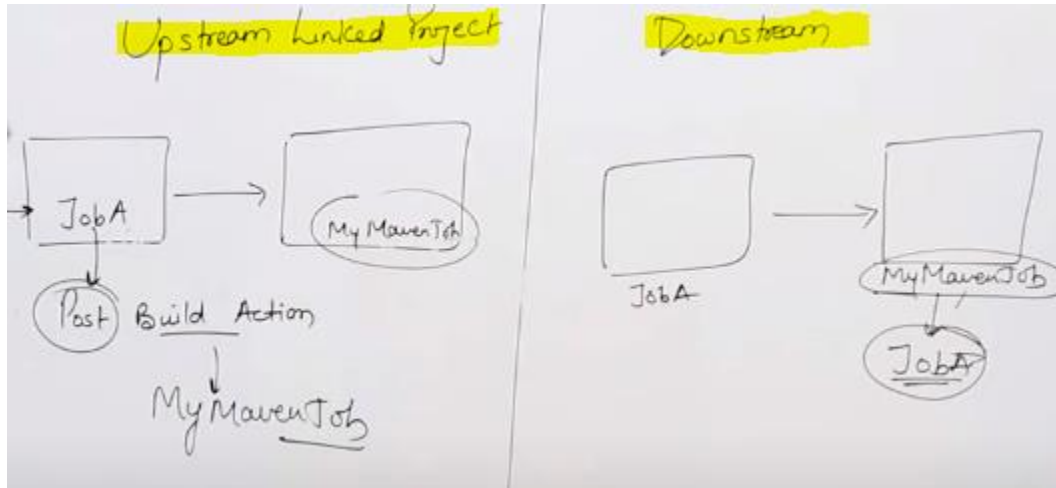


Jenkins-Linked Projects, Views, User Management and Master Slave Concept

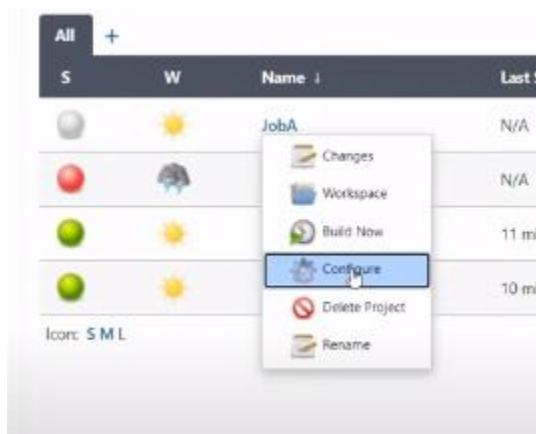
Linked Projects:

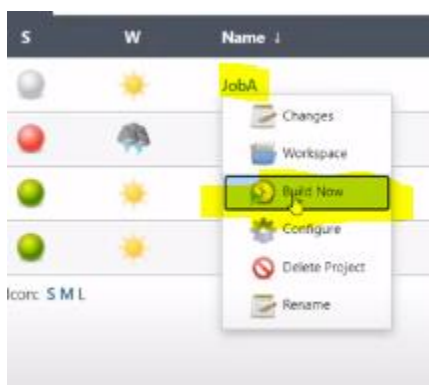
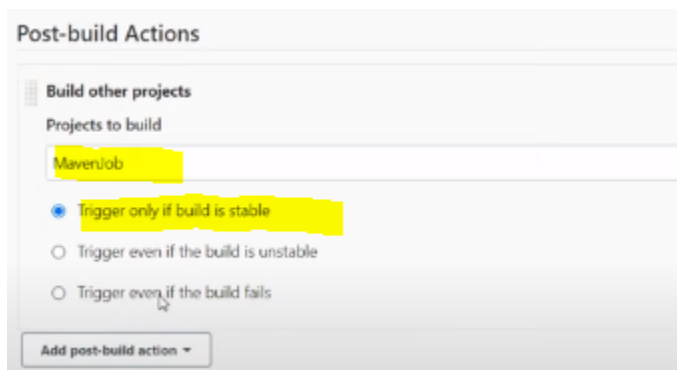
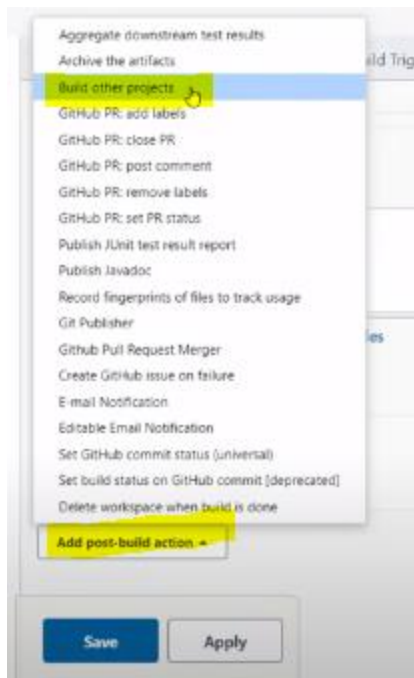


