1) Install jenkins and run jenkins on port number 8081.

**Download jenkins Repo**: sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo

**Import the jenkins key:**sudo rpm --import <https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key>

**Update ec2**:sudo yum upgrade

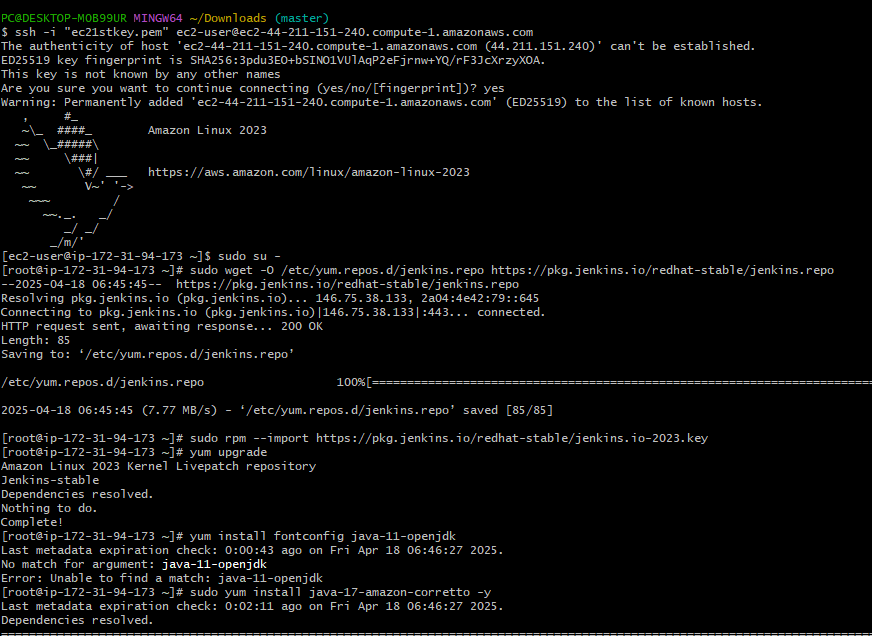
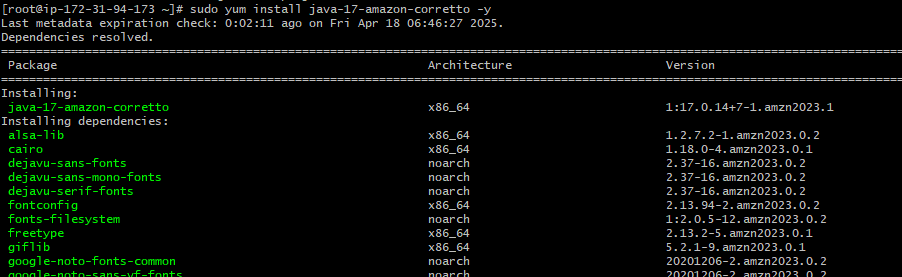
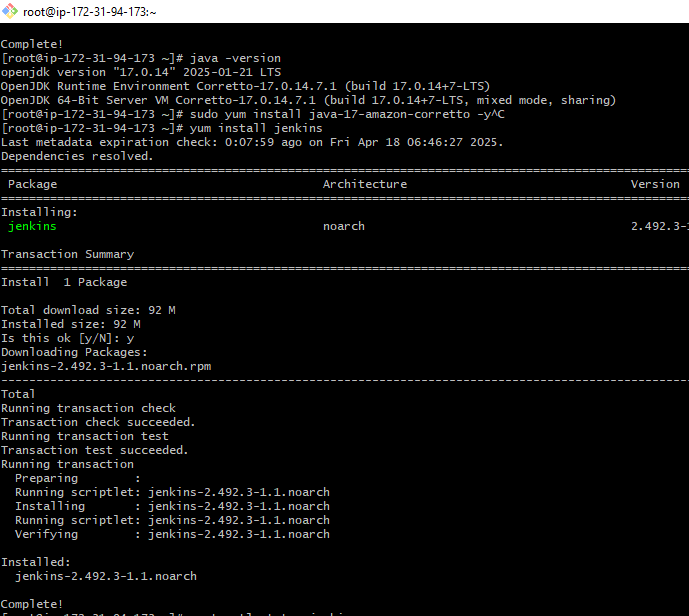
**For java installation**:sudo yum install java-17-amazon-corretto –y (Use only java version 11 or 17 only)

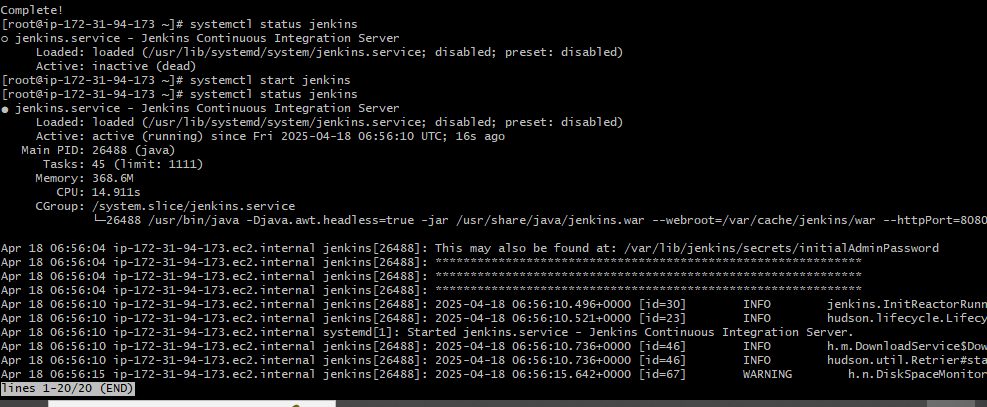
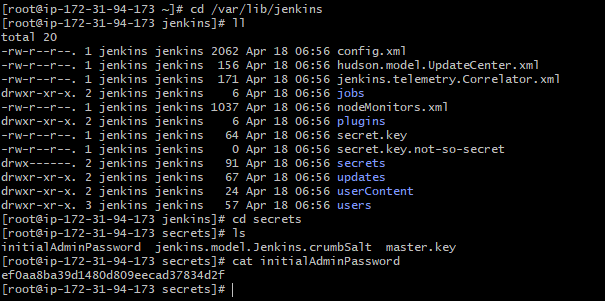
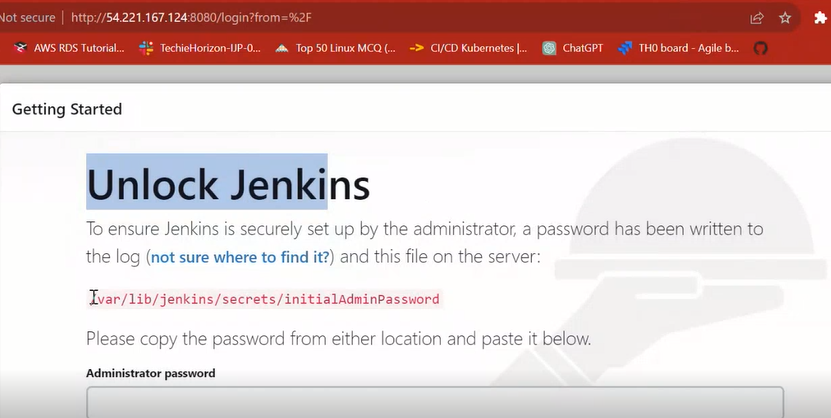
Jenkins supports the following Java versions, depending on the release:

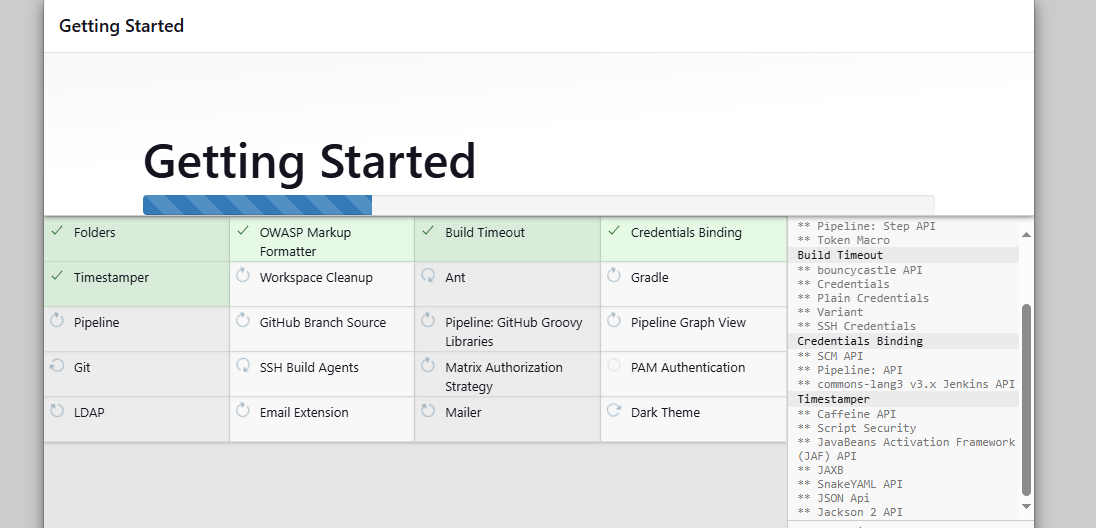
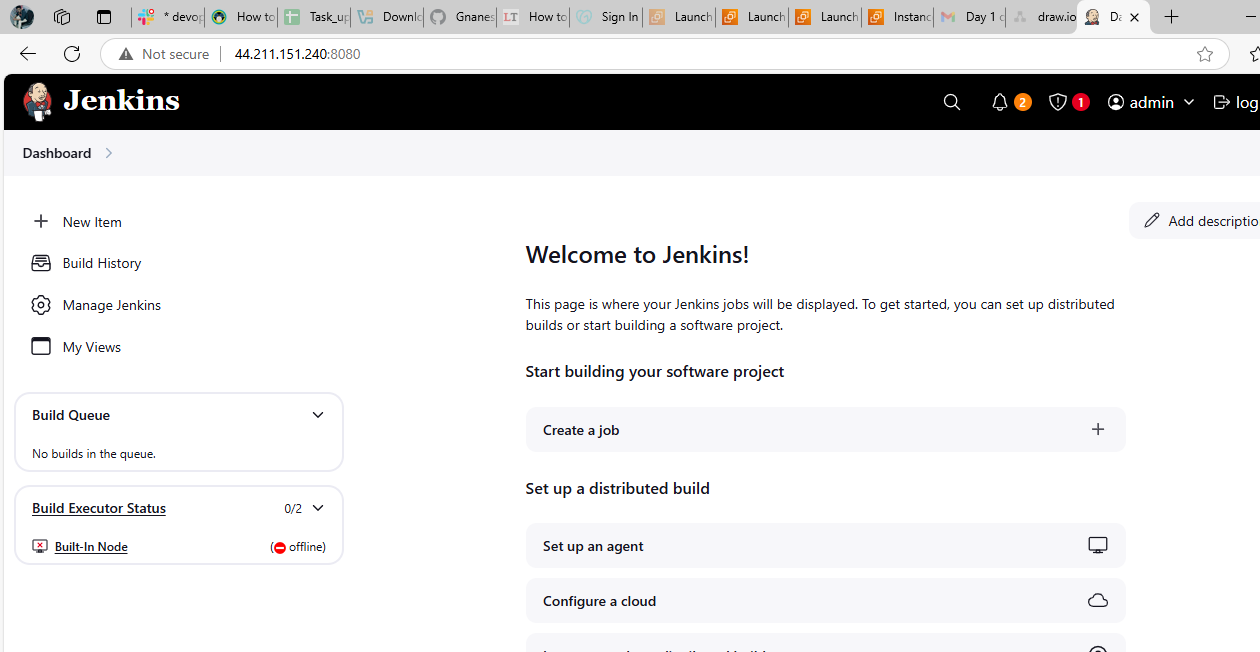
1. **Java 11**: Supported for older Jenkins versions and still widely used.
2. **Java 17**: Fully supported and recommended for modern Jenkins installations.
3. **Java 21**: Supported in newer Jenkins releases, especially for weekly updates and upcoming LTS versions.

It's important to note that Jenkins no longer supports Java 8 or earlier versions. If you're upgrading or installing Jenkins, it's best to use Java 17 or Java 21 for optimal compatibility and performance. Let me know if you'd like help setting up Jenkins with the right Java version!

