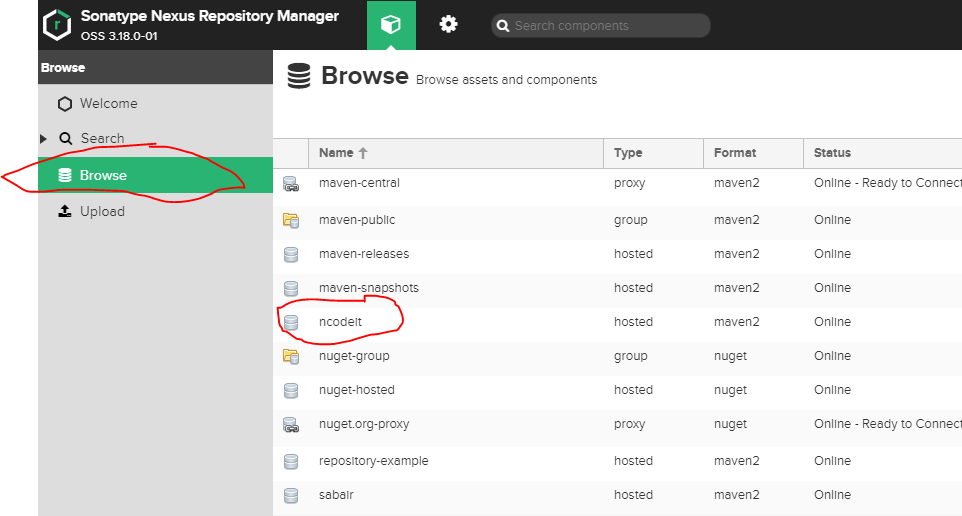
Click on create.

Your Repository got created.

To check your repostiry click on browser server contents



Click on browse and check your repository.



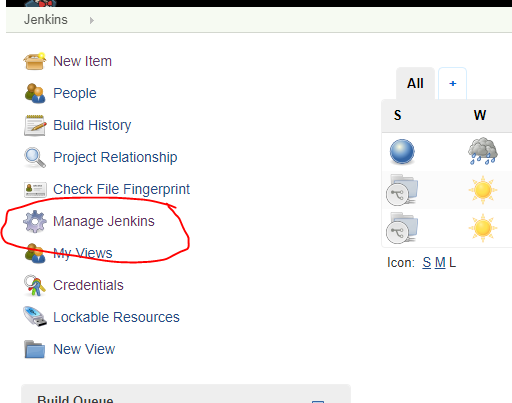
Done with Creation of Repository.

1. Install Nexus Plugin

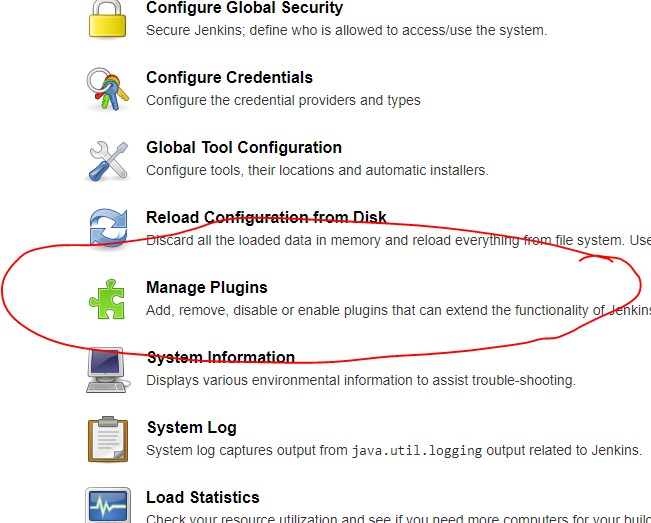
================

Got to Jenkins Server <your-Ip;Portno>  
Login with Credentials

Click on Manage Jenkins

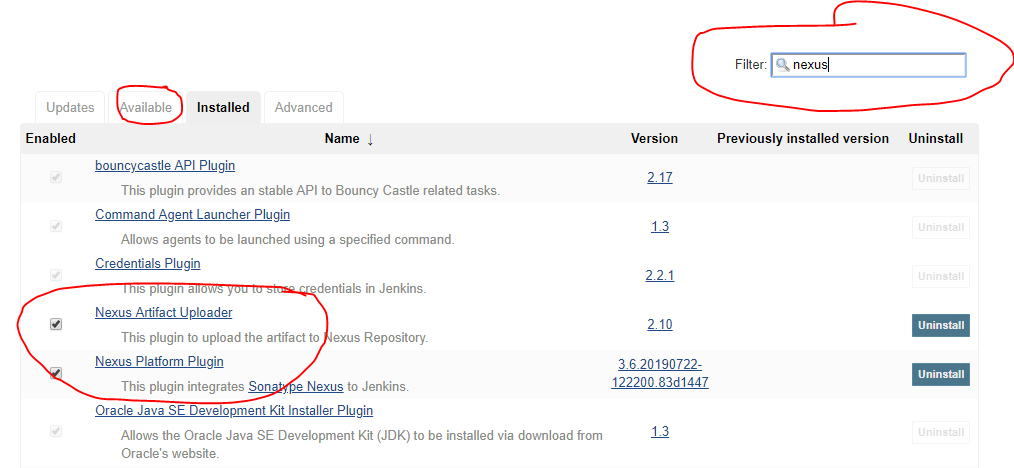


Click on manage plugins

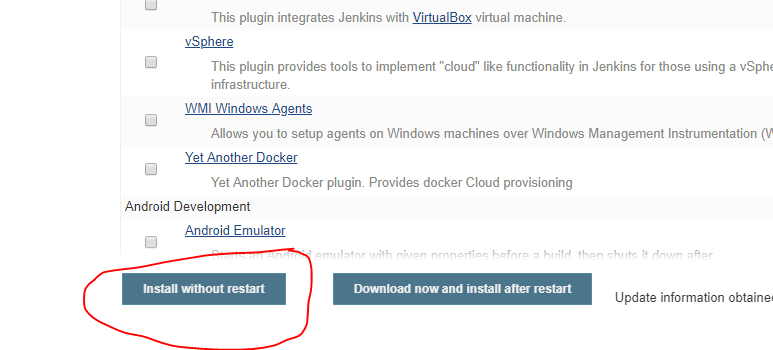


Click on available and search for nexus.

