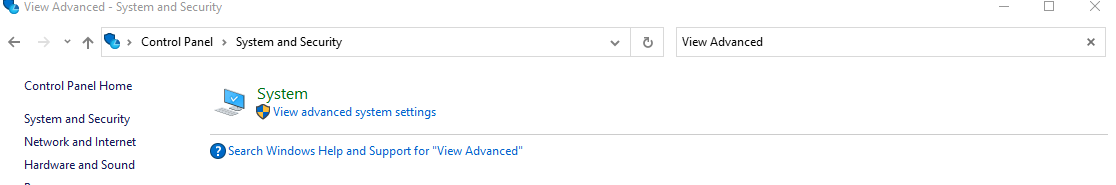
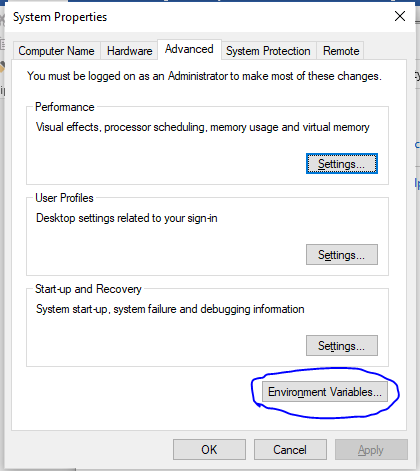
If u want to run the Jenkins u need Java 11 and above version

Java path setup

Open control panel and search view Advances system settings

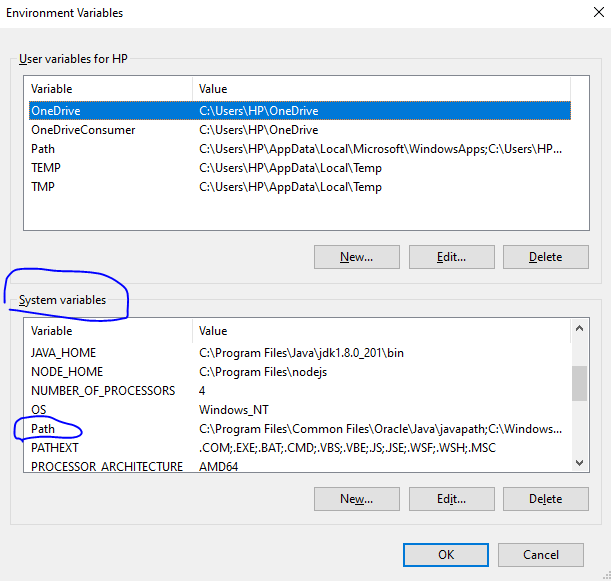


Click environment settings

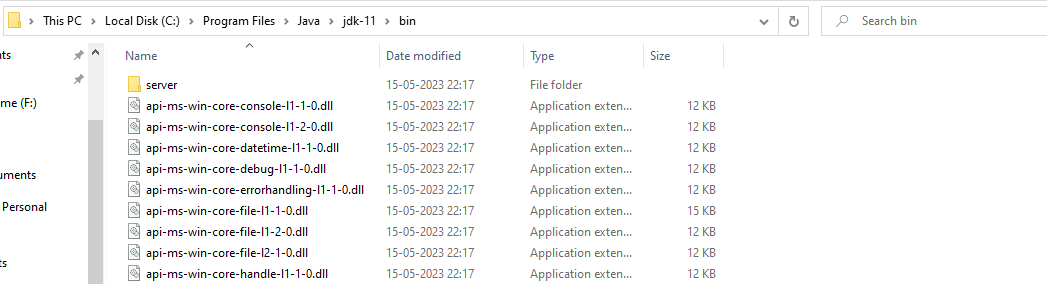


Under system variables

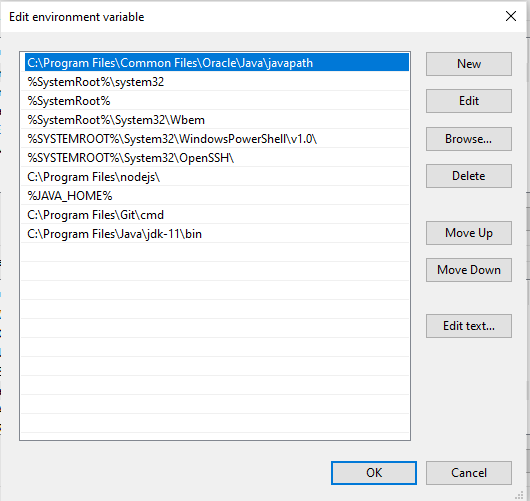
Click path if available if not click new button



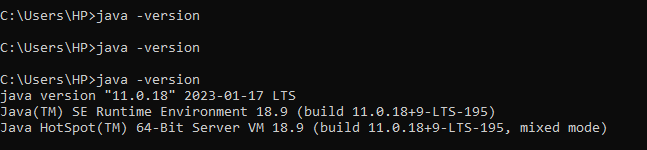
Open your java installed folder like this--🡪C:\Program Files\Java\jdk-11\bin



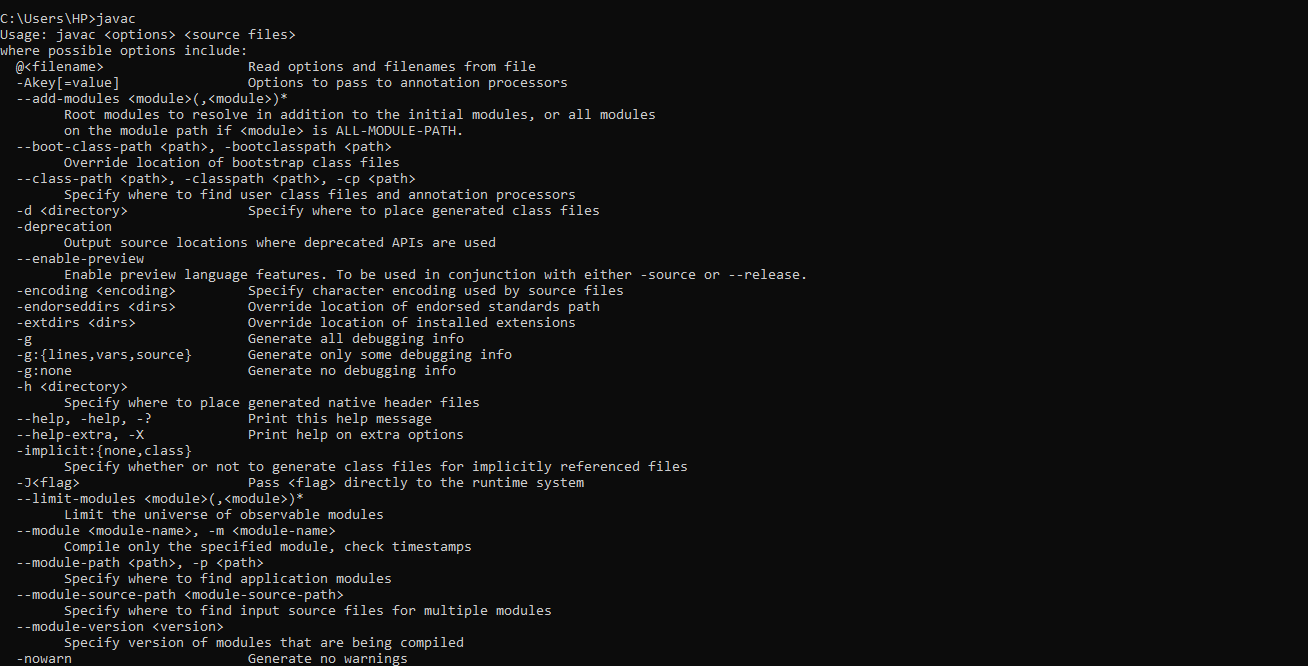
Copy the the folder path (C:\Program Files\Java\jdk-11\bin) and paste below any column & click Ok.



Java version checking command **java –version**

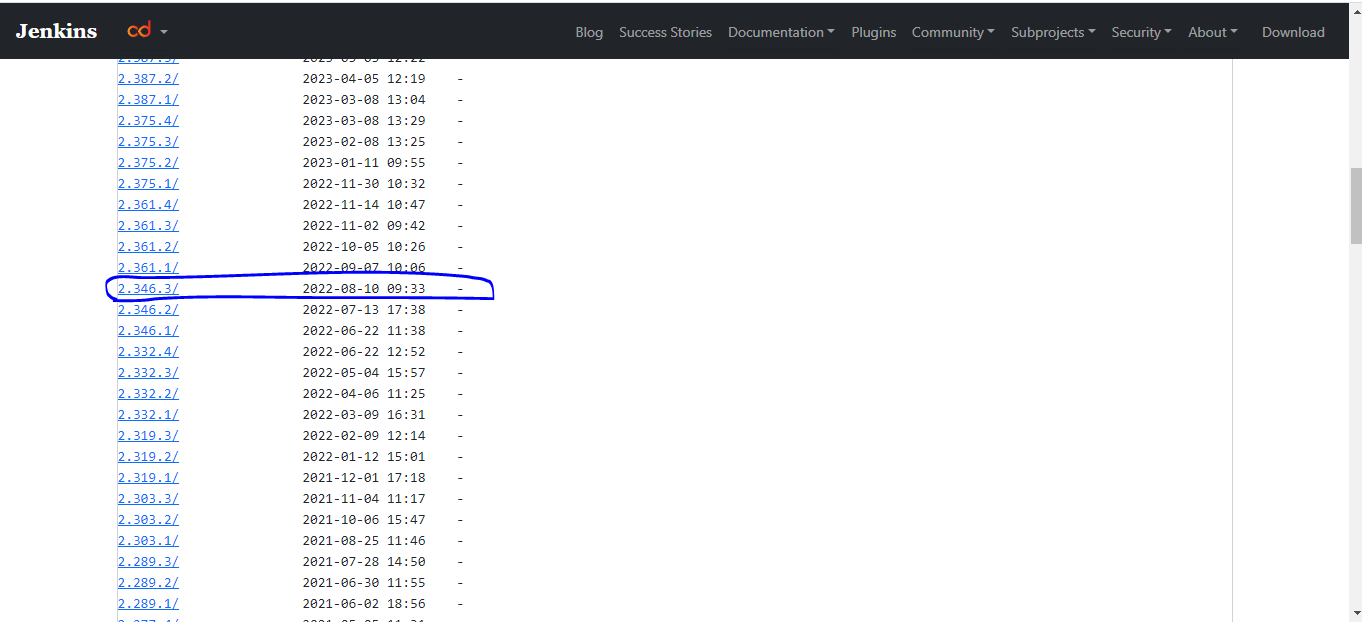


Compiler checking  **javac**

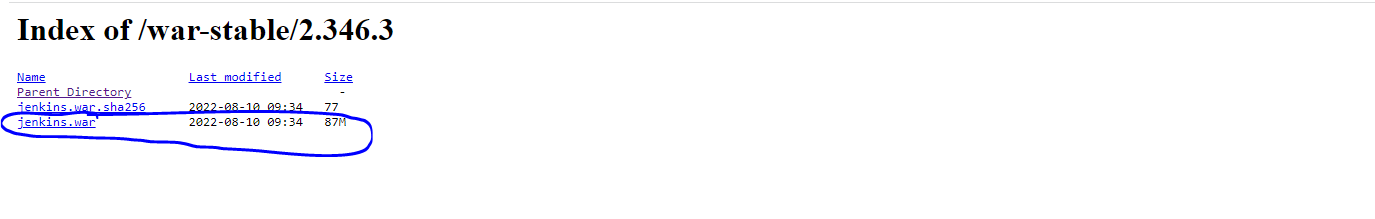


Jenkins download kink-- <https://get.jenkins.io/war-stable/>

Download Warfile



Select the war file

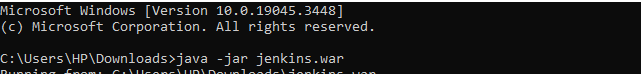


Got to Jenkins.war file downloaded location and type and open cmd promot

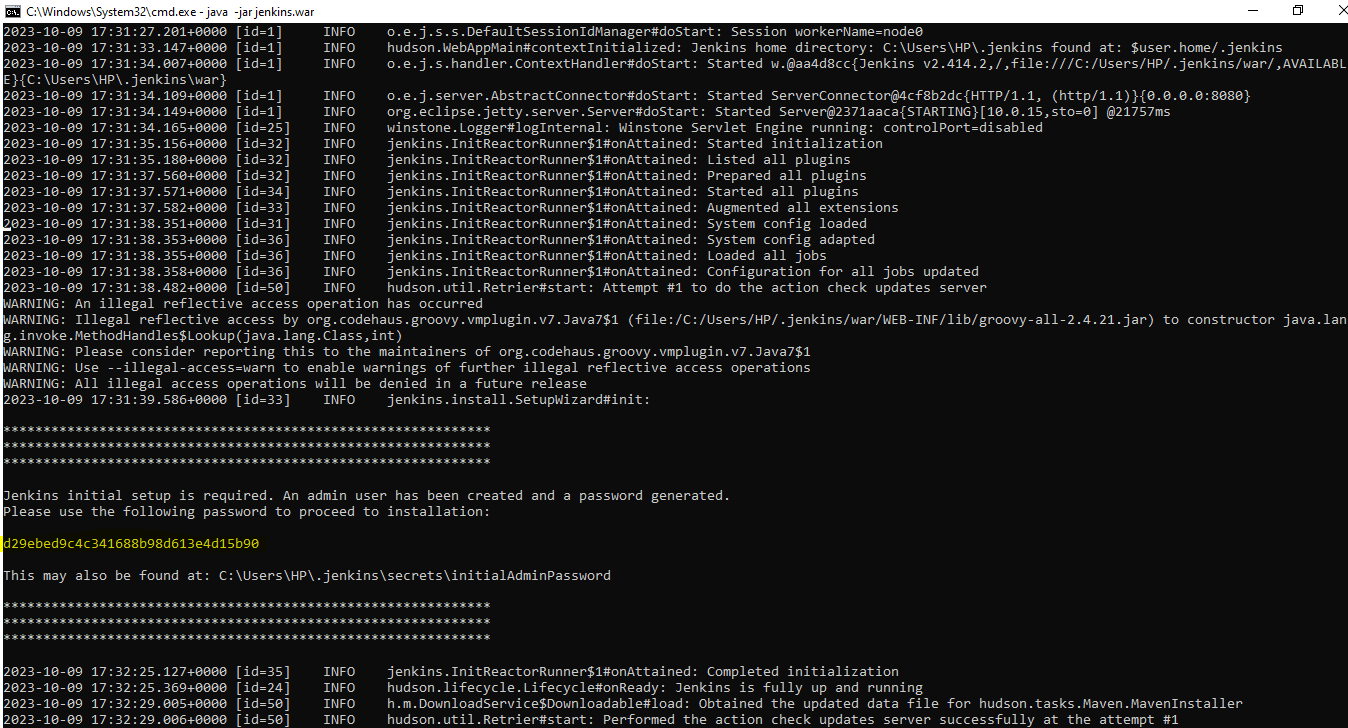
Open command promot and run the below command .

Note:don’t change the downloaded file path

Type: java –jar Jenkins.war



After it takes some time and it is showing below message.



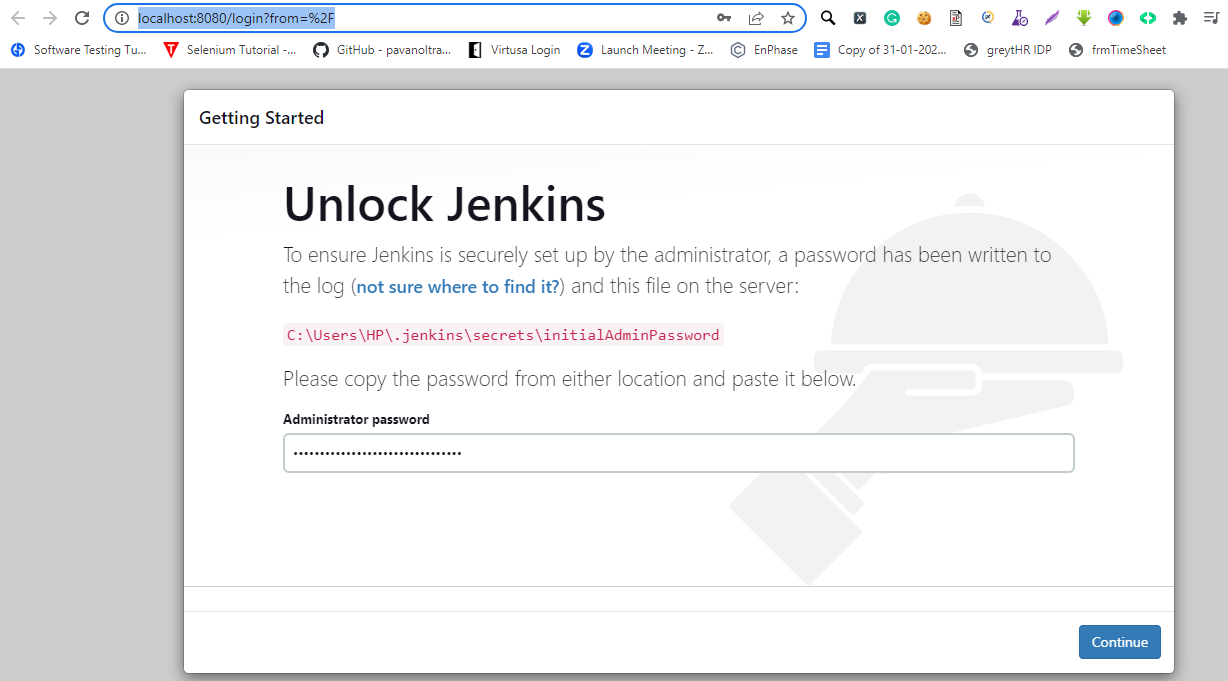
Above image It is showing **password** for initial setup and

Password also available Above image path.

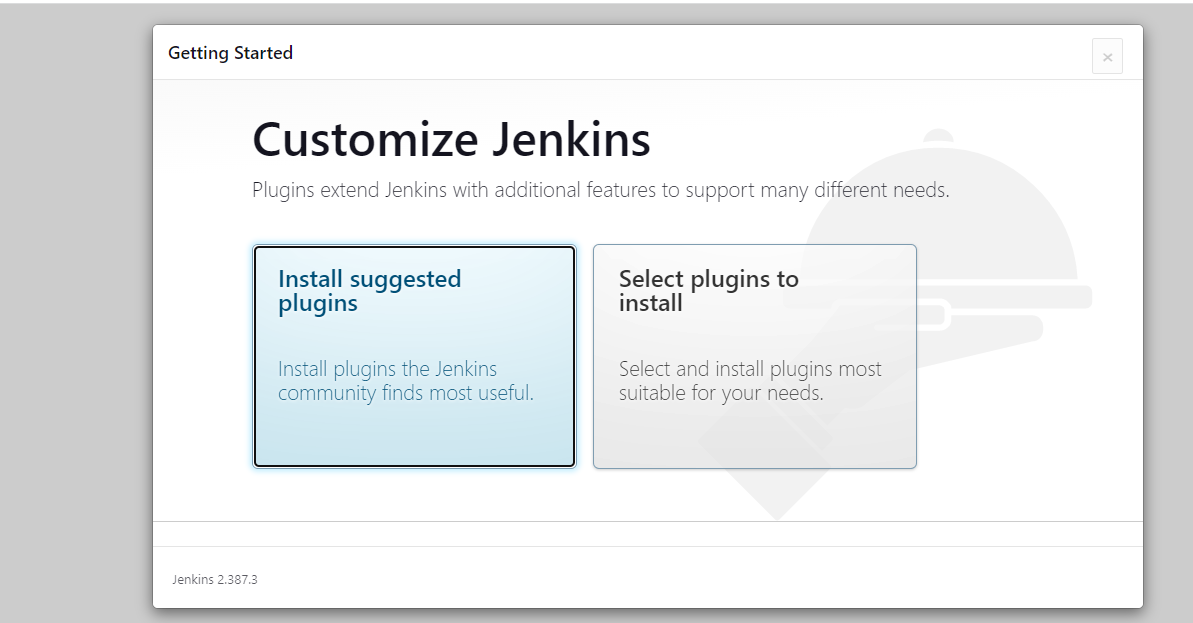
**d29ebed9c4c341688b98d613e4d15b90**

open chrome browser and type--http://localhost:8080/login?from=%2F

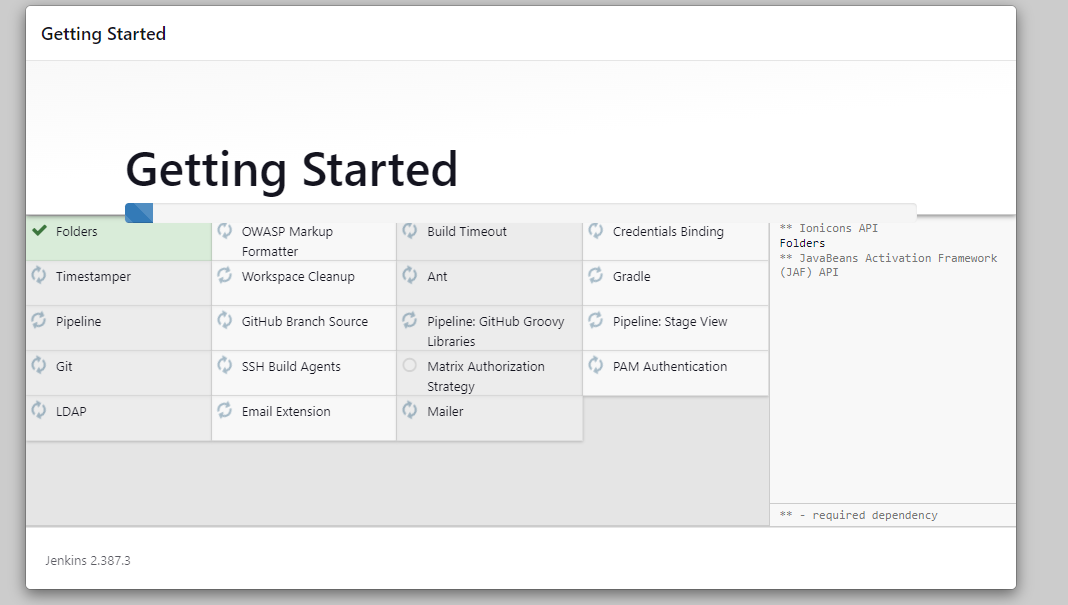
enter above password in password column and click continue button



After u are able to see below image and select Install Suggested plugins



After selection plugins are installed automatically



Click continue button after enter username and password click Save and Continue

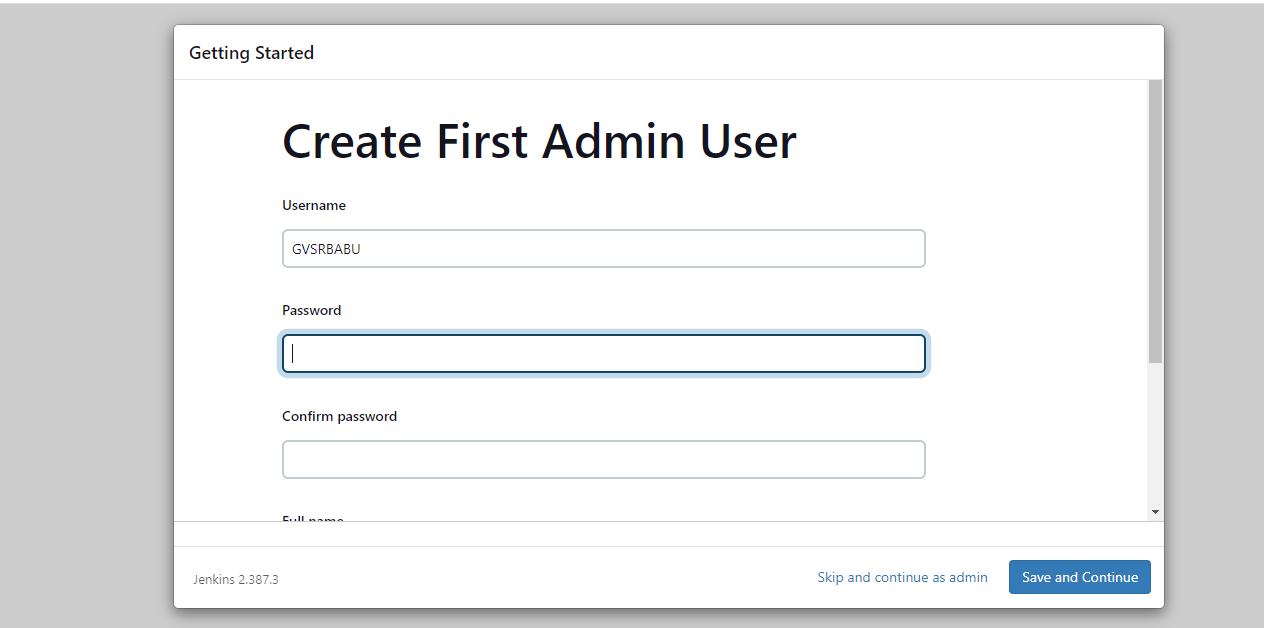
Username:GVSRBABU

Password: R@mesh123$

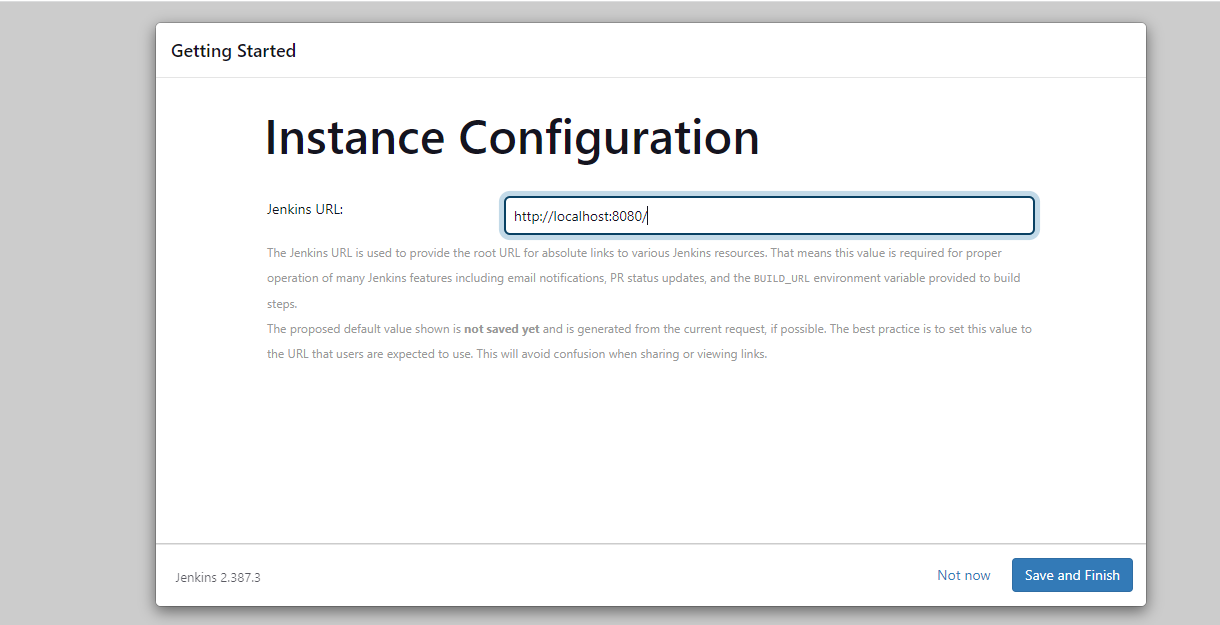
FullName:GVSRBABU

Emailaddress: ramgvs4@gmail.com

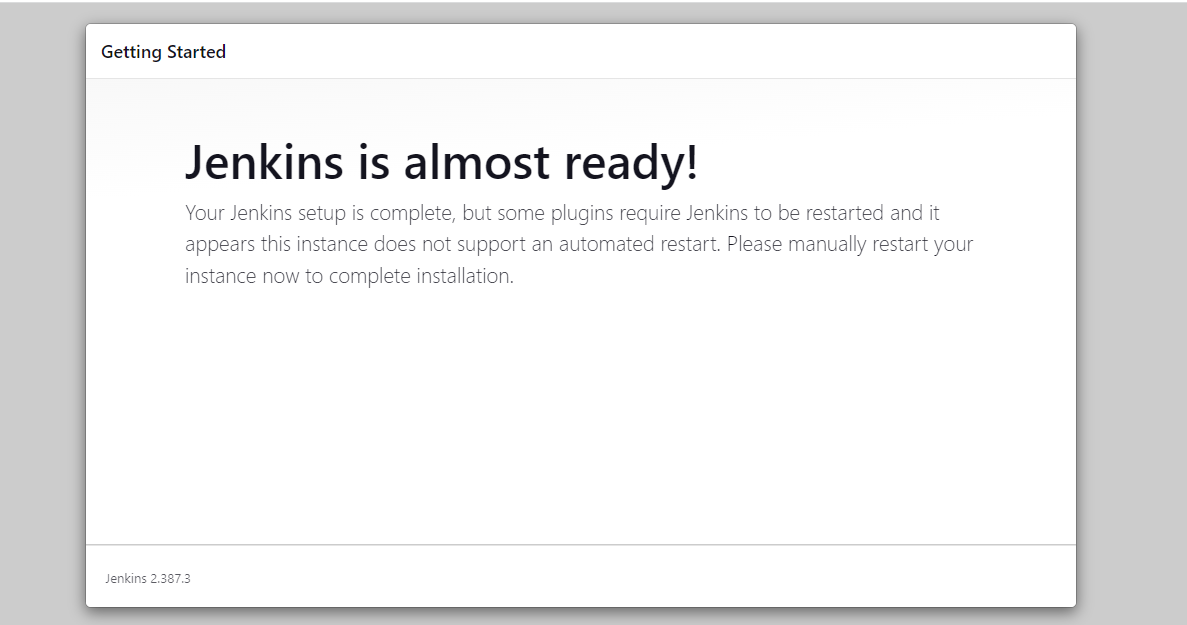
Click Save and Continue button



Click and Save and Finish button



After u got the successful message



During Jenkins setup don’t close cmd line.

