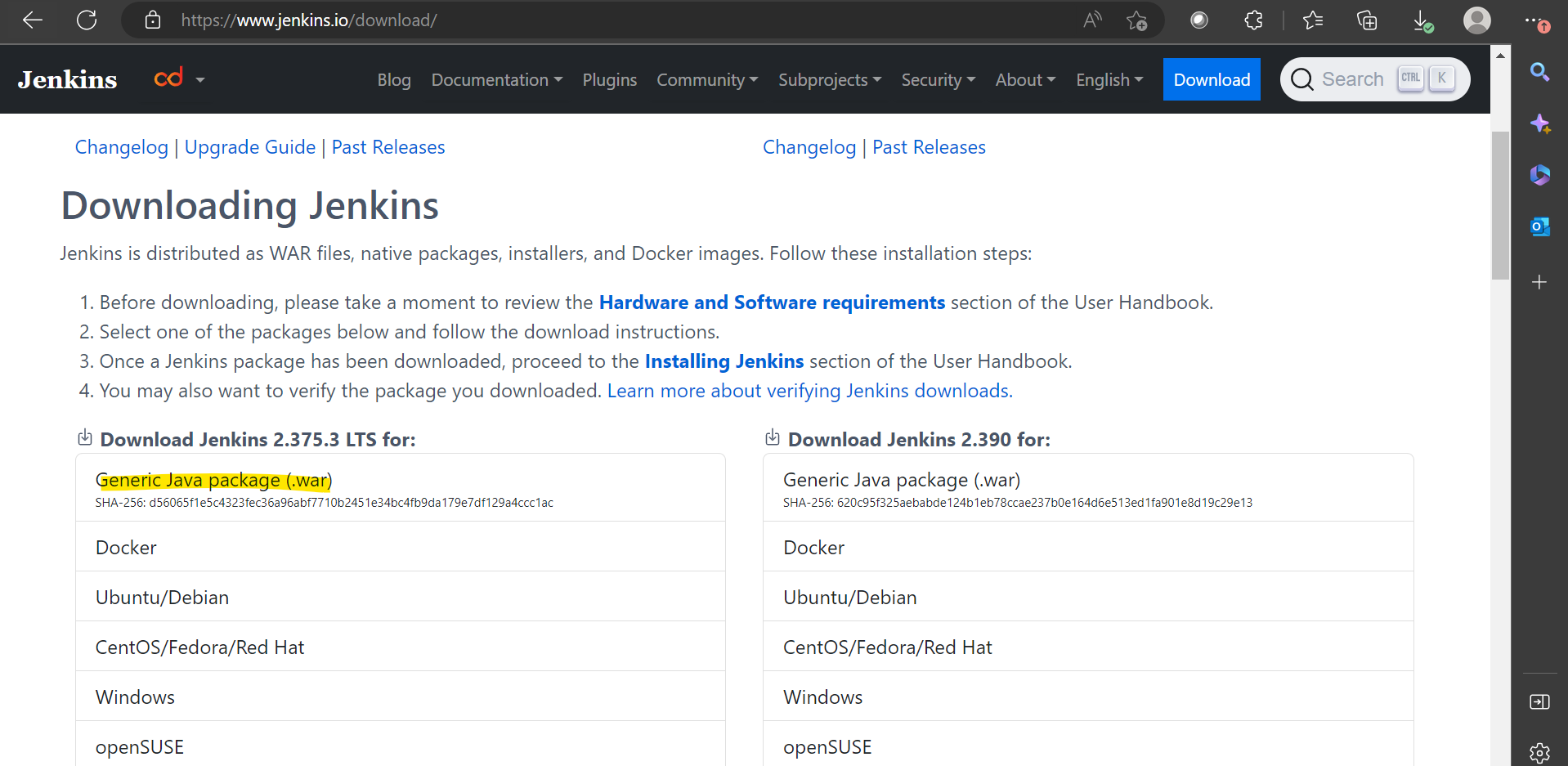
1. Install java in the system. Verify using java -version in the cmd prompt. (Jenkins expects java 11 version)
2. Download Jenkins.war file



1. Go to the folder where this war file is placed and run java -jar Jenkins.war

This will start the Jenkins on the default port 8080

Text

Description automatically generated

1. This is the initial password that is generated. Once you see the confirmation that Jenkins is fully up and running, visit localhost:8080 and paste the password