Install “Nexus artifact plugin” and “nexus platform plugin”

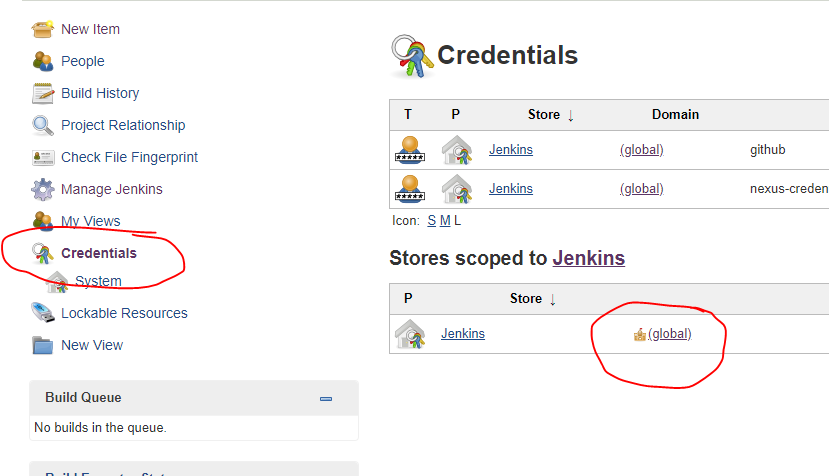


Click on “Install without restart”

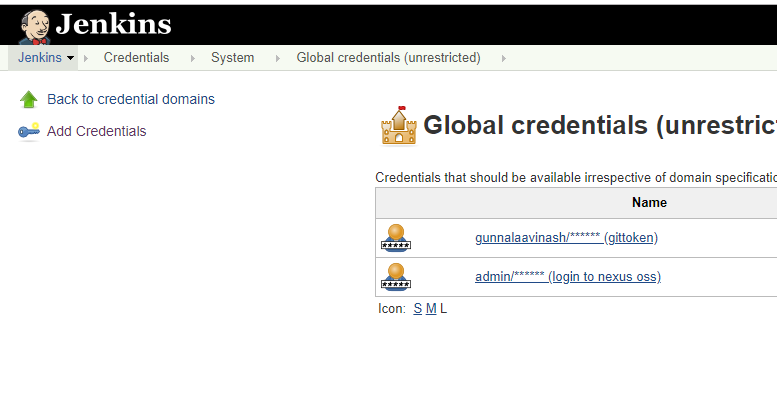


1. Add Credentials Globally

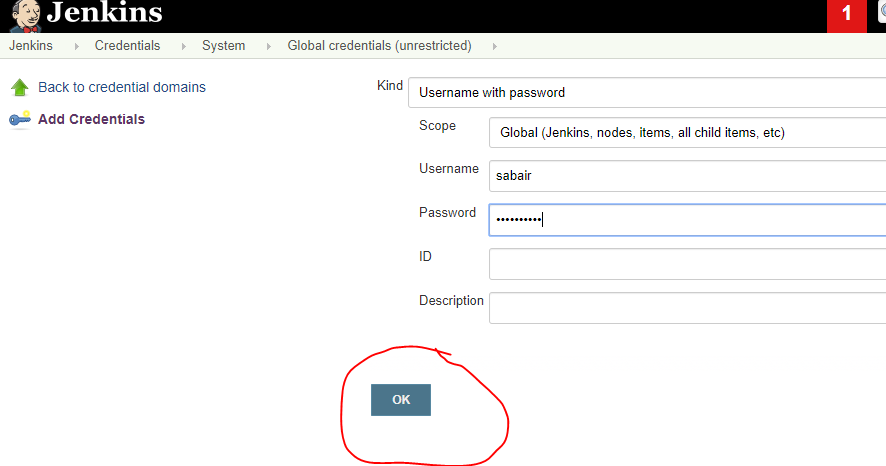
Click on credentials and click on global