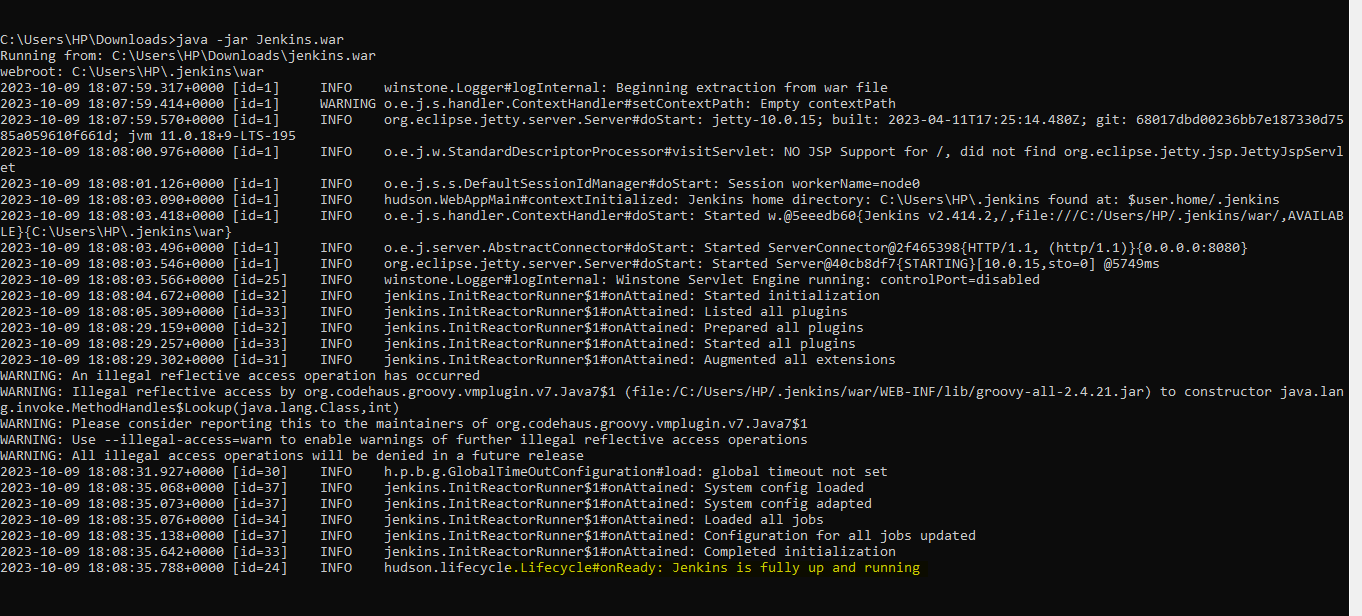
If u are suddenly close the project next time u are not able to open Jenkins

At the time u can go below location C:\Users\HP\.jenkins\config and open config file and in UseSecurity parameter change **true to false**

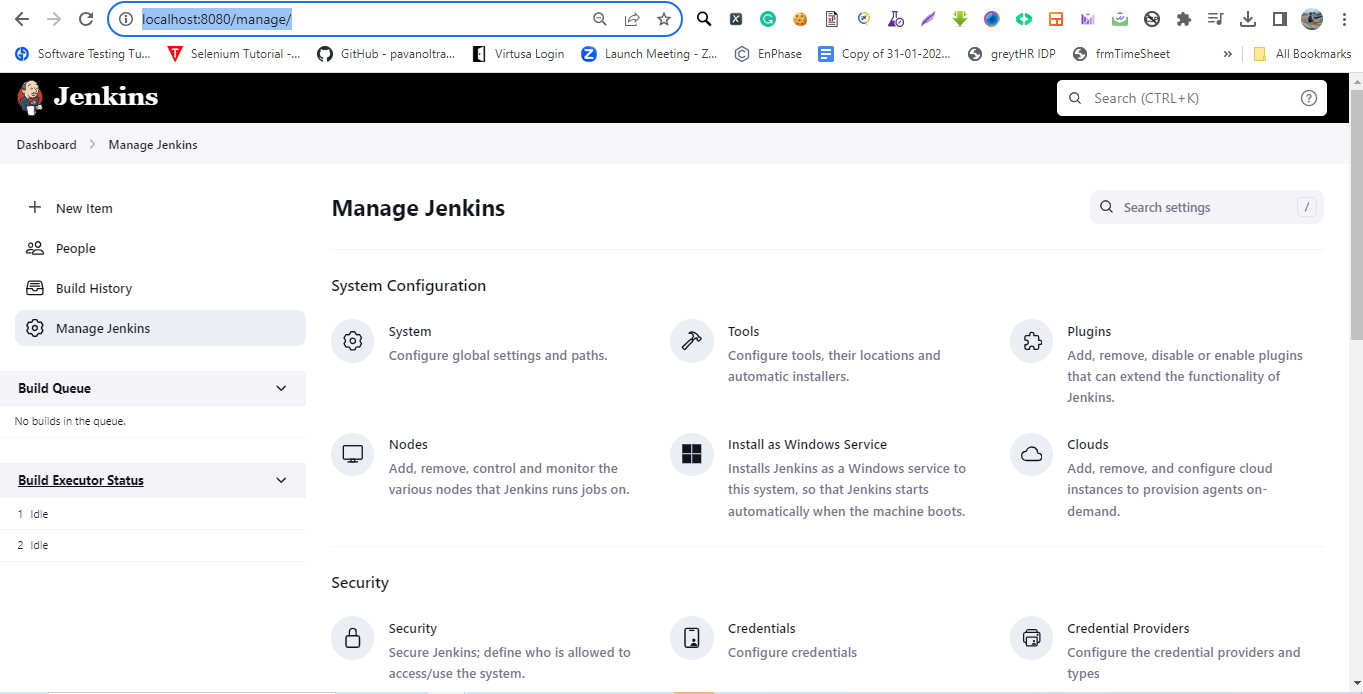
And click save and



Goto Jenkins.war downloaded file path(mostly every time it is in downloads file)again run below command in cmdjava –jar Jenkins.war and we are able to see success message “**Jenkins is fully up and running**”

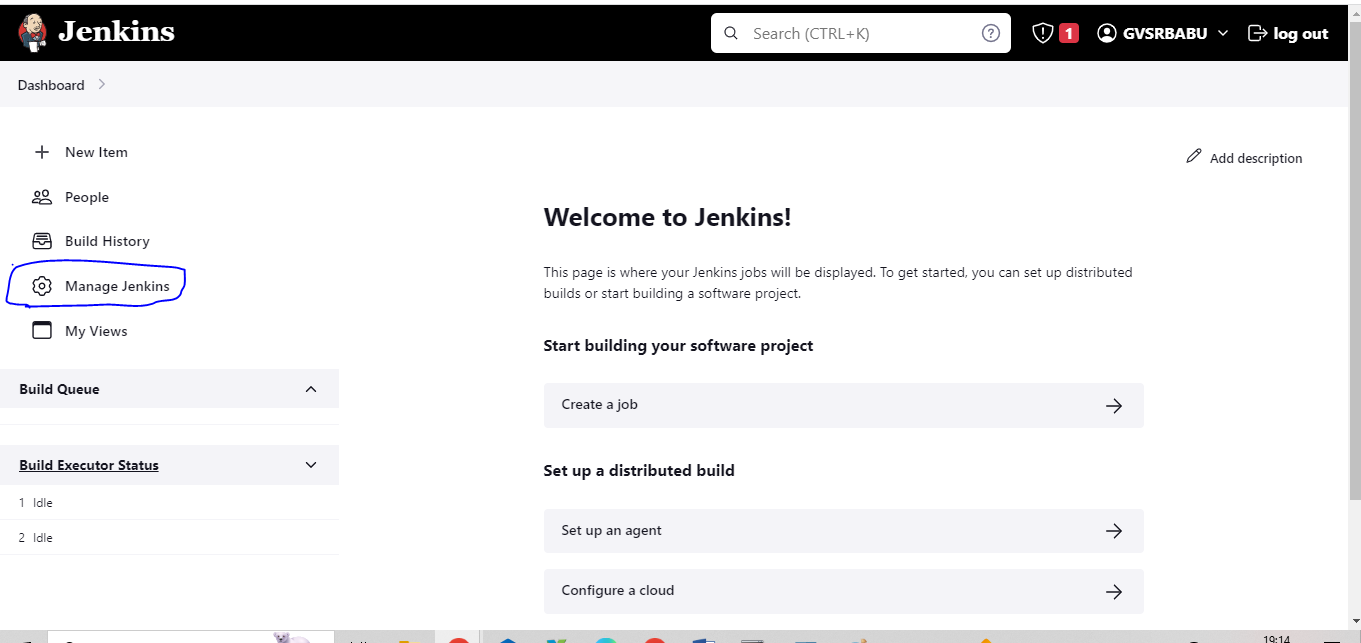


**Open** [**http://localhost:8080/manage/**](http://localhost:8080/manage/) **able to use Jenkins**

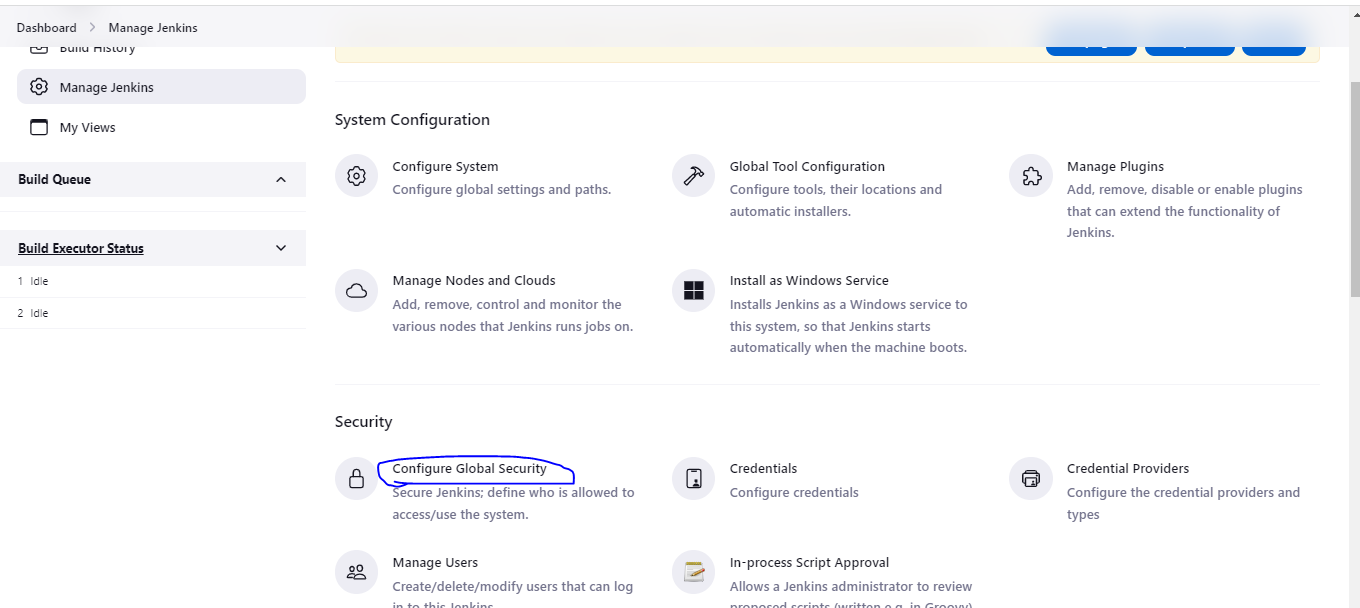


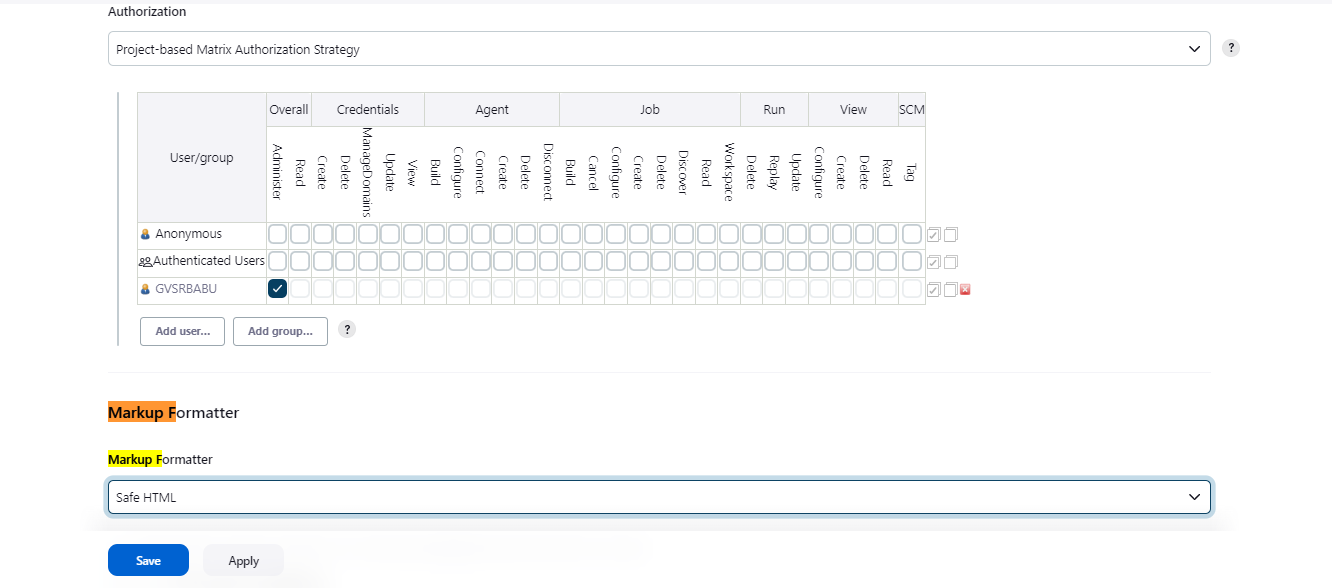
**2.How to Configure Jenkins in Windows**

Login Jenkins goto 🡪Manage Jenkins🡪Configure System

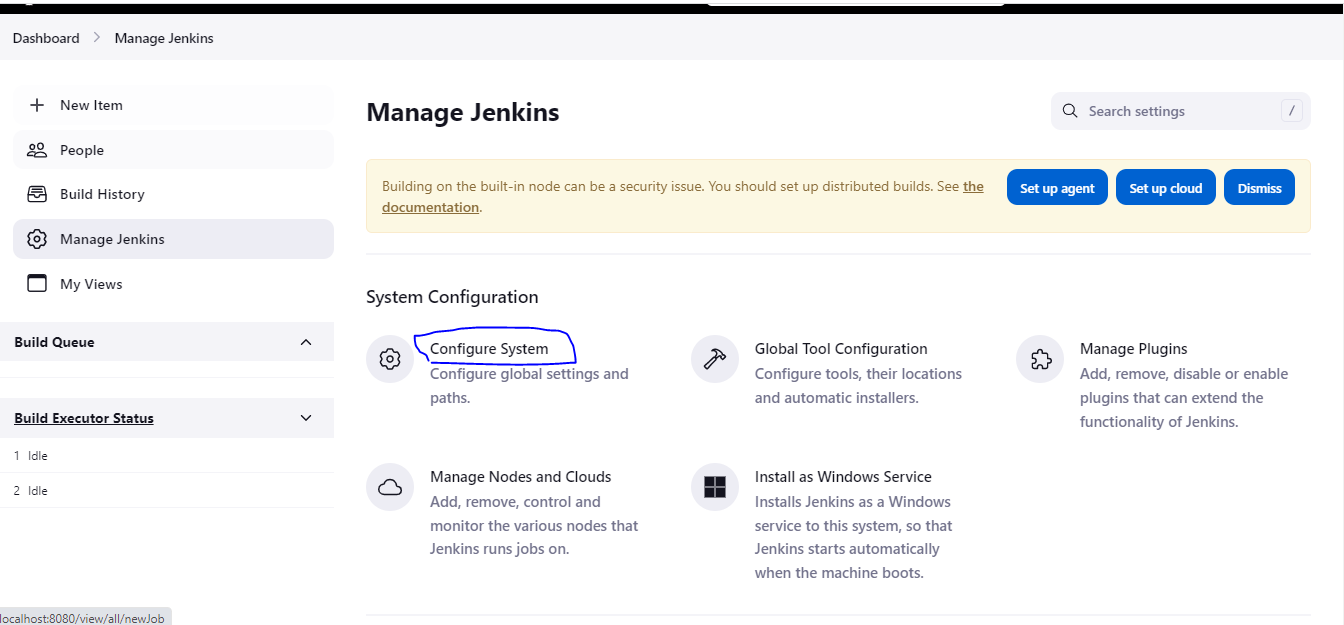


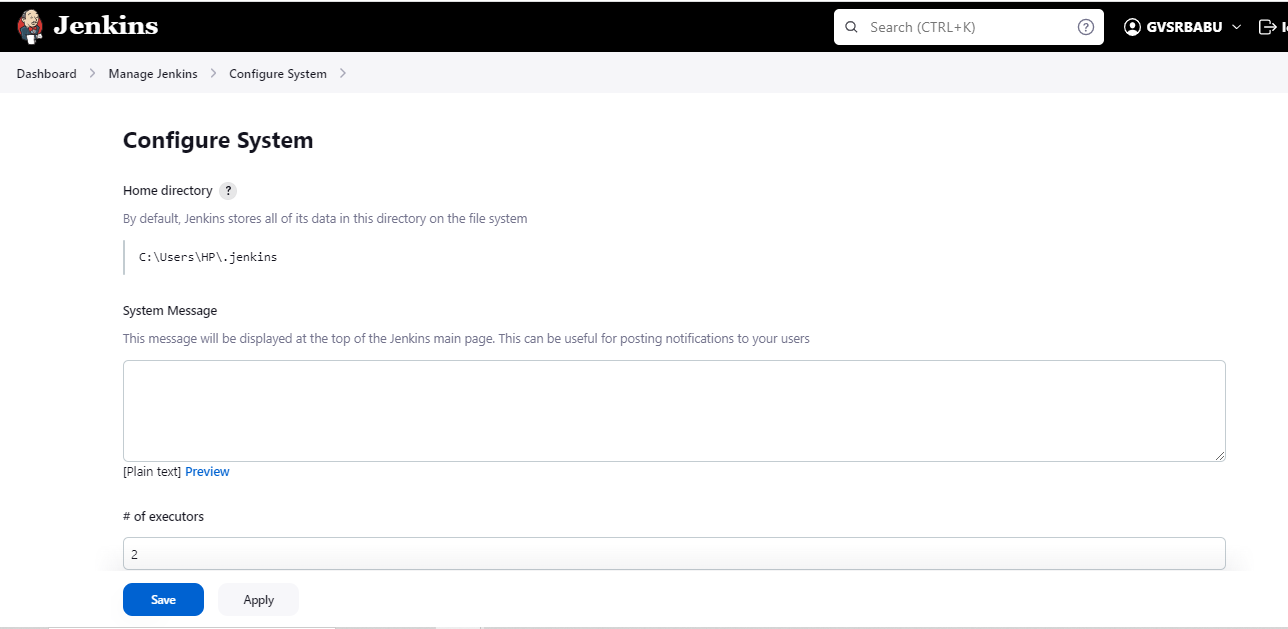
Set these properties configure global Security





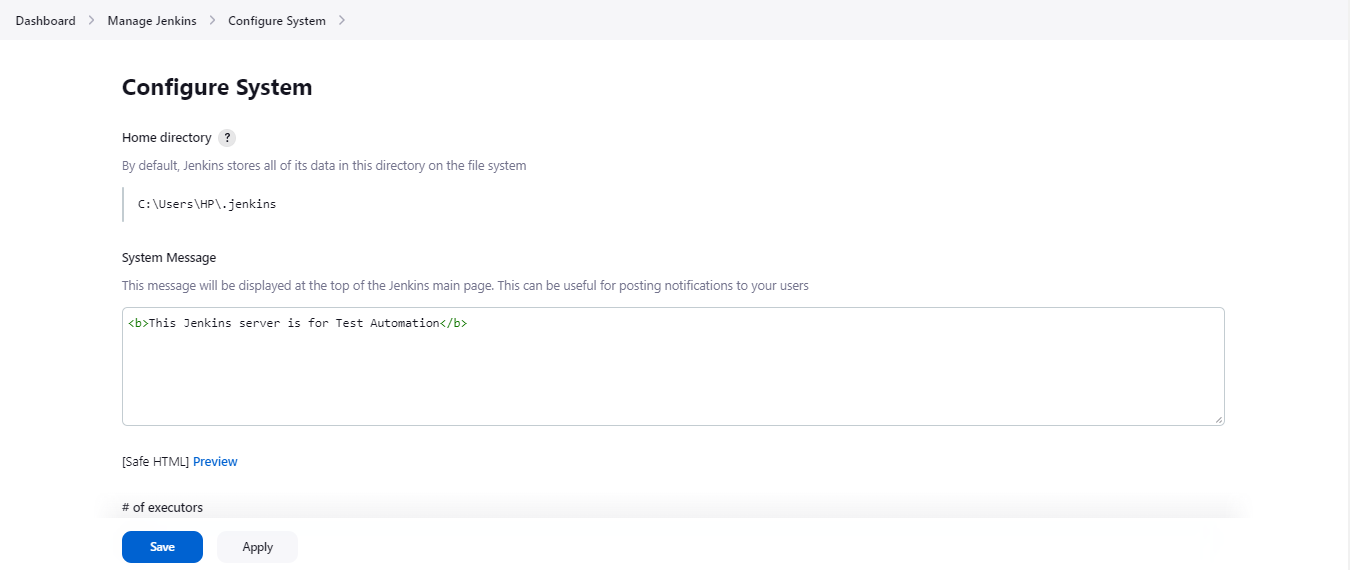
Save and Apply.