Graphical user interface, text, application

Description automatically generated

1. Click on Install suggested plugins and proceed.
2. Now create the admin user and password

Username : spraveen879

Pw : tapamth

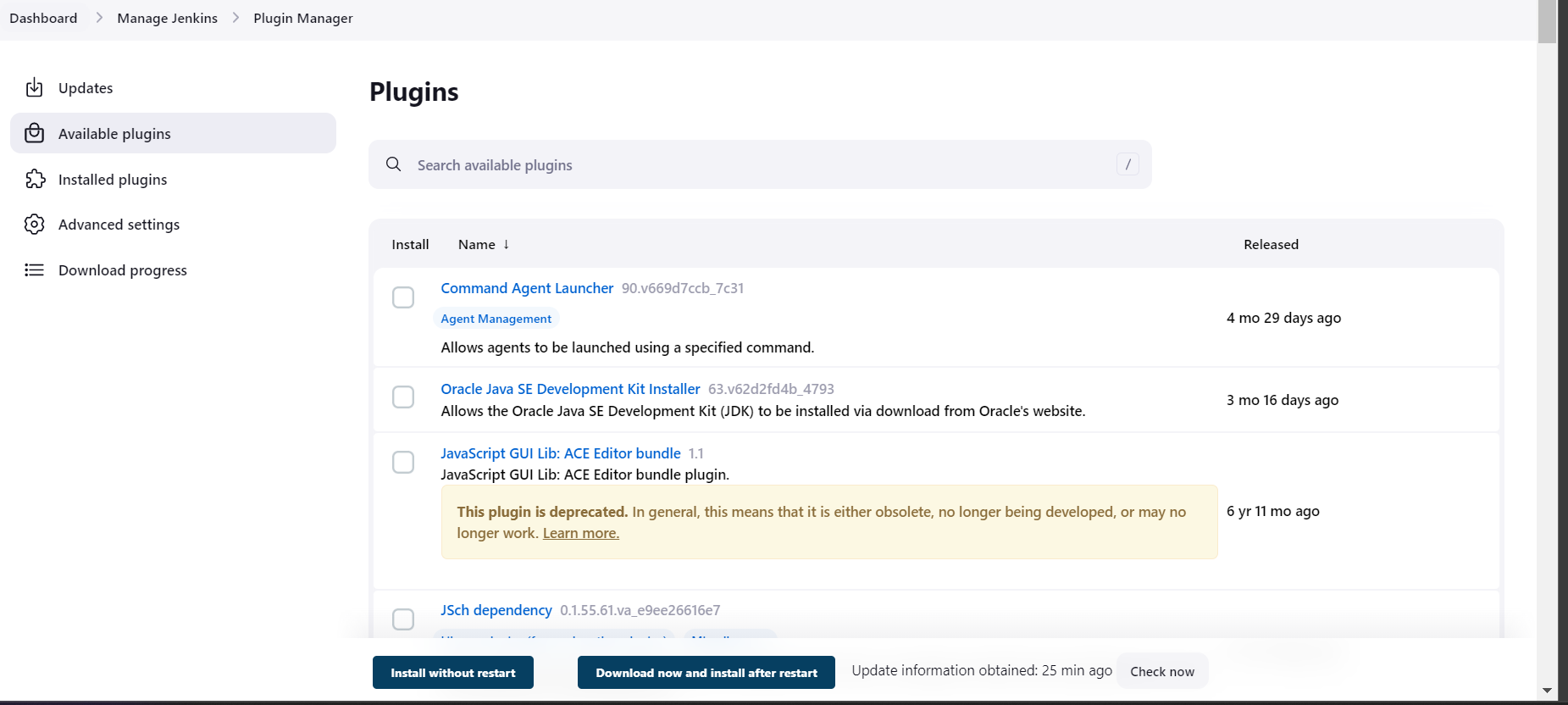
url : <http://localhost:8080/>

1. Go to Manage Jenkins – Configure System. You will see the home directory of jenkins

A picture containing graphical user interface

Description automatically generated

1. Go to Manage Jenkins and Plugin manager to manage the plugins



1. Change the home directory in Jenkins – Sometimes this is needed if there’s no enough storage space in the system or project requirements.
   1. Create a folder in the required location.
   2. Copy the contents from home directory to this location.
   3. Go to environment variables and specify this path in JENKINS\_HOME variable.  
      Graphical user interface, application

      Description automatically generated
   4. Restart Jenkins using cmd prompt.
   5. The home directory would be updated. THIS WOULD BE THE DEFAULT DIRECTORY INSTEAD OF .jenkins in C:Users
2. **Integrate Git with Jenkins**

**Pre requisite 1** : Go to plugin manager and check if git is installed.

Graphical user interface, application

Description automatically generated

**Pre requisite 2** : Go to Manage Jenkins -> Manage credentials ->System -> Global credentials(unrestricted) and add credentials

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Now add the username and password for git. Copy the username from github.com

Graphical user interface, text, application, email, Teams

Description automatically generated

Now, Create a new job by clicking New Item -> Freestyle project from the home page. Now the configuration for the job starts as below.