Again click on add credentials

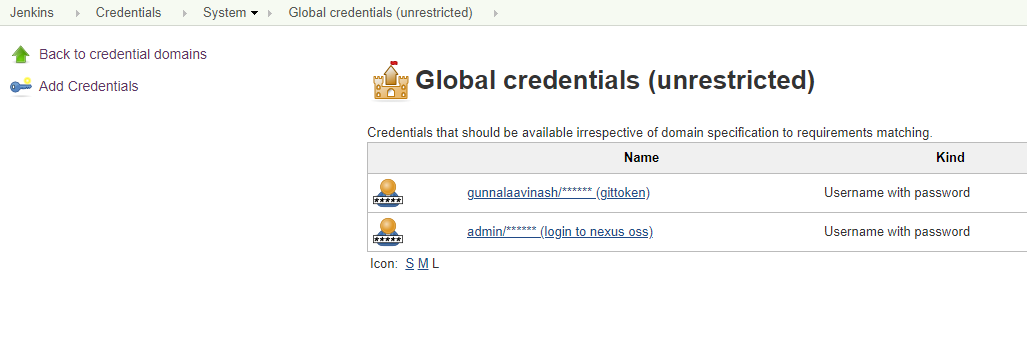


Enter Nexus username and password and click on ok



Repeat the same procdeure to store github credentials

Verify the global credentials



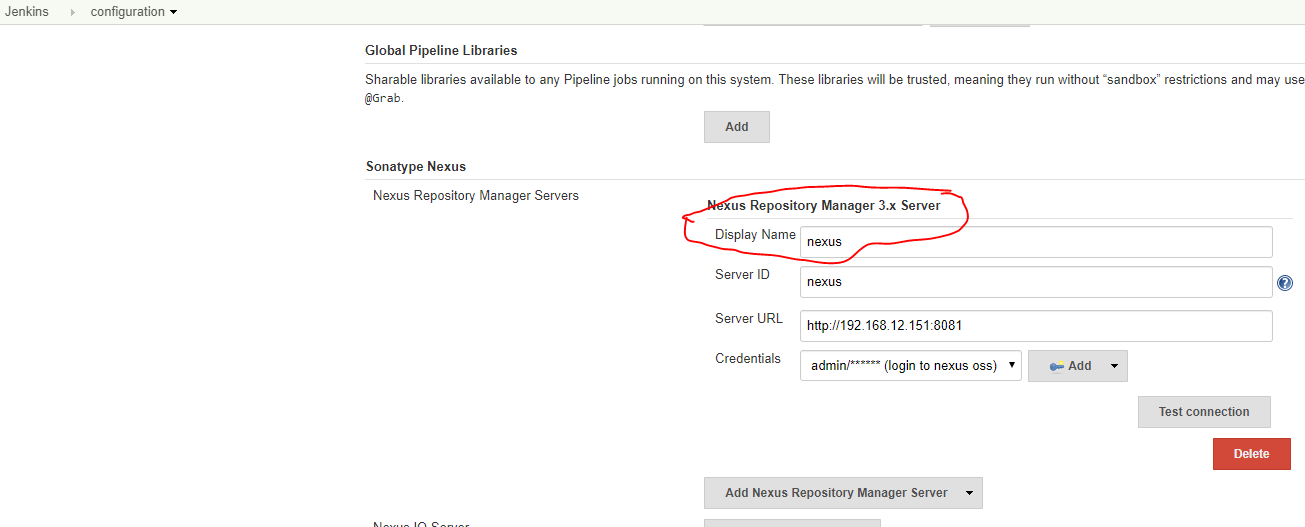
13) Integrate Nexus with jenkins

========================

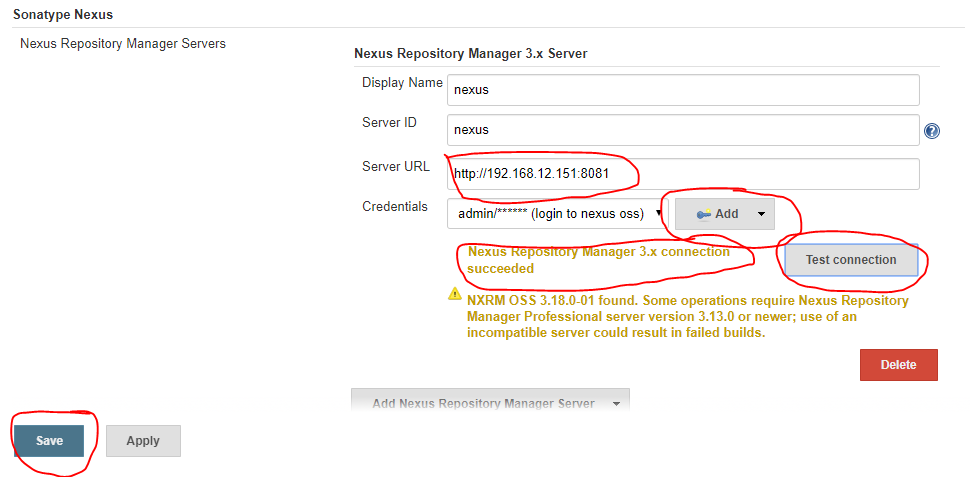
Click on manage jenkins and select configure system



Goto sonatype nexus and select for nexus 3.x server



Enter any server name and display name

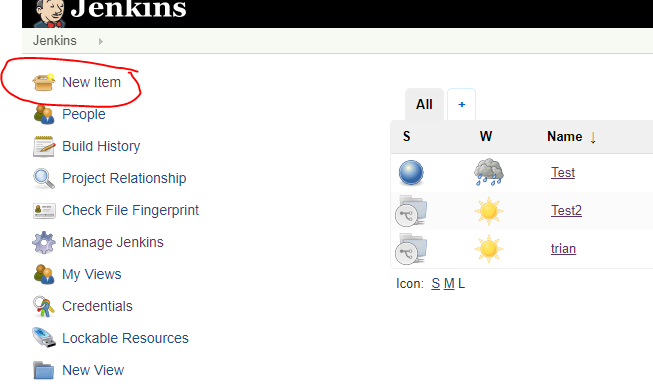
Enter the nexus server ip and portnumber,add credentials,click on test connection and check for the success messgae and click on save.  


