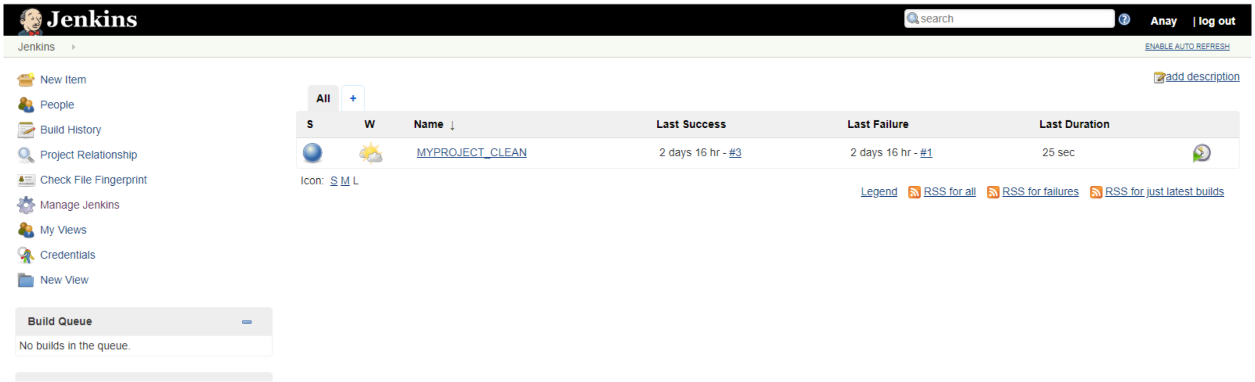
MASTER-SLAVE Configuration for Jenkins.

Step1: Install Jenkins in a host “Master_host”.

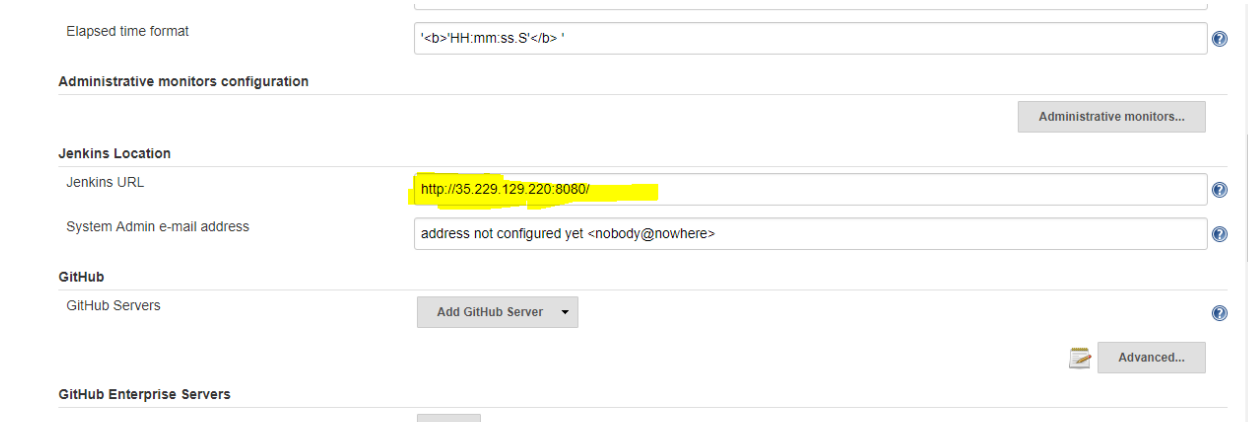


S	W	Name ↓	Last Success	Last Failure	Last Duration
●		MYPROJECT_CLEAN	2 days 16 hr - #3	2 days 16 hr - #1	25 sec

No builds in the queue.

Step2: Navigate to manage Jenkins → configure system and check the configuration on tab Jenkins location.

We should have entered our Jenkins URL



Elapsed time format: '<b>'HH:mm:ss.S'</b> '