**Install Jenkins**:sudo yum install jenkins

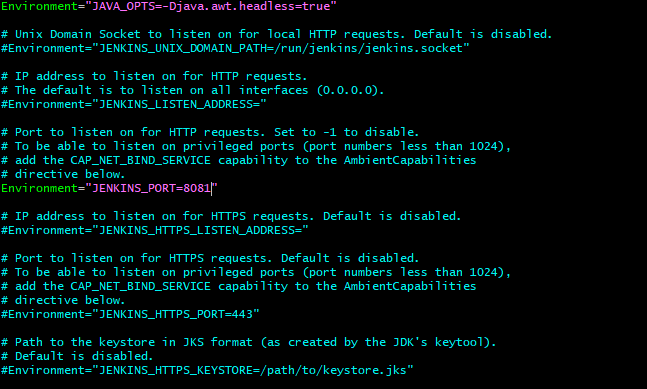
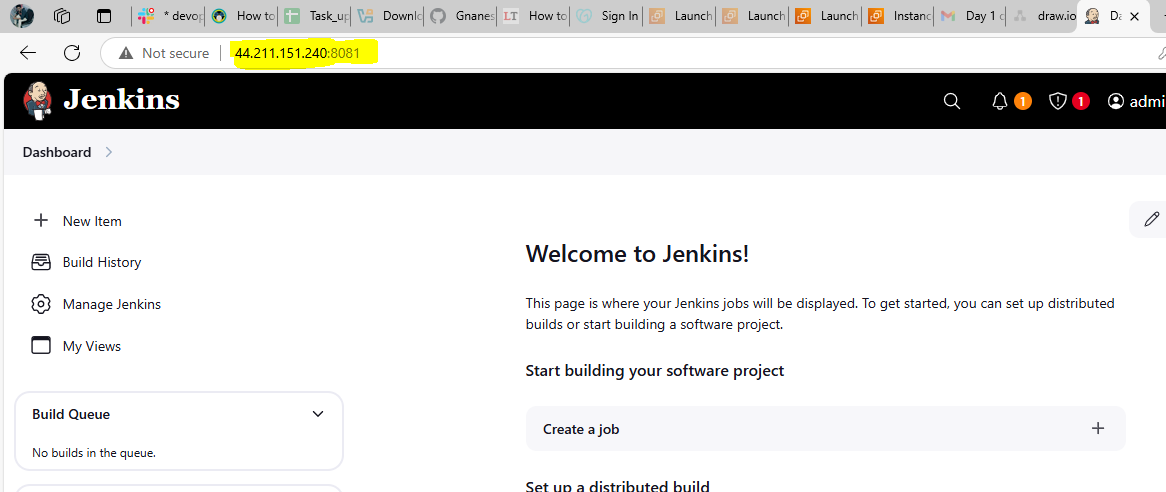
  

 Goto /var/lib/jenkins/secrets/intialAdminpasswords and copy the password to access to jenkins  Now access the Jenkins through the browser using **public IP:8080(Jenkins port number)** &paste password

  To change the port number of Jenkins goto /lib/systemd/system there you will find **Jenkins.service** ,open and edit the port number to 8081

After changing the port number restart the Jenkins using 'systemctl daemon-reload' to reload units and ‘systemctl restart Jenkins'

Run 'systemctl daemon-reload' to reload units.