1. Create First Jenkins Job.

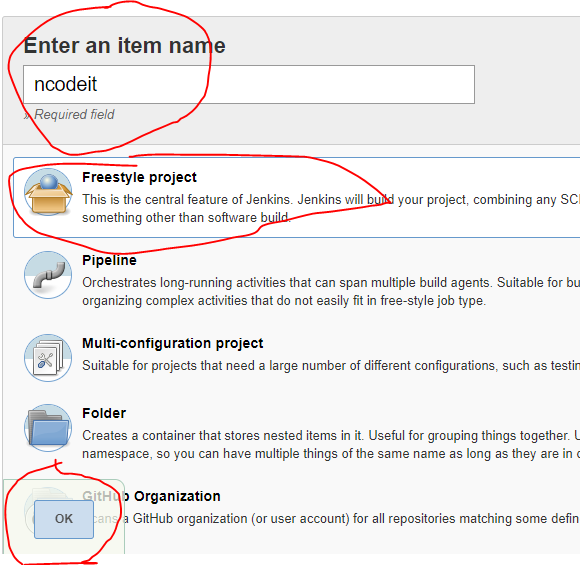
====================

Login to your jenkins sever

Click on new item

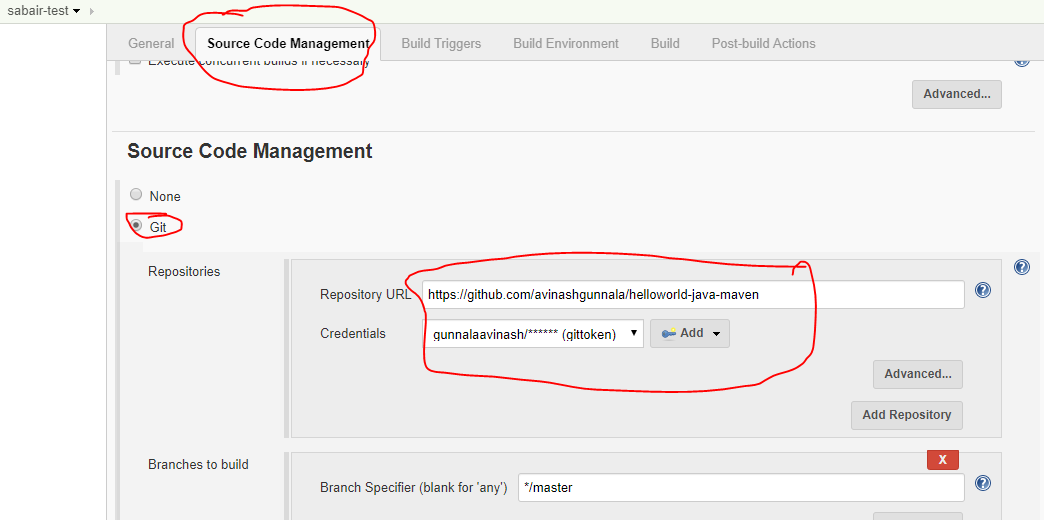


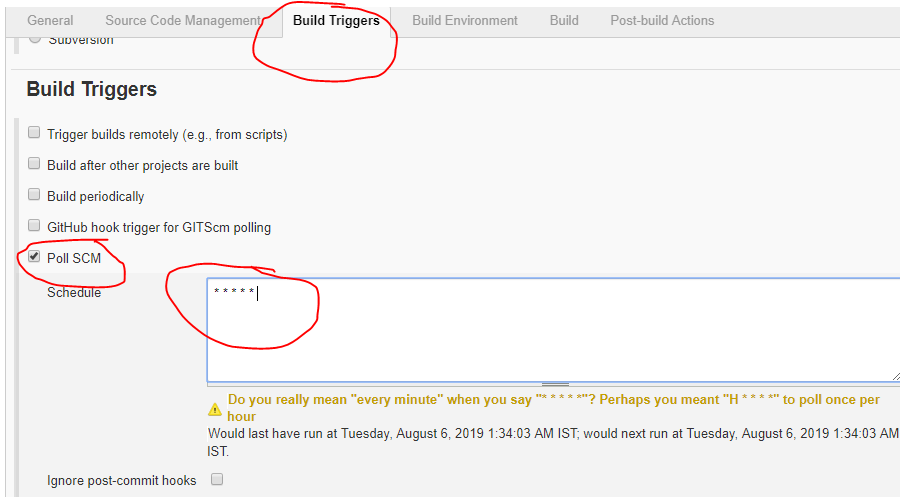
Enter project name,select free style project and click on okay.



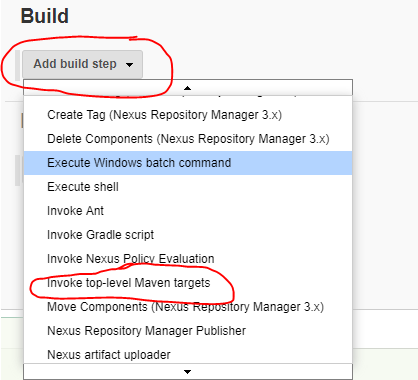
Under source control management,select git,enter the github repo url and add the credentials

Under build trigger,select poll scm and enter “\* \* \* \* \* ”



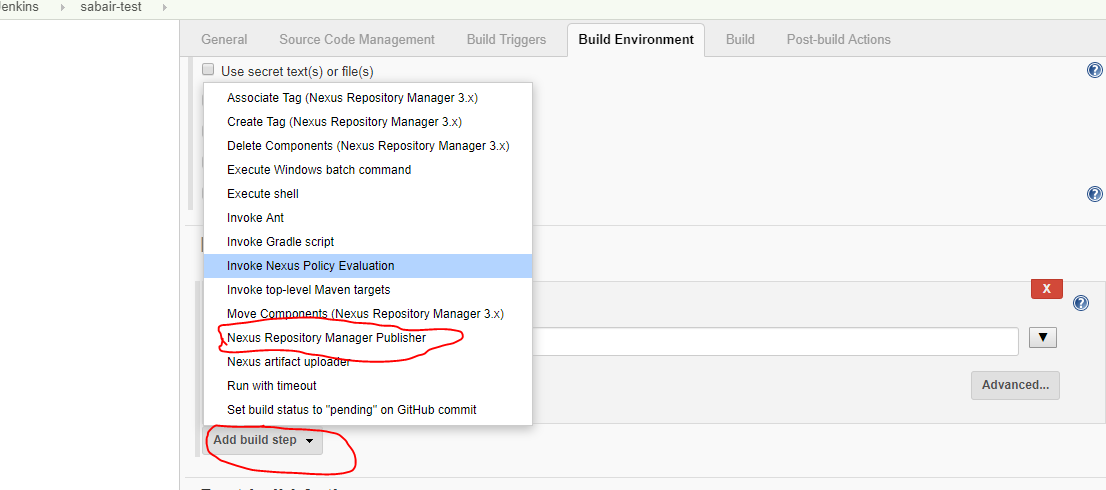
This is done to automatically pull the code from github if any new commit is done.  