In upstream linked project → In JOB A -post build action (my maven job will be mentioned)

In downstream linked project → In My maven job (Job A will be mentioned)

UPSTREAM PROJECT:





After JobA maven job will run.

DOWNSTREAM PROJECT:

Build Triggers

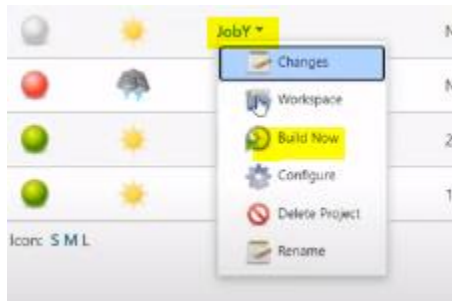
- ☒ Build whenever a SNAPSHOT dependency is built
- ☐ Schedule build when some upstream has no successful builds
- ☐ Trigger builds remotely (e.g., from scripts)
- ☒ Build after other projects are built
- ☐ Build periodically
- ☐ GitHub Branches
- ☐ GitHub Pull Request Builder
- ☐ GitHub Pull Requests
- ☐ GitHub hook trigger for GITScm polling
- ☐ Poll SCM

Build after other projects are built

Projects to watch

JobY

- ☒ Trigger only if build is stable
- ☐ Trigger even if the build is unstable
- ☐ Trigger even if the build fails



VIEW

New Item

People

Build History

Edit View

Delete View

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

AllNEW+

S	W	Name	Last Success	Last Failure	Last Duration	
		JobA	11 min - #1	N/A	1.1 sec	
		JobY	4 min 29 sec - #1	N/A	1.1 sec	
		MavenJob	N/A	4 min 18 sec - #4	9.9 sec	

Icon: S M L

Legend

Atom feed for all

Atom feed for failures

Atom feed for just latest builds

add description

User Management

JENKINS

Lec-39 → Linked Projects, Views, User Management & Master-Slave Concept

User Management

Go to Jenkins Homepage → Manage Jenkins
→ Manage Users

Create two Users → Bhupinder & Rajput

Now login as → Bhupinder

{ By default, you have all the permissions }

Login as "admin" again

Go to Manage Jenkins → Manage plugins
→ Search "Role-based authorization strategy"
→ Install Without Restart

Go to Jenkins home → Manage Jenkins
→ Configure Global Security
→ Select Role Based Strategy → Save

Login as "Bhupinder" → Access denied

Now, Attach permissions
go to jenkins → Manage Jenkins → Manage
and assign Role → Manage Roles

Role to add → Employee

go to item project → add Developer & Tester

Pattern → dev* test*

Then Assign Roles

User/group to add [Bhupinder] & [Rajput]

Item Roles

user/group	Developer	Tester
Bhupinder	<input checked="" type="checkbox"/>	
Rajput		<input checked="" type="checkbox"/>

The image shows two screenshots of the Jenkins web interface. The top screenshot displays the 'Security' configuration page, where 'Role-based authorization strategy' is selected. The 'Manage Users' link is highlighted. The bottom screenshot shows the 'Users' configuration page, listing the 'admin', 'Bhupinder', and 'Rajput' users. The 'Create User' form is also visible on the left side of the bottom screenshot.

Security

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

Manage Users
Create/delete/modify users that can log in to this Jenkins

Users

These users can log into Jenkins. This is a sub set of [this list](#).

User ID
admin

Create User

Username:
Password:
Confirm password:
Full name:
E-mail address:

Create User

Users

These users can log into Jenkins. This is a sub set of [this list](#).

User ID
admin
Bhupinder
Rajput

Jenkins Dashboard

Search:

log out

Dashboard

New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Lockable Resources

add description

S	W	Name	Last Success	Last Failure	Last Duration
●	☀	JobA	51 min - #1	N/A	1.1 sec
●	☀	JobY	43 min - #1	N/A	1.1 sec
●	☁	MavenJob	N/A	43 min - #4	9.9 sec
●	☀	MyFirstJob	1 hr 4 min - #1	N/A	0.69 sec
●	☀	MySecondJob	1 hr 3 min - #1	N/A	0.61 sec

Jenkins Dashboard

Search:

log out

Dashboard

New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Lockable Resources

add description

S	W	Name	Last Success	Last Failure	Last Duration
●	☀	JobA	50 min - #1	N/A	1.1 sec
●	☀	JobY	43 min - #1	N/A	1.1 sec
●	☁	MavenJob	N/A	42 min - #4	9.9 sec
●	☀	MyFirstJob	1 hr 3 min - #1	N/A	0.69 sec
●	☀	MySecondJob	1 hr 2 min - #1	N/A	0.61 sec

Legend

Atom feed for all

Atom feed for failures

Atom feed for just latest builds

To restrict the permission:

Check File Fingerprint

Manage Jenkins

My Views

Manage Plugins

Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

Installing Plugins/Upgrades

Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

Role-based Authorization Strategy

Success

Loading plugin extensions

Success

[Go back to the top page](#)

(you can start using the installed plugins right away)

☐ Restart Jenkins when installation is complete and no jobs are running

Updates

Available

Installed

Advanced

Install

Name

Role-based Authorization Strategy

Security

Authentication and User Management

Enables user authorization using a Role-Based strategy. It uses regular expressions.

Dashboard
Configure Global Security

- ☐ LDAP
- ☐ None
- Authorization
 - ☐ Anyone can do anything
 - ☐ GitHub Committer Authorization Strategy
 - ☐ Legacy mode
 - ☐ Logged-in users can do anything
 - ☐ Matrix-based security
 - ☐ Project-based Matrix Authorization Strategy
 - ☒ Role-based Strategy

Access Denied

bhupinder is missing the Overall/Read permission

Manage and Assign Roles

Handle permissions by creating roles and assigning them to users/groups

Manage Roles

Global roles

Role	Overall	Credentials	Agent
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Dashboard
Manage and Assign Roles

Role to add

Employee

Add

Item roles

Role	Pattern	Credentials	Job	Run	SCM	Lockable Resources
admin		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

New Item
People
Build History
Project Relationship
Check File Fingerprint

Assign Roles

Global roles

User/group	Employee	admin
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Anonymous	<input type="checkbox"/>	<input type="checkbox"/>

Item roles

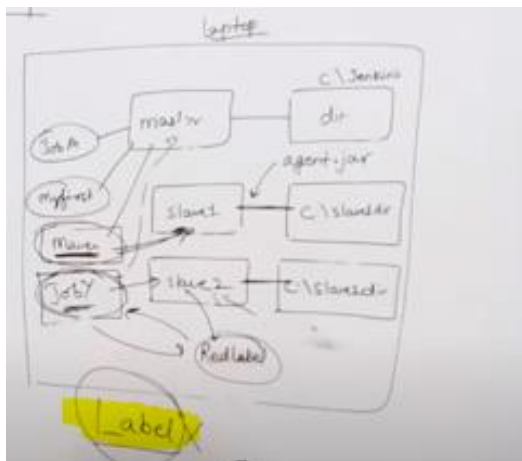
User/group developer tester

☒ Anonymous ☐ ☐

User/group to add

Bhupinder

MASTER SLAVE CONCEPT:



Dashboard


Build Executor Status


1 idle


2 idle

Jenkins 2.277.2 core and libraries:
Denial of service vulnerability in bundled Jetty

System Configuration






**Configure System**
Configure global settings and paths.

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

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

Jenkins

search

 admin  log out

Dashboard


Nodes

Back to Dashboard

Manage Jenkins

New Node

Configure Clouds

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	master	Windows 10 (amd64)	In sync	167.28 GB	13.25 GB	167.28 GB	0ms
Data obtained		29 min	29 min	29 min	29 min	29 min	29 min

Refresh status

Dashboard

Nodes

Back to Dashboard

Manage Jenkins


New Node

Configure Clouds


Node Monitoring

Node name

Slave1

 **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.



AGENT PATH NEEDS TO BE MENTIONED:

Launch method




Launch agent via execution of command on the master

Launch command

java -jar c:\agent.jar

No launch command specified

Single command to launch an agent program, which controls the agent computer and communicates with the master. Jenkins assumes that the executed program launches the agent . jar program on the correct machine.
A copy of agent . jar can be downloaded from [here](#).
In a simple case, this could be something like `ssh root@slave java -jar ~/bin/agent.jar`.

	master	Windows 10 (amd64)	In sync	167.28 GB	13.25 GB	167.28 GB	0ms	
	Slave1	N/A	N/A	N/A	N/A	N/A	N/A	
Data obtained		11 ms	11 ms	2 ms	7 ms	2 ms	1 ms	

Refresh status

REFRESH THE PAGE AND AGENT WILL BE ONLINE:

Jenkins Dashboard - Nodes

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
master	Windows 10 (amd64)	In sync	167.19 GB	10.50 GB	167.19 GB	0ms	
Slave1	Windows 10 (amd64)	In sync	N/A	10.55 GB	167.19 GB	173ms	
Data obtained			5.4 sec	5.4 sec	5.4 sec	4.8 sec	5.3 sec

Refresh status

Jenkins Dashboard - Nodes

Node name: Slave2

☐ Permanent Agent

☒ Copy Existing Node

Copy from: Slave1

Build Executor Status

- master
 - 1 MavenJob #5
 - 2 MySecondJob #2
- Slave1
 - 1 Idle
- Slave2
 - 1 Idle

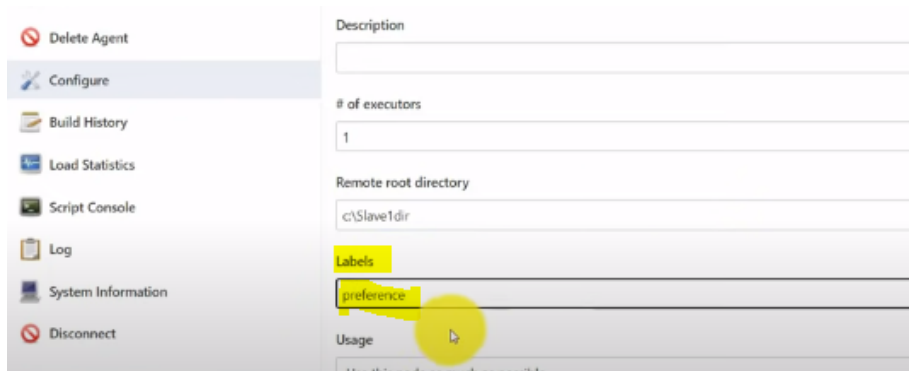
ANYONE CAN RUN THE JOB AS MASTER, SLAVE1, SLAVE2 IF YOU WANT PERTICULAR JOBS RUN BY PERTICULAR SLAVES OR MASTER YOU NEED TO LABEL IT.

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
master	Windows 10 (amd64)	In sync	167.14 GB	7.47 GB	167.14 GB	0ms	
Slave1	Windows 10 (amd64)	In sync	167.14 GB	7.47 GB	167.14 GB	66ms	
Slave2	Windows 10 (amd64)	In sync	167.14 GB	7.47 GB	167.14 GB	92ms	
Data obtained			58 sec	58 sec	58 sec	58 sec	58 sec

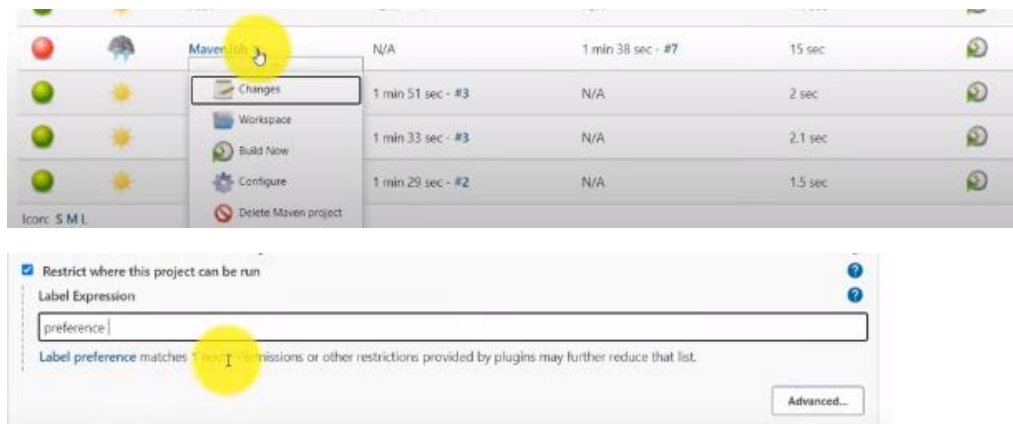
Refresh status

Context menu for Slave1:

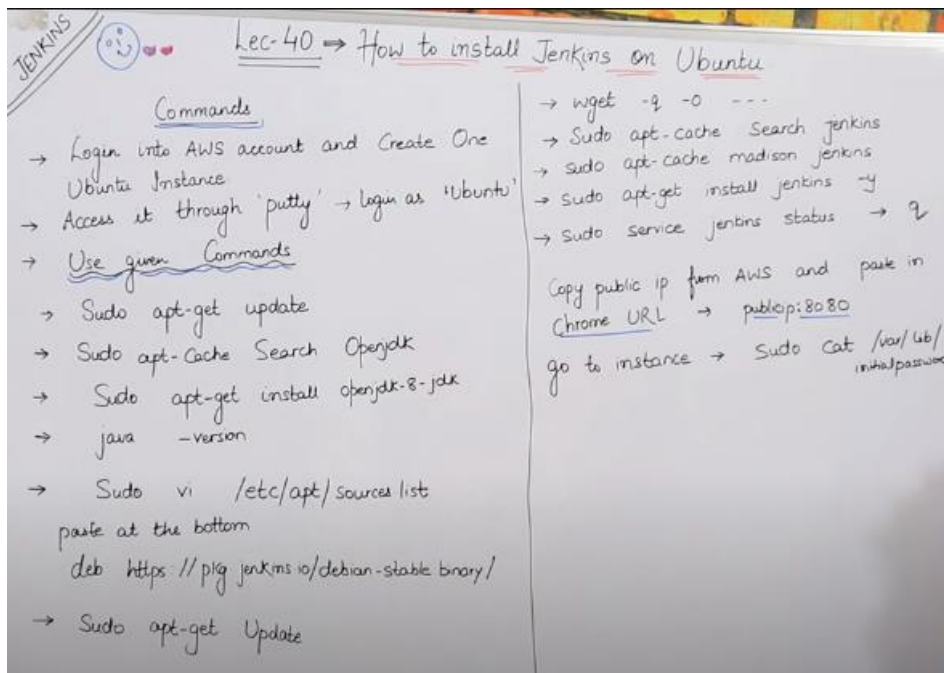
- Delete Agent
- Configure
- Build history
- Load Statistics
- Script Console



GOTO JOB→CONFIGURE→RESTRICIT WHERE THIS PROJECT CAN BE RUN



Jenkins installation on Linux



	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input type="checkbox"/>	jenkins-server	i-009f80e81255018a2	Terminated	t2.micro	-	No alarms	ap-south-1a	-
<input checked="" type="checkbox"/>	New-Server	i-09f9a70f1a105e6a32	Pending	t2.micro	-	No alarms	ap-south-1a	ec2-15-206-79-5

ubuntu@ip-172-31-37-70: ~

```
ubuntu@ip-172-31-37-70:~$ sudo apt-get update
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu focal InRelease [265 kB]
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [109 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease [1
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease
```

ubuntu@ip-172-31-37-70: ~

```
ubuntu@ip-172-31-37-70:~$ sudo apt-get install openjdk-8-jdk
```

ubuntu@ip-172-31-37-70: ~

```
ubuntu@ip-172-31-37-70:~$ java -version
openjdk version "1.8.0_292"
OpenJDK Runtime Environment (build 1.8.0_292-8u292-b10-0ubuntu1~20.04-b10)
OpenJDK 64-Bit Server VM (build 25.292-b10, mixed mode)
```

```
wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -
```

Then add the following entry in your `/etc/apt/sources.list`:

```
deb https://pkg.jenkins.io/debian-stable binary/
```

Update your local package index, then finally install Jenkins:

```
sudo apt-get update
sudo apt-get install jenkins
```

The apt packages were signed using this key:

```
pub  rsa4096 2020-03-30 [SC] [expires: 2023-03-30]
     62A07568FD780C377CF24BA8FCE32E745F2C3D5
uid           Jenkins Project
sub  rsa4096 2020-03-30 [E] [expires: 2023-03-30]
```