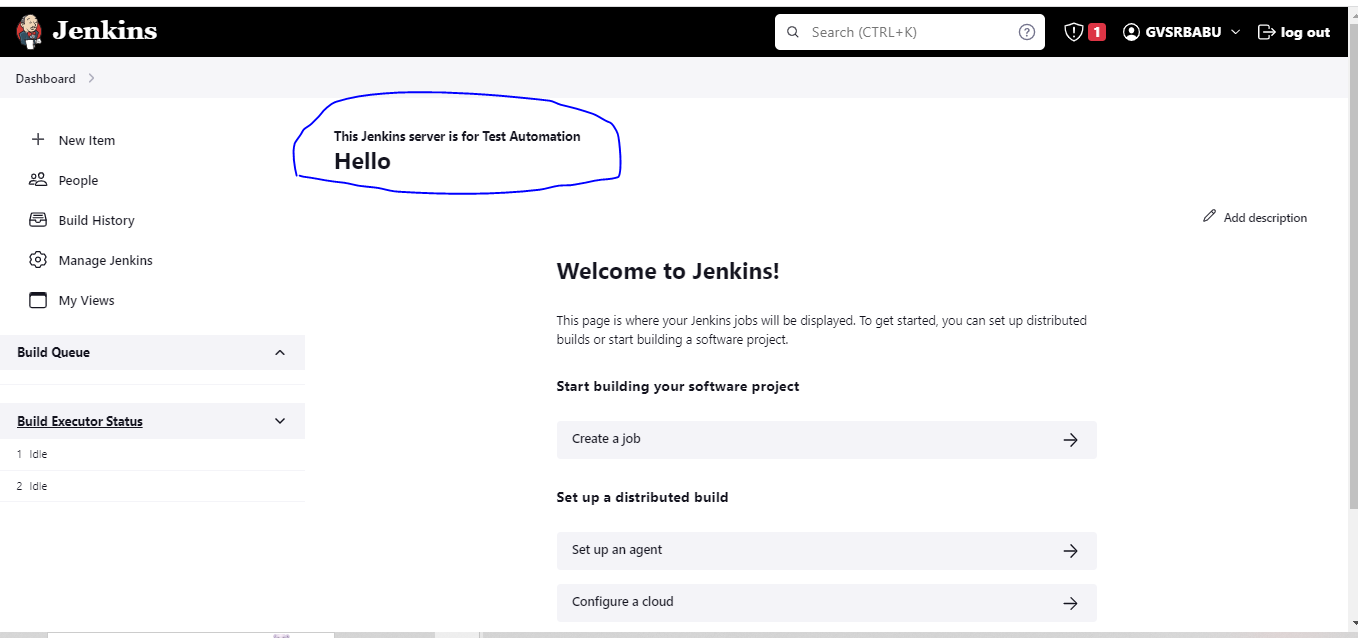




Enter text in System message box <b>This Jenkins server is for Test Automation</b>

Click Save and Observe in Home page it reflects

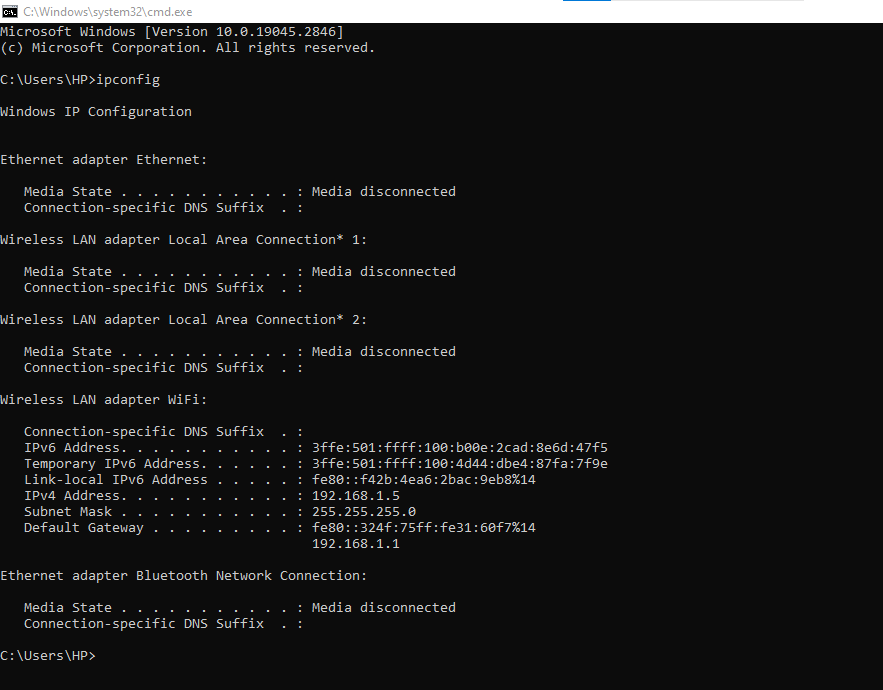




Configure jenkins url

If you are using windows machine open cmd promot

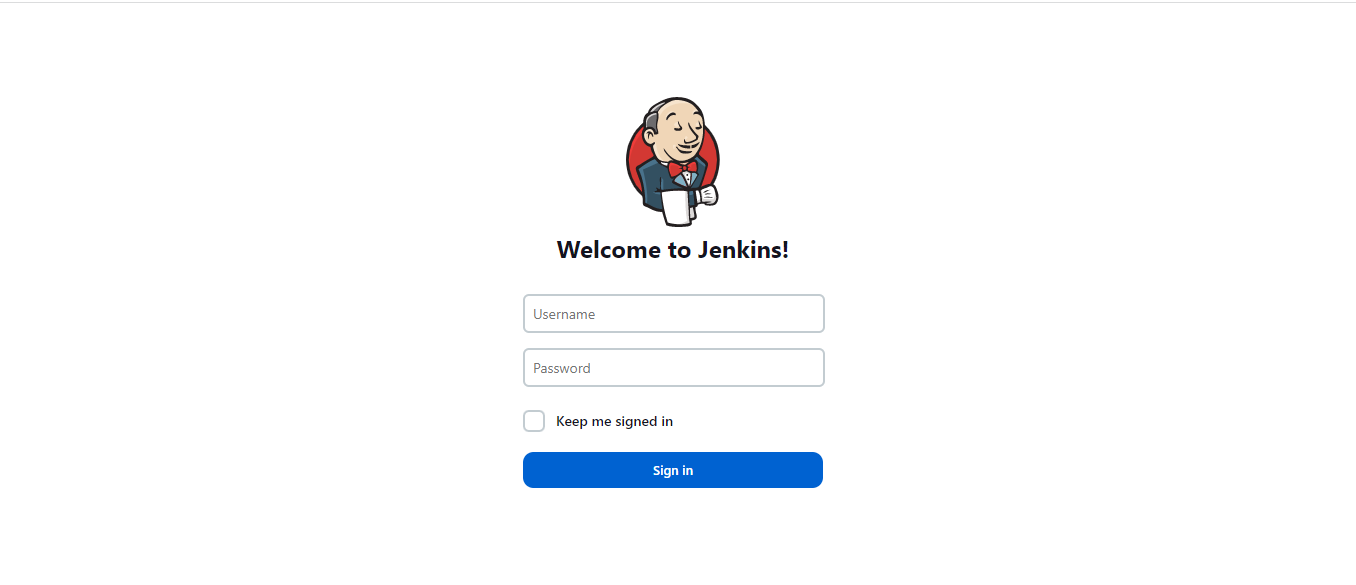
Type ipconfig



Copy IPV4 Address—192.168.1.5

Configure with local host with IPV4Address

Got chrome browser and edit <http://192.168.1.5:8080/login?from=%2Fmanage%2Fconfigure>

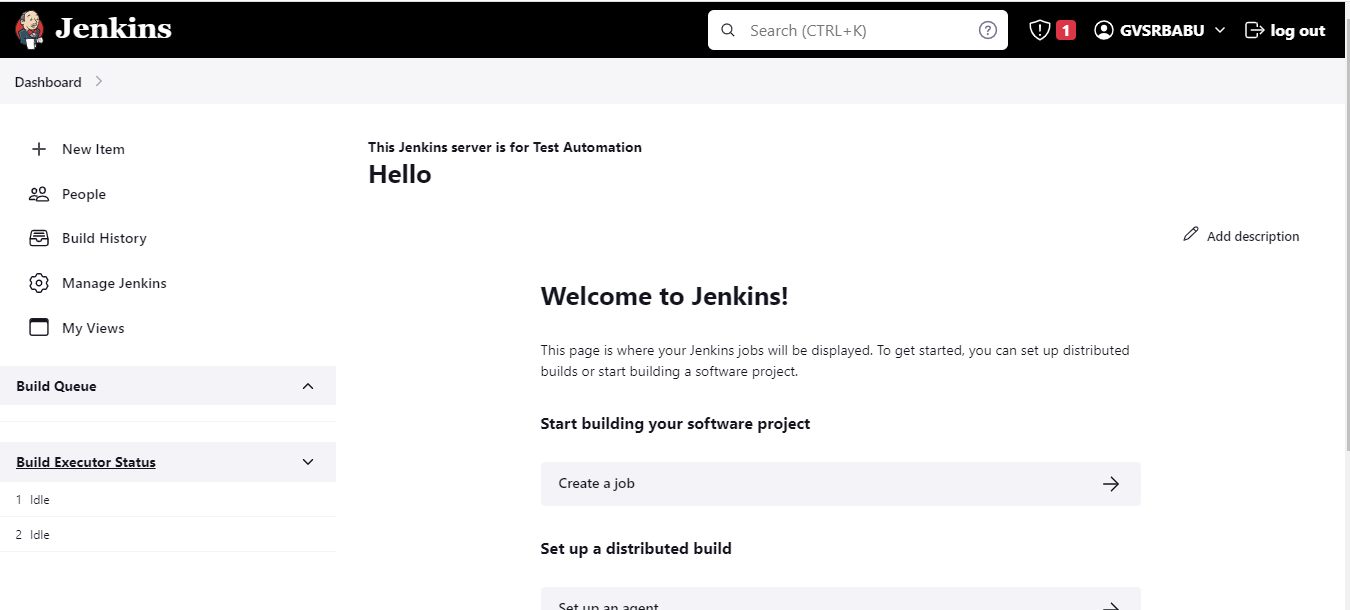


Enter username,Password what ever we are created

Username:GVSRBABU

Password: R@mesh123$

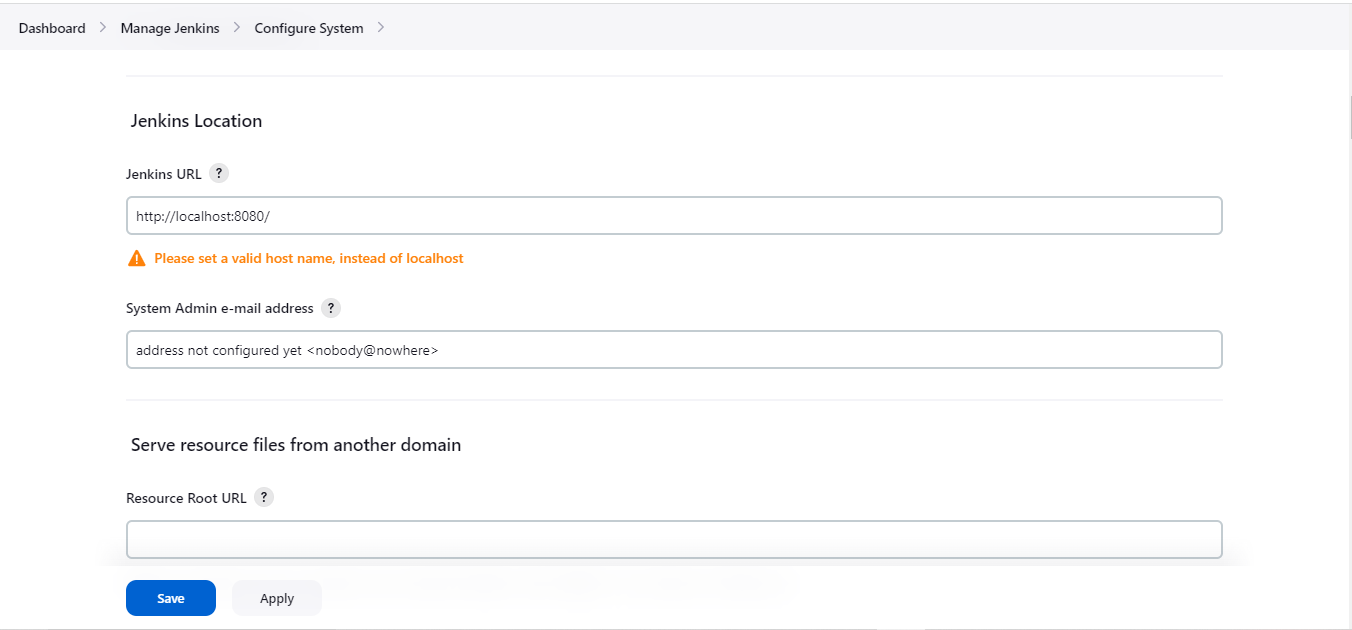
Userable to see the home page



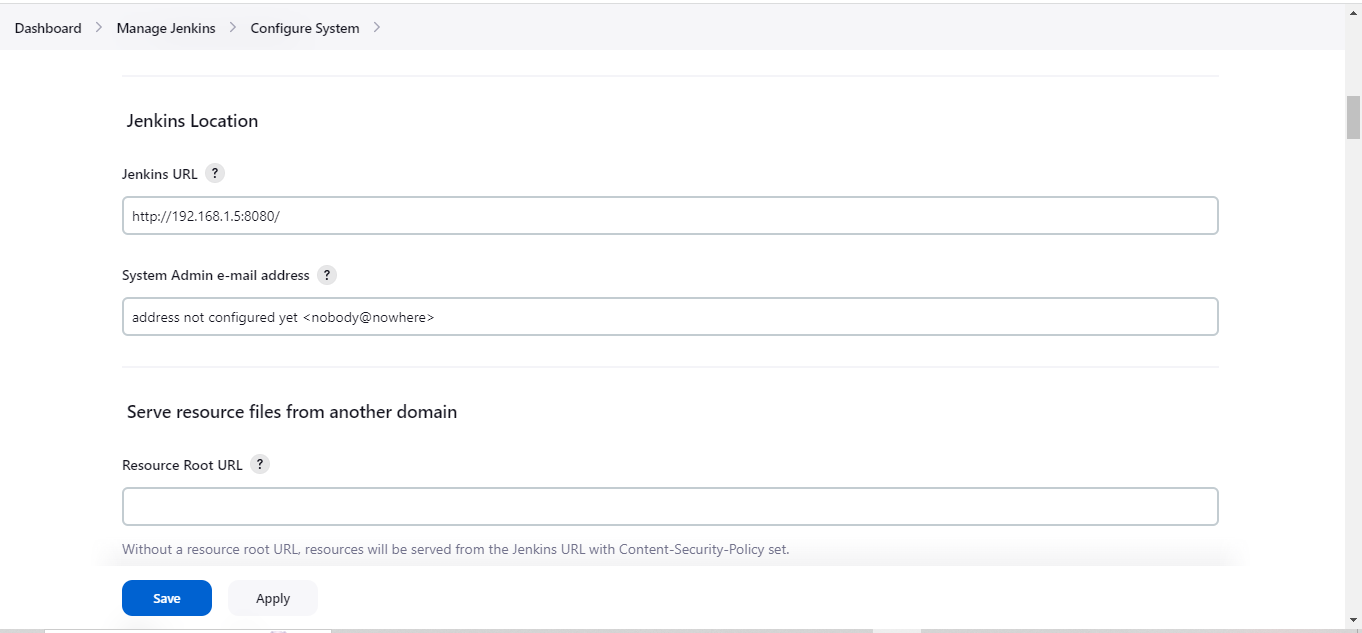
Change the local host address in configuration section <http://192.168.1.5:8080/>

Manage Jenkins🡪Configure System🡪Jenkins URL Text box

Add remove old url(http://localhost:8080/) and add new configured url(<http://192.168.1.5:8080/>)



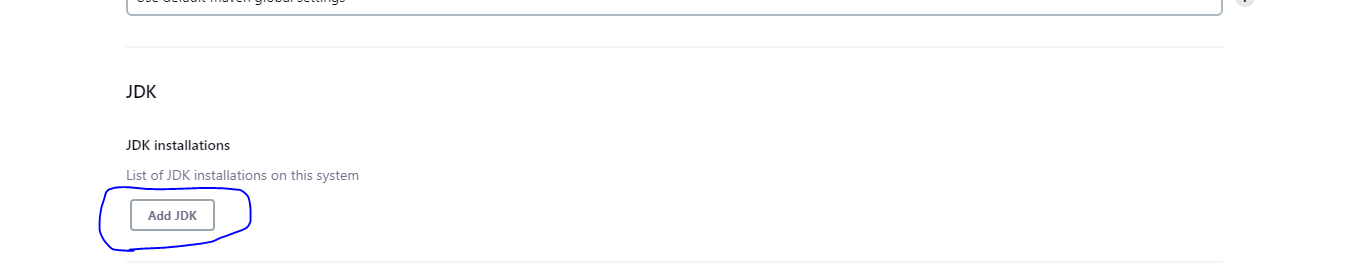
Click Save and Apply now u can easily able to access configured url



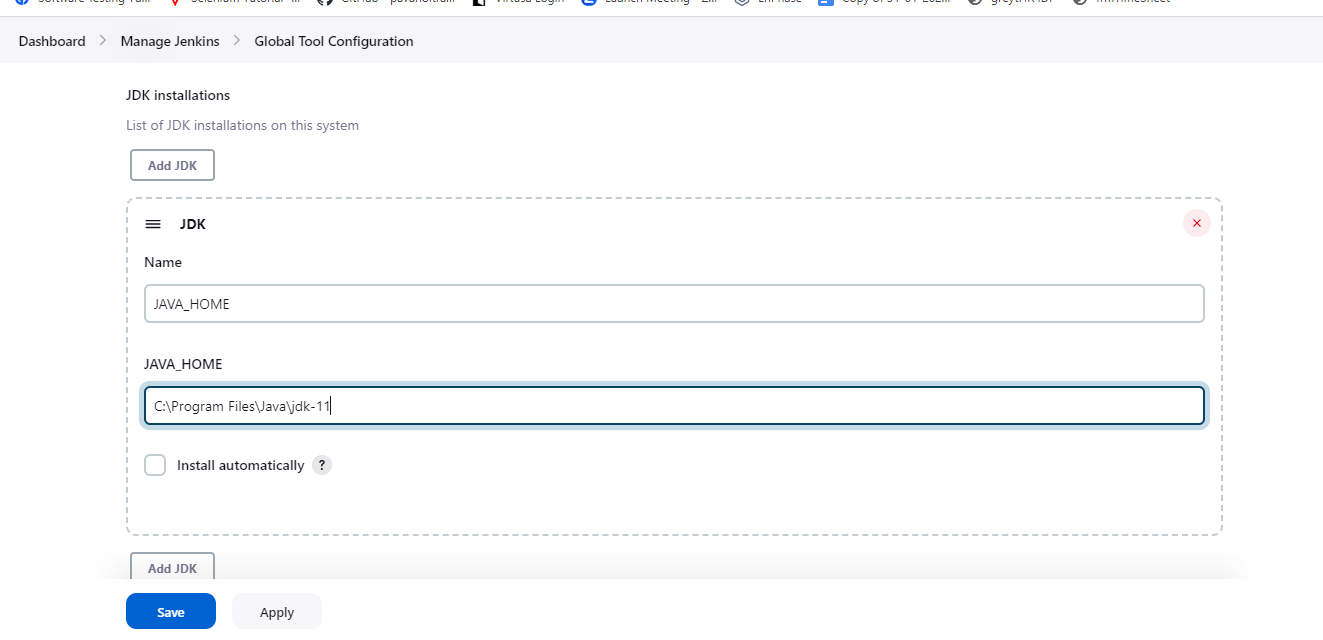
Global Tool Configuration

Manage jenkins🡪Global Tool configuration

Click Add JDK button



Goto java Installed folder copy the java file and paste in JAVA\_HOME text box

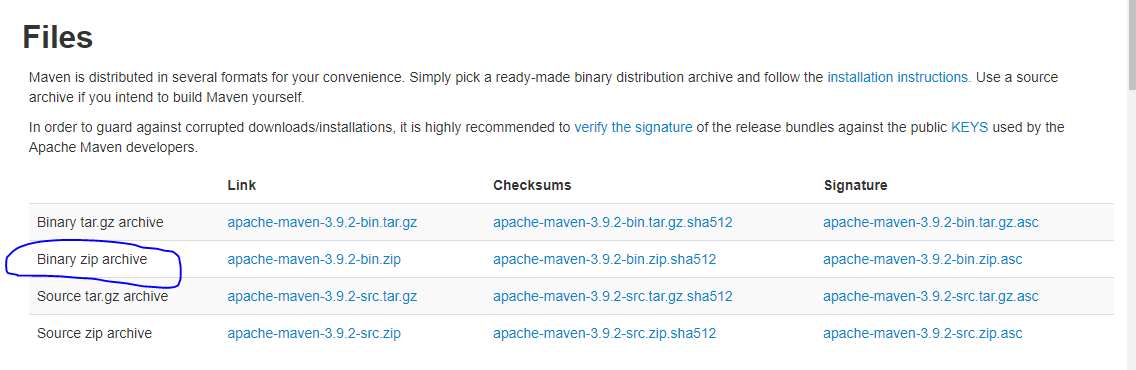


Same as GIT,

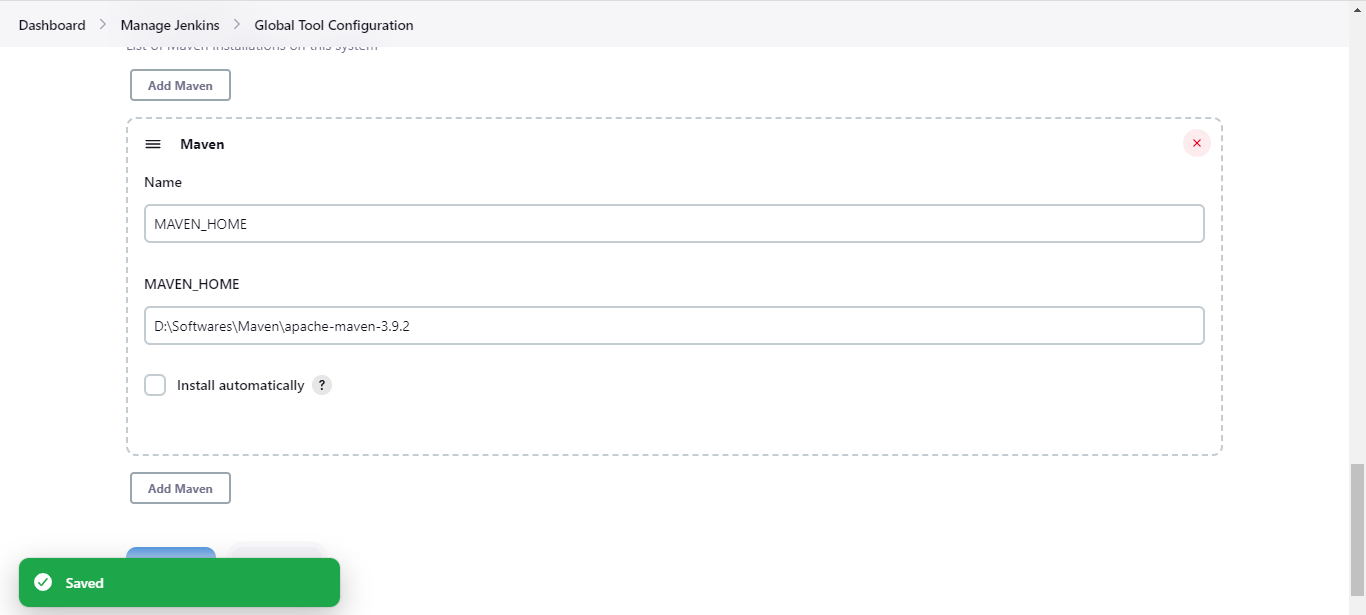
Maven

how to download maven 🡪https://maven.apache.org/download.cgi?.

select Binary Zip format and click download



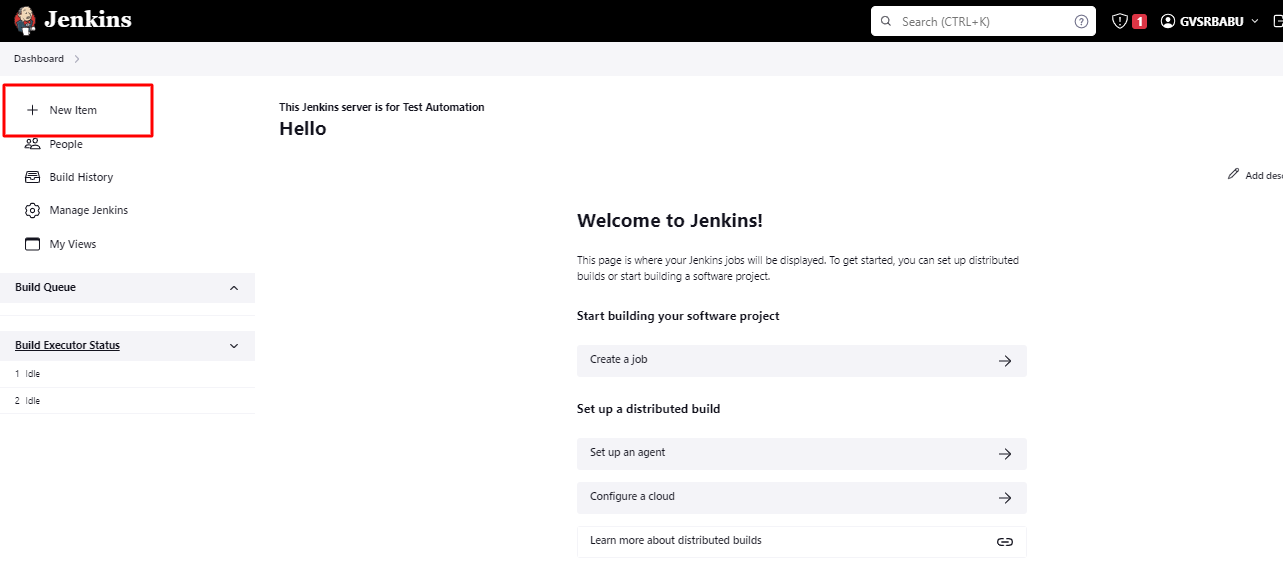
After extracting open the file and goto up bin folder and copy the path paste in below Maven text box ex: D:\Softwares\Maven\apache-maven-3.9.2



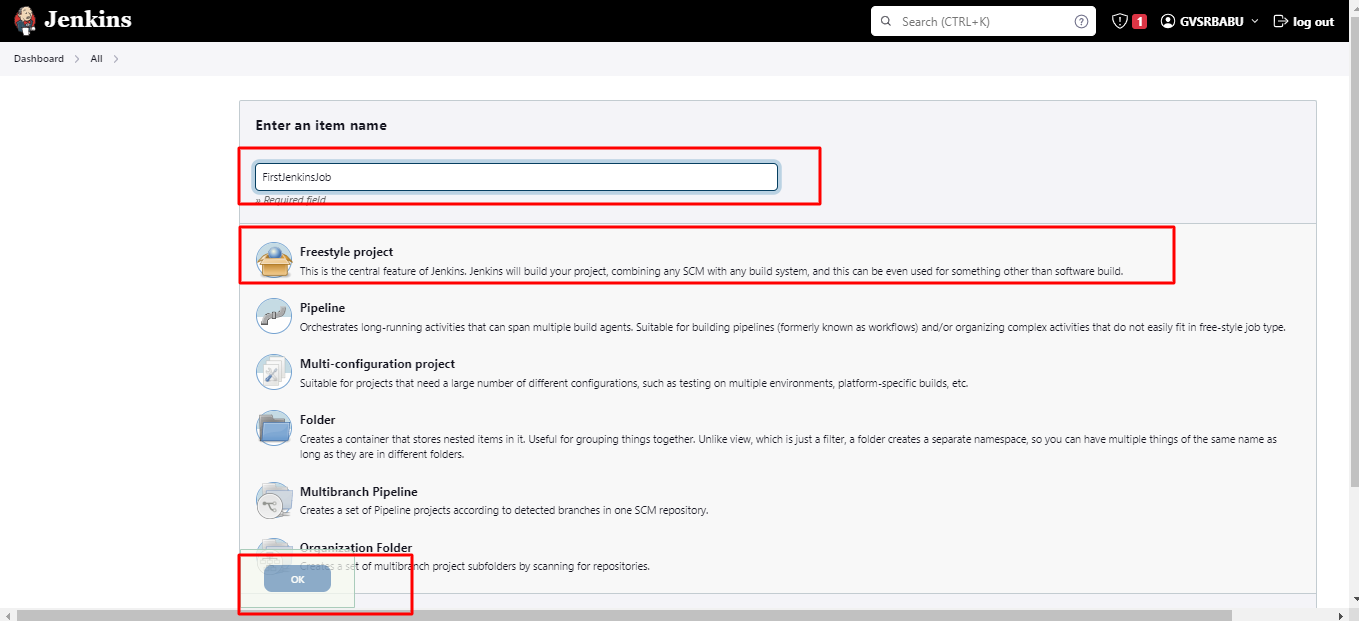
Click Apply and Save.