Administrative monitors configuration

Jenkins Location

Jenkins URL: http://35.229.129.220:8080/

System Admin e-mail address: address not configured yet <nobody@nowhere>

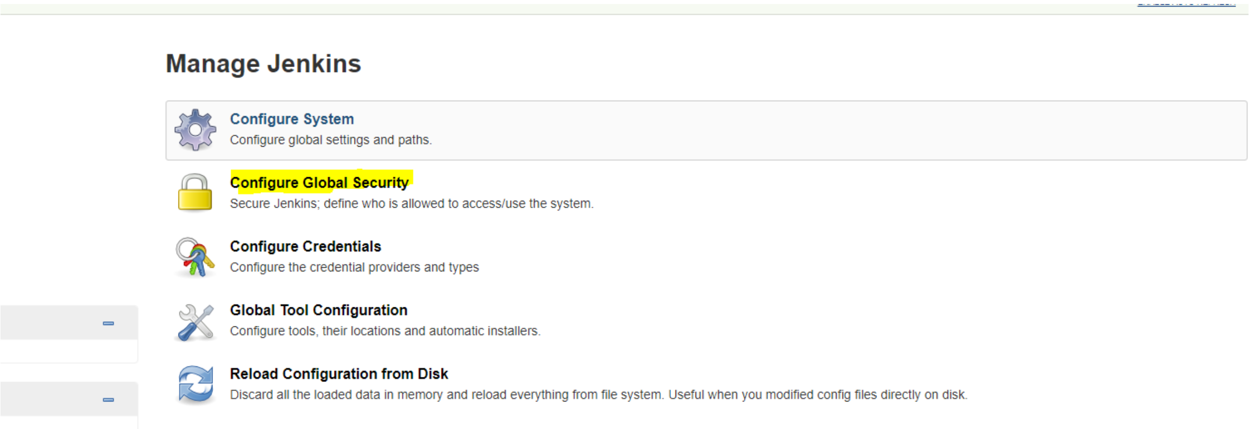
GitHub

GitHub Servers

GitHub Enterprise Servers

Step3: we will configure or global security settings.

Navigate to manage Jenkins → global security



Manage Jenkins

Configure System
Configure global settings and paths.

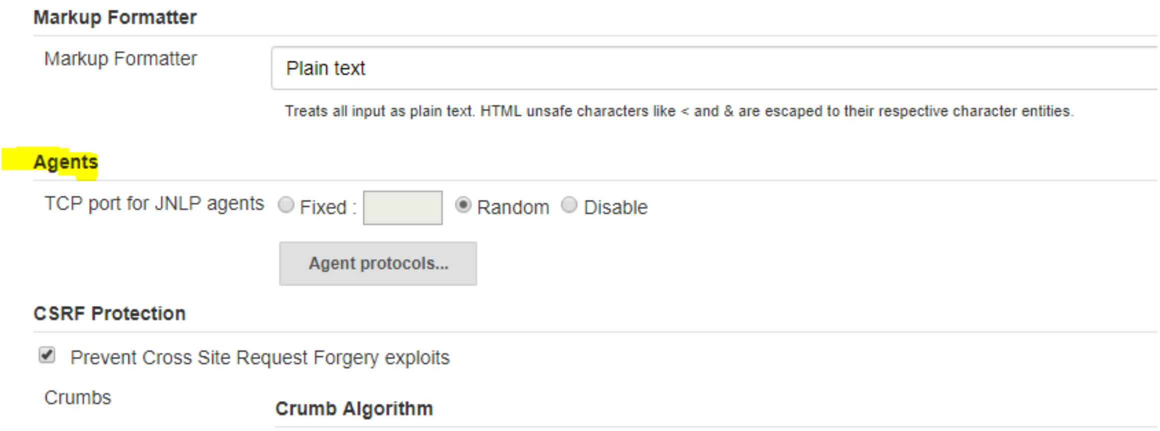
Configure Global Security
Secure Jenkins; define who is allowed to access/use the system.

Configure Credentials
Configure the credential providers and types

Global Tool Configuration
Configure tools, their locations and automatic installers.

Reload Configuration from Disk
Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.

Next, click on this and scroll down to Agent tab and select random if it is disabled.



Markup Formatter

Markup Formatter: Plain text

Treats all input as plain text. HTML unsafe characters like < and & are escaped to their respective character entities.

Agents

TCP port for JNLP agents ○ Fixed : ● Random ○ Disable

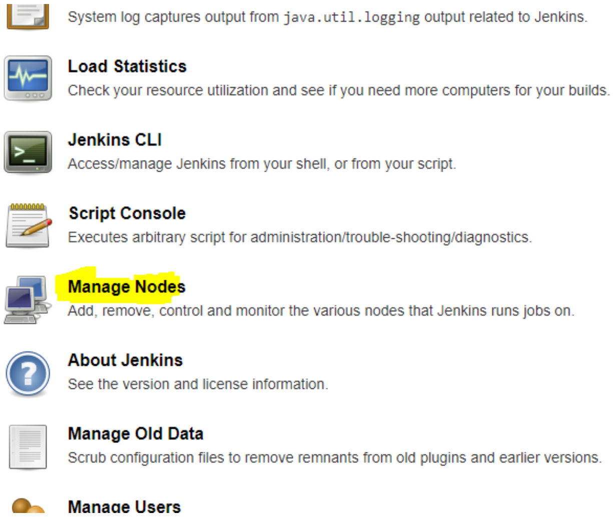
CSRF Protection

☑ Prevent Cross Site Request Forgery exploits

Crumbs

Crumb Algorithm

Step4: Navigate to manage Jenkins → Manage Nodes



System log captures output from `java.util.logging` output related to Jenkins.