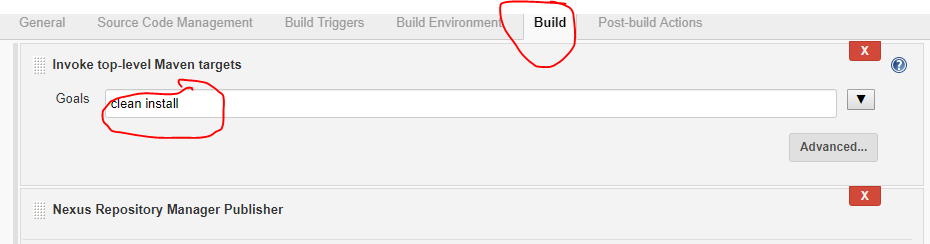
Under build tab,click on add build step,select invoke top-level maven tragets and write goal as clean install

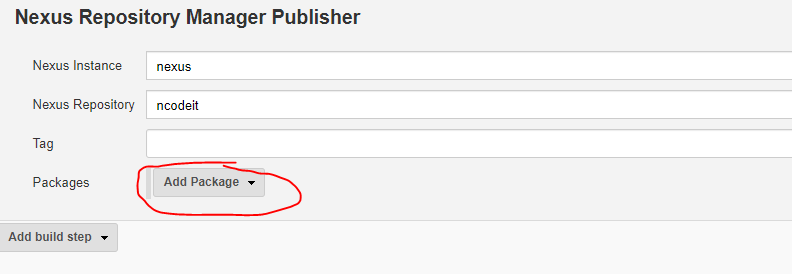


Click again on build steps and select nexus repository manager publisher

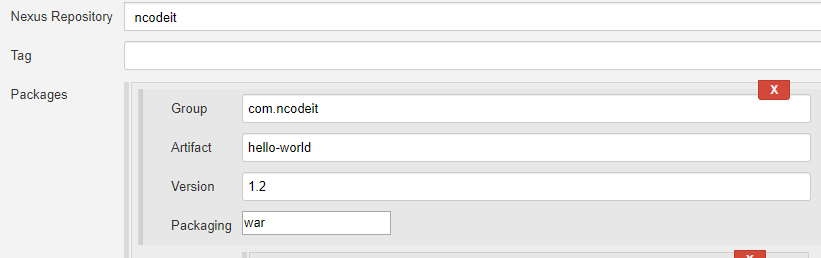
Enter Nexus instance name,nexus repositry which you created in nexus, and click on add package



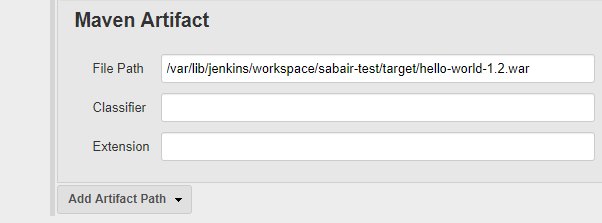




Enter the same details as shown in the image and click on “add artifact path”

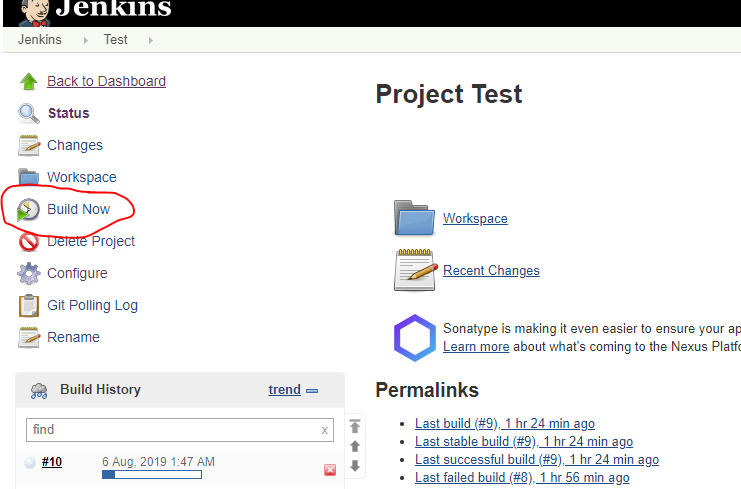


Check yout file path and enter in file path and then click on save.

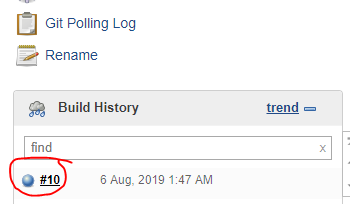


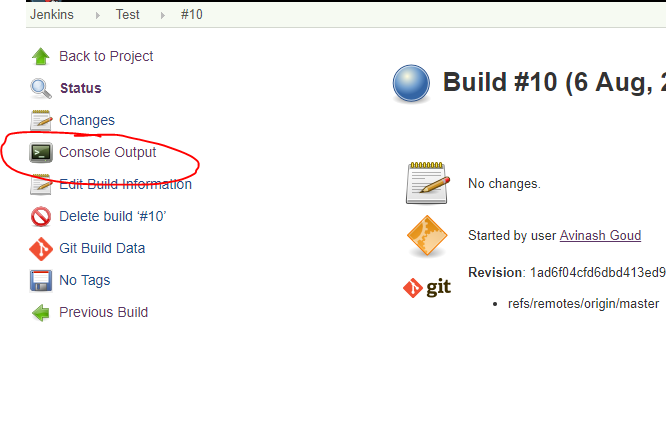
Your cane use this path and click on save“/var/lib/jenkins/workspace/your-jenkins-jobname/target/hello-world-1.2.war”