Create New Jenkin job

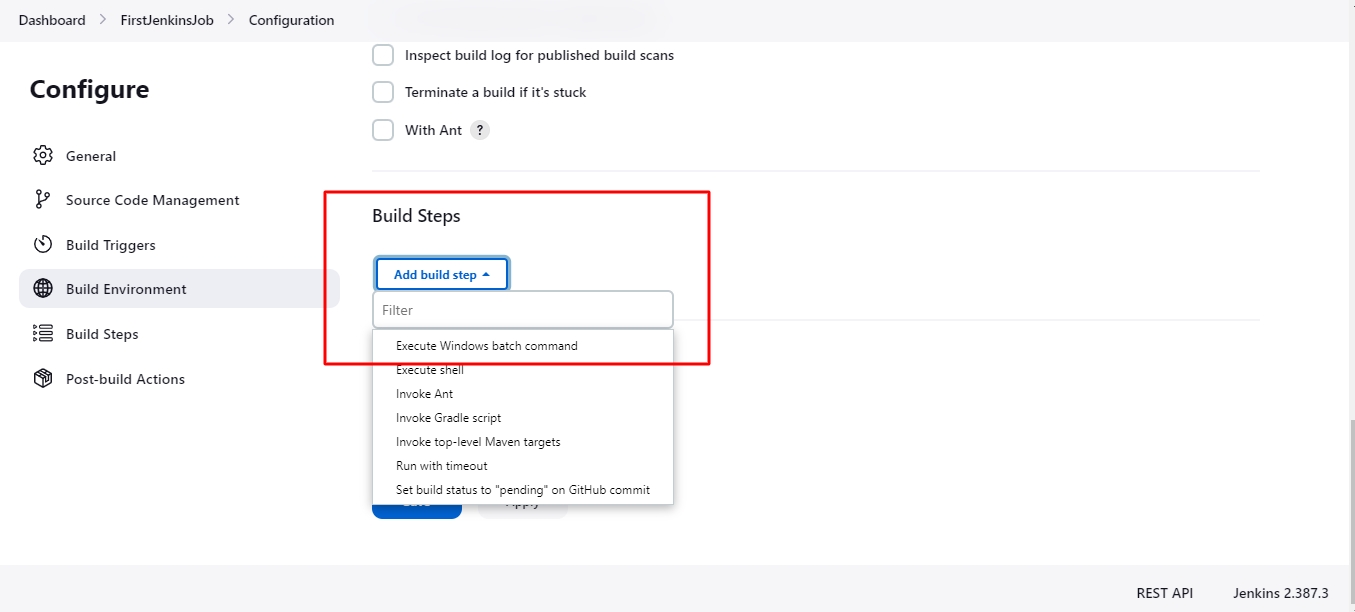
Select New Item and enter the job name



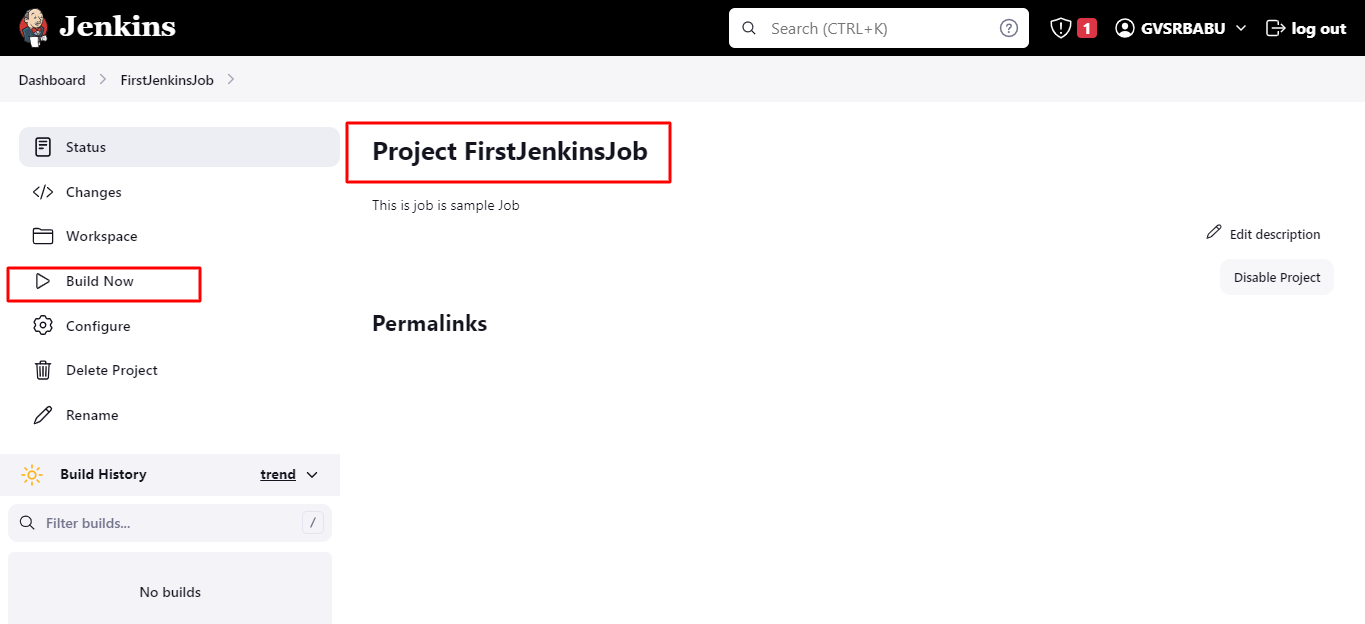
Select the freeStyle option and Click Ok button



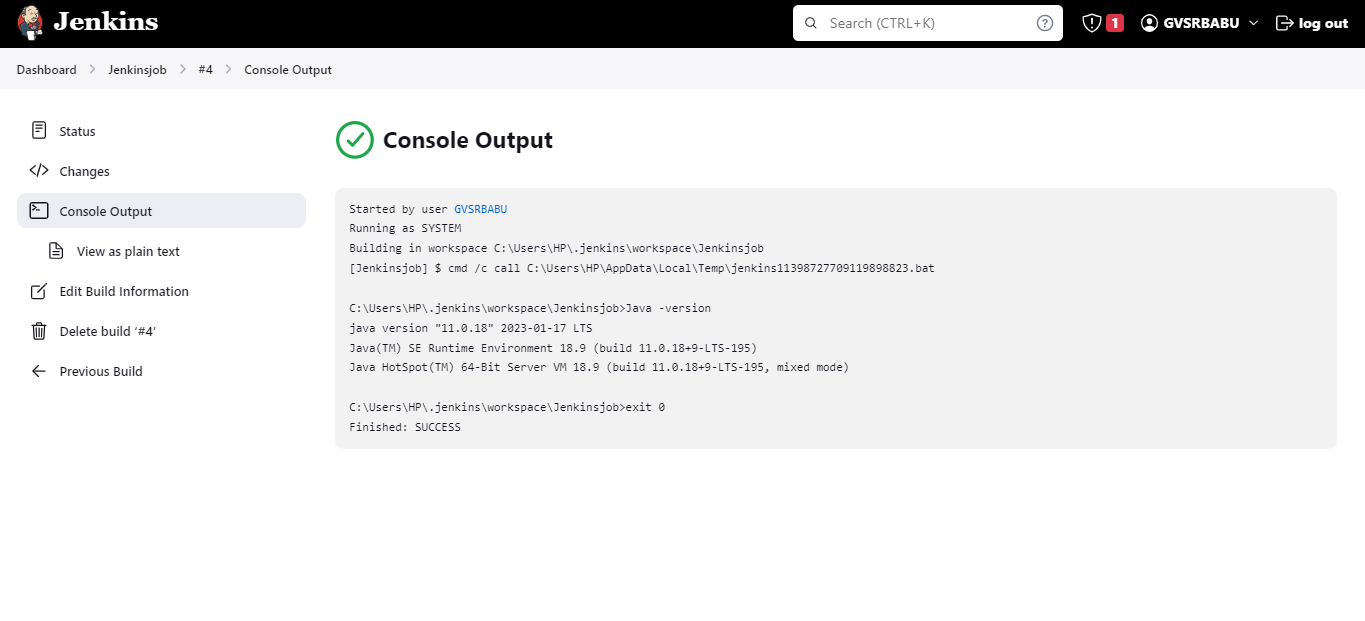
If u are working windows Select the below option in Build Steps



Now u are able to see the created job in Homepage and Build option right hand side. Click Build now option



After creation of the job user able to see the console

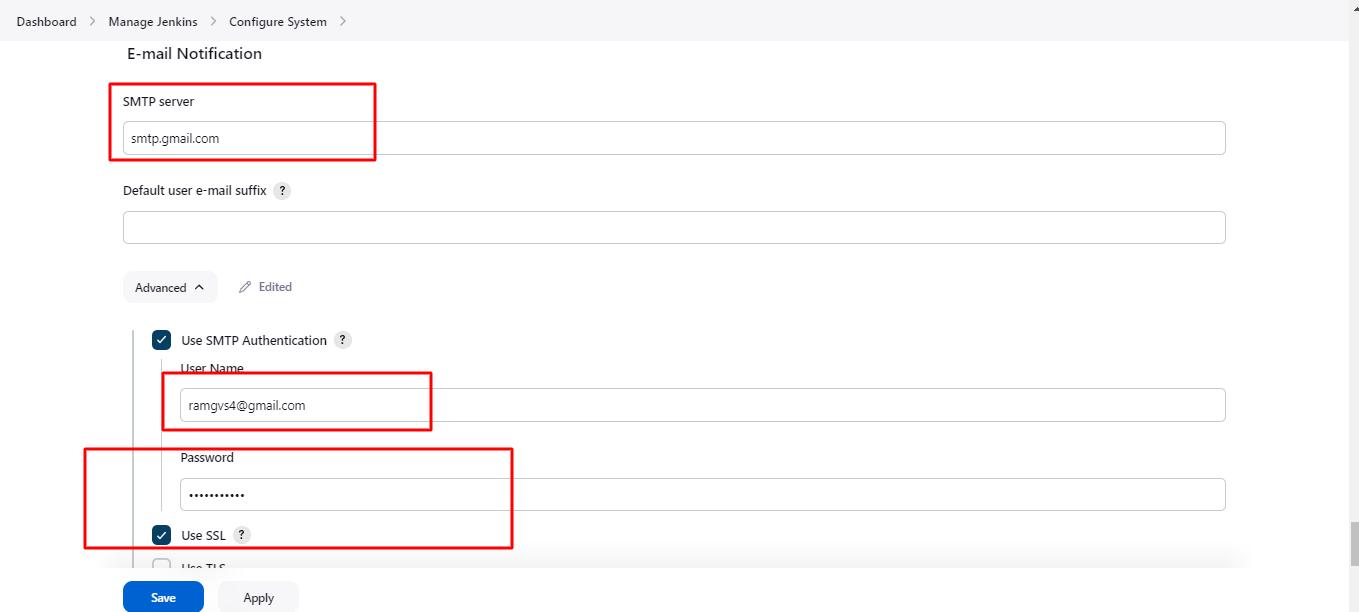


Email Notifications in Jenkins/How to send email from Jenkins Job

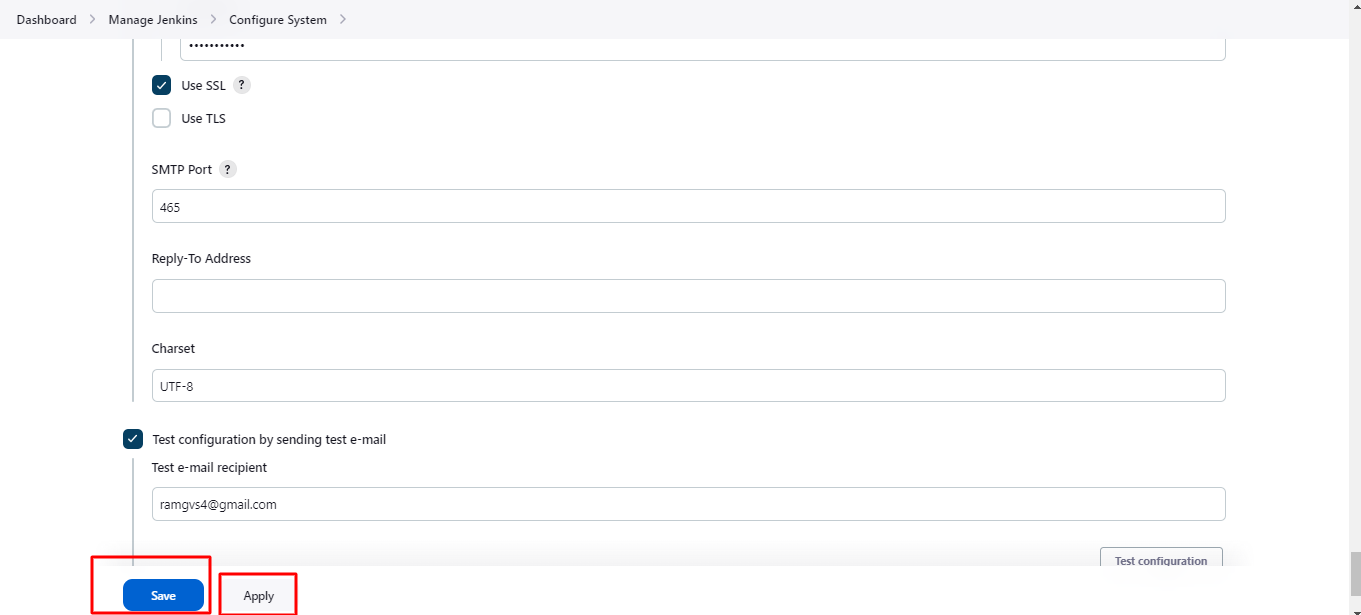
Home page🡪Go to Mange Jenkins🡪Configure system

Set the details in Email

Notification module

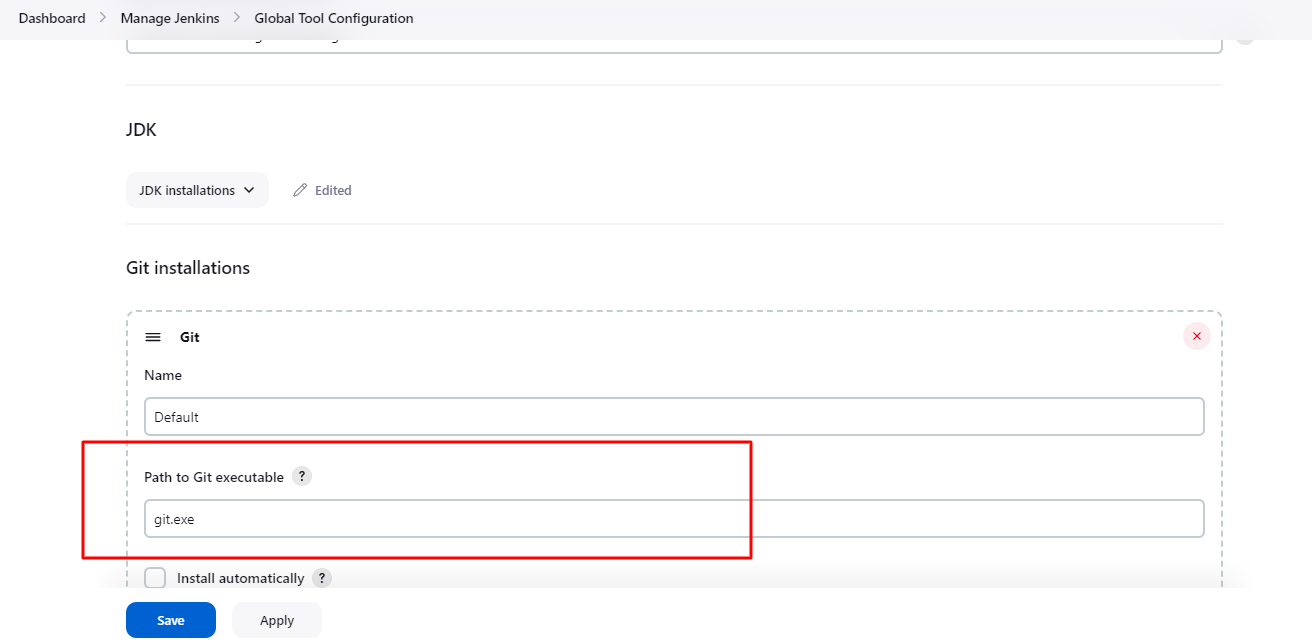


Click Apply and Save .if u are set once it saved permenantly

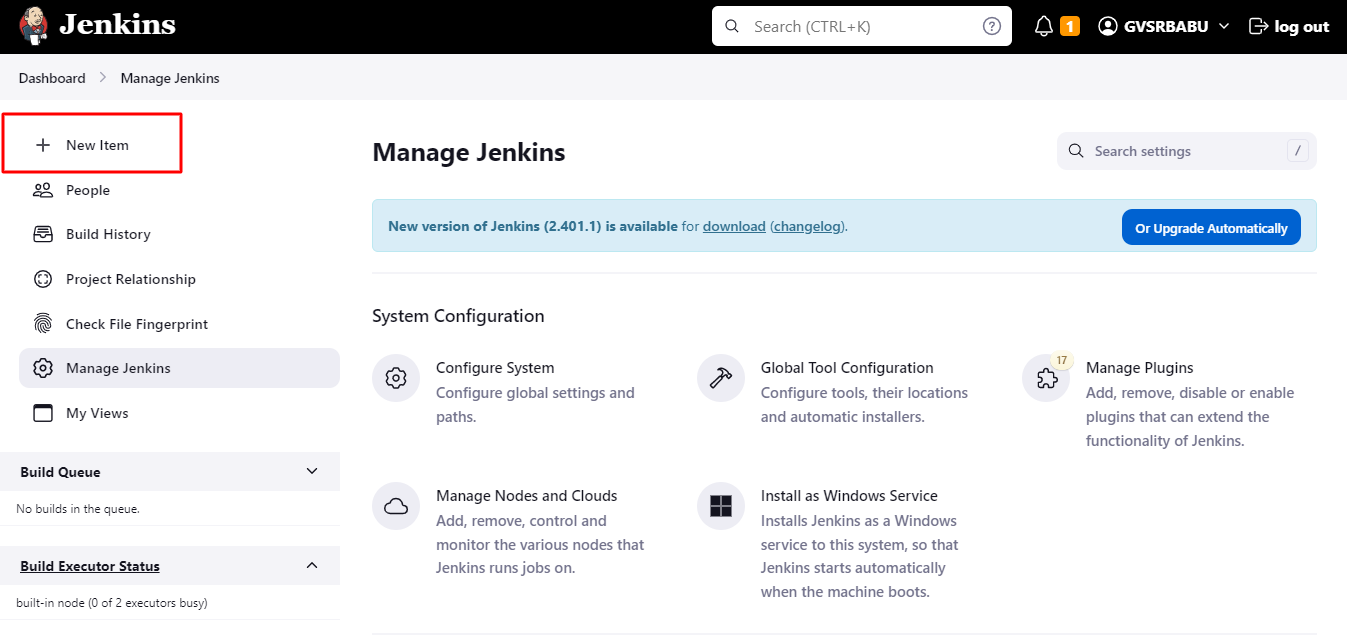


Maven Project Configure Jenkins with GITHUB

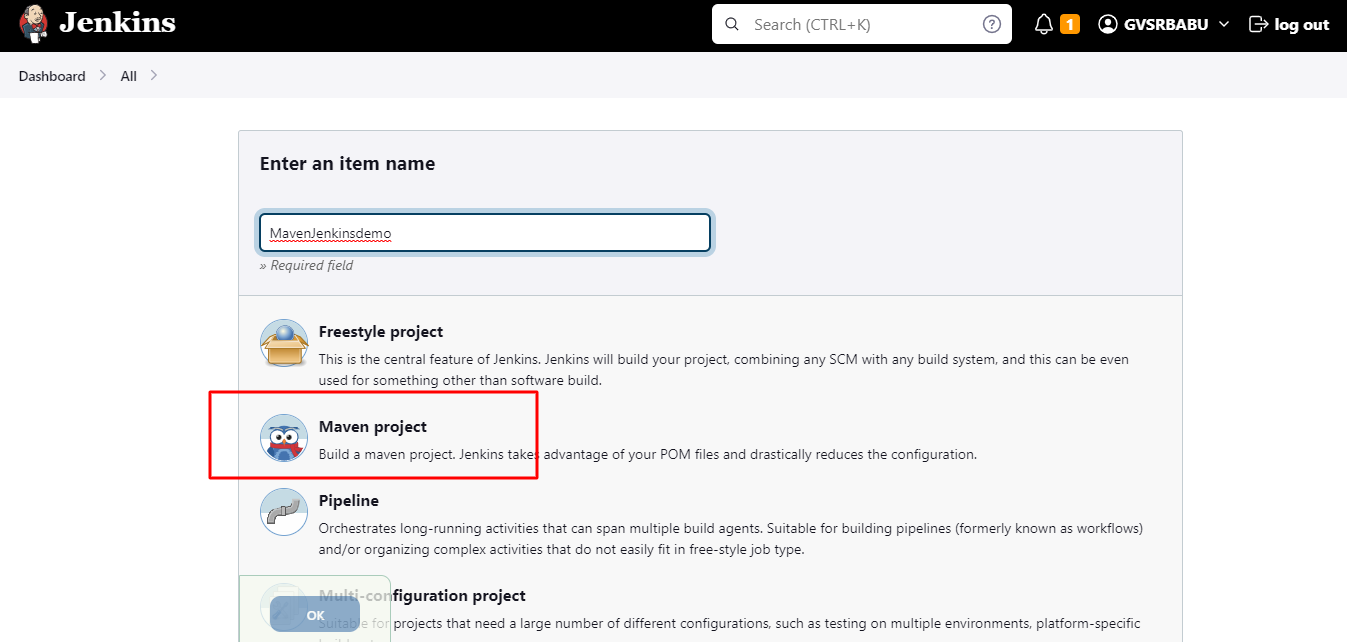
Follow the path:Jenkins Homepage🡪Manage Jenkins🡪Tools



Create a project: Click New Iteam

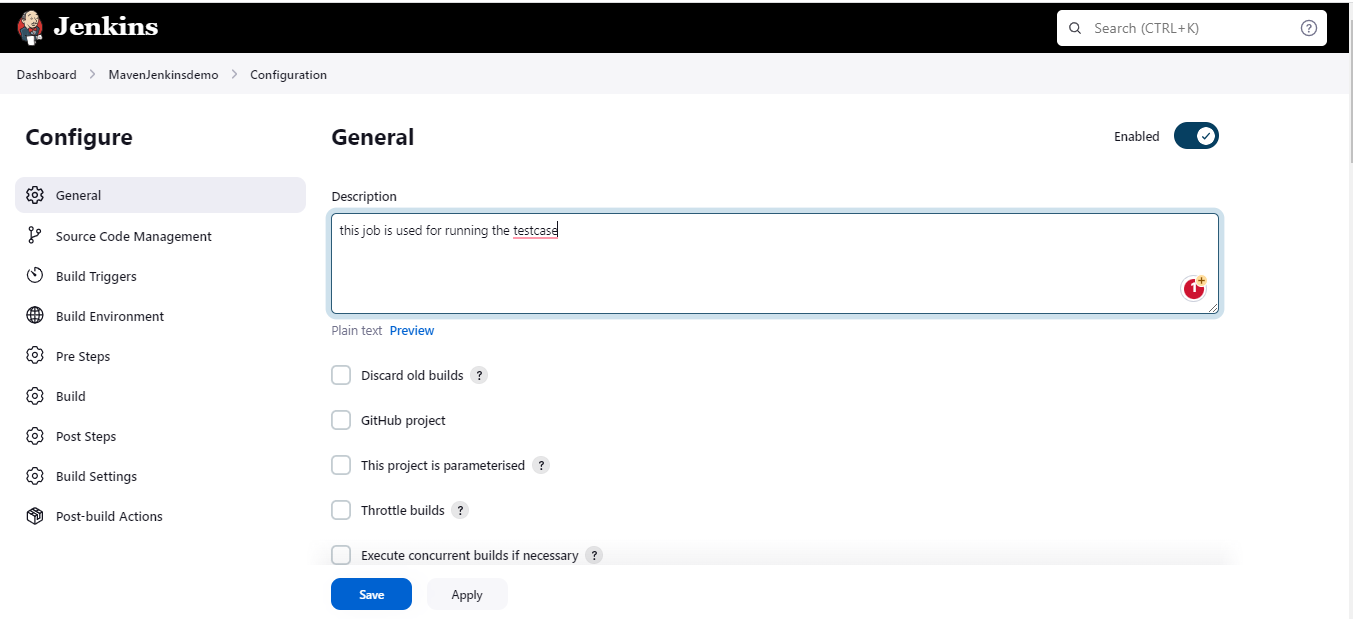


Selet maven project and Click Ok



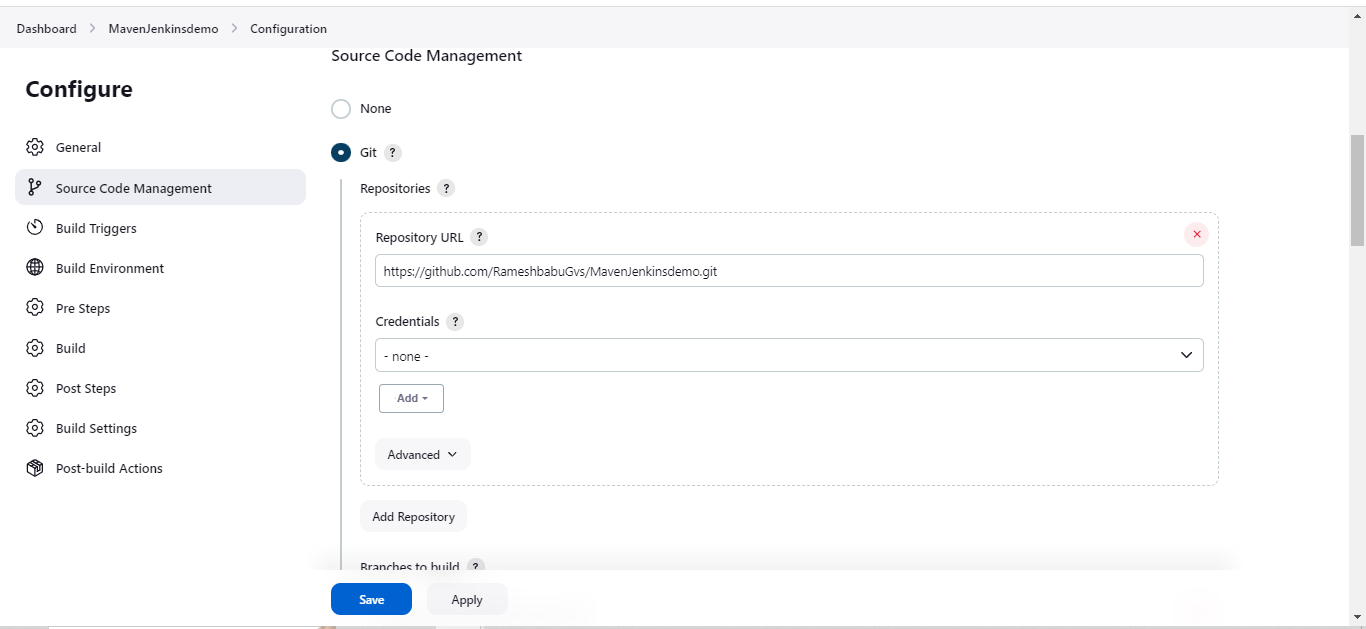
afte below page will be opend

Add descrption below text box

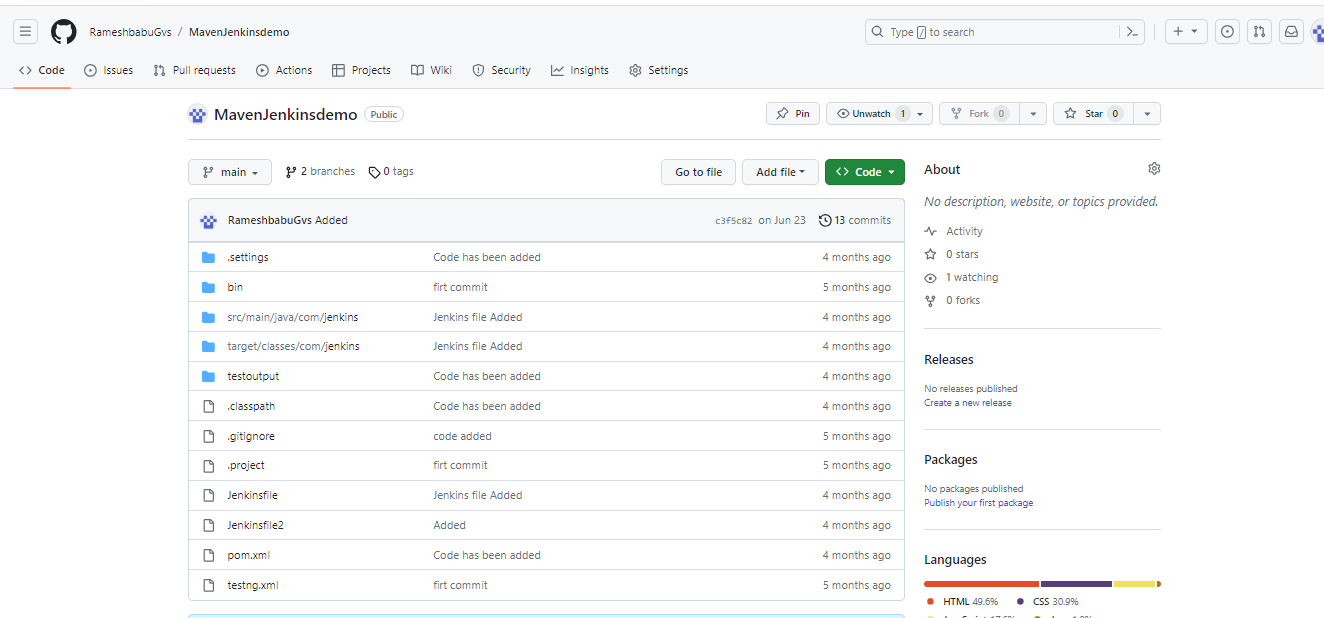


Select the source management (GIT)

And select Git Radio button and take same project github url from GITHUB and enter the same url click save button. If it is a public no need to enter the credentials.

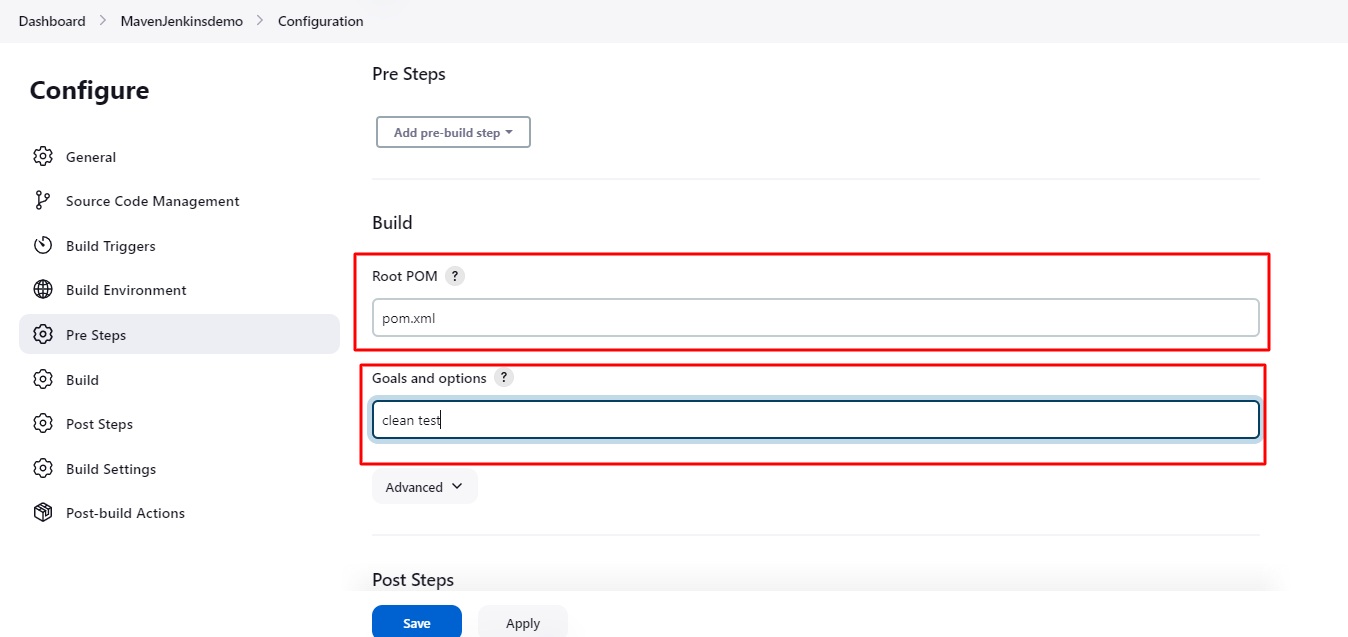


If u have multiple branches in git hub u need to mentioned the git branch name which branch where u are in

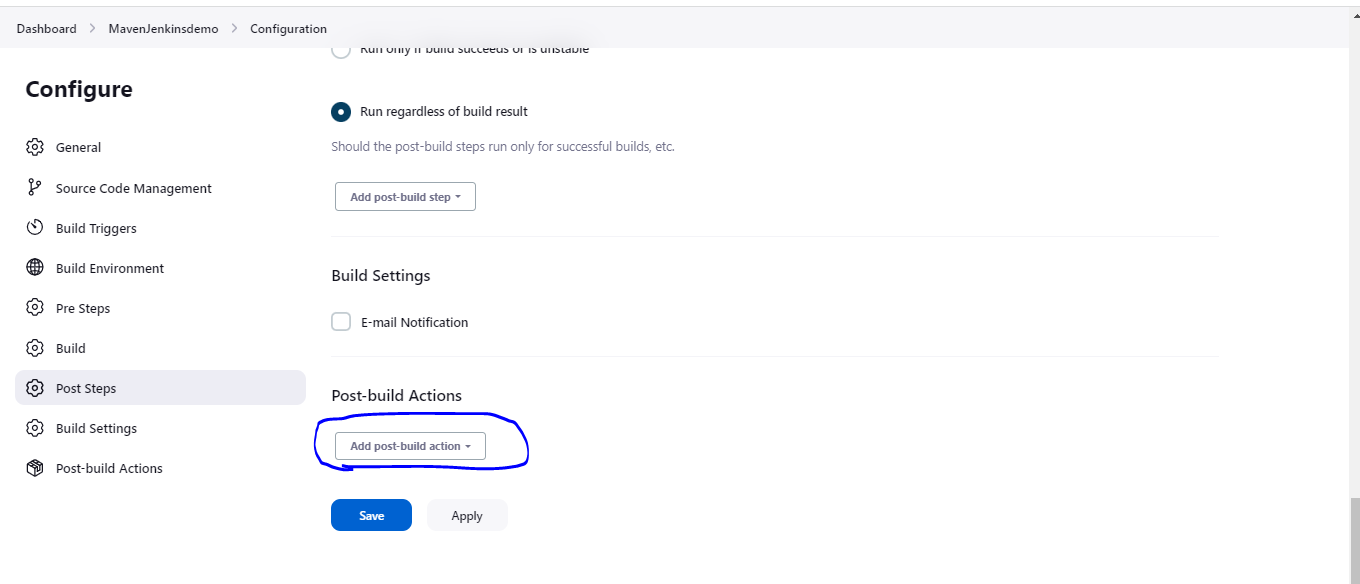


NO need to give the pom.xml file Jenkins takes file path automatically and download the code

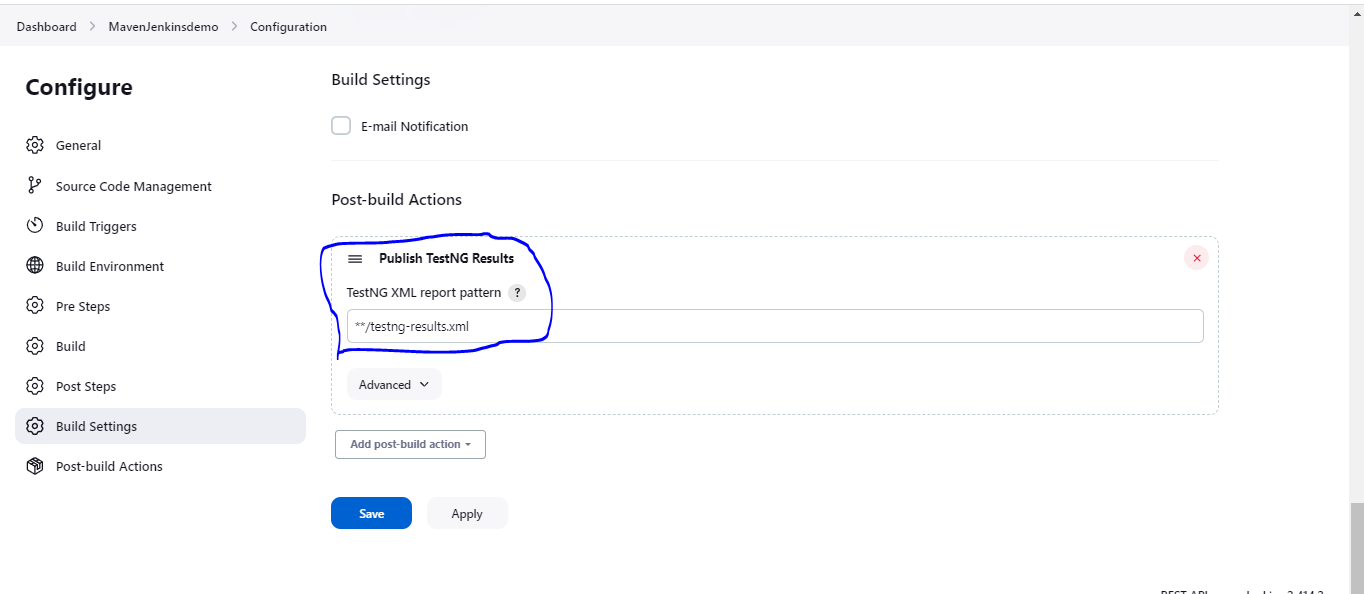
Enter commands below text box



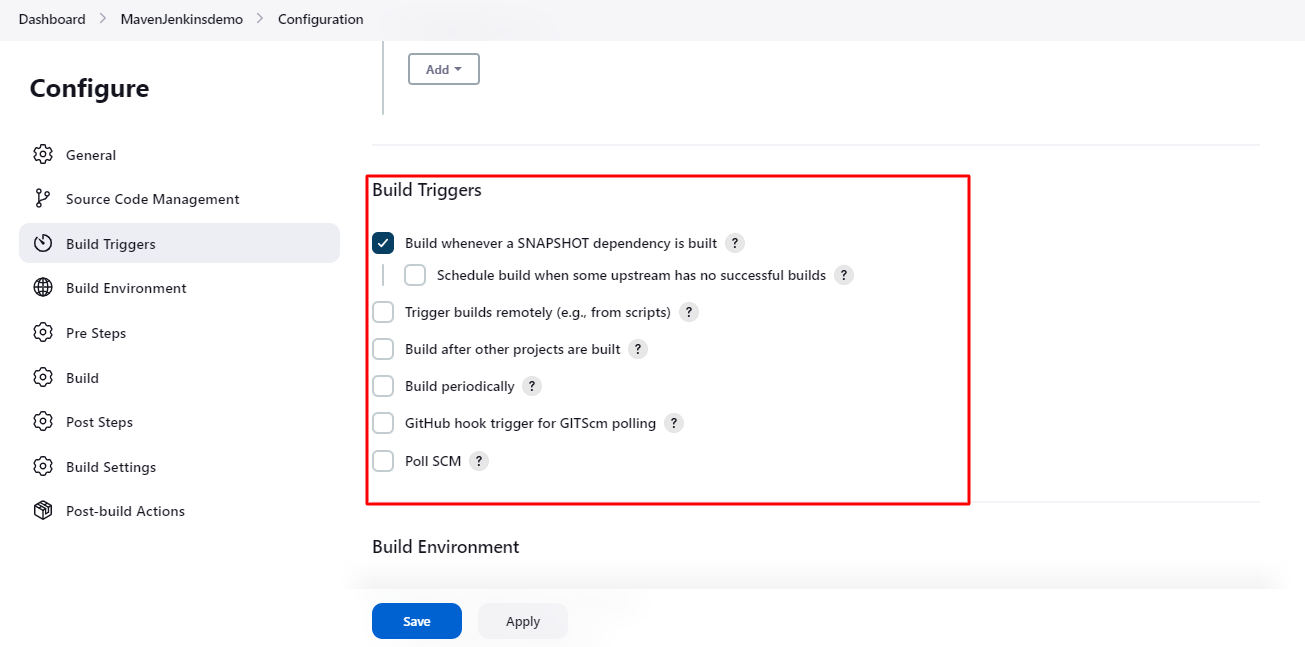
Go to Post-Build Notifications



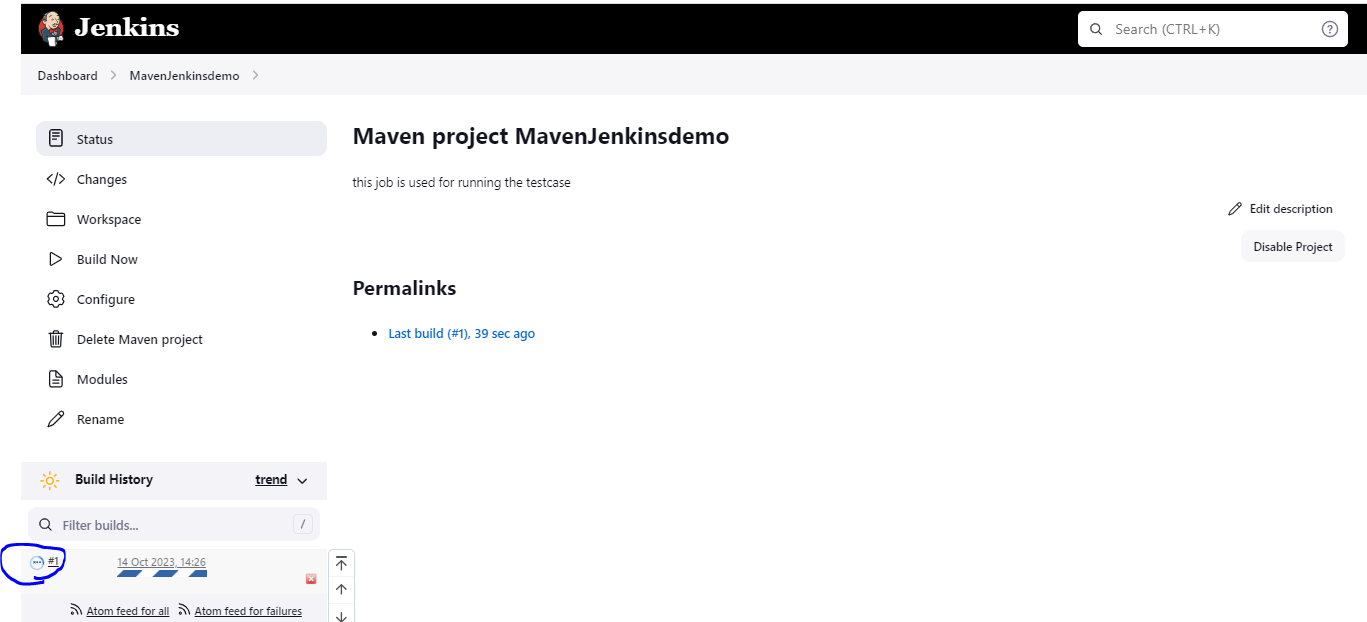
Select Publish TestNgResults



No need to change any thing in below part right now click Apply and Save

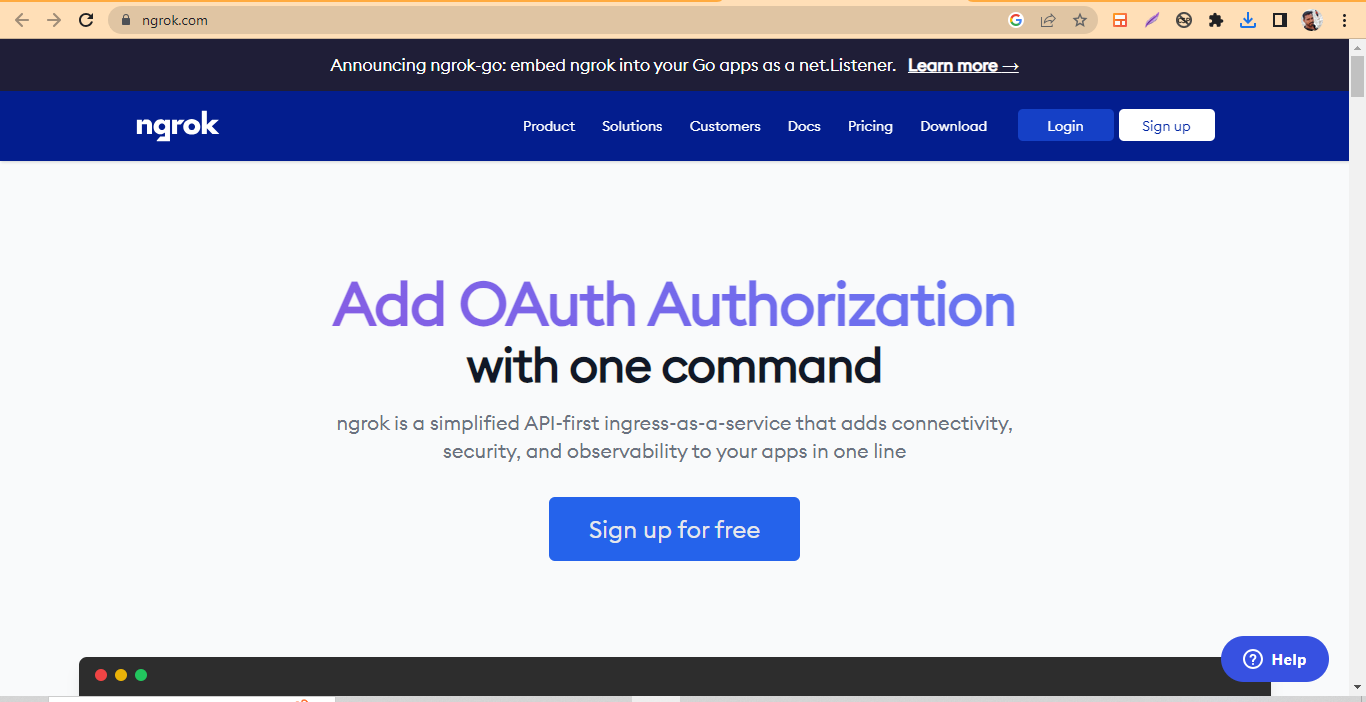


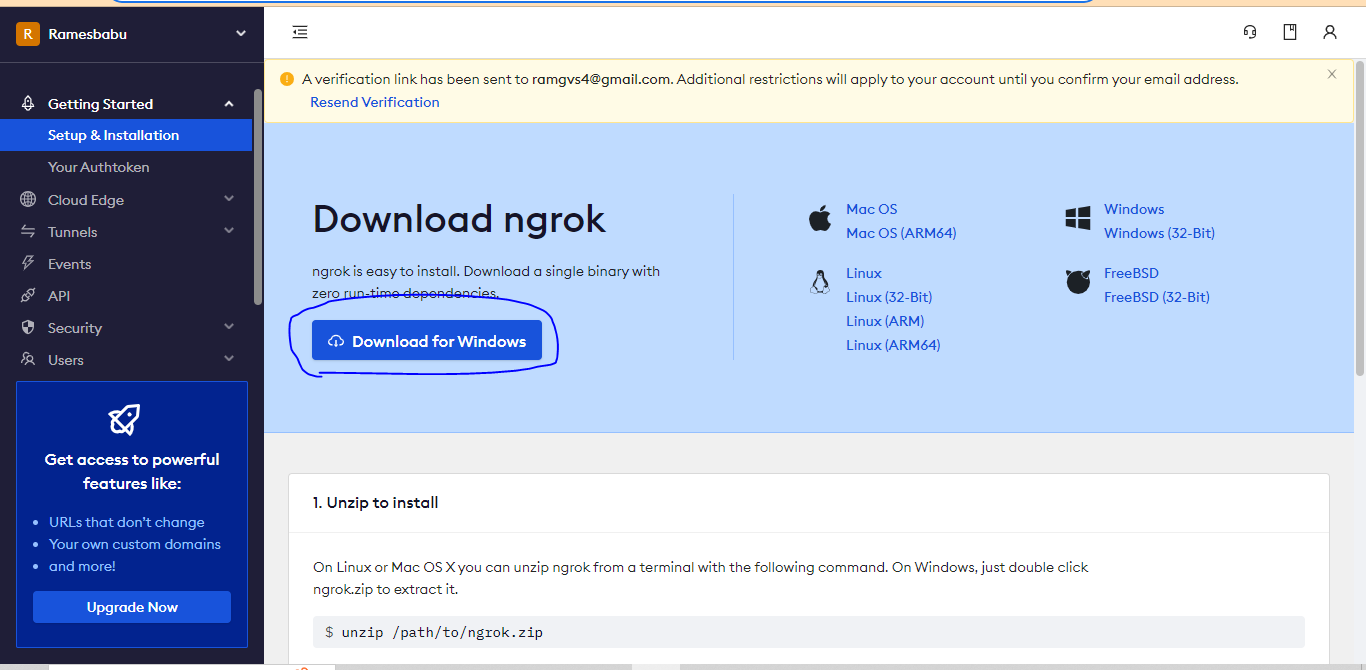
Click Build Now observe the Console output.Select rounded circle and observe console output



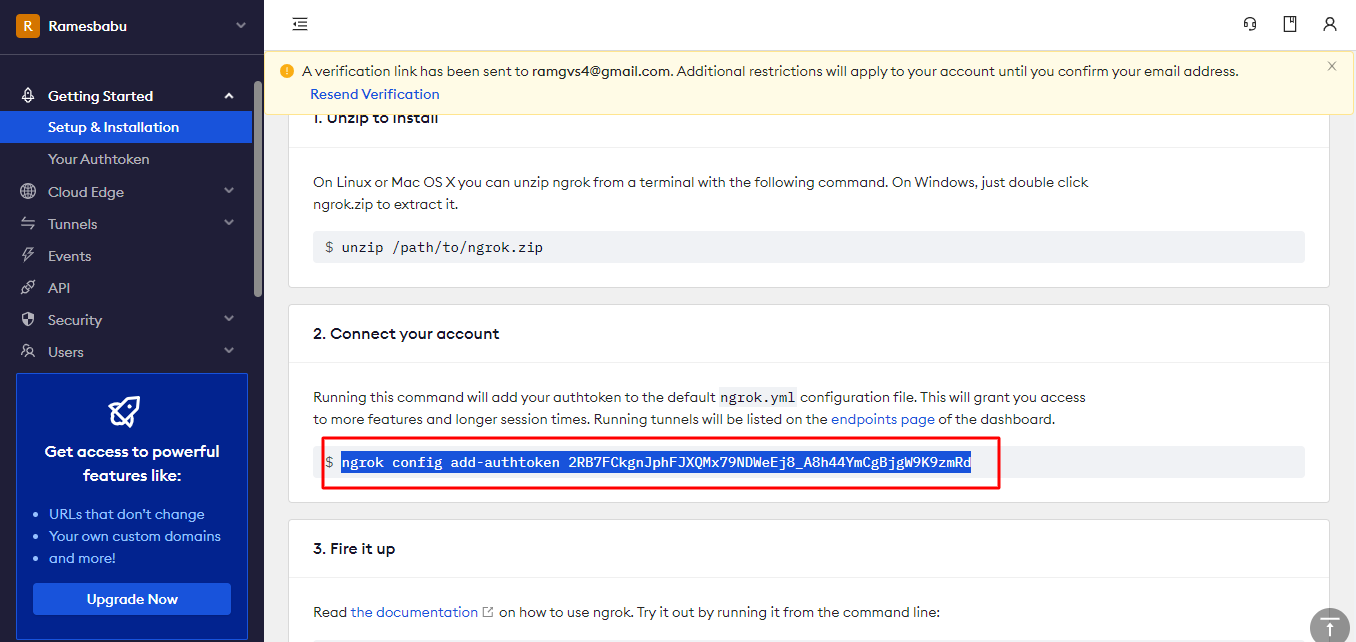
NGRockProxy

Open NGRock website and create account by using below url --https://ngrok.com/

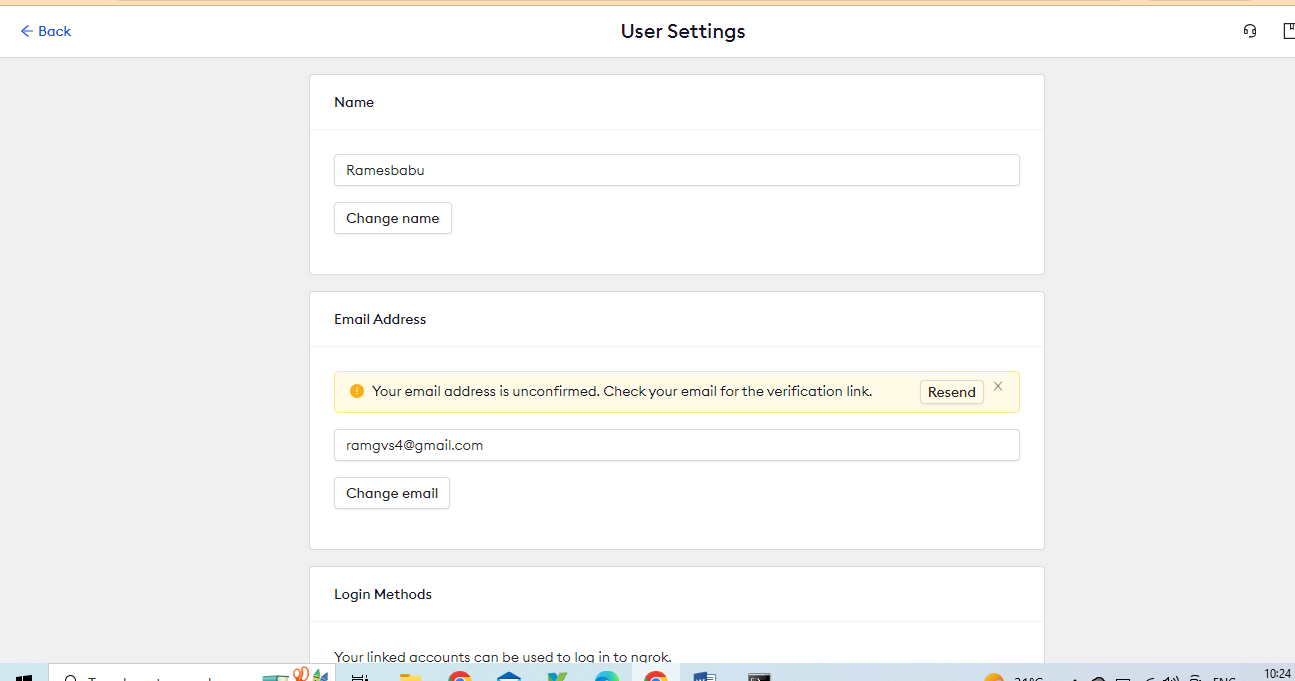


Download windows supported file and un zip 

Unzip the folder the open the cmd and goto NgRock account copy the Point NO:2 Auth token and paste in cmd it will generates the Token and saves in certain location

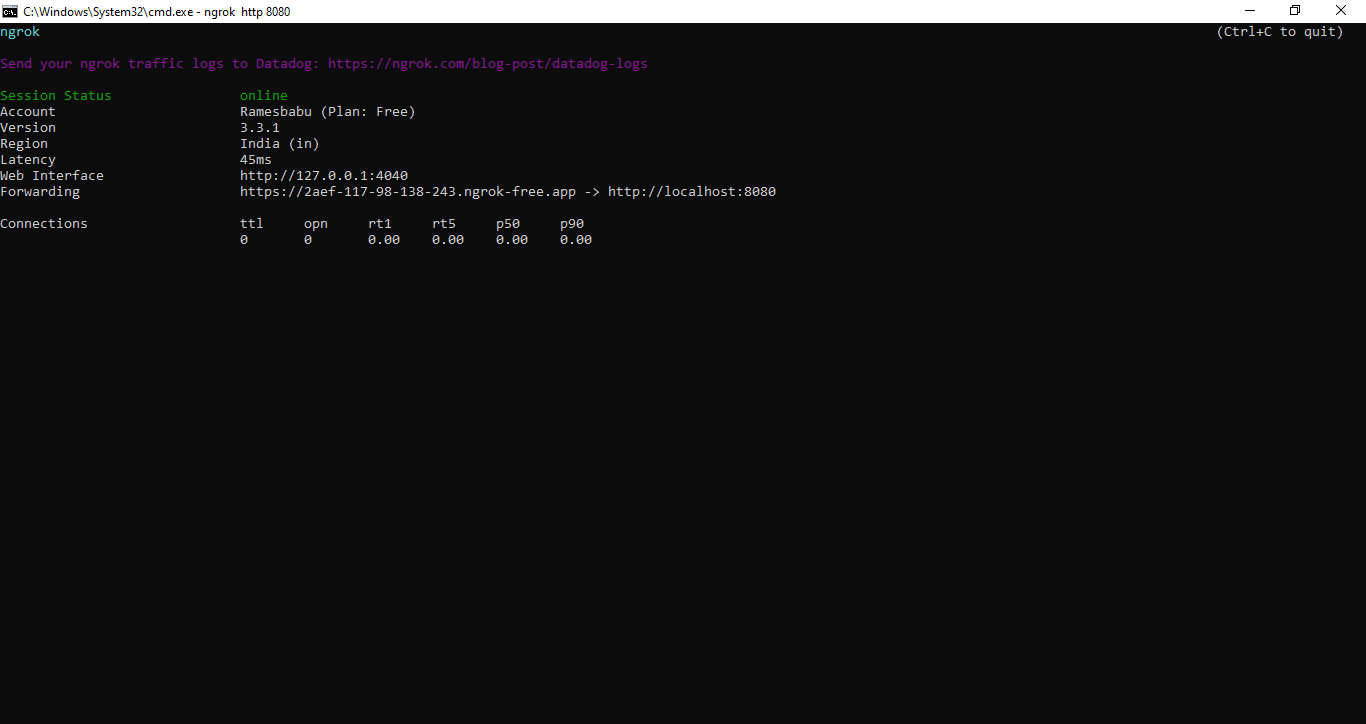


Uneed verify with your email



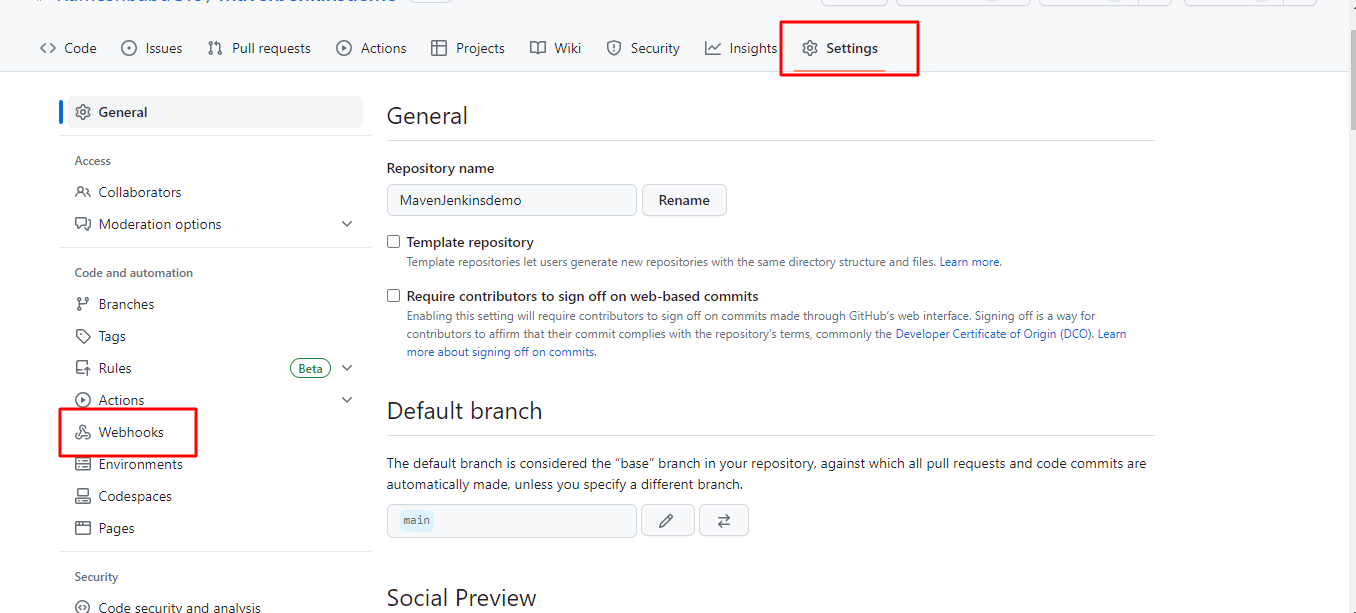
Next u enters below command for ngrok http 8080 for connecting u can able to see below image for successful connected.

Below imge shows ngrock proxy is successfully connected with localhost(Jenkins)



Come to GITHUB Account open the WebHook

WebHook🡪openRepositry🡪Settings🡪Webhook

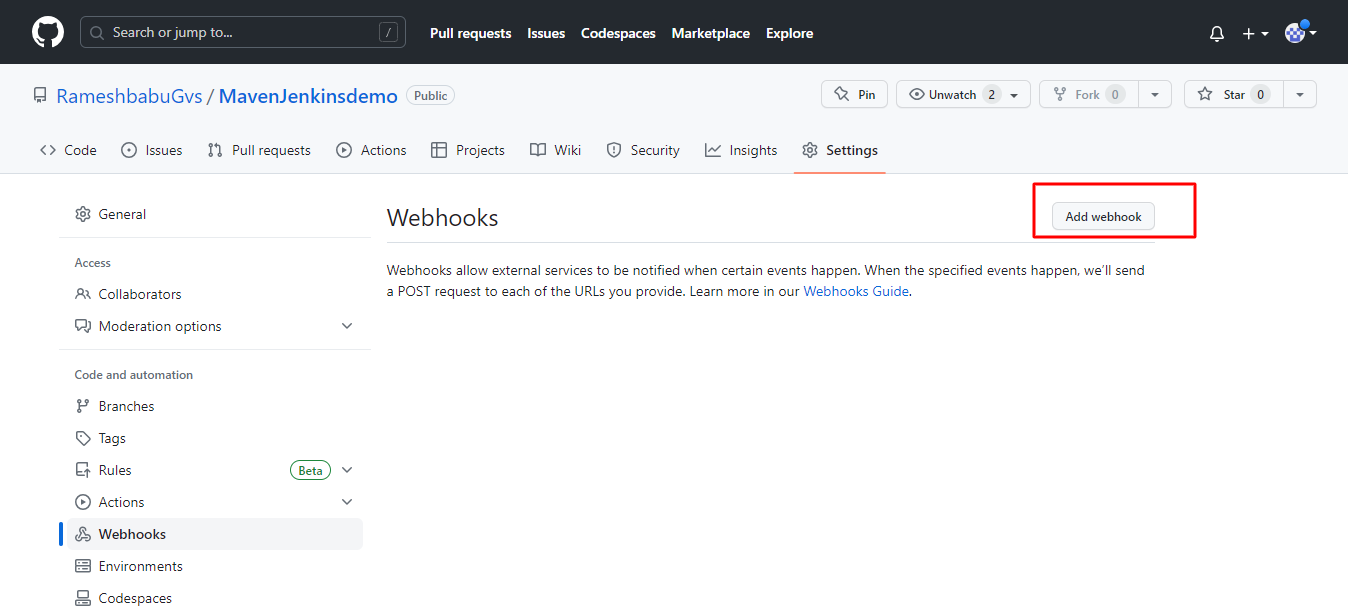


## Webhooks

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we’ll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](https://docs.github.com/webhooks/" \t "_blank).

You can add Payload url in github Webhook

Click Add Webhook

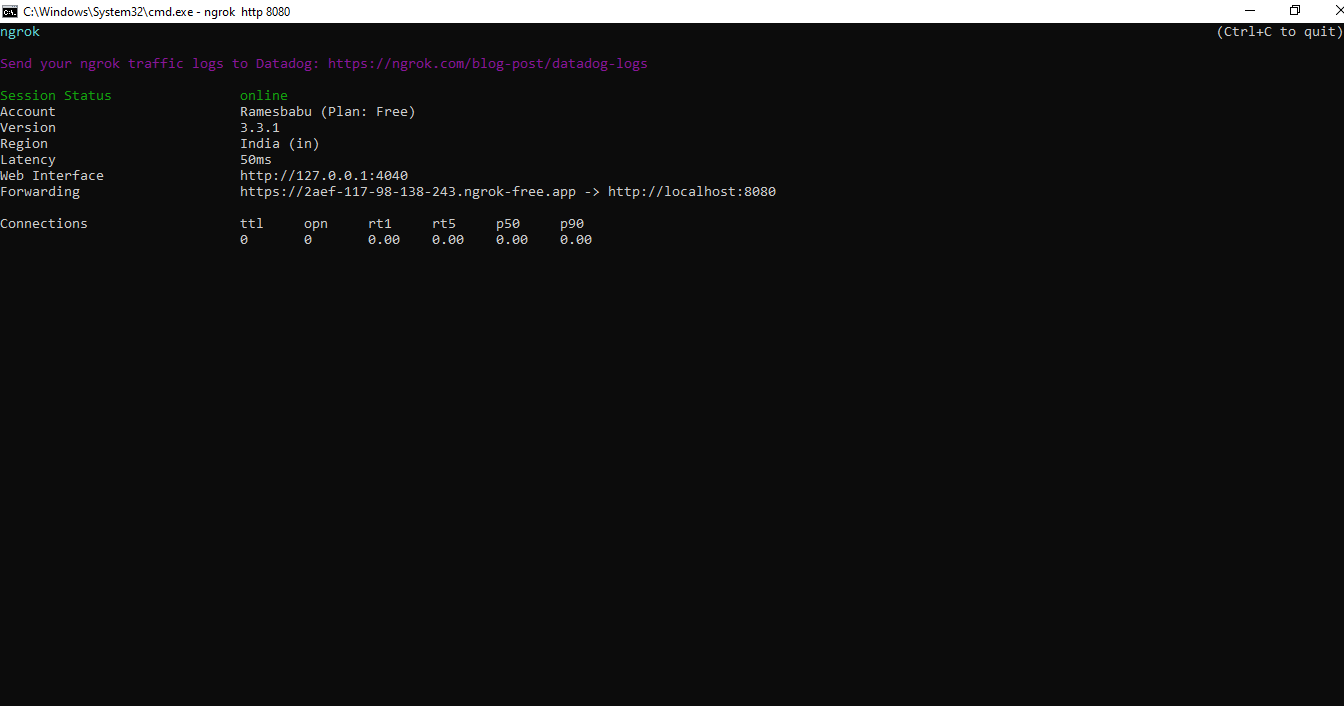


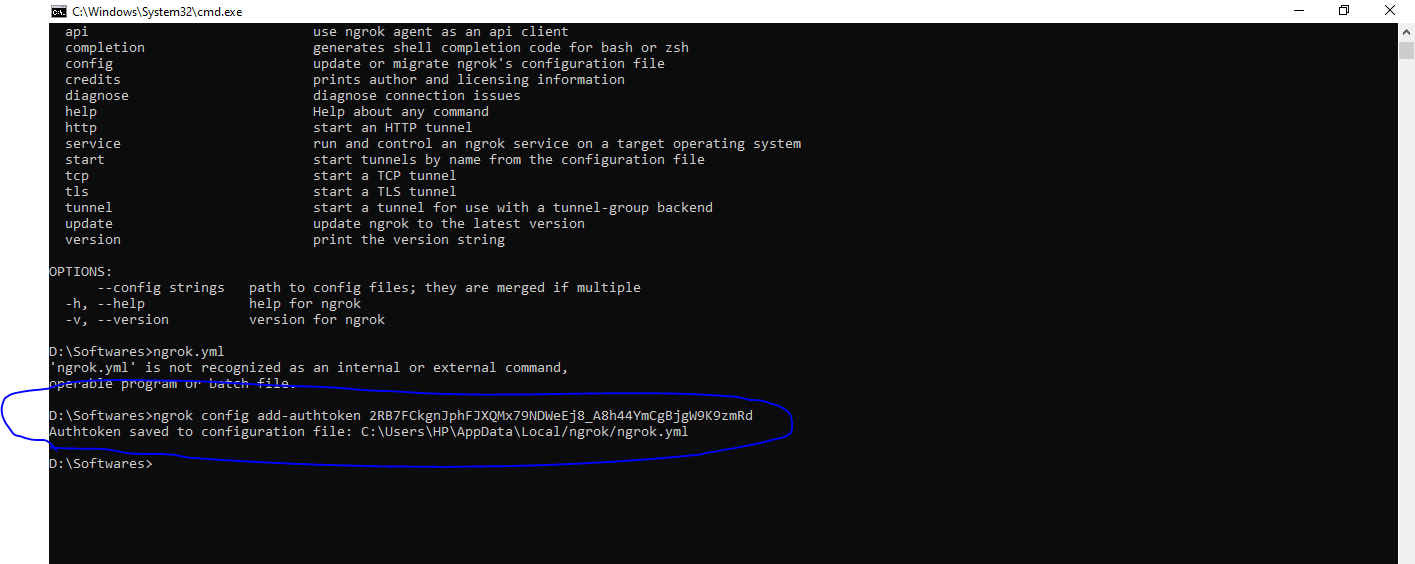
Payload url is nothing but NgrockProxy url

Take the https url🡪 <https://2aef-117-98-138-243.ngrok-free.app>

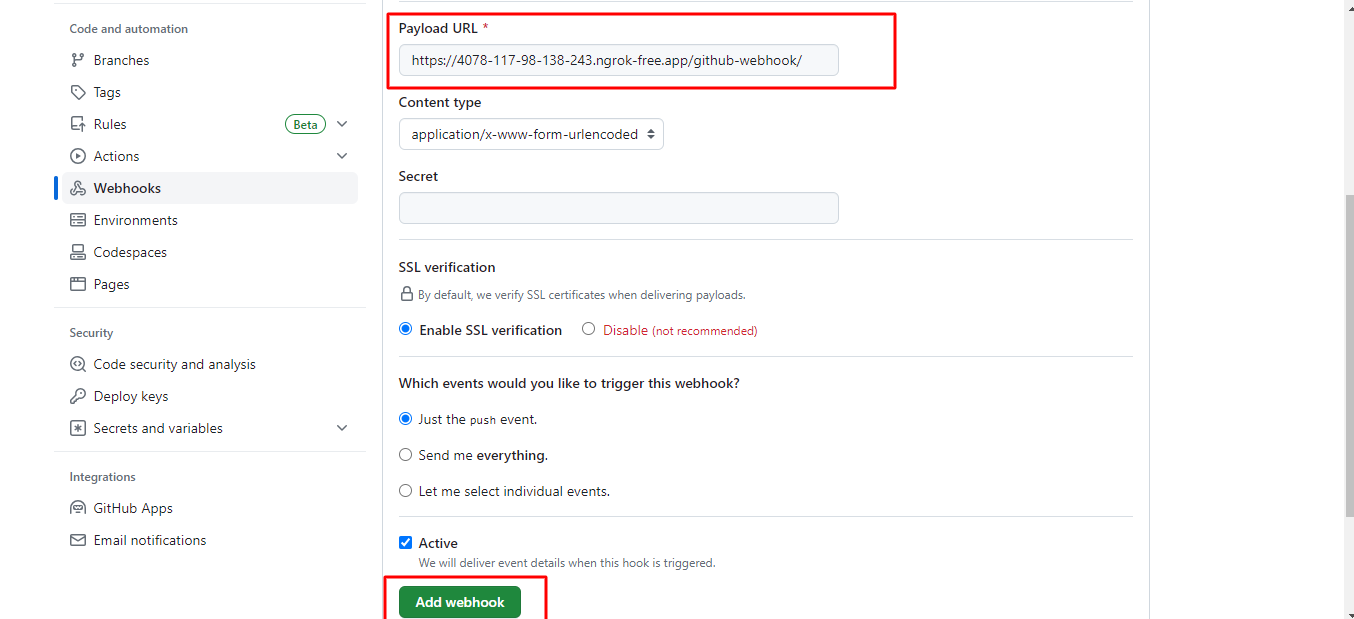
Add /github-webhook/

Like this 🡪https://e377-117-98-138-243.ngrok-free.app/github-webhook/

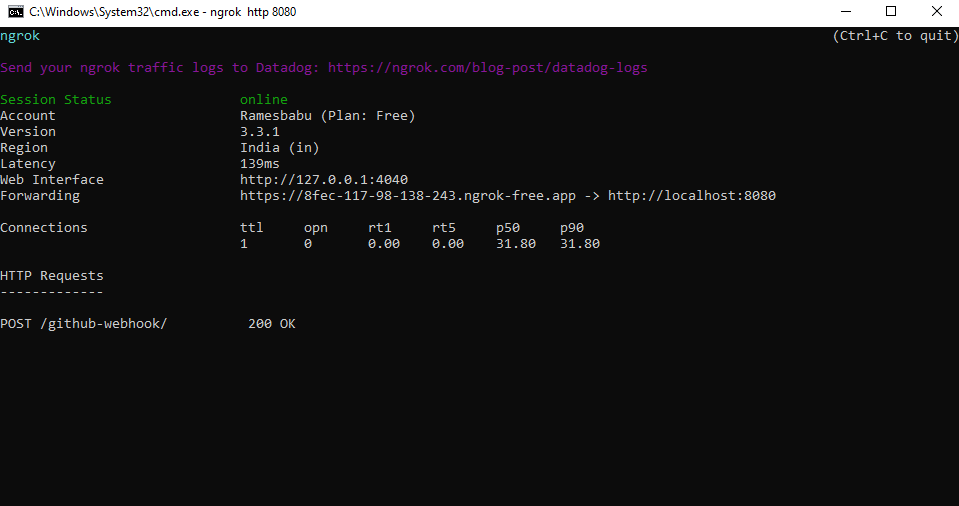


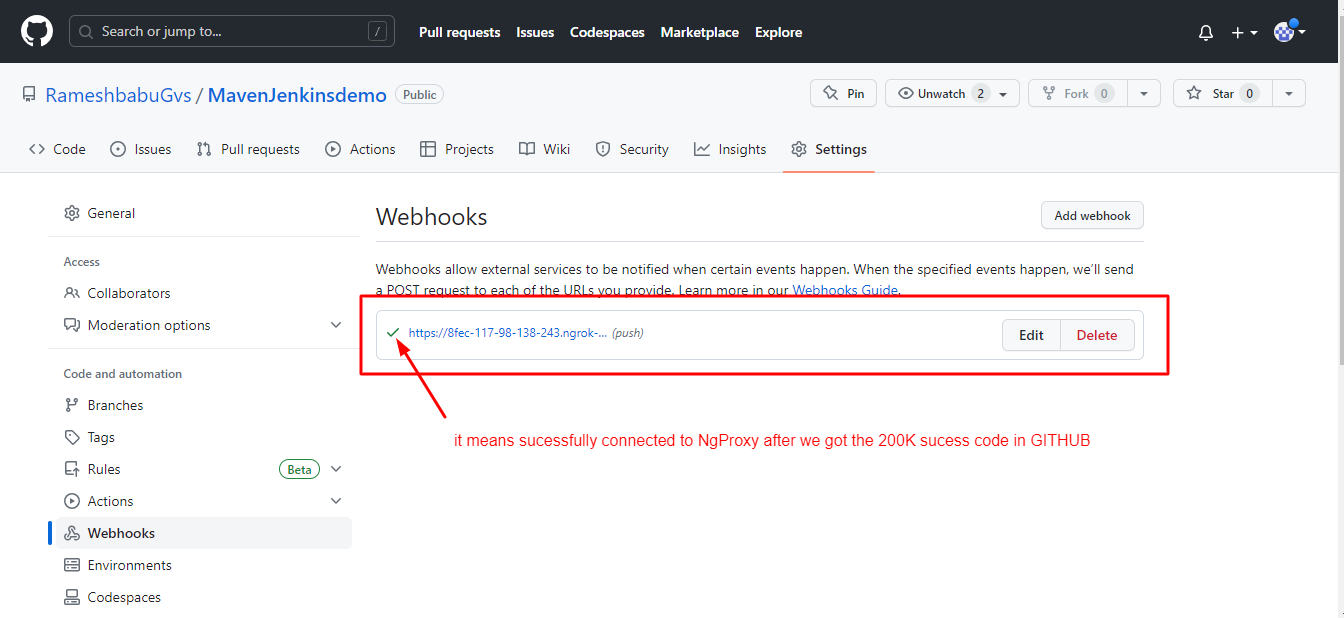


U can see below image



After you get the 200k Success message in CMD





Jenkins for Automatic triggering follow below steps

Open Jenkins job🡪click configure🡪click **GitHub hook trigger for GITScm polling** check box

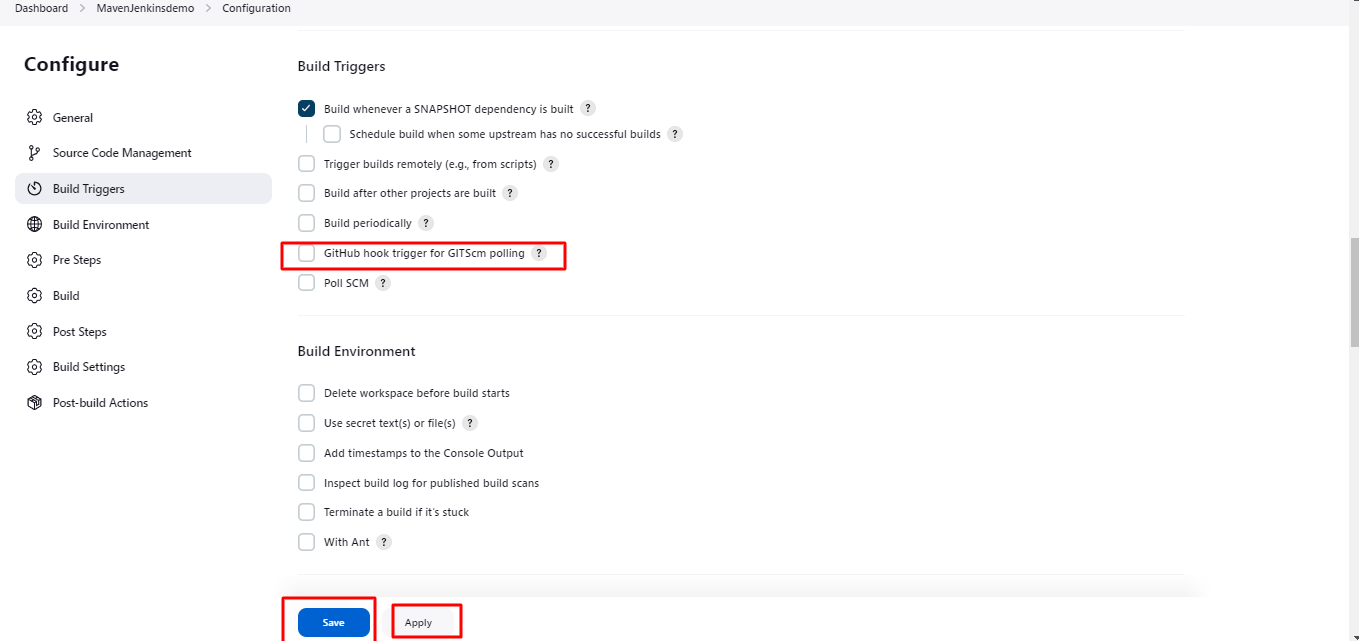
Apply and Save.



If u want to stop the Jenkins for Automatic triggering follow below steps

Open Jenkins job🡪click configure🡪Uncheck the **GitHub hook trigger for GITScm polling** check box

Apply and Save.



**If u want schedule the jobs and running schedule time follow below steps**

Open Jenkins job🡪click configure🡪check the **PollScm** check box

And enter the time period Apply and Save.

**If you want to schedule your build every 5 minutes, this will do the job : \*/5 \* \* \* \***

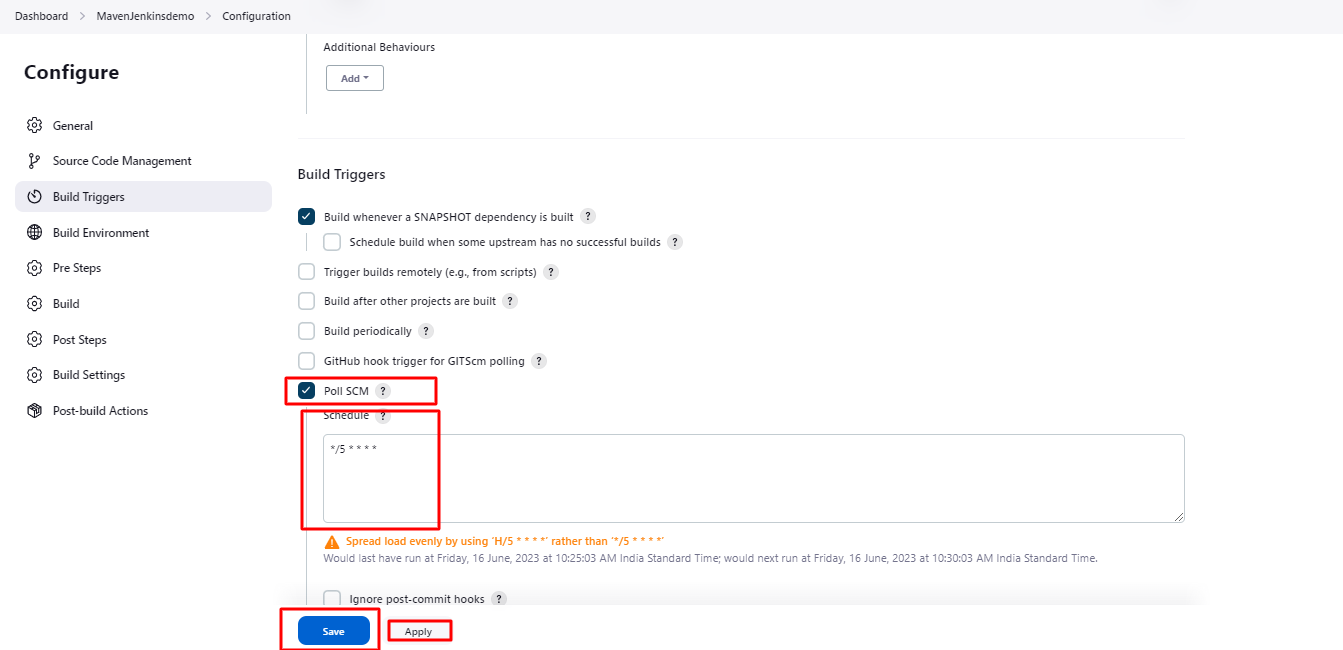
**If you want to schedule your build every day Hours u can use 24 hours format**

**If u want daily at 8h00, this will do the job : 0 8 \* \* \***

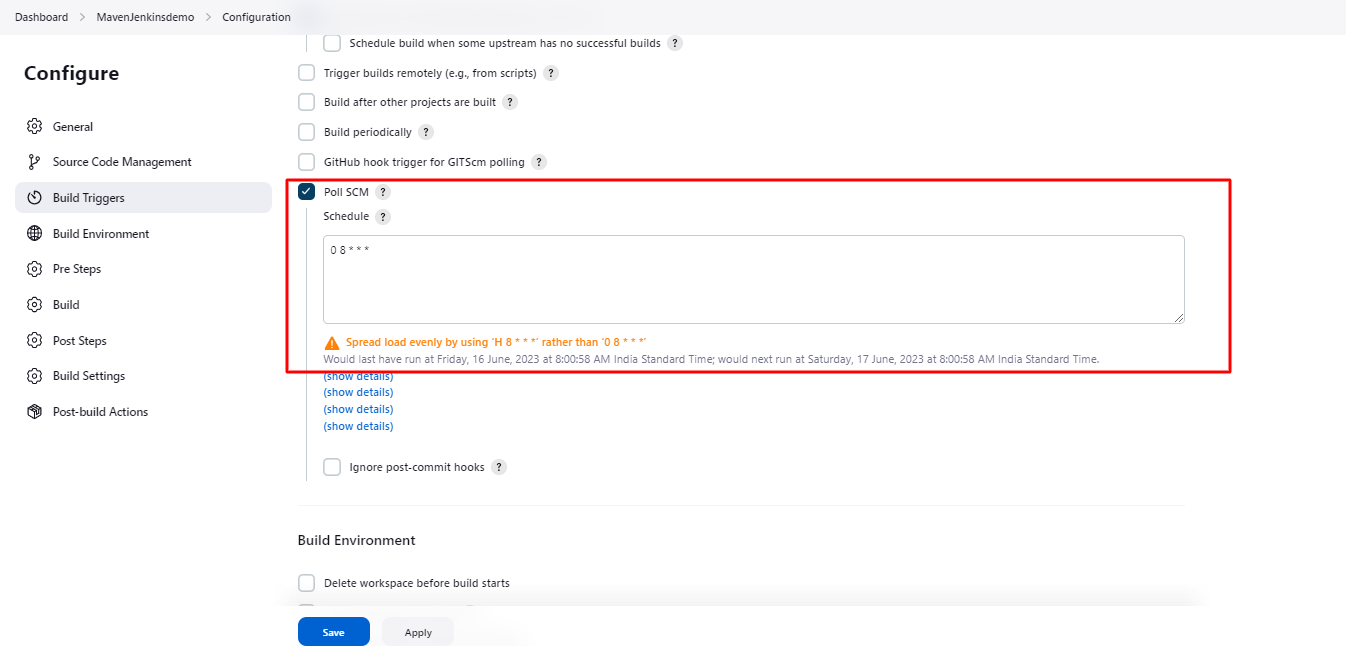
By setting the schedule period to 15 13 \* \* \* you tell Jenkins to schedule the build every day of every month of every year at the 15th minute of the 13th hour of the day.

Jenkins used a [cron expression](https://en.wikipedia.org/wiki/Cron" \l "CRON_expression), and the different fields are:

1. MINUTES Minutes in one hour (0-59)
2. HOURS Hours in one day (0-23)
3. DAYMONTH Day in a month (1-31)
4. MONTH Month in a year (1-12)
5. DAYWEEK Day of the week (0-7) where 0 and 7 are sunday



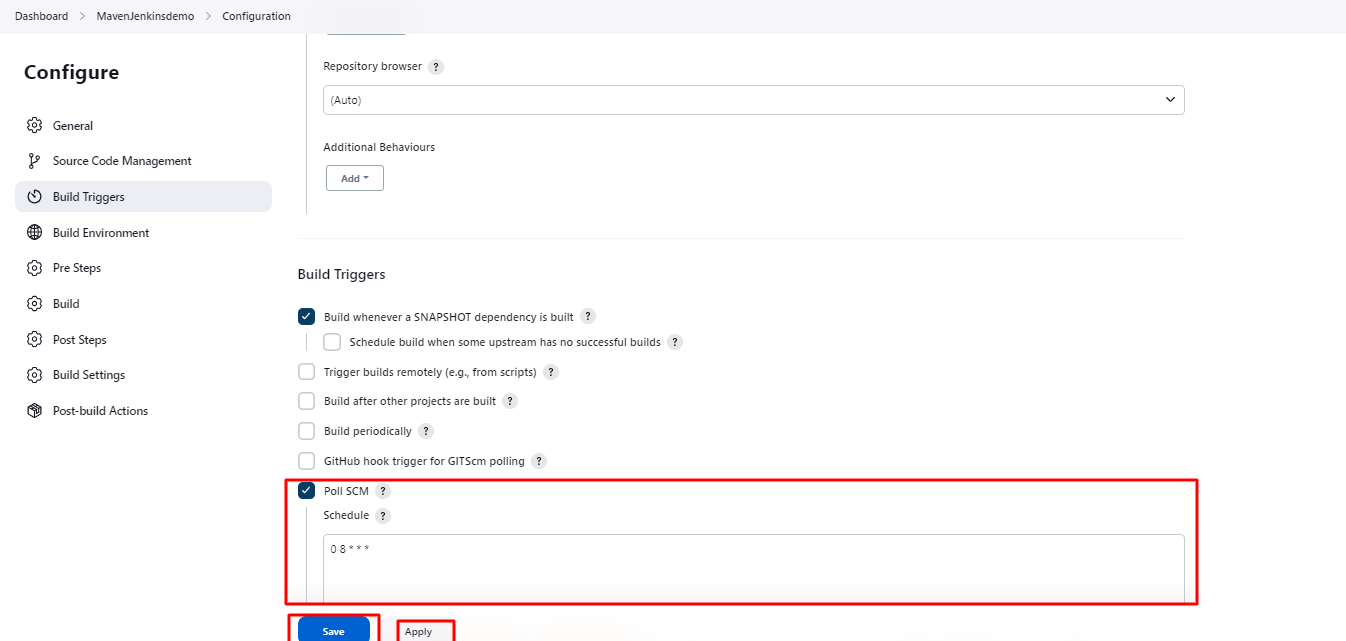
**If u want daily at 8h00, this will do the job : 0 8 \* \* \***



**If u want to stop the scheduling the running jobs in follow below steps**

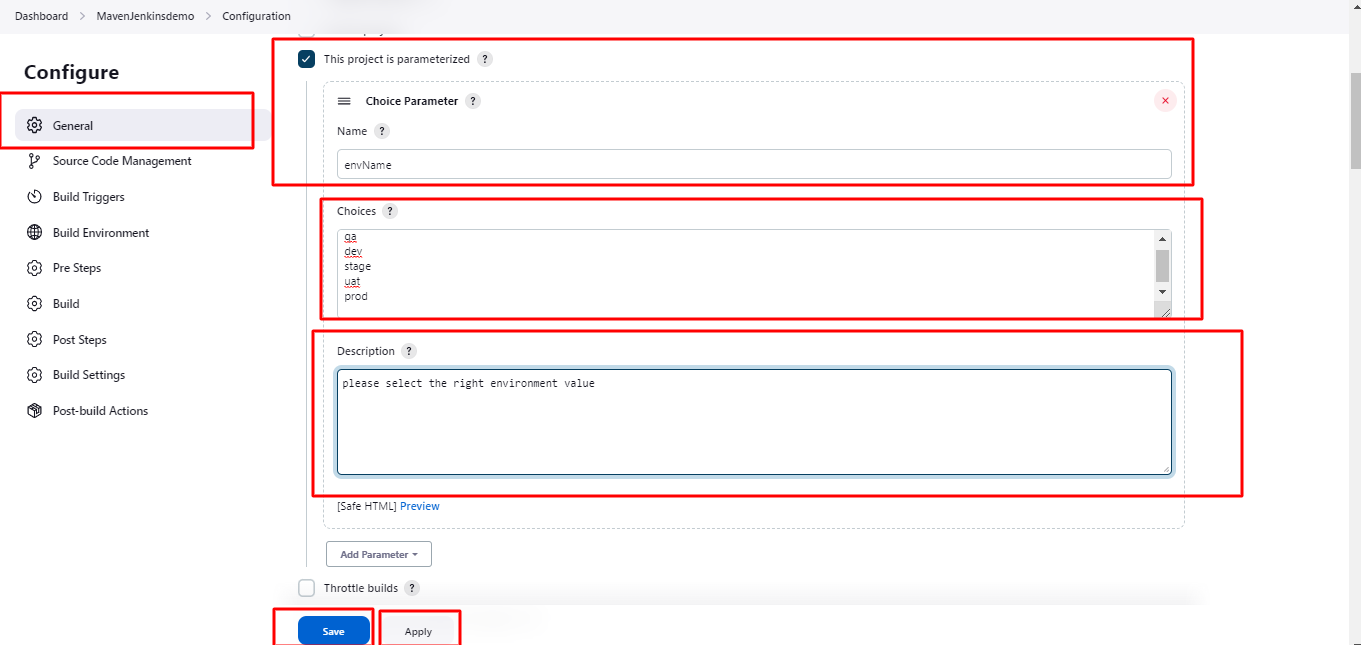
Open Jenkins job🡪click configure🡪Uncheck the **PollScm** check box

And enter the time period Apply and Save.



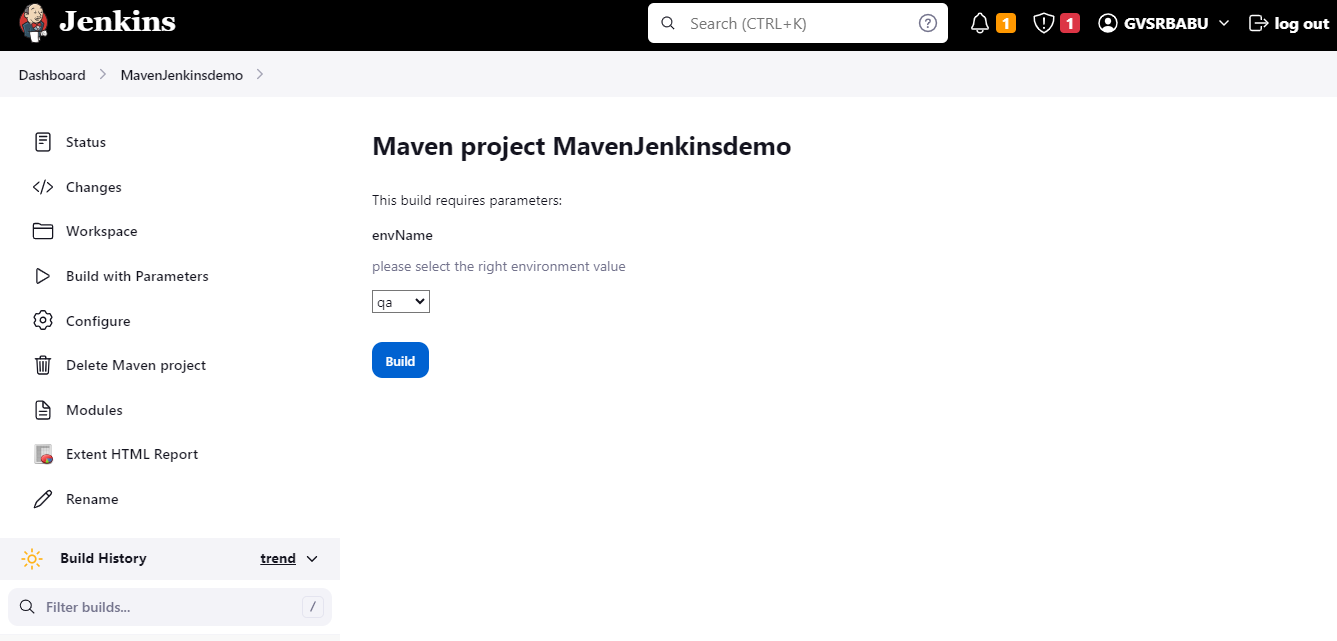
If u want set the parameter wise like:Environment,browser

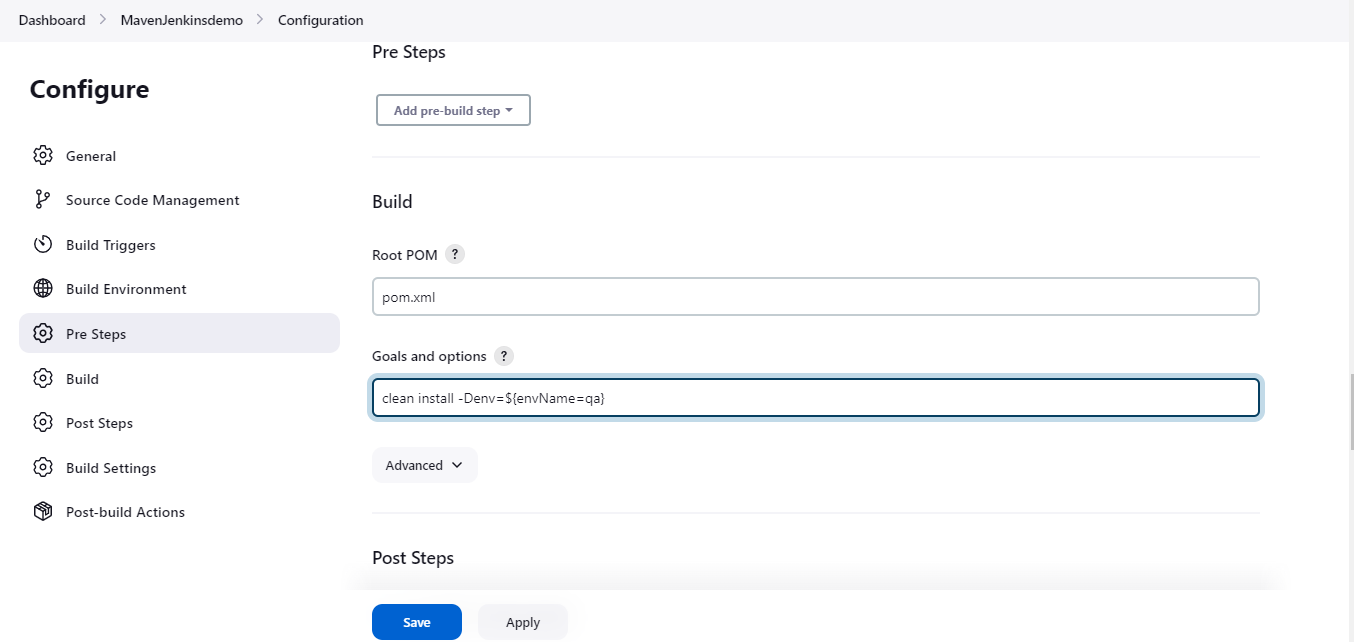
First u need to install the First choice parameter plugin



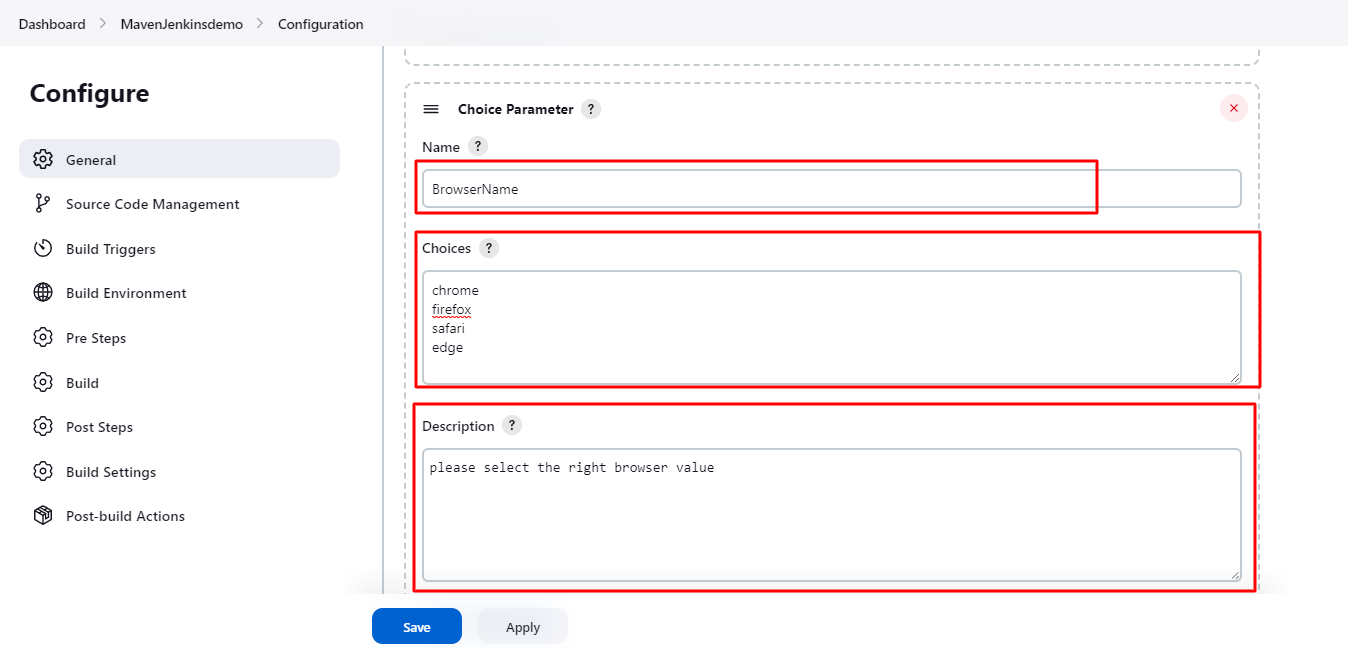
Enter envName and click Apply and Save.

Now u need to select the right env goto Build with Parameters 🡪select the dropdown





Select the right browser

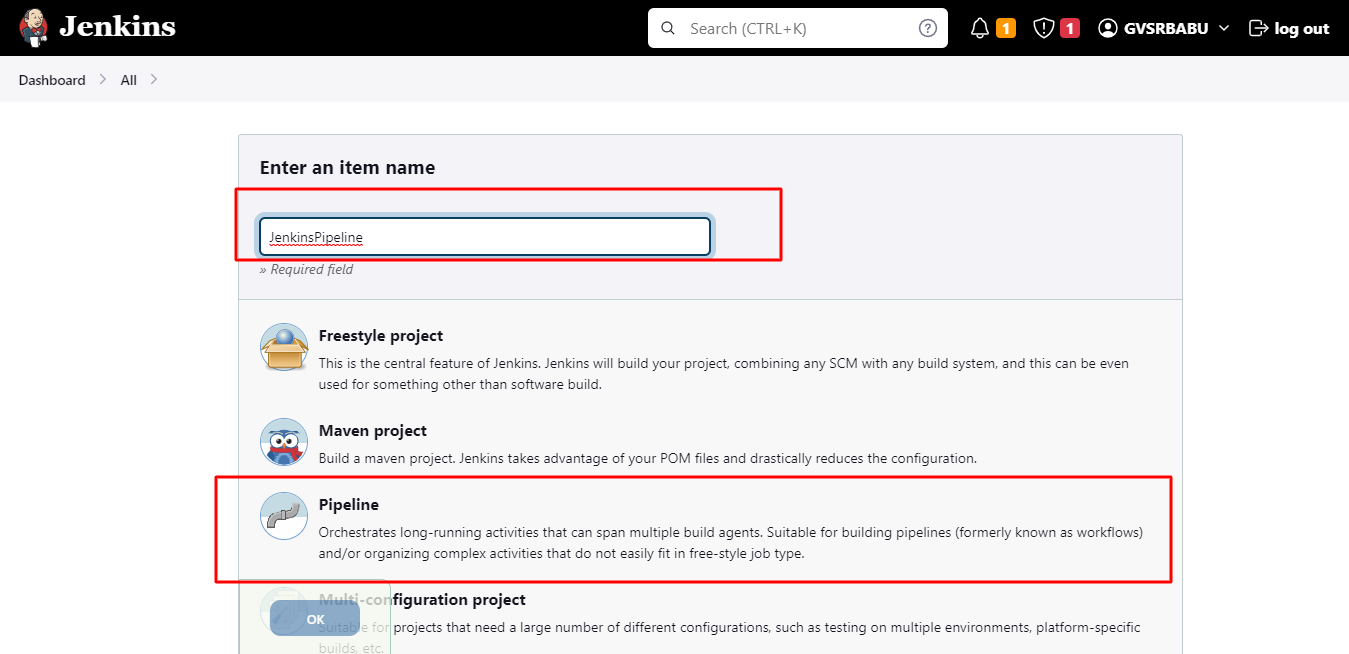


**Pipelines In Jenkins**

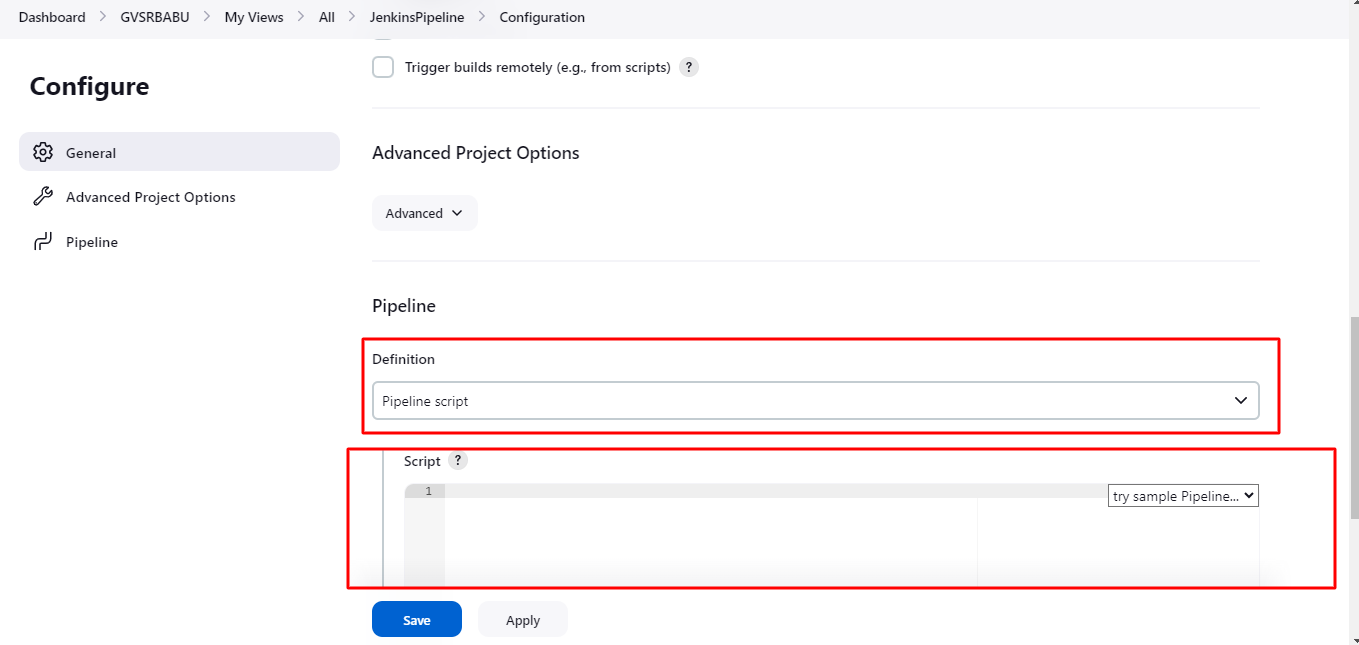
**1.Classic Pipeline**

**2.Blue ocean Pipeline**

**Open Jenkins🡪Dashboard🡪New Item🡪Enter project Name🡪Select Pipeline option🡪Click Ok button.**

****

**Select Pipeline script dropdown.**

****

**Enter basic code in**

**pipeline{**

**agent any**

**stages{**

**stage("Build"){**

**steps{**

**echo("Build the project")**

**}**

**}**

**stage("Deploy on STAGE"){**

**steps{**

**echo("Deploy on STAGE")**

**}**

**}**

**stage("Run Sanity Testcases on STAGE"){**

**steps{**

**echo("Run Sanity Testcases on STAGE")**

**}**

**}**

**stage("Deploy on PROD"){**

**steps{**

**echo("Deploy on PROD")**

**}**

**}**

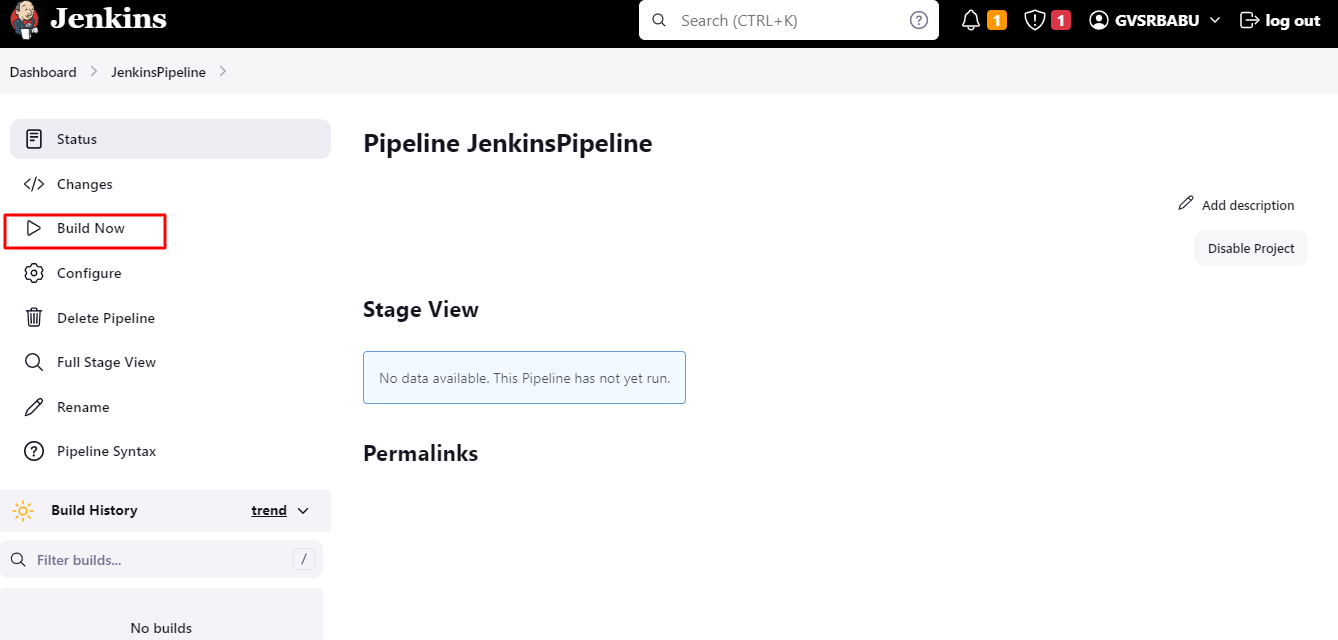
**}**

**}**

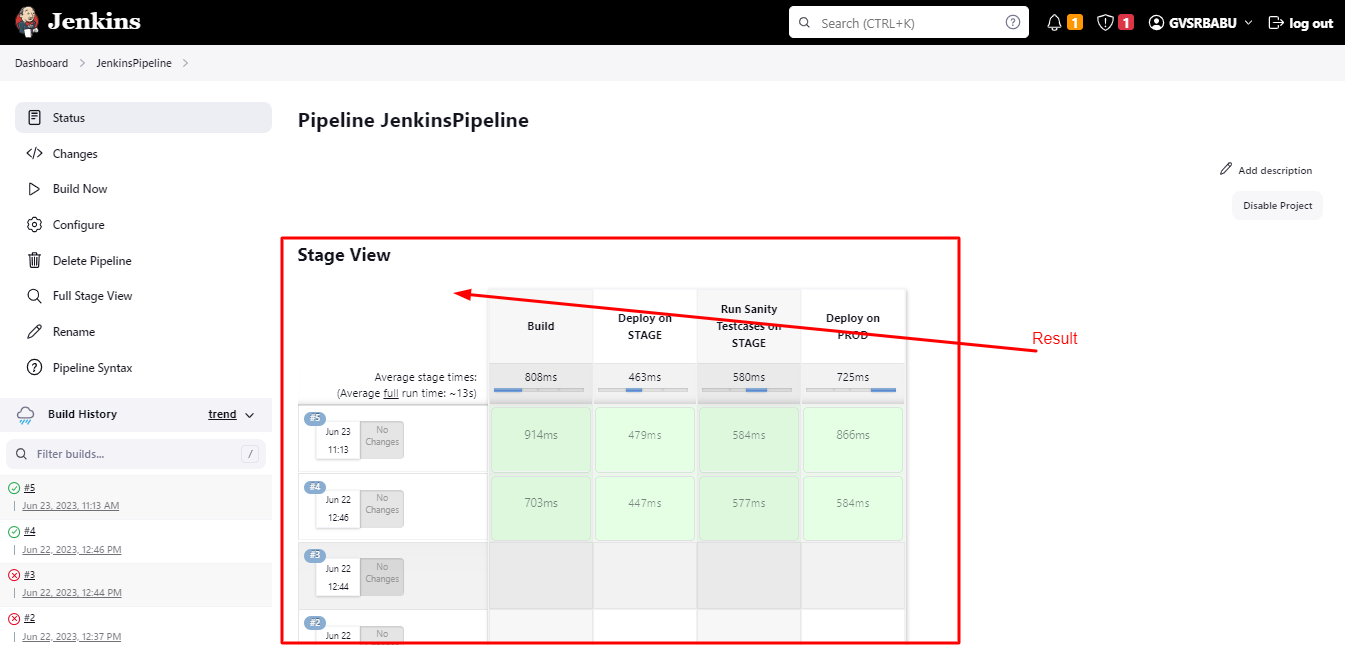
**Click on Apply and Save**

****

**Click Build Now**

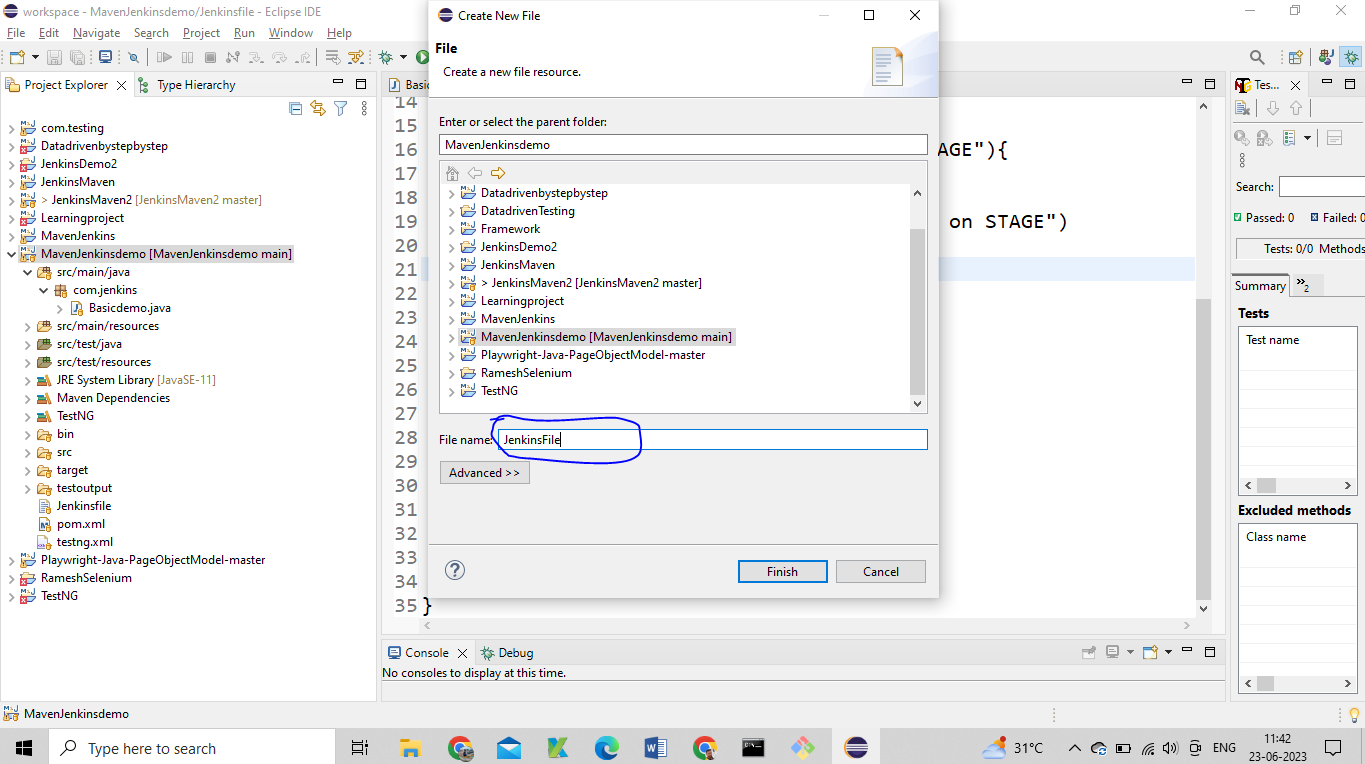
****

**Result**

****

**If I want run the Eclipse script from Pipe line**

**U need to create the file in Eclipse project🡪Right click the project 🡪New 🡪File🡪Create file🡪Click Finish button🡪Add the Code.**



pipeline{

agent any

stages{

stage("Build"){

steps{

echo("Build the project")

}

}

stage("Deploy on STAGE"){

steps{

echo("Deploy on STAGE")

}

}

stage("Run Sanity Testcases on STAGE"){

steps{

echo("Run Sanity Testcases on STAGE")

}

}

stage("Deploy on PROD"){

steps{

echo("Deploy on PROD")

}

}

stage("Deploy on UA"){

steps{

echo("Deploy on UA")

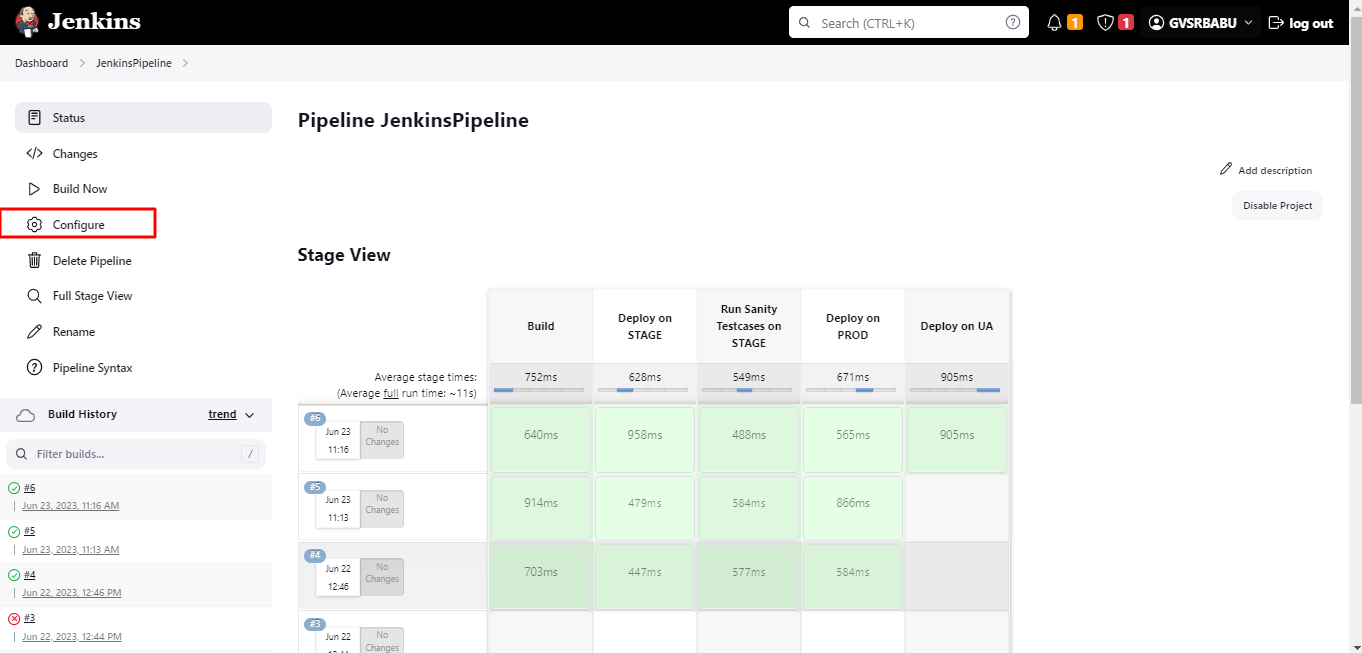
}

}

}

}

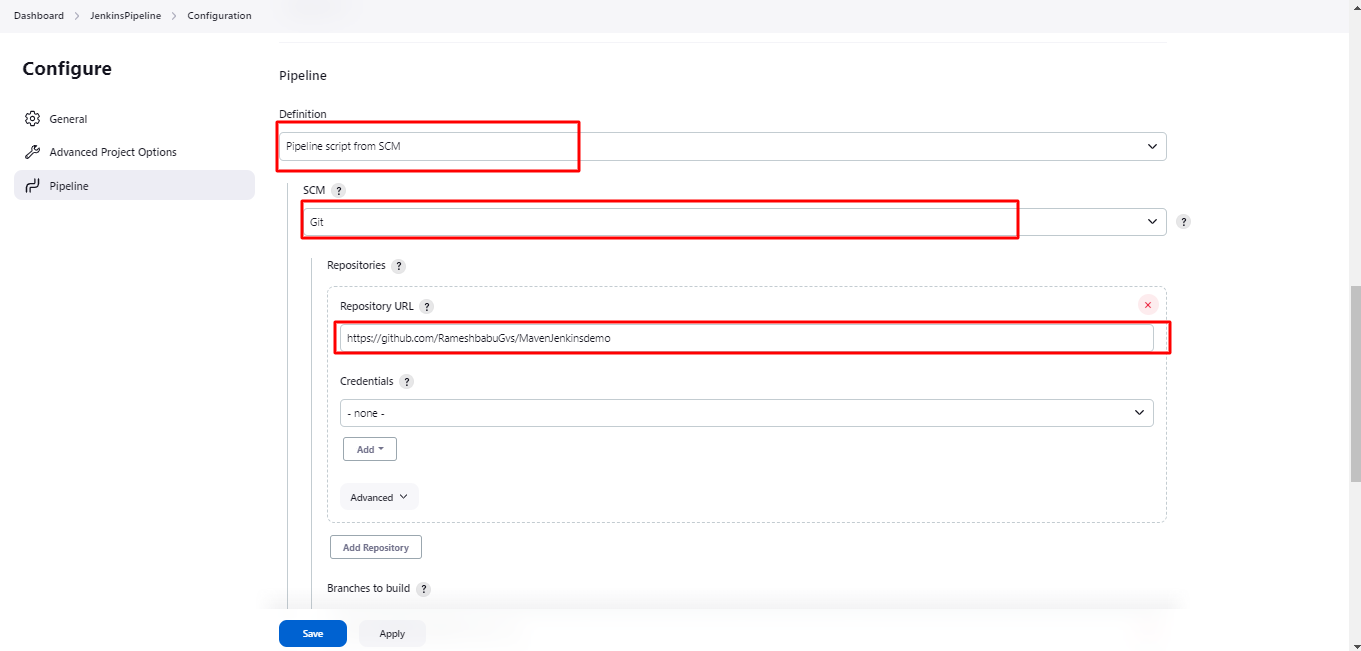
**Open Jenkins🡪Got the Created Pipeline🡪Click Configure🡪**

****

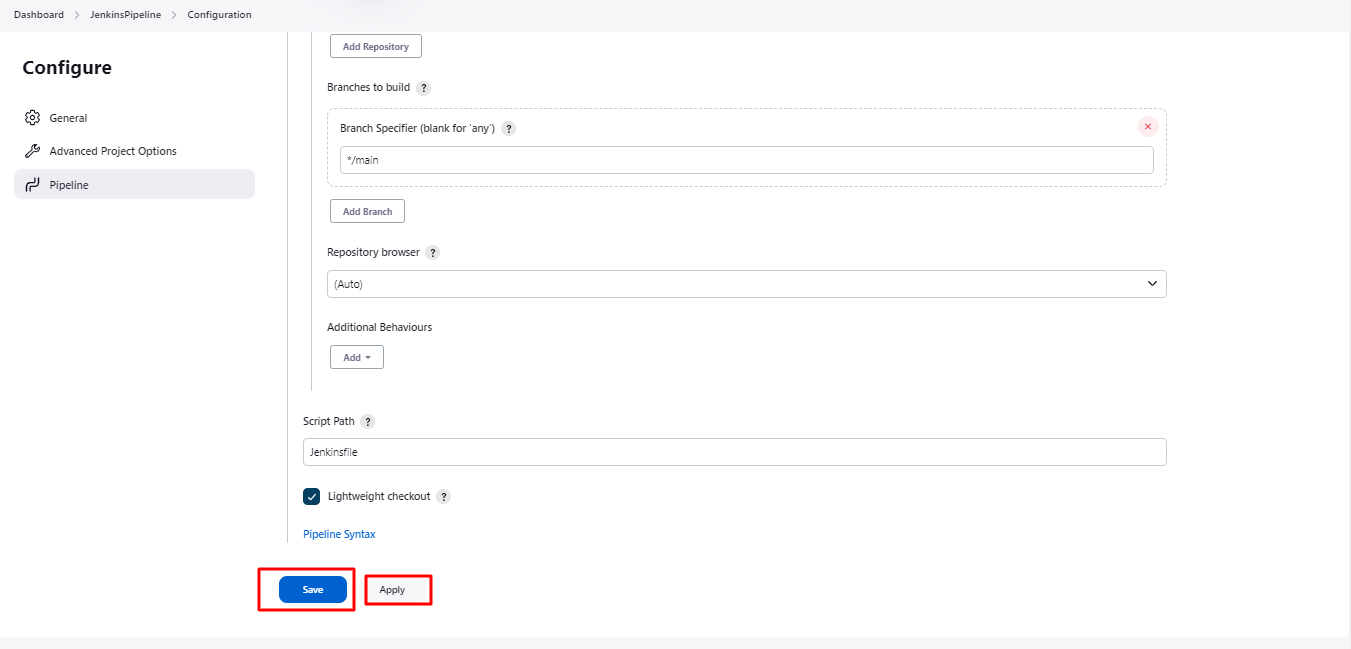
**Change the Definition🡪 Pipeline script from SCM**

**SCM🡪GIT**

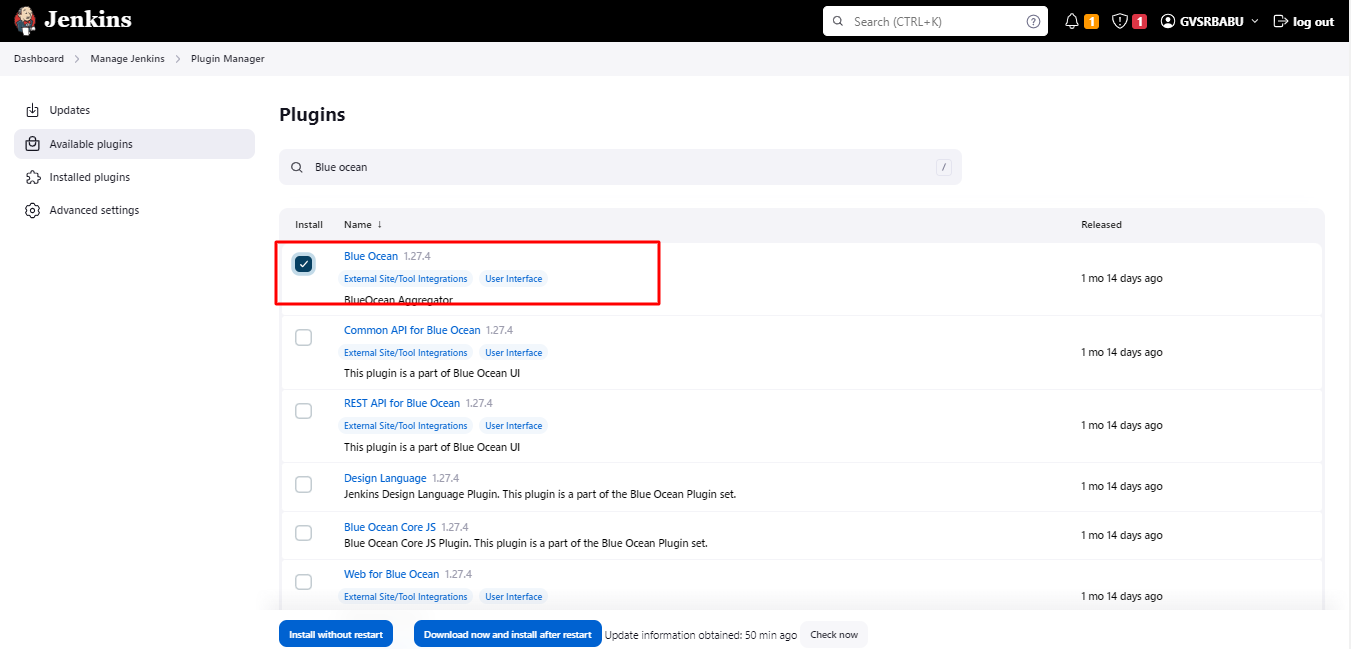
Repository URl🡪 <https://github.com/RameshbabuGvs/MavenJenkinsdemo>

****

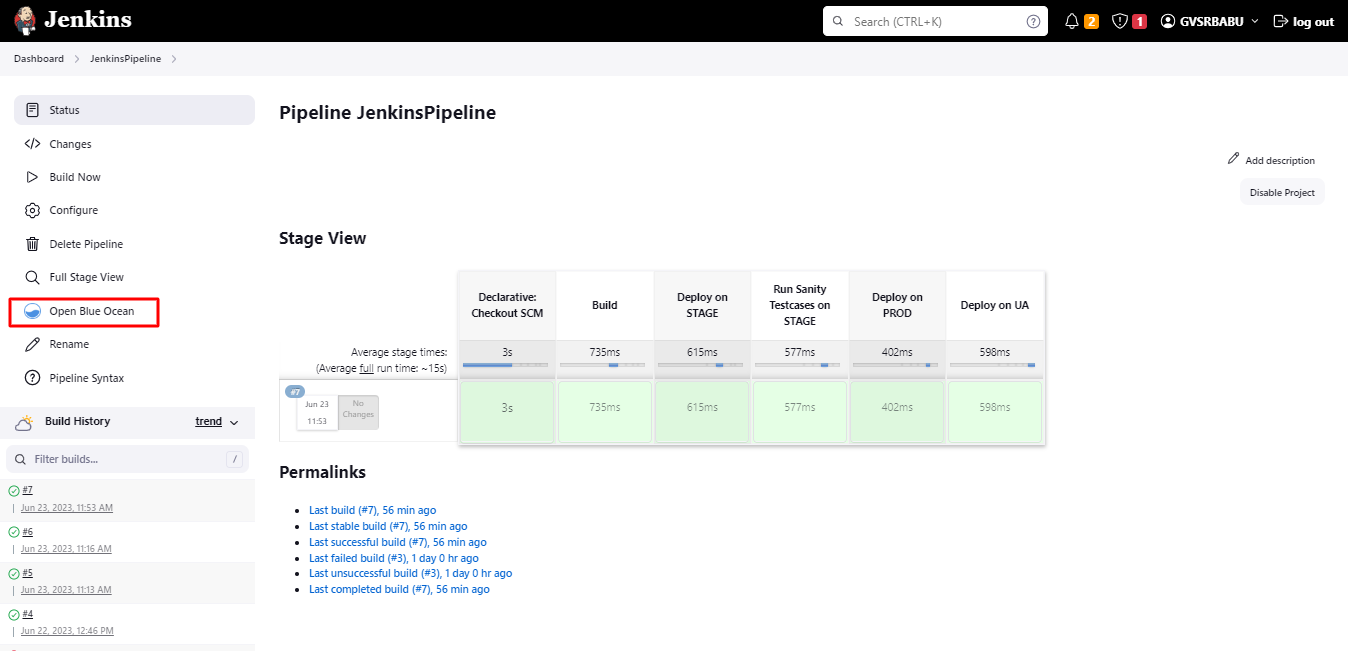
**Click Apply and Save**

****

**Install Blue ocean Plugin –** Blue Ocean is a new user experience for Jenkins based on a personalizable, modern design that allows users to graphically create, visualize and diagnose Continuous Delivery (CD) Pipelines "

****

**After Installing the plugin Inside the pipe line Blue ocean Icon is Visible**

****