Load Statistics
Check your resource utilization and see if you need more computers for your builds.

Jenkins CLI
Access/manage Jenkins from your shell, or from your script.

Script Console
Executes arbitrary script for administration/trouble-shooting/diagnostics.

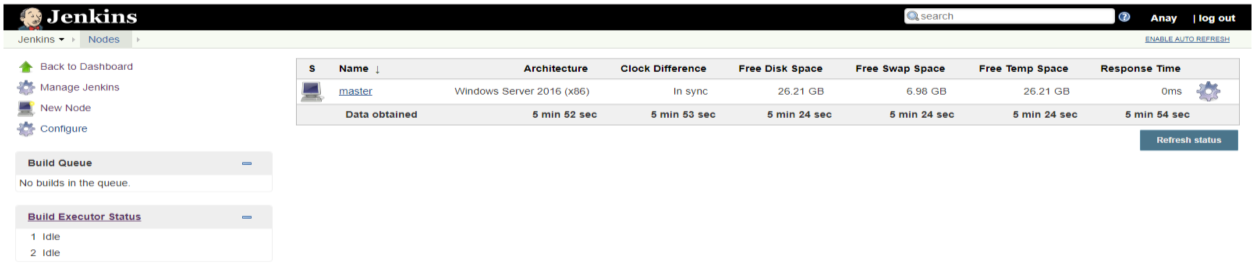
Manage Nodes
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

About Jenkins
See the version and license information.

Manage Old Data
Scrub configuration files to remove remnants from old plugins and earlier versions.

Manage Users

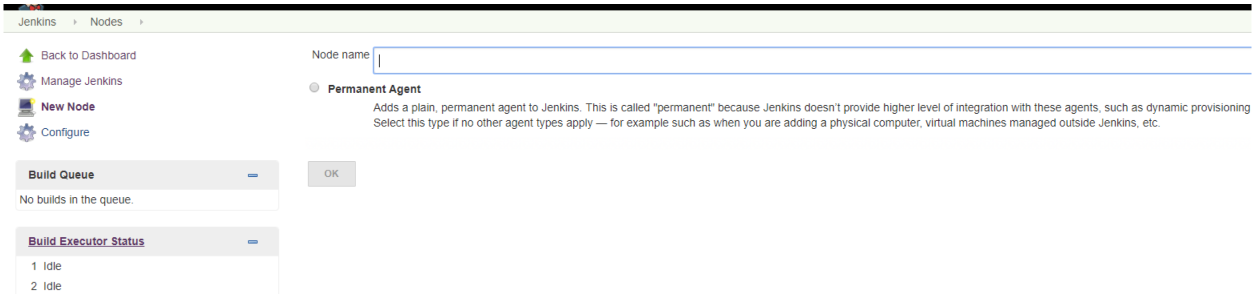
Next, click on this and we will get master node as given below:



S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	master	Windows Server 2016 (x86)	In sync	26.21 GB	6.98 GB	26.21 GB	0ms
	Data obtained	5 min 52 sec	5 min 53 sec	5 min 24 sec	5 min 24 sec	5 min 24 sec	5 min 54 sec

Step5: We will configure slave node and launch it.

Now, click on new node and we will get a page like this:

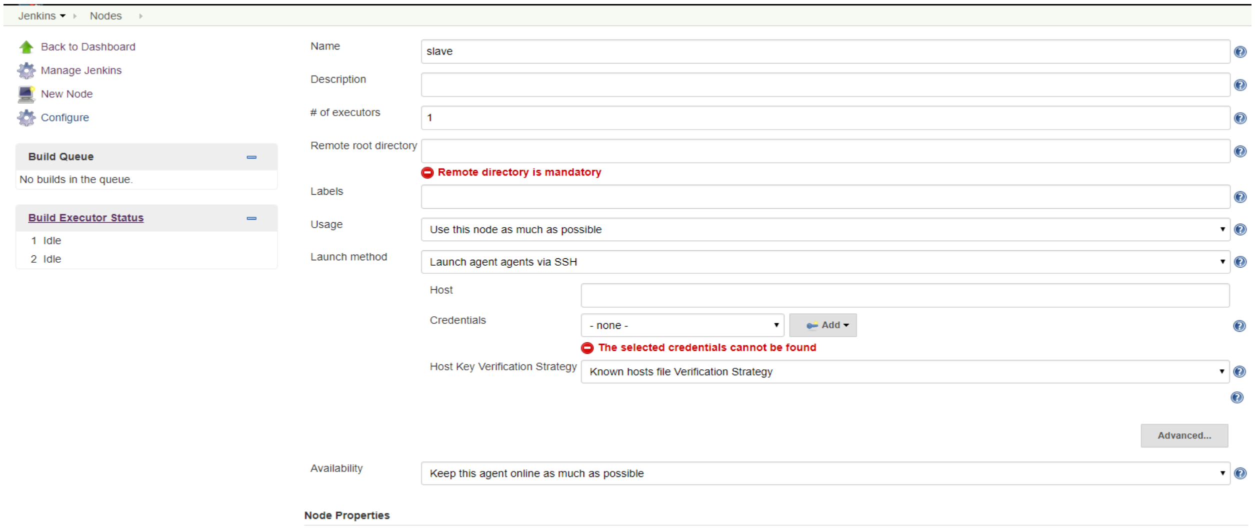


Node name

○ Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning
Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

Next, select permanent agent and give a name to agent and click on ok.



Name: slave

Description:

# of executors: 1

Remote root directory:

Remote directory is mandatory

Labels:

Usage: Use this node as much as possible

Launch method: Launch agent agents via SSH

Host:

Credentials: - none -

The selected credentials cannot be found

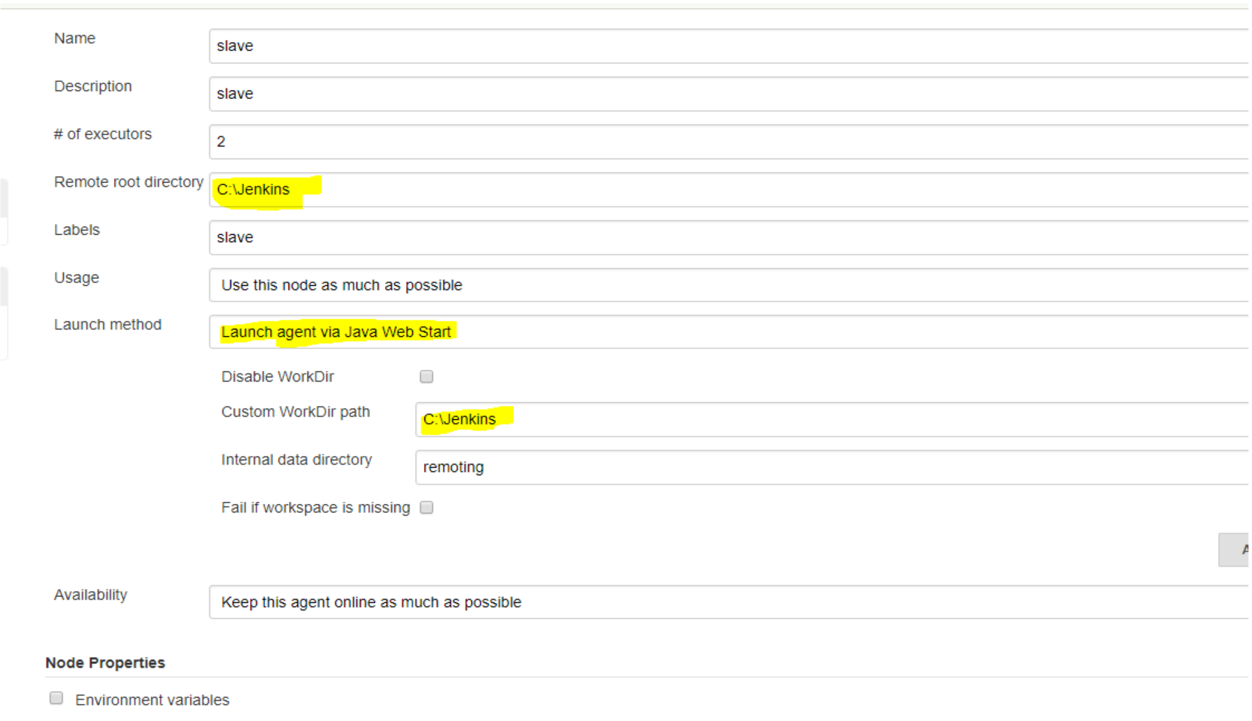
Host Key Verification Strategy: Known hosts file Verification Strategy

Availability: Keep this agent online as much as possible

Node Properties

Now, we need to have another virtual machine which we want to configure as node.