Goto your jenkins job now and click on buildnow

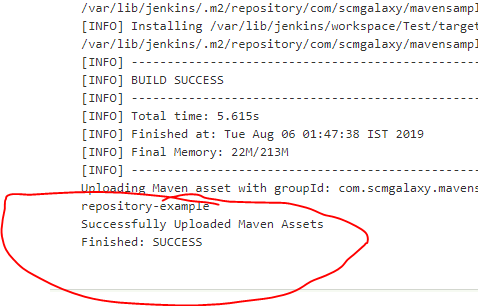


Click on job and check console output





Check the output status of your jenkins job



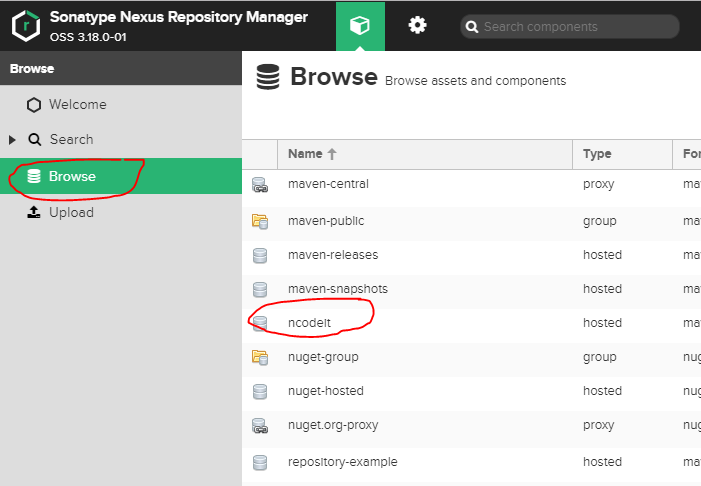
1. Goto nexus and confirm the jar file

============================

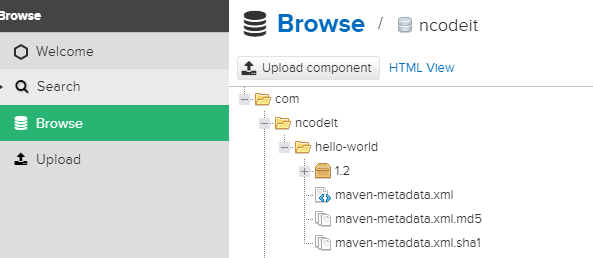
Enter your-nexus ip:portno

Enter Nexus Credentials and login

Click on browse and your repository.



You can find you artifacats in repository



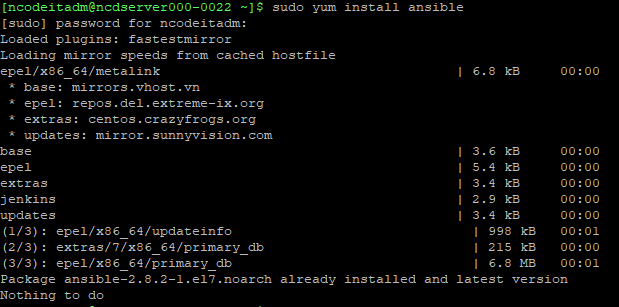
Done.

1. **Ansible Installation**

**==================**

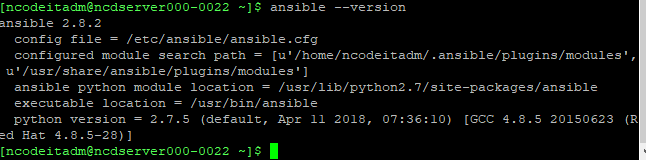
Install Ansible using below command

Sudo yum install ansible



Check wether ansible is installed succesfully or not.

ansible --version



**17) Tomcat Installation**

==================

sudo wget <http://mirrors.estointernet.in/apache/tomcat/tomcat-8/v8.5.43/bin/apache-tomcat-8.5.43.tar.gz>

With the above command tomcat gets installed.