Graphical user interface, text, application, email, Teams

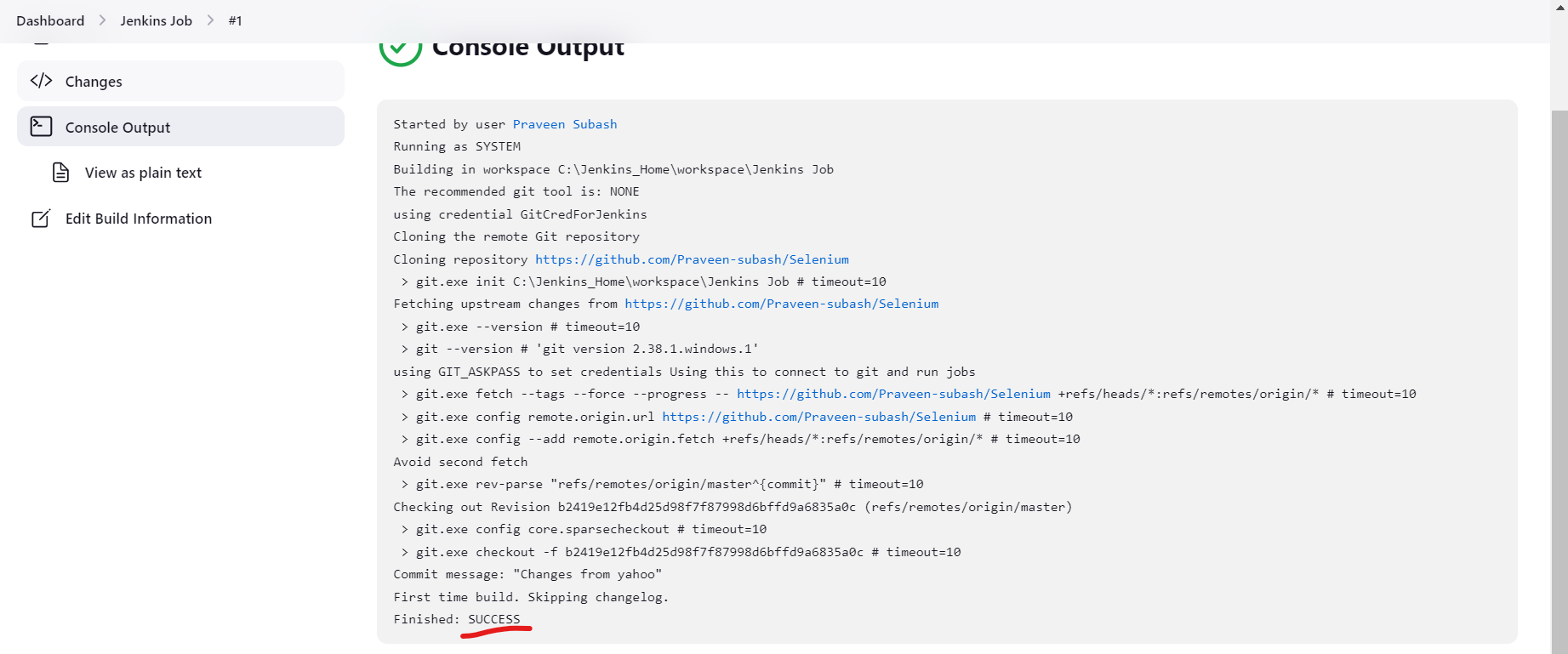
Description automatically generated

Go to source code management and enter the details as below. WE can select which branch has to be run. Here, we are selecting master. Click on apply and save

Graphical user interface, text, application, email

Description automatically generated

Click on Build now to see if the repository is linked with Jenkins.



Graphical user interface, text, application

Description automatically generated

NOTE : IF YOU DON’T WANT TO CONNECT GIT IN DEFAULT DIRECTORY AS SET, USE THE BELOW OPTION TO GIVE CUSTOM WORKSPACE INSTEAD OF CONFIGURING DEFAULT DIRECTORY

Graphical user interface, text, application, email

Description automatically generated

1. **How to use command line in Jenkins CLI**

Go to Manage Jenkins -> Configure Global Security

Graphical user interface, text

Description automatically generated

Apply and save

Graphical user interface, text, application, Teams

Description automatically generated

Go to cli page of Jenkins - [***http://localhost:8080/cli/***](http://localhost:8080/cli/)and download the cli jar

Graphical user interface, text, application, Teams

Description automatically generated

Open the command line location of the folder and paste the command shown above

Text

Description automatically generated

If you are seeing such authentication issues, run the below command

***java -jar jenkins-cli.jar -s*** [***http://localhost:8080***](http://localhost:8080) ***-auth username:password***

java -jar jenkins-cli.jar -s <http://localhost:8080> -auth spraveen879:tapamth

Text

Description automatically generated

Now to get the list of commands, open this link

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Now run the below command

***java -jar jenkins-cli.jar -s http://localhost:8080/ -auth username:password build* <jobname>**

java -jar jenkins-cli.jar -s http://localhost:8080/ -auth spraveen879:tapamth build "Jenkins Job"



Graphical user interface, text, application, email

Description automatically generated

1. **How to create users and assign roles**

Go to Manage Jenkins – Manage users – Create user and add users

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

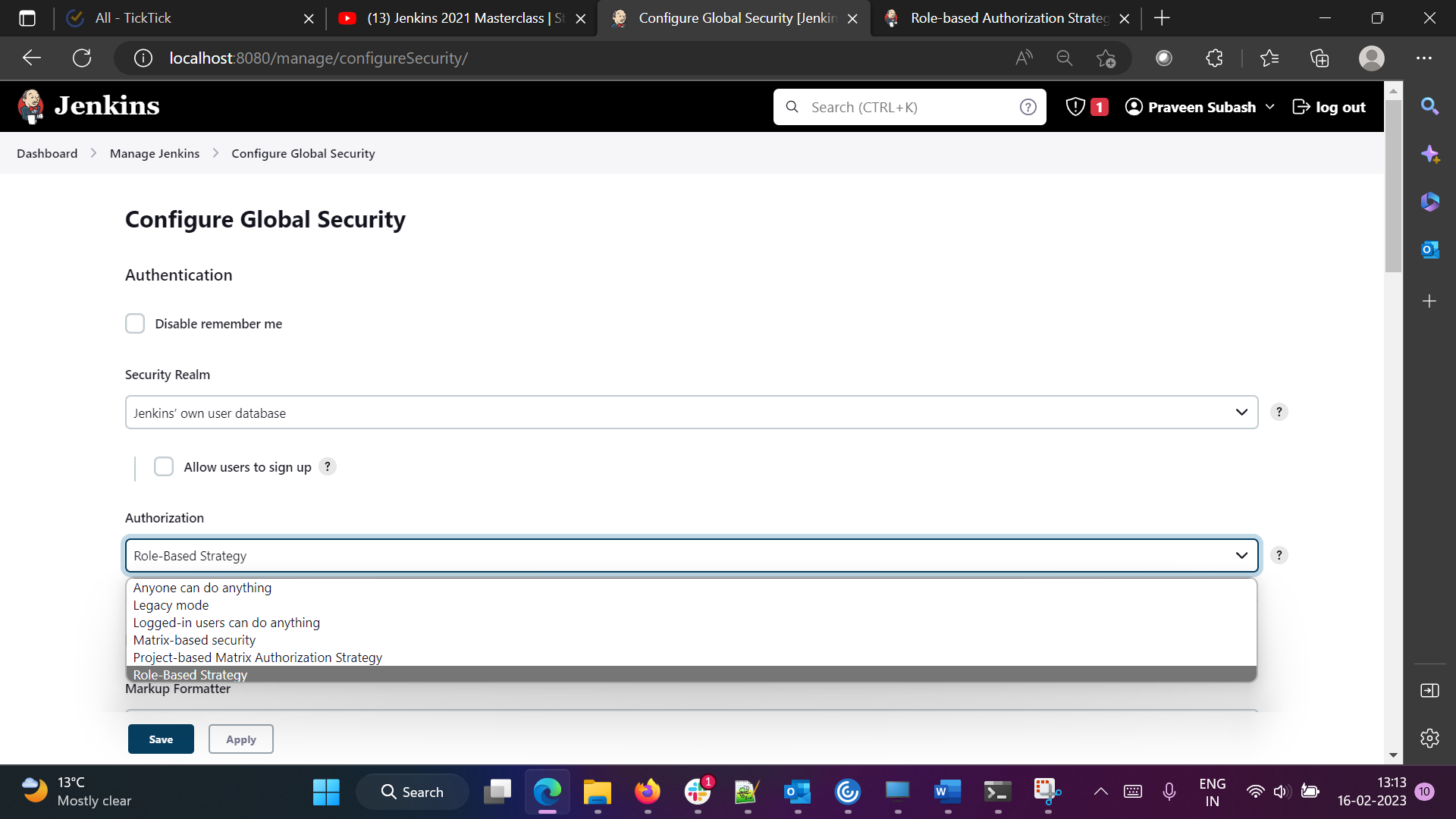
To give role based access to users, go to Manage Jenkins -> Plugins and search for [Role-based Authorization Strategy](https://plugins.jenkins.io/role-strategy)