1. We must a have a dir in slave node, So create a dir “C:\Jenkins”
2. Make sure we have opened our Jenkins master server in that slave node because later in this tutorial we need to download some software’s.

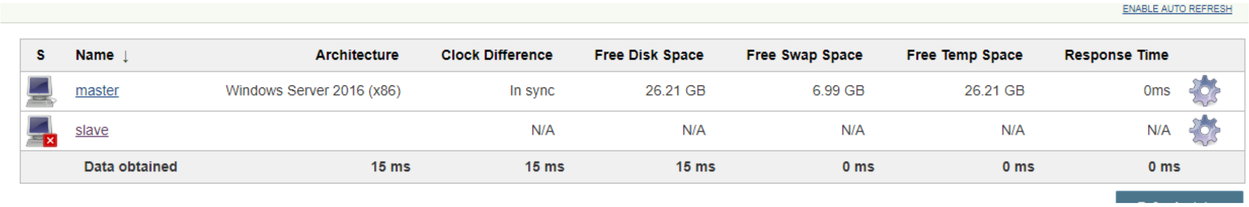


Name	slave
Description	slave
# of executors	2
Remote root directory	C:\Jenkins
Labels	slave
Usage	Use this node as much as possible
Launch method	Launch agent via Java Web Start
Disable WorkDir	□
Custom WorkDir path	C:\Jenkins
Internal data directory	remoting
Fail if workspace is missing	□
Availability	Keep this agent online as much as possible

Node Properties

□ Environment variables

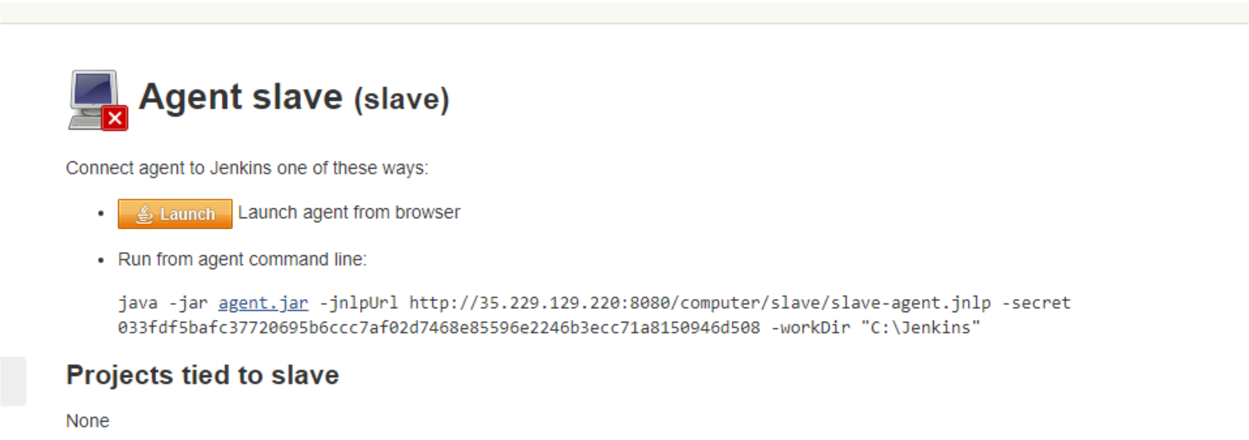
Fill all the labels as given above and click on save.



ENABLE AUTO REFRESH

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time	
	master	Windows Server 2016 (x86)	In sync	26.21 GB	6.99 GB	26.21 GB	0ms	
	slave		N/A	N/A	N/A	N/A	N/A	
	Data obtained	15 ms	15 ms	15 ms	0 ms	0 ms	0 ms	

Next, we need to launch this slave node to launch this we need to visit at http://<masterurl>/computer/slave/ and we will get:



Agent slave (slave)

Connect agent to Jenkins one of these ways:

- Launch | Launch agent from browser
- Run from agent command line:

```
java -jar agent.jar -jnlpUrl http://35.229.129.220:8080/computer/slave/slave-agent.jnlp -secret
033fdf5bafc37720695b6ccc7af02d7468e85596e2246b3ecc71a8150946d508 -workDir "C:\Jenkins"
```