cd /etc/tomcat

tar -xvf apache-tomcat-8.5.43.tar.gz

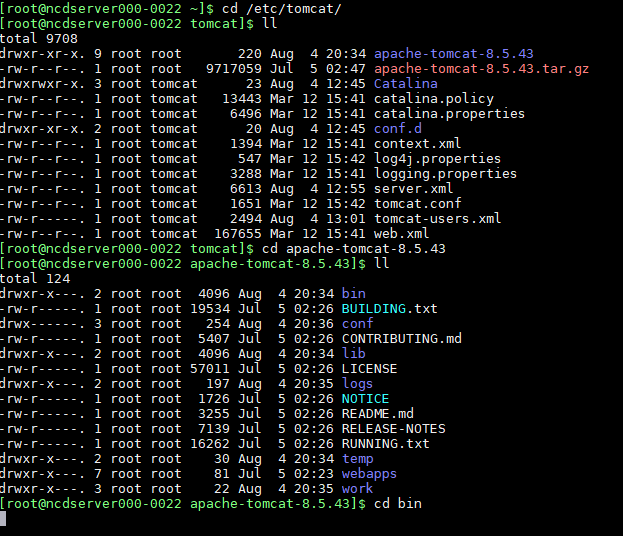
cd apache-tomcat-8.5.43.tar.gz

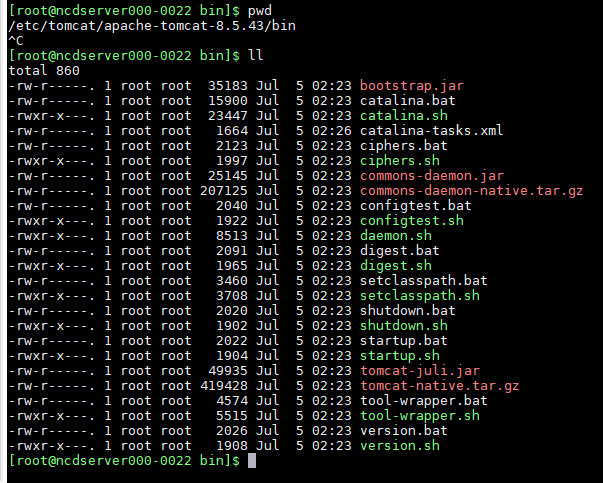
ls -lrth

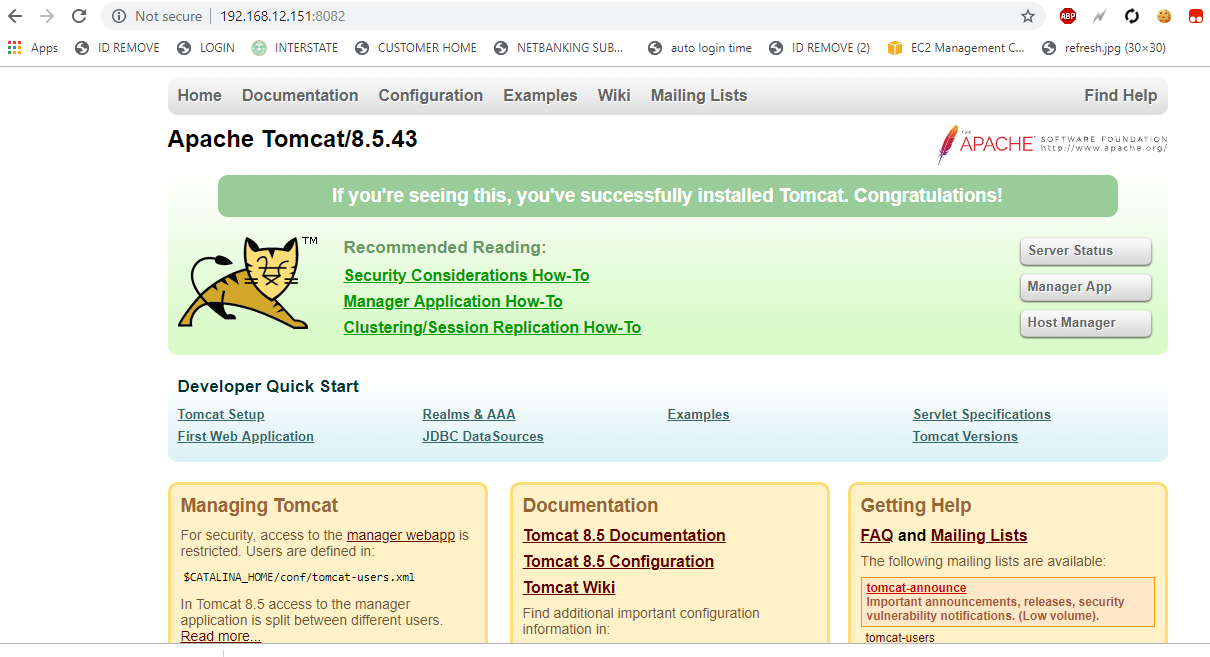
To start tomcat run the below command

./startup.sh

To shutdown the tomcat use below command

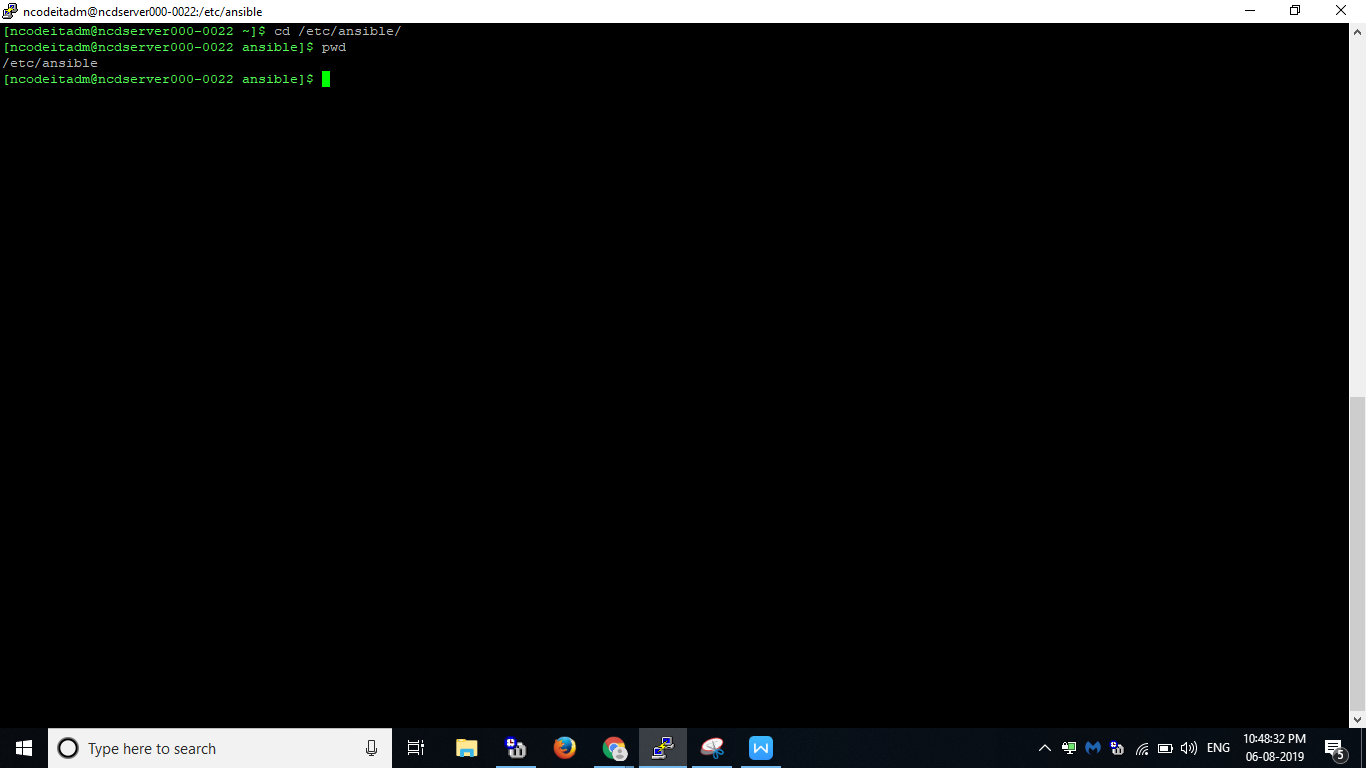
./shutdown.sh





1. Configure Ansible Inventory file and sshkey generation

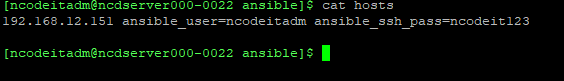
===========================================

--> Go to ansible home “/etc/ansible”  


--> specify the ip address of the remote system in hosts file(ansible-inventory).

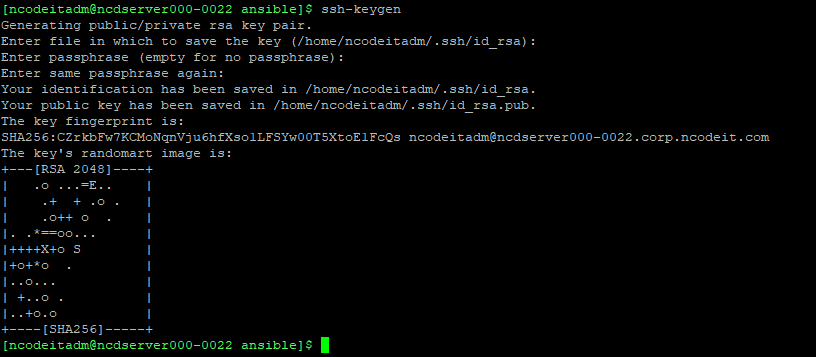
sudo vi /etc/ansible/hosts

1. 168.12.151 ansible\_user=ncodeitadm ansible\_ssh\_pass=ncodeit123



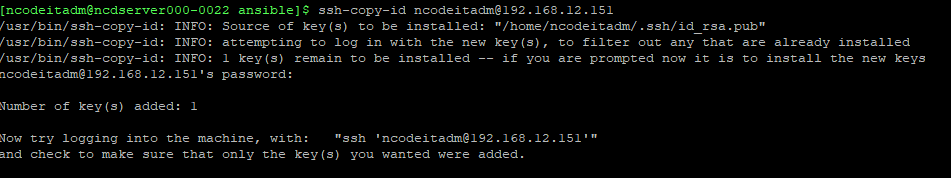
--> and generate ssh-key in your master and and copy that to remote machine.

ssh-keygen



And next copy that key to remote machine

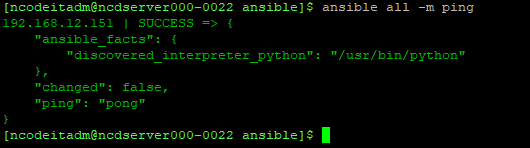
ssh-copy-id [ncodeitadm@192.168.12.151](mailto:ncodeitadm@192.168.12.151)



Then the key will be copied to your remote machine.