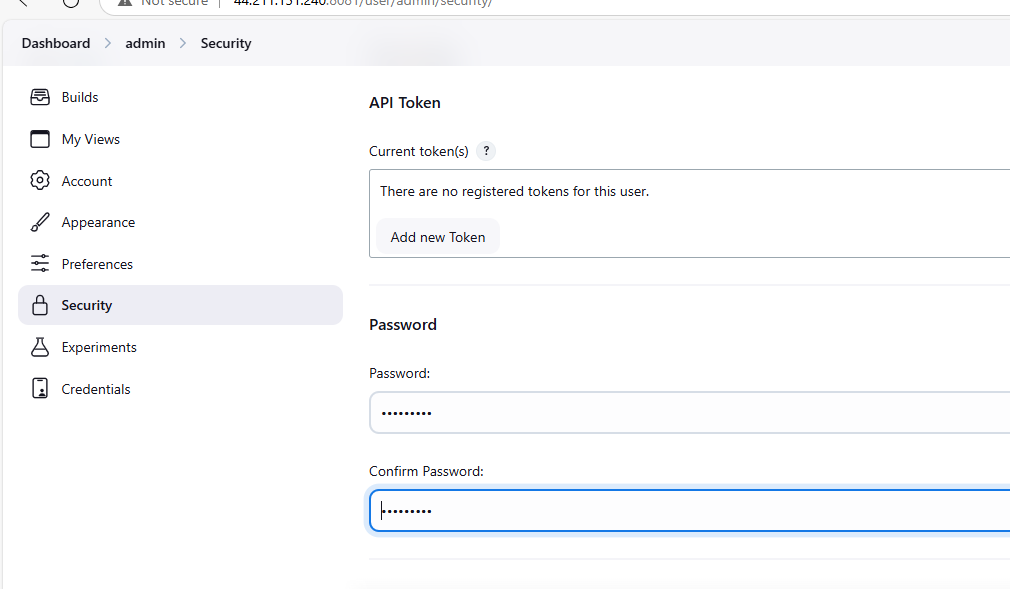
 

2) Secure Jenkins server

**To secure Jenkins we must change initial admin password to our custom password and save**

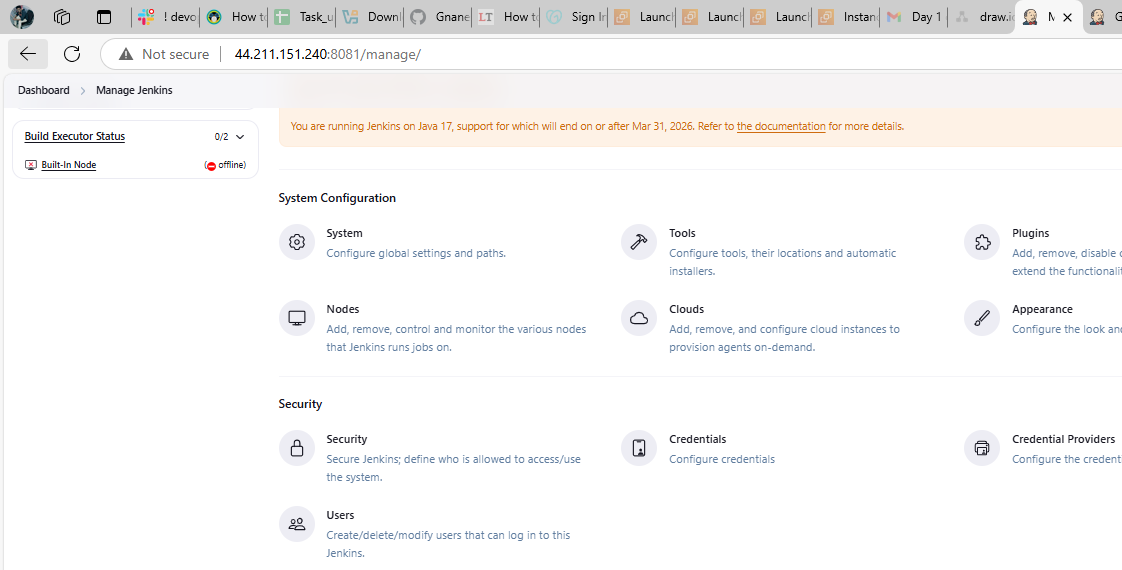
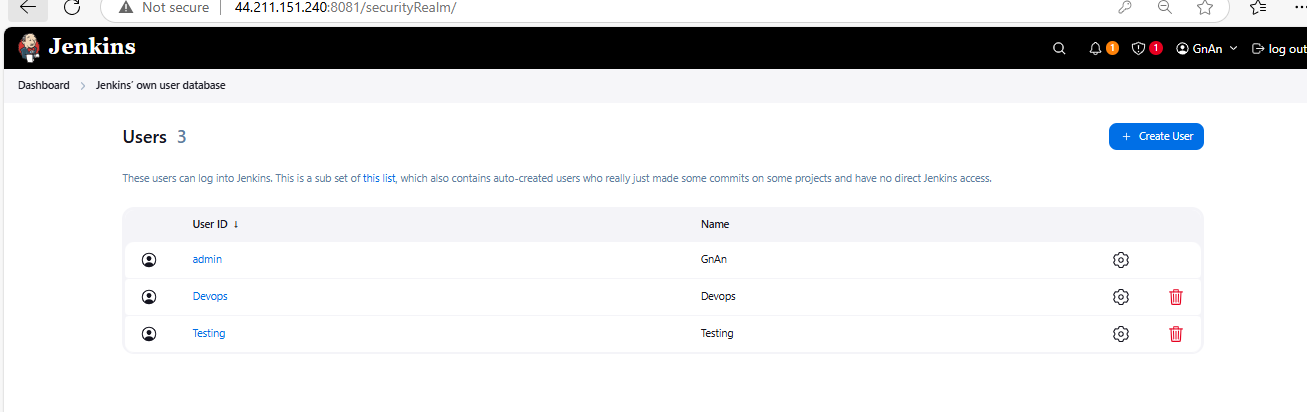
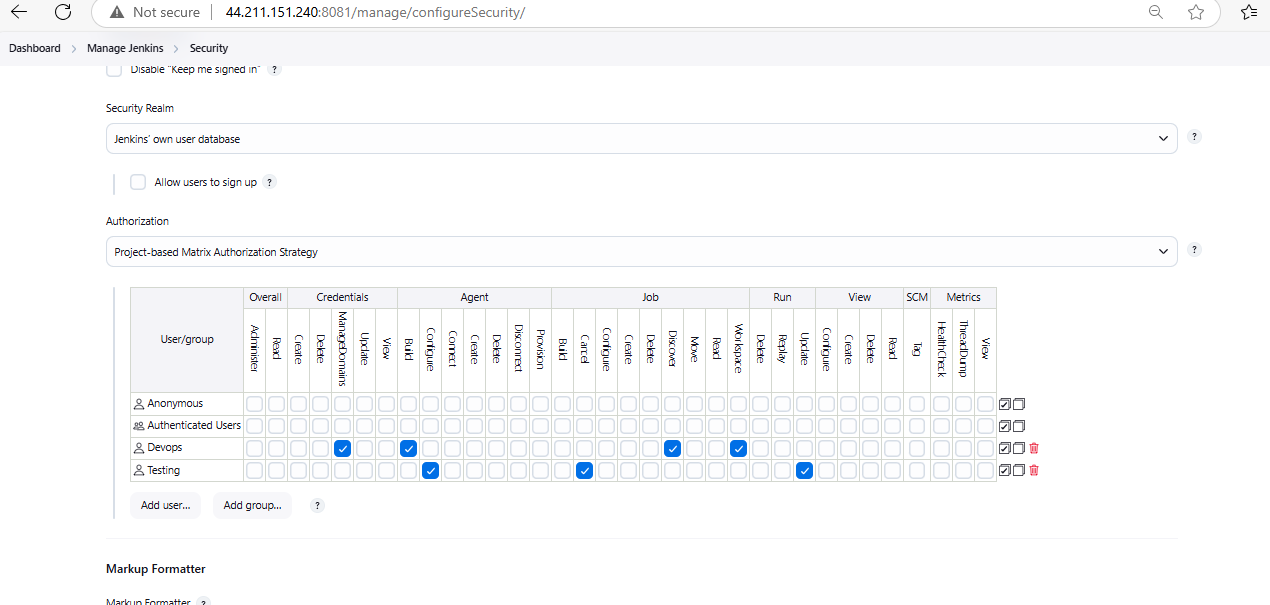
**Go to Jenkins profile-🡪Admin--🡪Securituy--🡪ChAnge passwords AND SAVE**

**AND VERIFY if the password is changed and able to access.**



3) Create users called Devops, Testing in Jenkins with Limited access.

Goto Jenkins Dashboard--🡪Manage Jenkins-🡪Users--🡪Create Users

  manage Jenkins 🡪security 🡪authorization 

4) Configure labels and restrict the jobs to execute based on label only.