Graphical user interface, text, application, email

Description automatically generated

Once it is installed, go to Manage Jenkins -> Configure Global security ->Configure Global Security.

Under authentication select Role Based strategy. This will be visible only after plugin is installed.



Now go to Manage Jenkins -> Manage and assign roles ->Manage roles

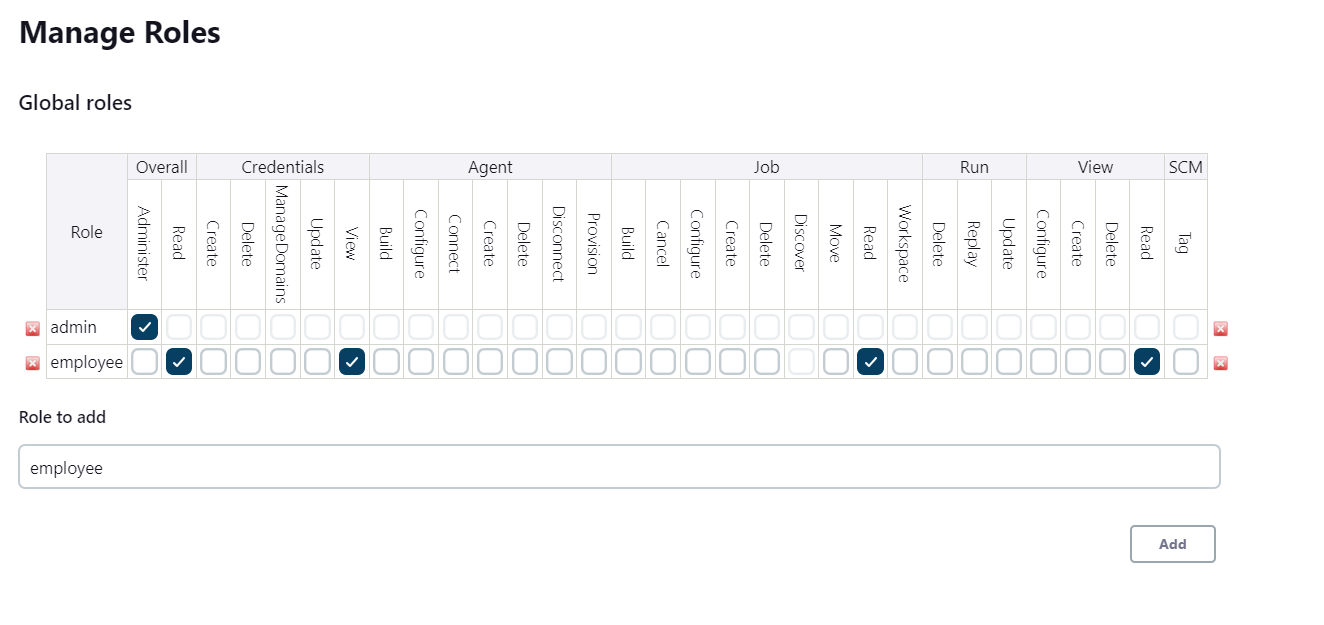
Graphical user interface, text, application, Teams

Description automatically generated

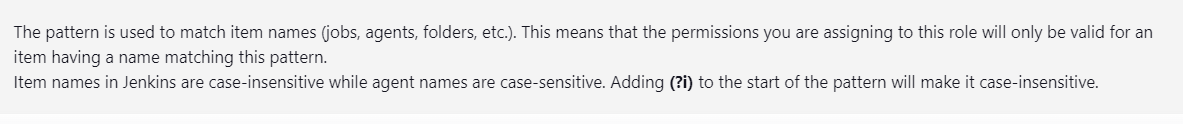
Graphical user interface, text, application, email