And then check whether that both master and remote machine are got connected or not by running following command.

ansible all -m ping



Once you run the above command you will get a pong reponse from the remote machine, that will be confirmation that both are talking each other.

1. Ansible Playbook

Write a ansible playbook to download artifactory from nexus and push that artifactory to tomcat server.

Copy the below content and paste in file in “**/etc/ansible”**

---

- hosts: all

tasks:

- name: download the artifacts from the nexus

get\_url:

url: 192.168.12.151:8081/repository/ncodeit/com/ncodeit/hello-world/1.2/hello-world-1.2.war

dest: /home/ncodeitadm/artifacts

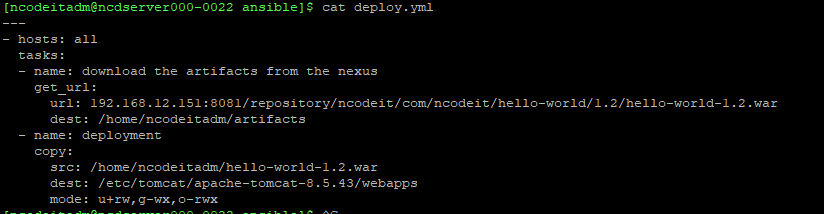
- name: deployment

copy:

src: /home/ncodeitadm/hello-world-1.2.war

dest: /etc/tomcat/apache-tomcat-8.5.43/webapps

mode: u+rw,g-wx,o-rwx

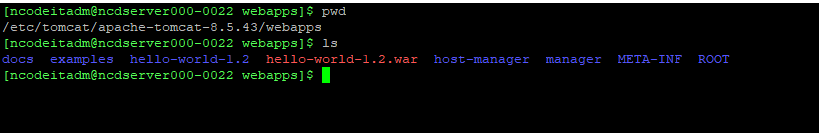


Now run your playbook to deploy code to tomcat.

ansible-playbook deploy.yml

After executing the playbook please find the war file at below path

**/etc/tomcat/apache-tomcat-8.5.43/webapps**



Now the application is deployed successfully and running .

URL : <http://192.168.12.151:8082/hello-world-1.2/>