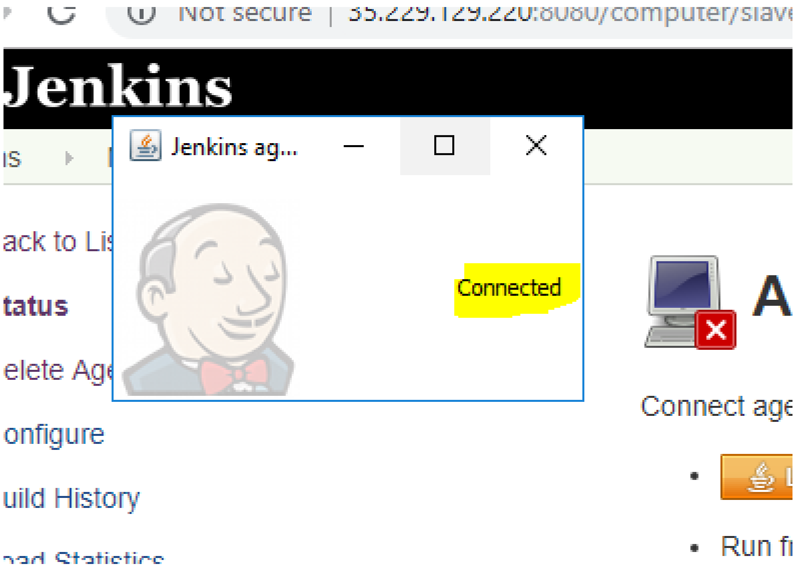
Projects tied to slave

None

Now, click on launch and it will download a .jnlp file.

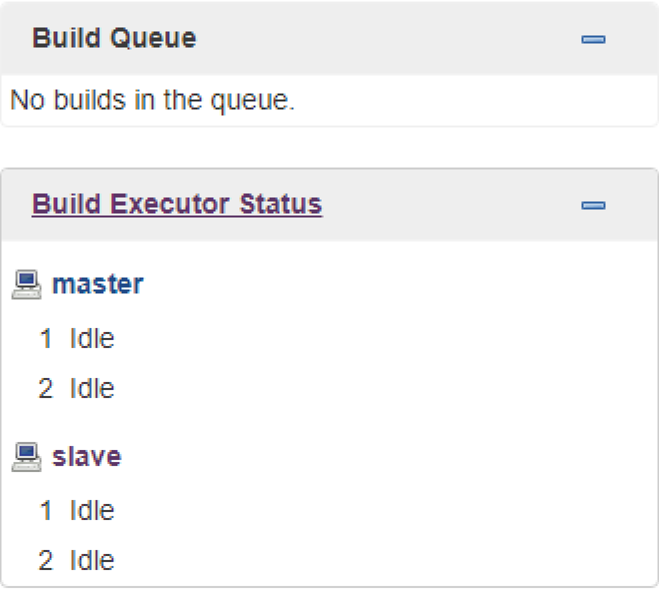
Note: The slave node must be having JDK.

Next, double click on that file and if we have jdk and the Dir that we have created previously, we will get the output as connected.



Connected

Now, refresh the master url and we can see that slave node is ready.



Build Queue

No builds in the queue.

Build Executor Status

master

1 Idle

2 Idle

slave

1 Idle

2 Idle

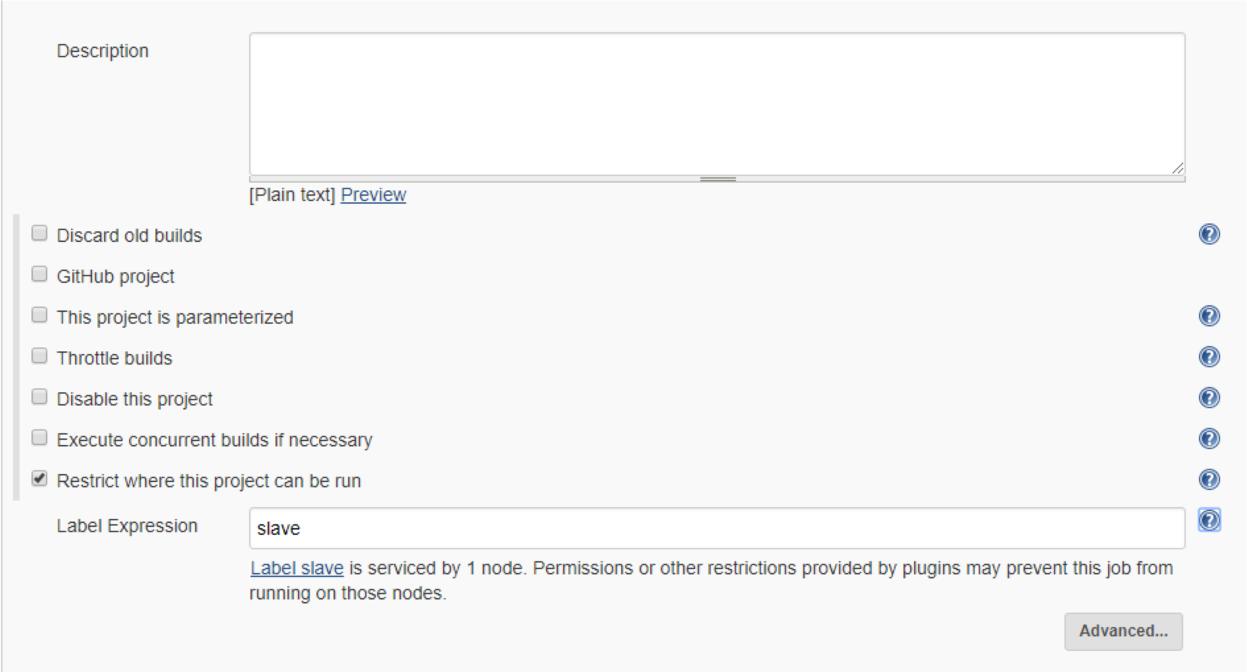
Example:

Now in this example we will restrict our Jenkins server to build a job on slave node only.

I’m assuming that you have gone through my pervious tutorials. And you are aware about how we can build a maven project in Jenkins.

We will be building a maven project. So, click on new item and configure all the steps.

General:



Description

[Plain text] Preview

□ Discard old builds

□ GitHub project

□ This project is parameterized

□ Throttle builds

□ Disable this project

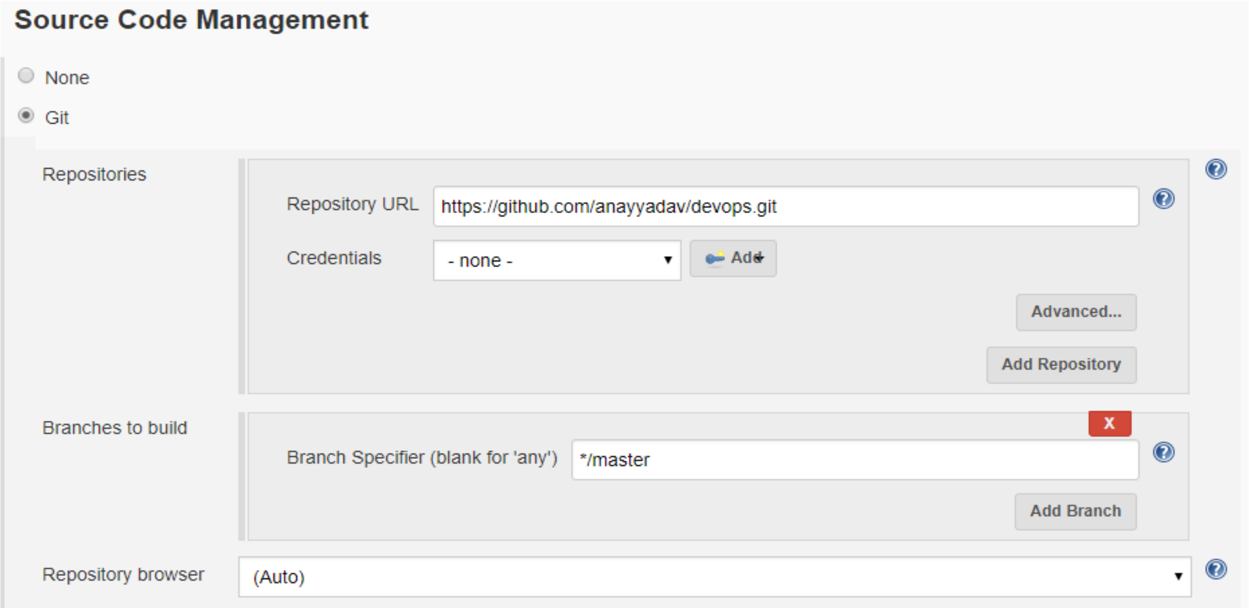
□ Execute concurrent builds if necessary

☑ Restrict where this project can be run

Label Expression: slave

Label slave is serviced by 1 node. Permissions or other restrictions provided by plugins may prevent this job from running on those nodes.