Description automatically generated

Create a role called employee and give the below access

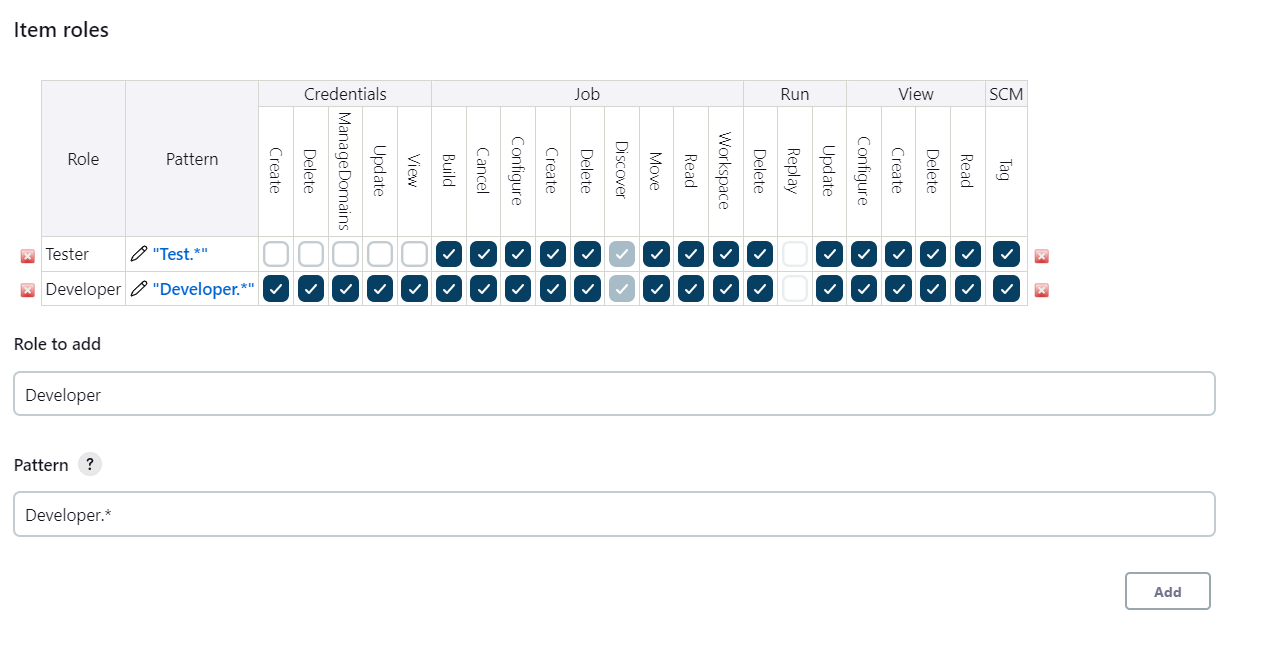


Under Item roles, create Developer and Tester roles. They can run jobs with prefix Test.\* and Developer.\*



A picture containing table

Description automatically generated



Now go to Assign roles and add the created user and roles

Here, I have added the users created and assigned them employee roles

Graphical user interface, application

Description automatically generated

We can add item based roles and give the permissions

Graphical user interface, application

Description automatically generated

Now create two new jobs DevJob and TestingJob

A picture containing graphical user interface

Description automatically generated

Now login with spraveen\_test1 which has roles set for Test.\* He has all access to test job

Graphical user interface, application

Description automatically generated

But he does not have access to dev job due to the permissions set. Similar for spraveen\_test2, dev job will be given access where testing job does not have.

Graphical user interface, application, Teams

Description automatically generated

1. **Chain Jobs**

We are now going to chain two jobs. For now instead of running anything on git, I am just going to print the content.

***So, go to Testing and Dev Job and add this step***

Job – Configure – Build Environment – Add build step – Execute windows batch command

A screenshot of a computer

Description automatically generated

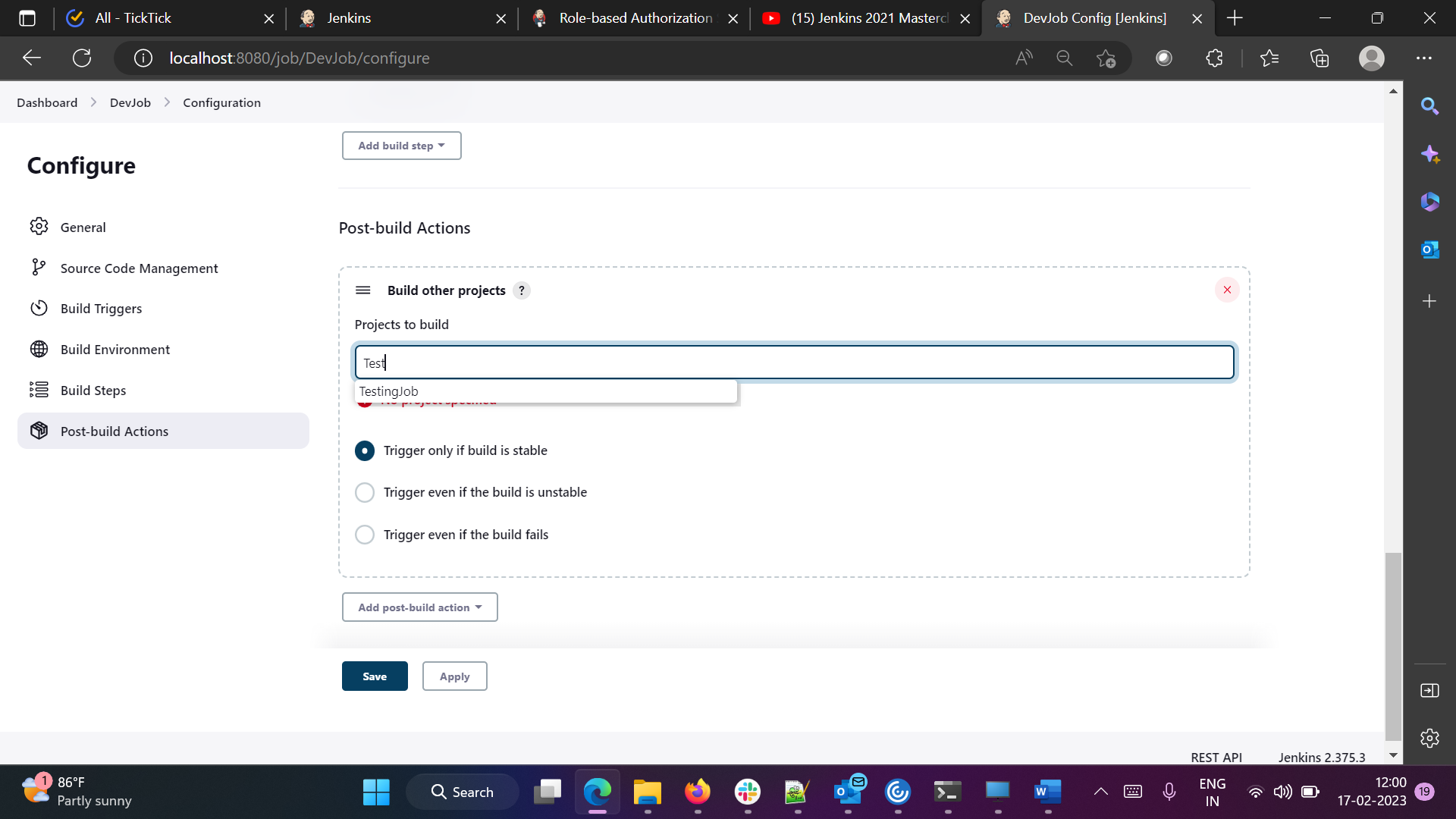
Graphical user interface, text, application, email

Description automatically generated

Replicate the same in Dev job as well.