Use below commands to increase Temp file space

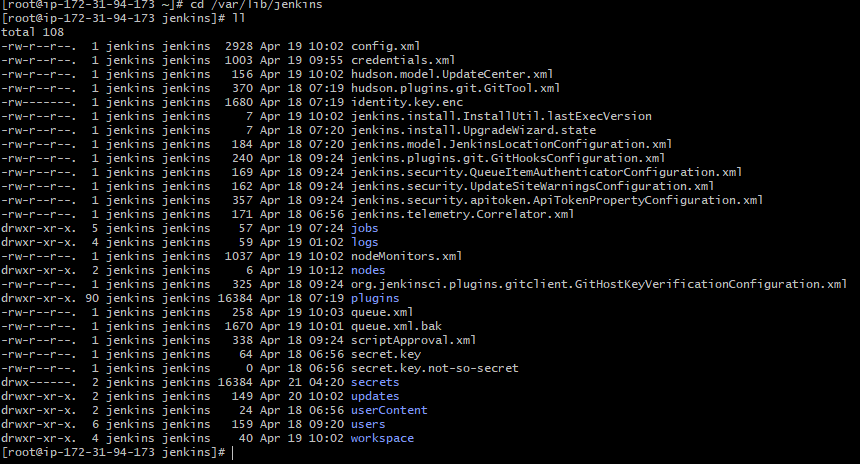
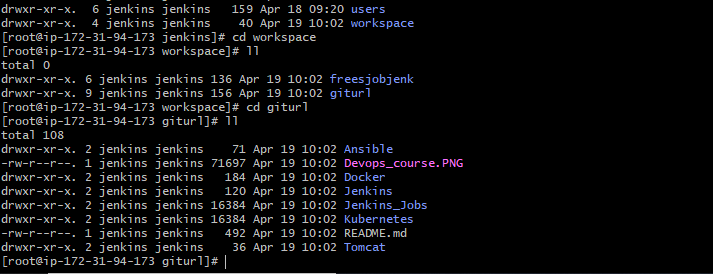
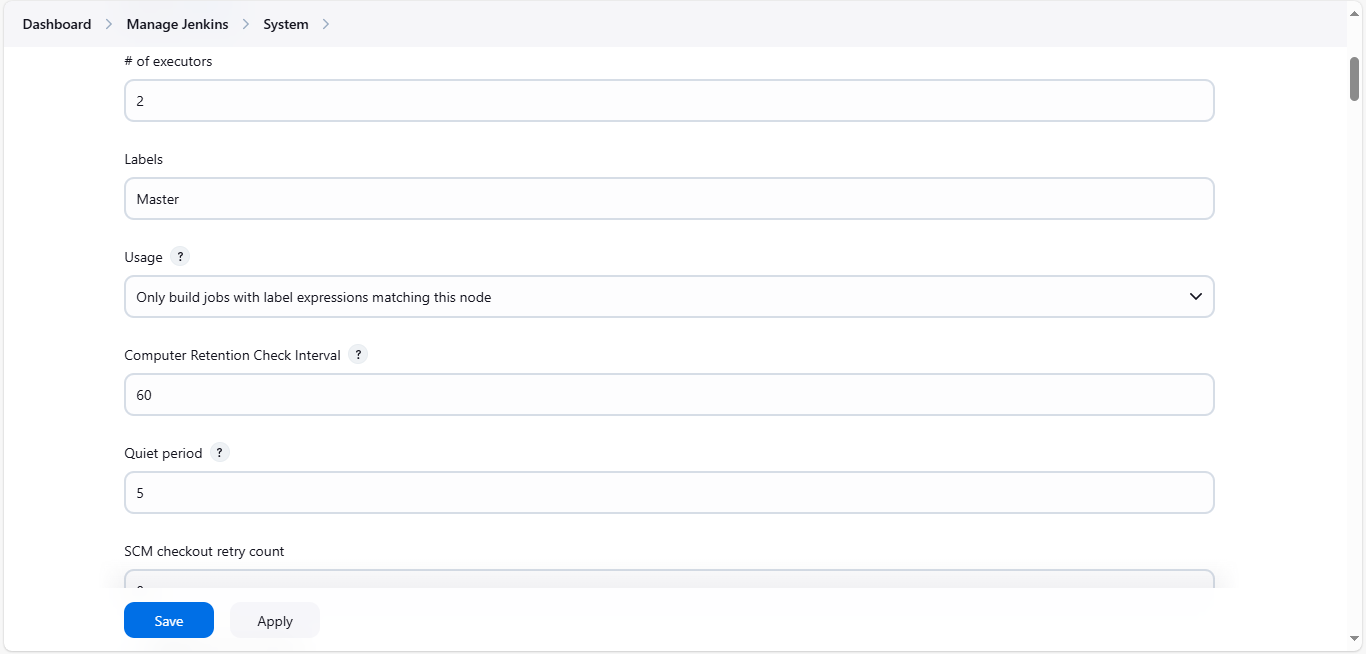
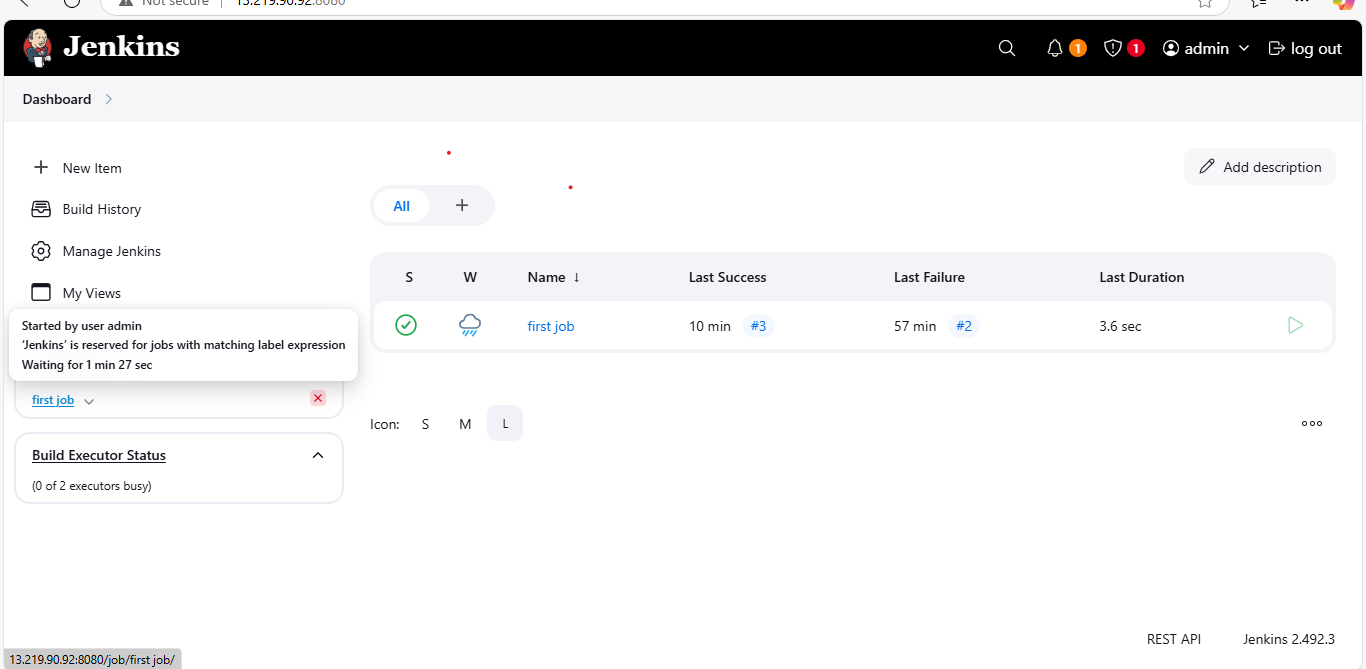
sudo mount -o remount,size=4G /tmp