SCM:



Source Code Management

○ None

● Git

Repositories

Repository URL: https://github.com/anayyadav/devops.git

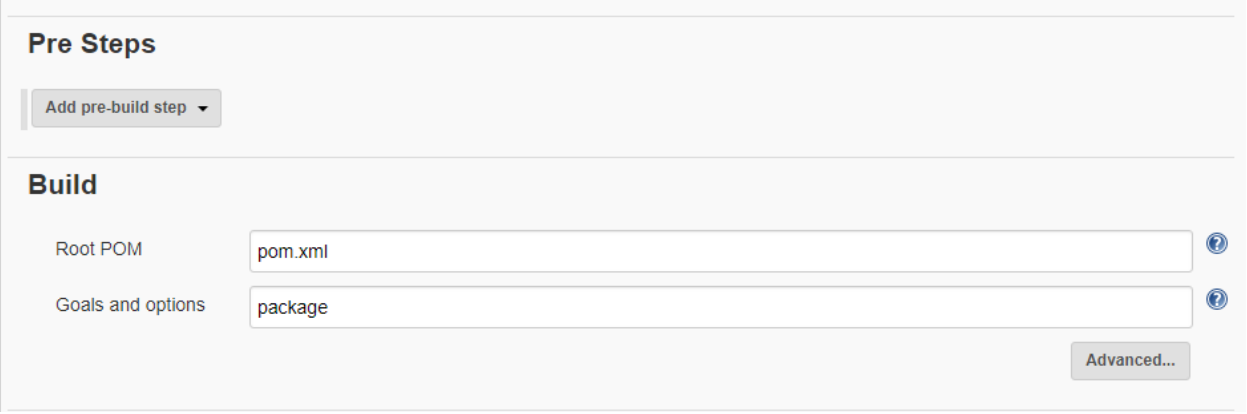
Credentials: - none -

Branches to build

Branch Specifier (blank for 'any'): */master

Repository browser: (Auto)

BUILD:



Pre Steps

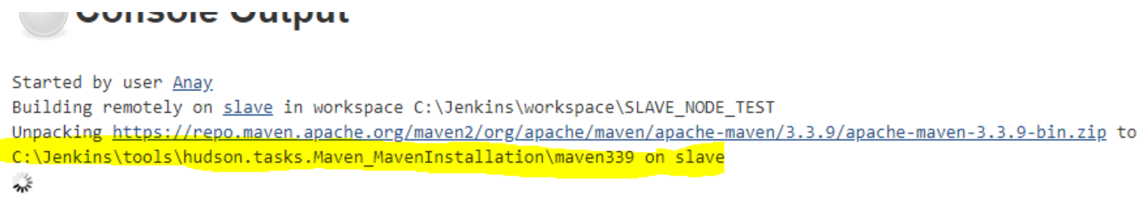
Build

Root POM: pom.xml

Goals and options: package

Finally save this job and click on build.

We can see that this process is running on slave.



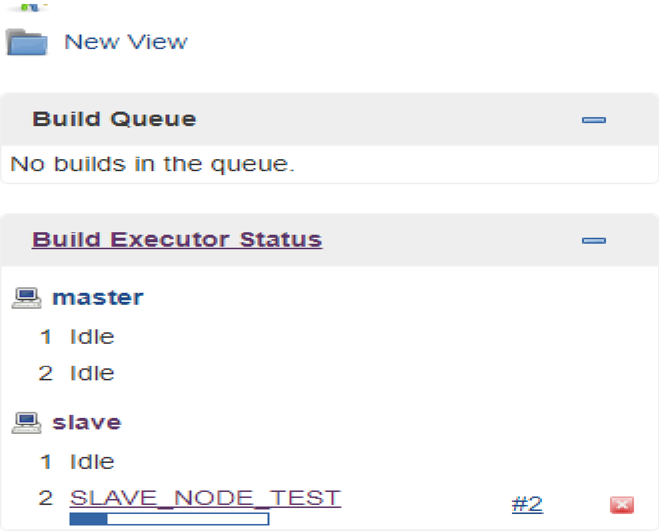
Console Output

```
Started by user Anay
Building remotely on slave in workspace C:\Jenkins\workspace\SLAVE_NODE_TEST
Unpacking https://repo.maven.apache.org/maven2/org/apache/maven/apache-maven/3.3.9/apache-maven-3.3.9-bin.zip to
C:\Jenkins\tools\hudson.tasks.Maven_MavenInstallation\maven339 on slave
```

NOTE: Make sure one must be having Git on their slave node.

Verification:

We can also verify that this job is running on Slave node. To verify we can check the main page of Jenkins server.



Build Queue

No builds in the queue.

Build Executor Status

master
1 Idle
2 Idle

slave
1 Idle
2 SLAVE_NODE_TEST #2