Chaining of jobs is necessary because we need to specify which job to run first. In our case, Dev job should run first and then the test job should run

To do this, go to Dev Job, click on Post Build Actions – Add post build action – Build other projects. Specify which job to run after this job.Select appropriate radio button and save



To verify this, if yu go to testing job you will see this.

Graphical user interface, text, application

Description automatically generated

1. **Build pipeline**

Go to Manage Plugins and download Build Pipeline plugin

Graphical user interface, text, application

Description automatically generated

Table

Description automatically generated

Now go to homepage and click on (+) button

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email, Teams

Description automatically generated

Select the initial job as devjob. SO when it starts first, next testing job runs.

Graphical user interface, text, application, email, Teams

Description automatically generated

Enter the below details and create the pipeline

Text, timeline

Description automatically generated with medium confidence

The below shows how your pipeline looks.

Graphical user interface, website

Description automatically generated

Click on Run at the top. Now both gets run and shows in green

Graphical user interface, website

Description automatically generated

1. **Jenkins Pipeline**

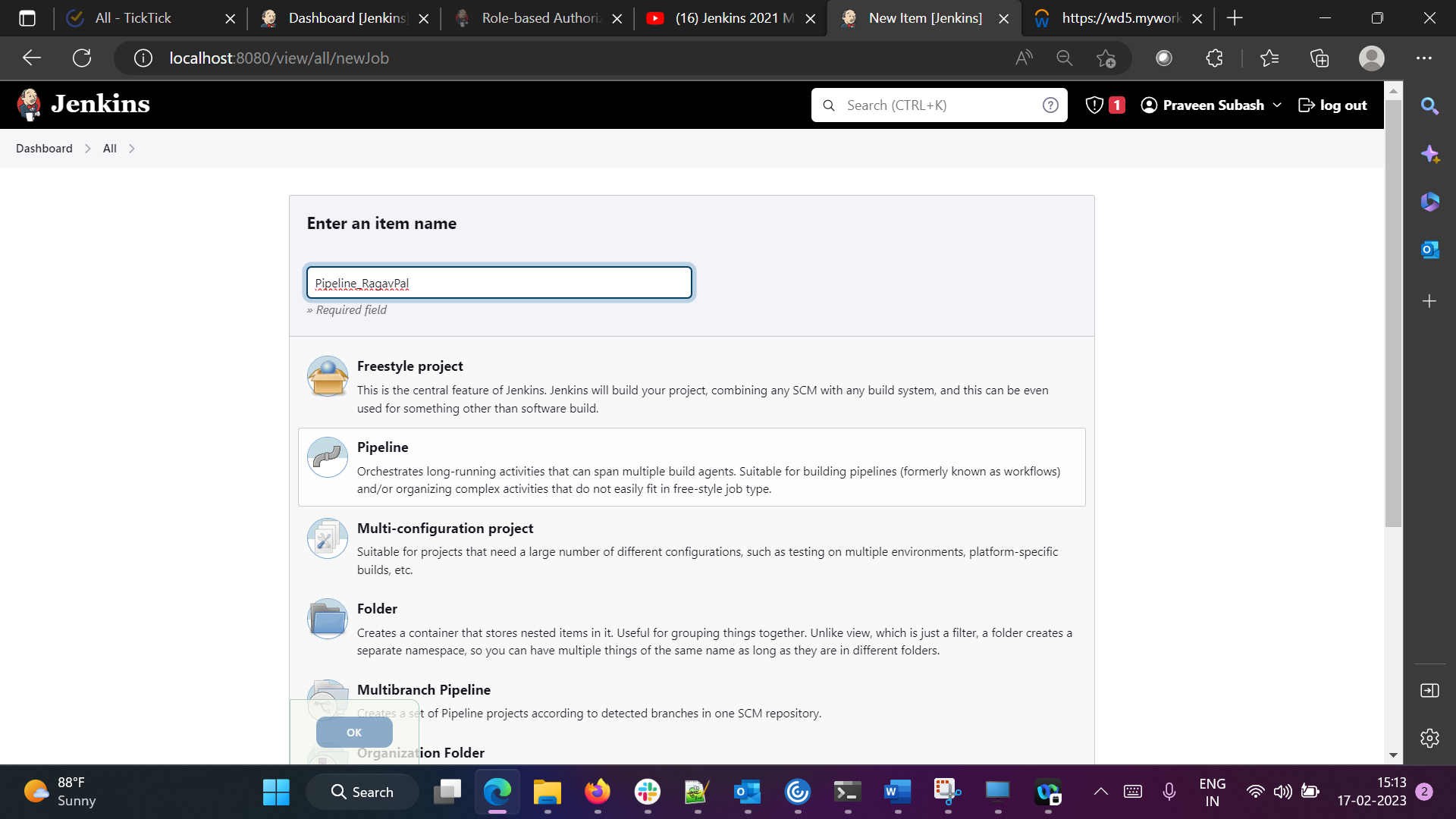
Jenkins File – Pipeline as a code is Jenkins File. With this we can implement Jenkins Pipeline.

Make sure that the below plugin is already installed else install it.