sudo mount -t tmpfs -o size=4G tmpfs /tmp

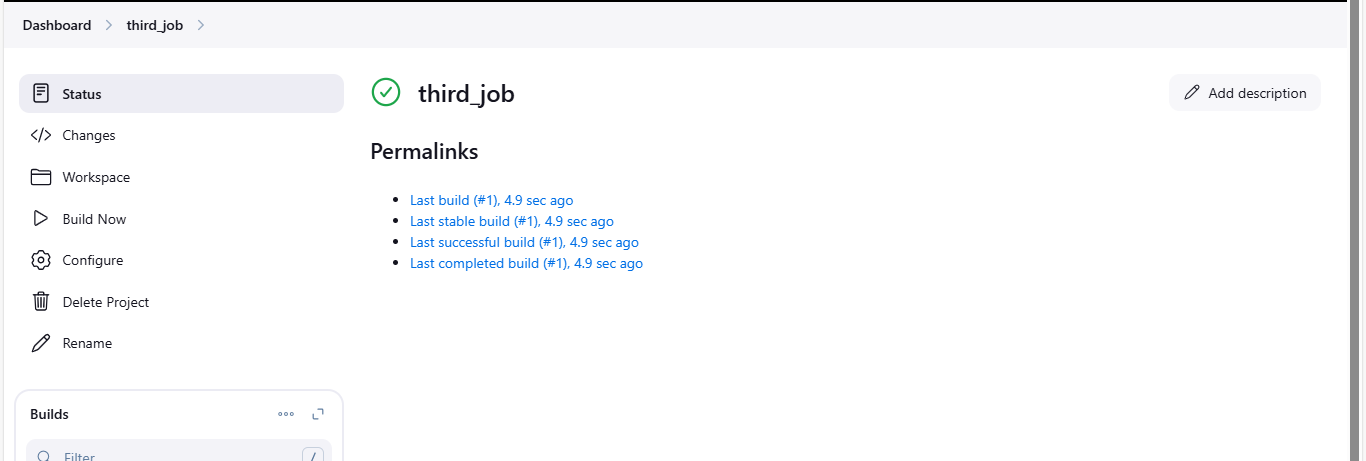
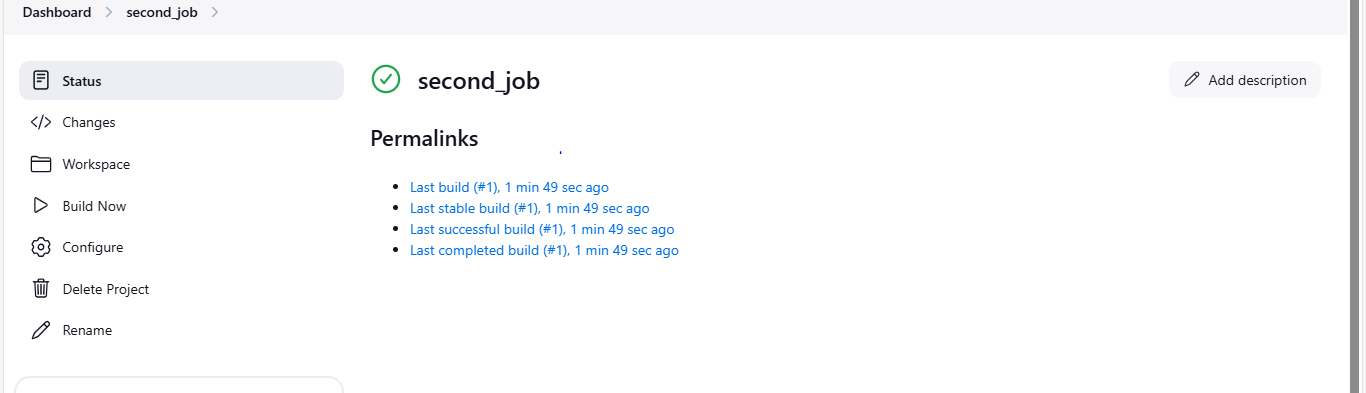
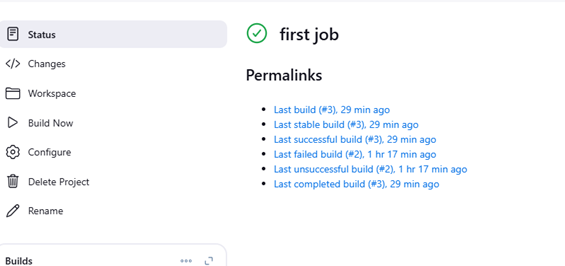
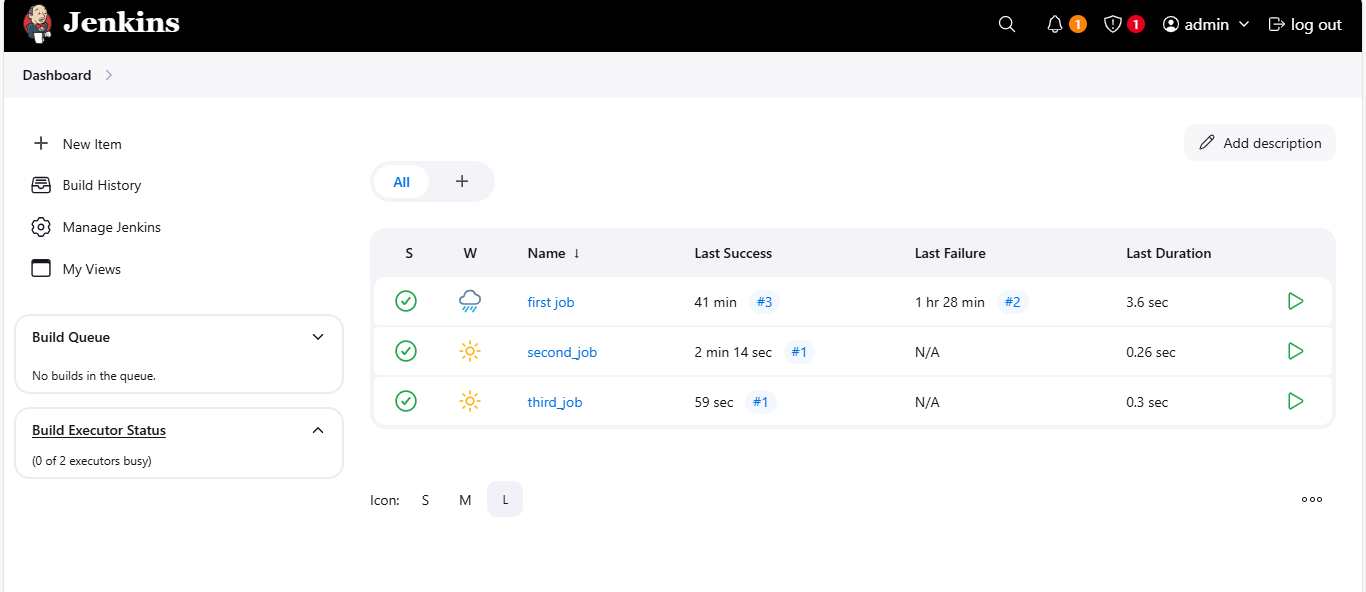
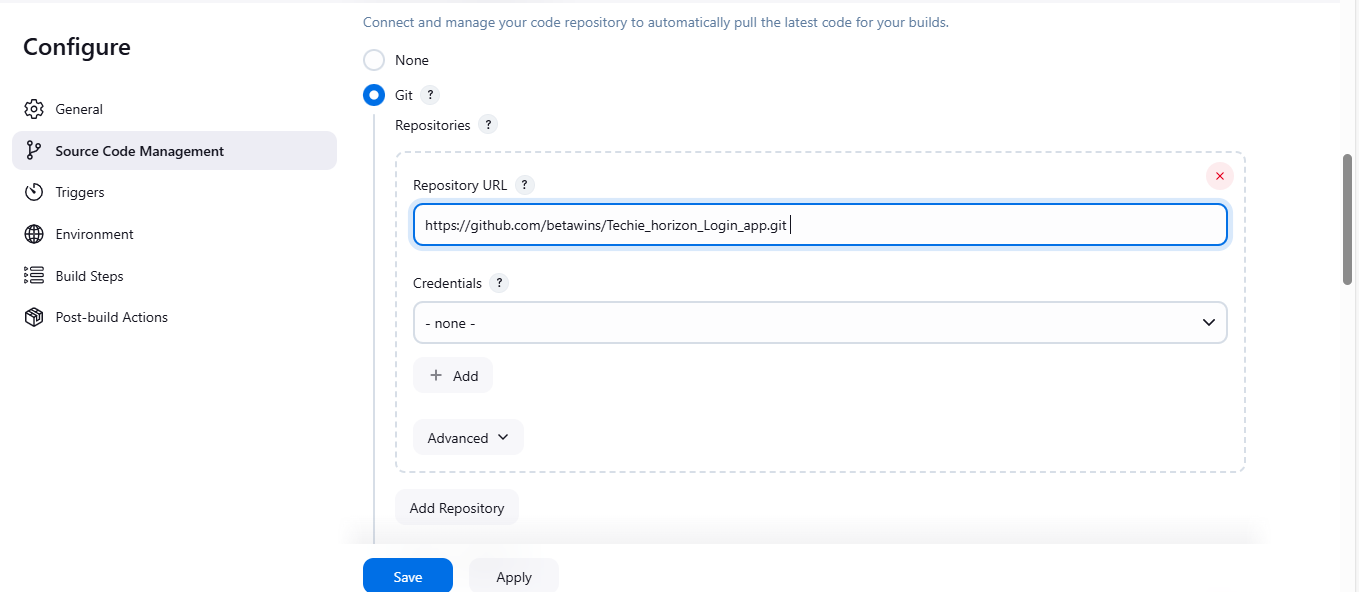
sudo systemctl daemon-reload

sudo systemctl restart Jenkins

Craete a new job with git hub url and click on build now when you check cd /var/lib/jenkins you will a find a new directory created with **Workspace**

 now you are able to view all files present in the project if we configured labels to this job we can resttrict this actions  

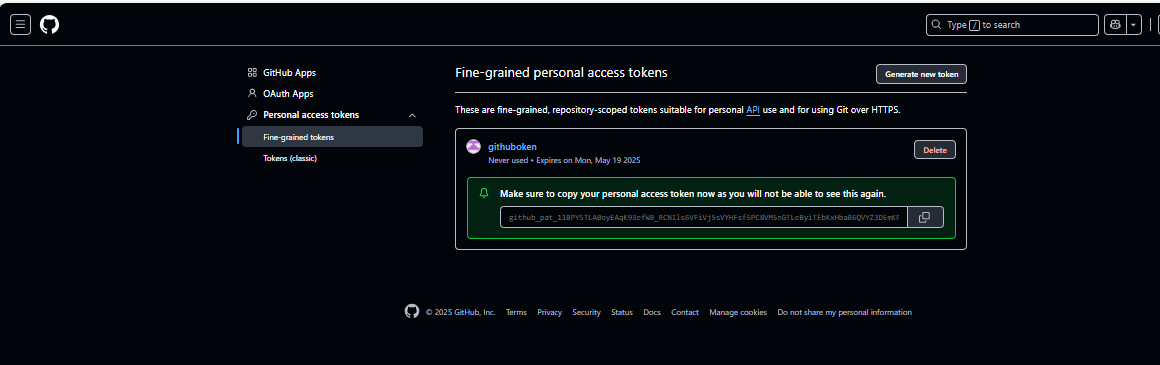
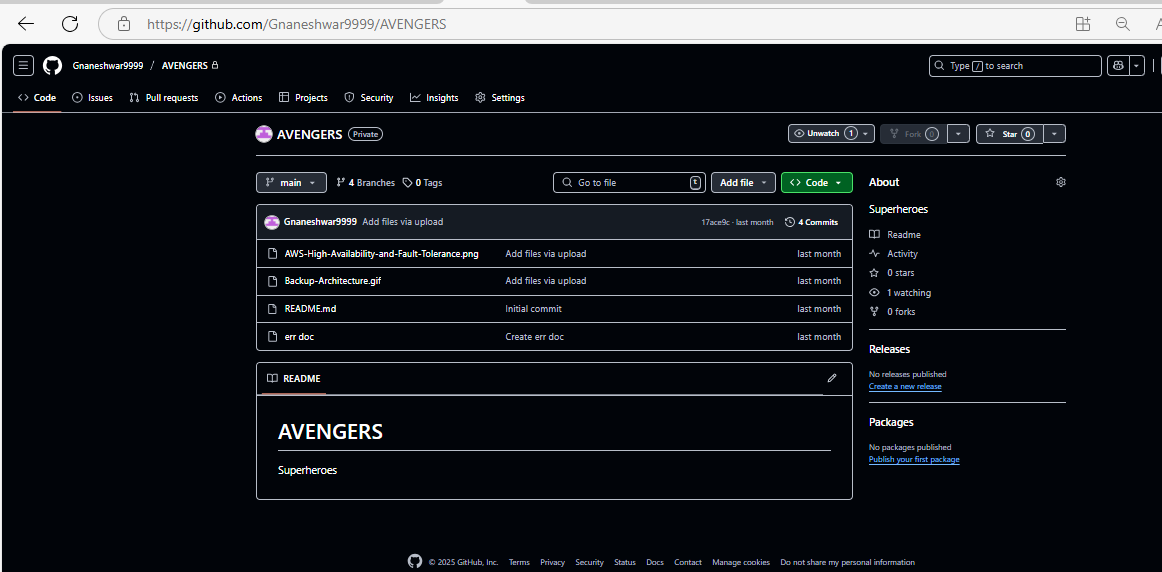
5) Create Three sample jobs using the below URL. <https://github.com/betawins/Techie_horizon_Login_app.git>



6) Create one jenkins job using git hub Private repository.

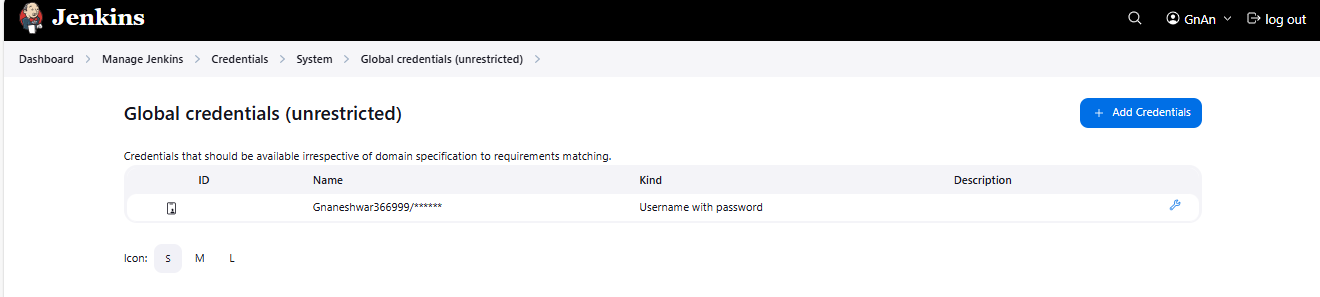
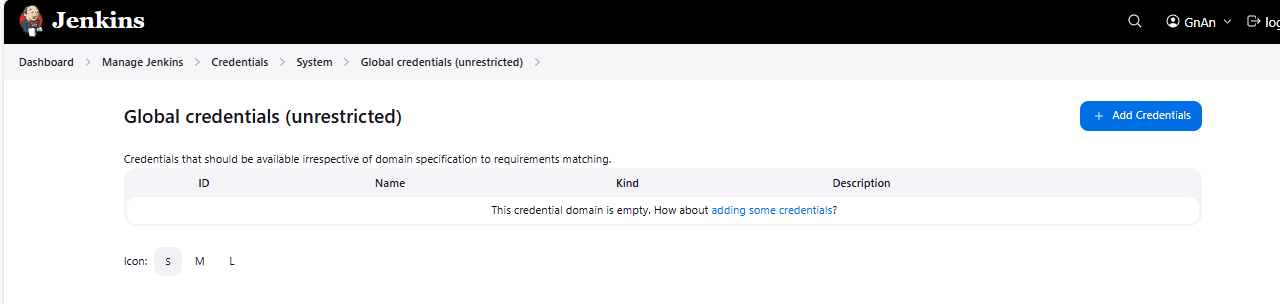
Create new job with github private URL and create 1 token in guthub

🡪Go to settings 🡪Developer settings 🡪tokens 🡪Tokens(classic) 🡪Give name and required permission for repo ,workfolow and click on create



Next Goto Jenkins Dashboard -🡪manage jenkins 🡪Credentials🡪

Global 🡪add credentials

🡪Username with password 🡪 Username and Password 🡪TokenURL 🡪Create

Got git bash add ssh-keygen and install git (if not)

git clone url :asks username as github username and password as taken url

🡪cd workspace 🡪cd job 🡪ll

Now we can see the files in github private repo