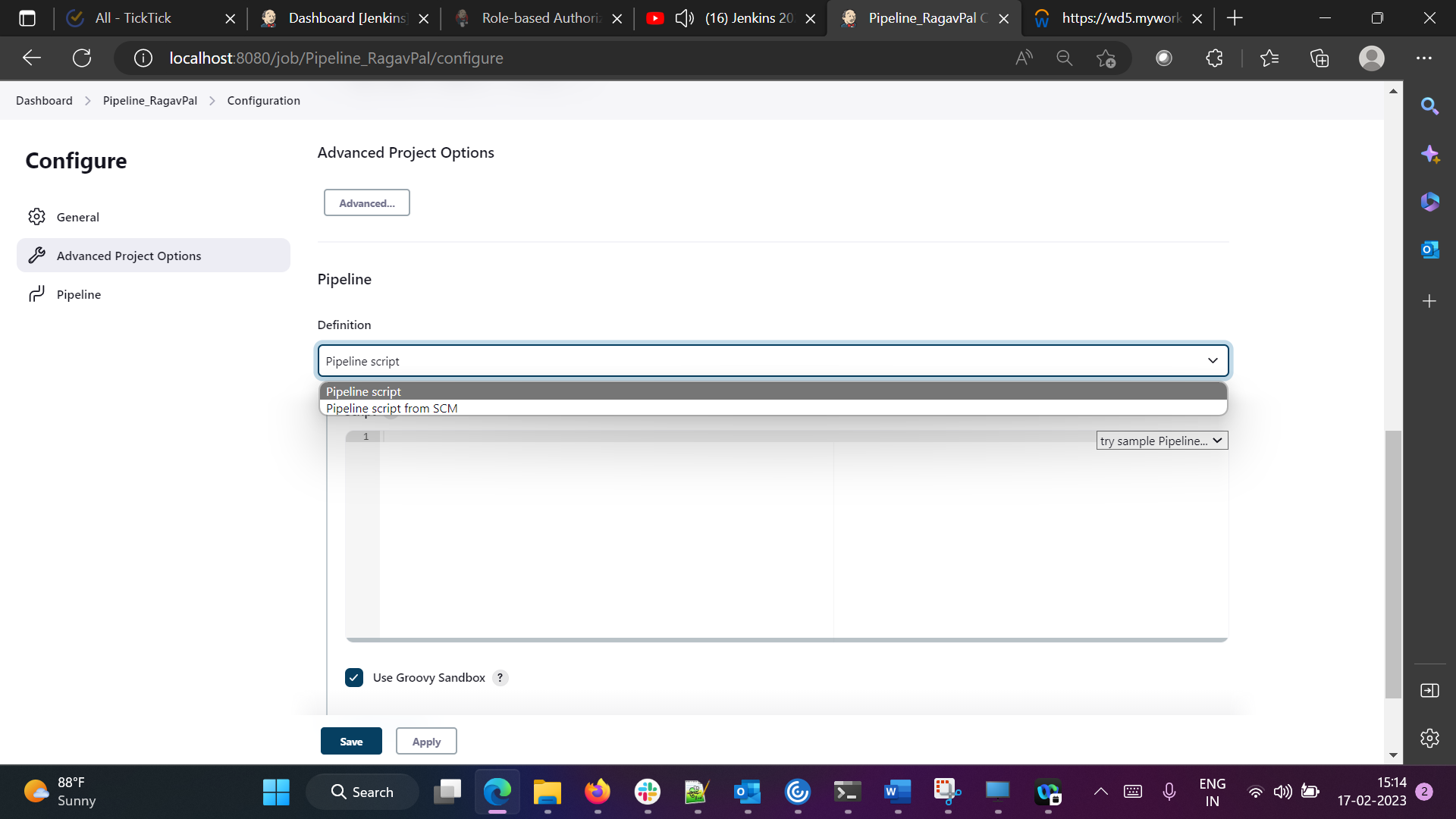
Graphical user interface, text, application, email, Teams

Description automatically generated

Click on New Item -> Pipeline



Here we can see 2 options. Either we can write our own Pipeline script or get it from Source Code Management like git.



We can write our own pipeline script here and try the traditional scripted pipeline option. Below is the example

A screenshot of a computer

Description automatically generated

Our own pipleline script

Agent : we can use any specific agent. Here we use any

Stage : Build or deploy or dev or qa

Steps : what that pipeline should do

A picture containing graphical user interface

Description automatically generated

NOTE : If yu are not sure how to write a pipeline script, click on Pipeline Syntax

Graphical user interface

Description automatically generated

If you give the requirement, pipeline code gets generated

Graphical user interface, application, Teams

Description automatically generated

Now click on save. Pipleline home page gets generated

Graphical user interface, application, Teams

Description automatically generated

Now click on Build Now

Graphical user interface, application

Description automatically generated

Make changes like these and build

Graphical user interface

Description automatically generated with medium confidence

You can see different stages like below

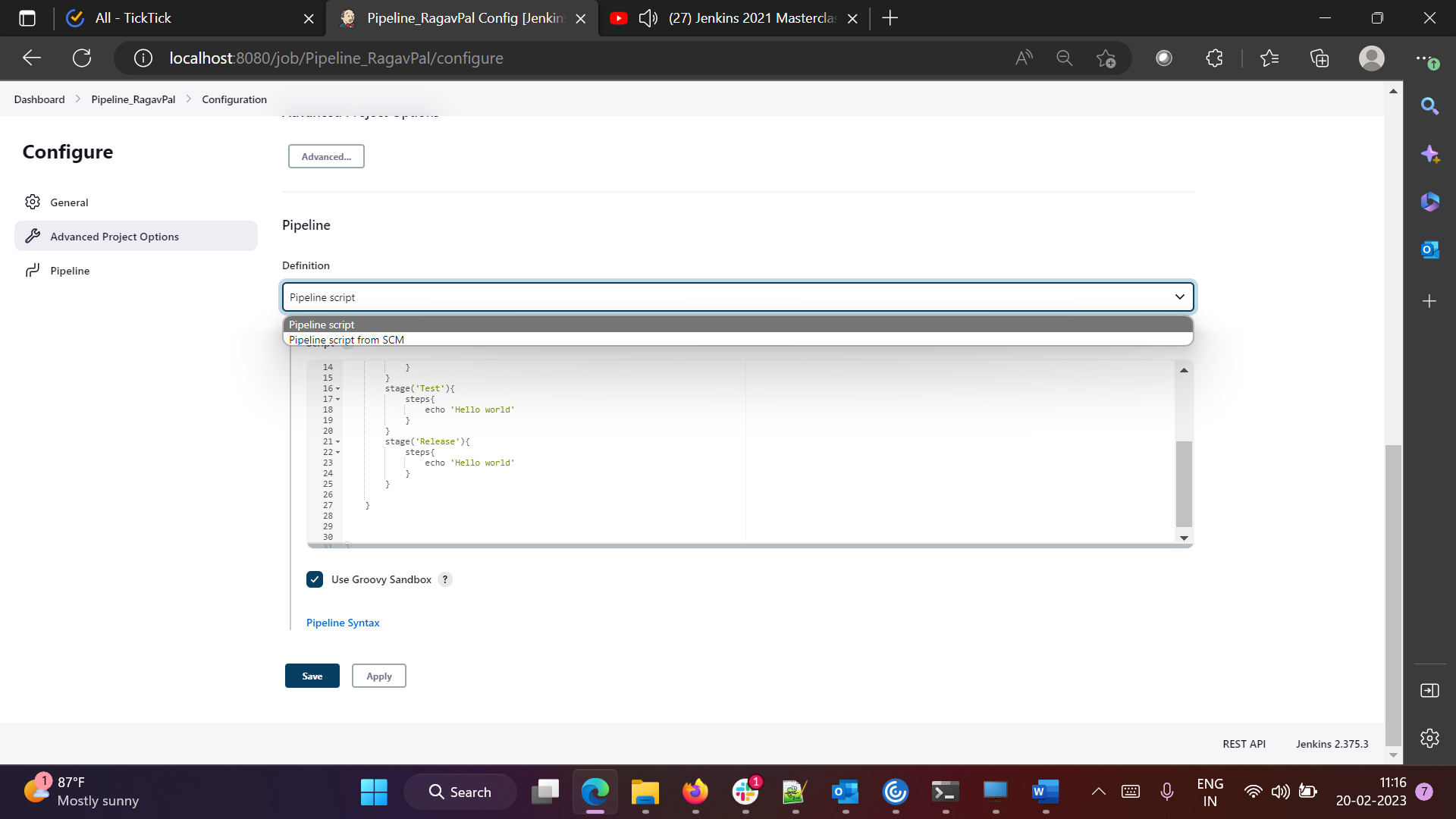
Graphical user interface, application, table

Description automatically generated

1. **How to get Jenkins File from git SCM**

Step 1 : Create a pipeline. Under configuration you can see options like this

Choose pipeline script from SCM and select the options as below



Graphical user interface, text, application

Description automatically generated

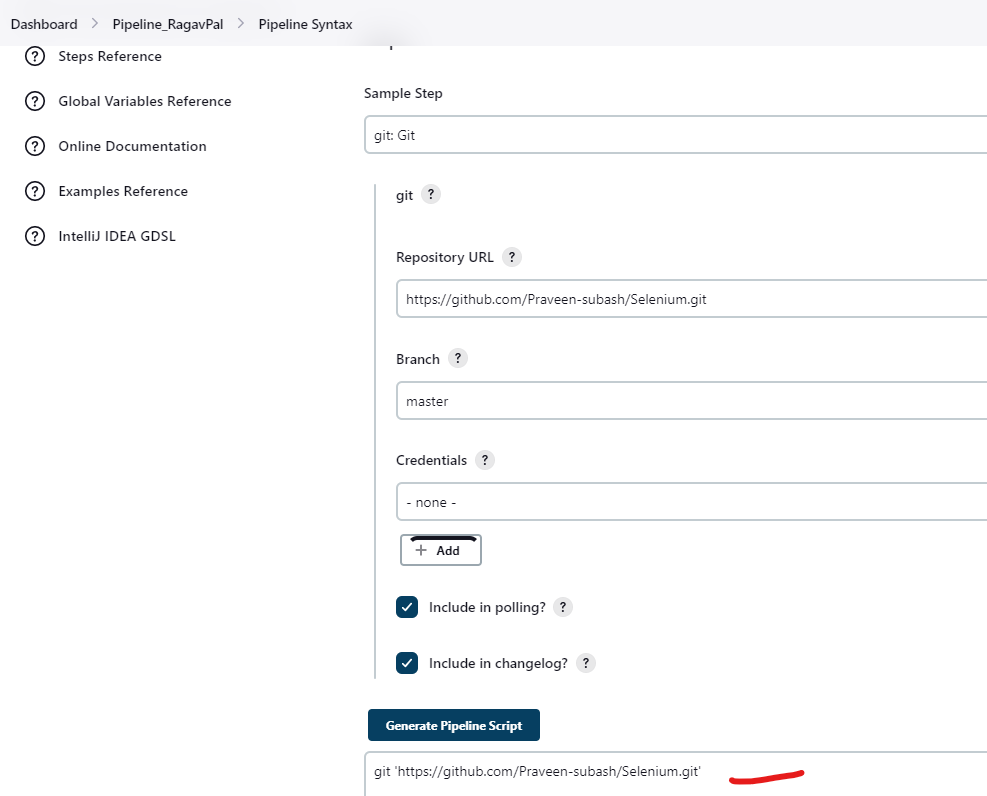
1. **How to clone a git repo using Jenkins File**

Take the pipeline and go to this link

Graphical user interface, text, application

Description automatically generated

Give the necessary details and get the jenkinsfile groovy script



Now add it in the JenkinsFile and then build it

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated