

# DeVops Jenkins

## Assignment 2

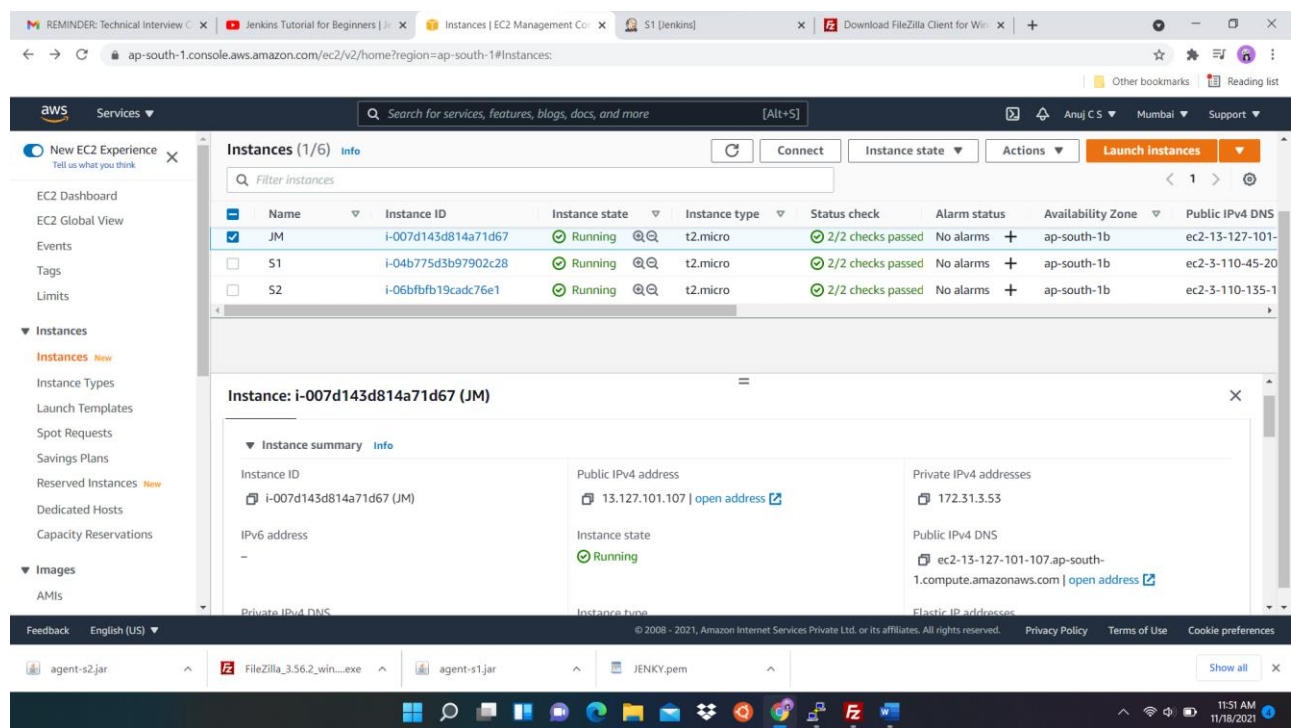
BY:

ANKITHA S MADANBHAVI (18BCS008)

ANUJ C SHIRAGAVE(18BCS009)

Assignment-02: Jenkins Master Slave pipeline intellipaat.

Step-1: Firstly, we are creating three aws ec2 instances and naming them as JM, S1, S2.



The screenshot shows the AWS Management Console interface. The top navigation bar includes the AWS logo, a search bar, and user information. The left sidebar contains navigation links for various AWS services. The main content area displays a list of EC2 instances under the heading 'Instances (1/6)'. The list includes three instances: JM, S1, and S2, all of which are in the 'Running' state. Below the list, a detailed view for instance JM is shown, including its ID, public IP address, private IP address, and instance type.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
JM	i-007d143d814a71d67	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1b	ec2-13-127-101-
S1	i-04b775d3b97902c28	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1b	ec2-3-110-45-20
S2	i-06bfbfb19cad76e1	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1b	ec2-3-110-135-1

**Instance: i-007d143d814a71d67 (JM)**

Instance summary	Info
Instance ID i-007d143d814a71d67 (JM)	Public IPv4 address 13.127.101.107   <a href="#">open address</a>
IPv6 address -	Instance state Running
Private IPv4 DNS -	Private IPv4 addresses 172.31.3.53
Private IPv4 DNS -	Public IPv4 DNS ec2-13-127-101-107.ap-south-1.compute.amazonaws.com   <a href="#">open address</a>
Private IPv4 DNS -	Elastic IP addresses -

Step-2: Creating three elastic IP addresses and associating these elastic IP addresses to the ec2 instances.

Step-3: Install Jenkins on jenkins\_master ec2 instance.

Commands to install Jenkins on ubuntu ec2 instance.

```
sudo apt-get update -y
sudo apt-get install openjdk-8-jdk
wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -
sudo sh -c 'echo deb https://pkg.jenkins.io/debian-stable binary/ >
/etc/apt/sources.list.d/jenkins.list' sudo apt-get update
apt-get install jenkins
sudo service jenkins start
```



## Step-5:

Download agent.jar from Slave-1 node and using Filezilla, transfer that file to slave-1 ec2 instance.

The screenshot shows the Jenkins web interface for 'Agent Slave-1'. The sidebar on the left contains links: 'Back to List', 'Status', 'Delete Agent', 'Configure', 'Build History', 'Load Statistics', and 'Log'. The main panel displays instructions for connecting the agent to Jenkins, including a 'Launch' button and a command line snippet. Below this, it says 'Projects tied to Slave-1' with 'None' listed. At the bottom, there's a status bar showing 'REST API' and 'Jenkins 2.303.3'.

Uploading agent file .jar file on slave 1 using Filezilla

The screenshot shows the FileZilla interface with a local site connected to 'sftp://ubuntu@3.110.45.208'. The local site shows the directory structure of the user's home directory. The remote site shows the directory structure of the remote server. The 'agent-s1.jar' file is highlighted in the remote site list.

Filename	Filesize	Filetype	Last modified	Permissions	Owner/Group
..					
.cache		File folder	11/18/2021...	drwx-----	ubuntu u..
.ssh		File folder	11/18/2021...	drwx-----	ubuntu u..
.bash_logout	220	BASHRC...	2/25/2020...	-rw-r--r--	ubuntu u..
.bashrc	3,771	BASHRC...	2/25/2020...	-rw-r--r--	ubuntu u..
.profile	807	PROFILE...	2/25/2020...	-rw-r--r--	ubuntu u..
.sudo_as_admin...	0	SUDO_A...	11/18/2021...	-rw-r--r--	ubuntu u..
agent-s1.jar	1,507,813	Executa...	11/18/2021...	-rw-rw-r--	ubuntu u..
secret-file	0	File	11/18/2021...	-rw-rw-r--	ubuntu u..

## Step-5:

Download agent.jar from Slave-2 node and using Filezilla, transfer that file to slave-2 ec2 instance.

The screenshot shows the Jenkins web interface in a browser. The address bar indicates the URL is 13.127.101.107:8080/computer/Slave-2/. The Jenkins dashboard is visible, with the 'Slave-2' node selected. The 'Agent Slave-2' configuration page is shown, providing instructions on how to connect the agent to Jenkins. It includes a 'Launch agent from browser' button and a 'Run from agent command line' section with the following commands:

```
java -jar agent.jar -jnlpUrl http://13.127.101.107:8080/computer/Slave-2/jenkins-agent.jnlp -secret e4f1ee00914a17f51e7bd4964da3c2f929ed34fb368f25a035e8263aaccd7a -workDir "C:\Users\Anuj\Desktop\DeVops"
```

Below the commands, it says 'Run from agent command line, with the secret stored in a file:' and provides another set of commands:

```
echo e4f1ee00914a17f51e7bd4964da3c2f929ed34fb368f25a035e8263aaccd7a > secret-file
java -jar agent.jar -jnlpUrl http://13.127.101.107:8080/computer/Slave-2/jenkins-agent.jnlp -secret @secret-file -workDir "C:\Users\Anuj\Desktop\DeVops"
```

The 'Projects tied to Slave-2' section shows 'None'. The bottom of the browser window shows the taskbar with various application icons.

Uploading agent file .jar file on slave 2 using Filezilla

The screenshot shows the FileZilla interface. The 'Local site' is C:\Users\Anuj\ and the 'Remote site' is /home/ubuntu. The 'agent-s2.jar' file is selected in the local site pane. The 'Remote site' pane shows the directory structure of the remote machine, including files like .cache, .ssh, .bash\_logout, .bashrc, .profile, .sudo\_as\_admin..., and agent-s2.jar. The 'agent-s2.jar' file is highlighted in blue, indicating it is selected for upload. The status bar at the bottom shows 'Selected 1 file. Total size: 1507,813 bytes'.

Step-6:

Run the command on the ec2 instances so that we can connect the nodes to the Jenkins.

SLAVE-1 CONNECTED

The image shows a terminal window and a web browser window. The terminal window displays the command to start the Jenkins agent and the subsequent logs showing the agent connecting to the Jenkins server. The browser window shows the Jenkins dashboard with the 'slave-1' node connected.

```
ubuntu@ip-172-31-12-71:~$ java -jar agent.jar -jnlpUrl http://13.127.101.107:8080/computer/slave-1/jenkins-agent.jnlp -secret e1c6e4e0ba16e700c66653012e2fb6ecfb0b8c9390dc227a9221811534127a3 -workDir "/home/ubuntu/jenkins"
Nov 18, 2021 7:22:54 AM org.jenkinsci.remoting.engine.WorkDirManager initializeW
orkDir
INFO: Using /home/ubuntu/jenkins/remoting as a remoting work directory
Nov 18, 2021 7:22:54 AM org.jenkinsci.remoting.engine.WorkDirManager setupLoggin
g
INFO: Both error and output logs will be printed to /home/ubuntu/jenkins/remotin
g
Nov 18, 2021 7:22:55 AM hudson.remoting.jnlp.Main createEngine
INFO: Setting up agent: slave-1
Nov 18, 2021 7:22:55 AM hudson.remoting.jnlp.Main$CuiListener <init>
INFO: Jenkins agent is running in headless mode.
Nov 18, 2021 7:22:55 AM hudson.remoting.Engine startEngine
INFO: Using Remoting version: 4.10.1
Nov 18, 2021 7:22:55 AM org.jenkinsci.remoting.engine.WorkDirManager initializeW
orkDir
INFO: Using /home/ubuntu/jenkins/remoting as a remoting work directory
Nov 18, 2021 7:22:55 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Locating server among [http://13.127.101.107:8080/]
Nov 18, 2021 7:22:55 AM org.jenkinsci.remoting.engine.JnlpAgentEndpointResolver
resolve
INFO: Remoting server accepts the following protocols: [JNLP4-connect, Ping]
Nov 18, 2021 7:22:55 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Agent discovery successful
Agent address: 13.127.101.107
Agent port: 38463
Identity: f8:7e:6c:89:fd:c8:68:7e:f9:36:46:5f:cc:23:20:62
Nov 18, 2021 7:22:55 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Handshaking
Nov 18, 2021 7:22:55 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Connecting to 13.127.101.107:38463
Nov 18, 2021 7:22:55 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Trying protocol: JNLP4-connect
Nov 18, 2021 7:22:55 AM org.jenkinsci.remoting.protocol.impl.BIONetworkLayer$Rea
dex run
INFO: Waiting for ProtocolStack to start.
Nov 18, 2021 7:22:55 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Remote identity confirmed: f8:7e:6c:89:fd:c8:68:7e:f9:36:46:5f:cc:23:20:62
Nov 18, 2021 7:22:56 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Connected
```

The browser window shows the Jenkins dashboard. The top navigation bar includes the Jenkins logo, a search bar, and the user name 'ANUJ C SHIRAGAVE' with a 'log out' button. The main content area shows the 'Dashboard' view with a list of nodes. The 'slave-1' node is highlighted, showing its status as 'connected'. The 'Projects tied to slave-1' section shows 'None'. The 'Build Executor Status' section shows '1 idle'.

## SLAVE-2 CONNECTED.

```
ubuntu@ip-172-31-15-239: ~
ubuntu@ip-172-31-15-239:~$ ls
agent-s2.jar  agent.jar
ubuntu@ip-172-31-15-239:~$ java -jar agent.jar -jnlpUrl http://13.127.101.107:8080/computer/slave-2/jenkins-agent.jnlp -secret 6bb4f5e4ac4be0a950f706759aa001fd3cbb74d47c0695b8bcb10ada35b36baa -workDir "/home/ubuntu/jenkins"
Nov 18, 2021 7:24:17 AM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/ubuntu/jenkins/remoting as a remoting work directory
Nov 18, 2021 7:24:17 AM org.jenkinsci.remoting.engine.WorkDirManager setupLogging
INFO: Both error and output logs will be printed to /home/ubuntu/jenkins/remoting
Nov 18, 2021 7:24:17 AM hudson.remoting.jnlp.Main createEngine
INFO: Setting up agent: slave-2
Nov 18, 2021 7:24:17 AM hudson.remoting.jnlp.Main$CuiListener <init>
INFO: Jenkins agent is running in headless mode.
Nov 18, 2021 7:24:17 AM hudson.remoting.Engine startEngine
INFO: Using Remoting version: 4.10.1
Nov 18, 2021 7:24:17 AM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/ubuntu/jenkins/remoting as a remoting work directory
Nov 18, 2021 7:24:18 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Locating server among [http://13.127.101.107:8080/]
Nov 18, 2021 7:24:18 AM org.jenkinsci.remoting.engine.JnlpAgentEndpointResolver resolve
INFO: Remoting server accepts the following protocols: [JNLP4-connect, Ping]
Nov 18, 2021 7:24:18 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Agent discovery successful
Agent address: 13.127.101.107
Agent port: 38463
Identity: f8:7e:6c:89:fd:c8:68:7e:f9:36:46:5f:cc:23:20:62
Nov 18, 2021 7:24:18 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Handshaking
Nov 18, 2021 7:24:18 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Connecting to 13.127.101.107:38463
Nov 18, 2021 7:24:18 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Trying protocol: JNLP4-connect
Nov 18, 2021 7:24:18 AM org.jenkinsci.remoting.protocol.impl.BIONetworkLayer$Reader run
INFO: Waiting for ProtocolStack to start.
Nov 18, 2021 7:24:18 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Remote identity confirmed: f8:7e:6c:89:fd:c8:68:7e:f9:36:46:5f:cc:23:20:62
Nov 18, 2021 7:24:19 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Connected
```

REMINDER: Technical | Jenkins Tutorial for B | Instances | EC2 Manas | slave-2 [Jenkins] | Download FileZilla C | (1) WhatsApp | Jenkins assignment 2 |

Not secure | 13.127.101.107:8080/computer/slave-2/ | Other bookmarks | Reading list

# Jenkins

search

Dashboard > Nodes > slave-2

Back to List

Status

Delete Agent

Configure

Build History

Load Statistics

Script Console

Log

System Information

Disconnect

## Agent slave-2

Agent is connected.

### Projects tied to slave-2

None

Mark this node temporarily offline


Build Executor Status






1 idle

REMINDER: Technical Inter...Jenkins Tutorial for Begin...Instances | EC2 Managemen...Nodes [Jenkins]WhatsAppJenkins assignment 2+


Not secure | 13.127.101.107:8080/computer/


Other bookmarksReading list


Jenkins


ANUJ C SHIRAGAVE log out


DashboardNodes

Back to Dashboard

Manage Jenkins

New Node


Configure Clouds

Node Monitoring

Build Queue


No builds in the queue.

Build Executor Status


\_master







1 idle

2 idle


\_slave-1

1 idle




\_slave-2

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	master	Linux (amd64)	In sync	4.93 GB	 0 B	4.93 GB	0ms
	slave-1	Linux (amd64)	In sync	5.64 GB	 0 B	5.64 GB	46ms
	slave-2	Linux (amd64)	In sync	5.64 GB	 0 B	5.64 GB	60ms
Data obtained		4 min 51 sec	4 min 51 sec	4 min 51 sec	4 min 51 sec	4 min 51 sec	4 min 51 sec

Refresh status



12:59 PM11/18/2021





## Step-7:

Install docker on Slave-1(orange) and Slave-2(green) ec2 instances.

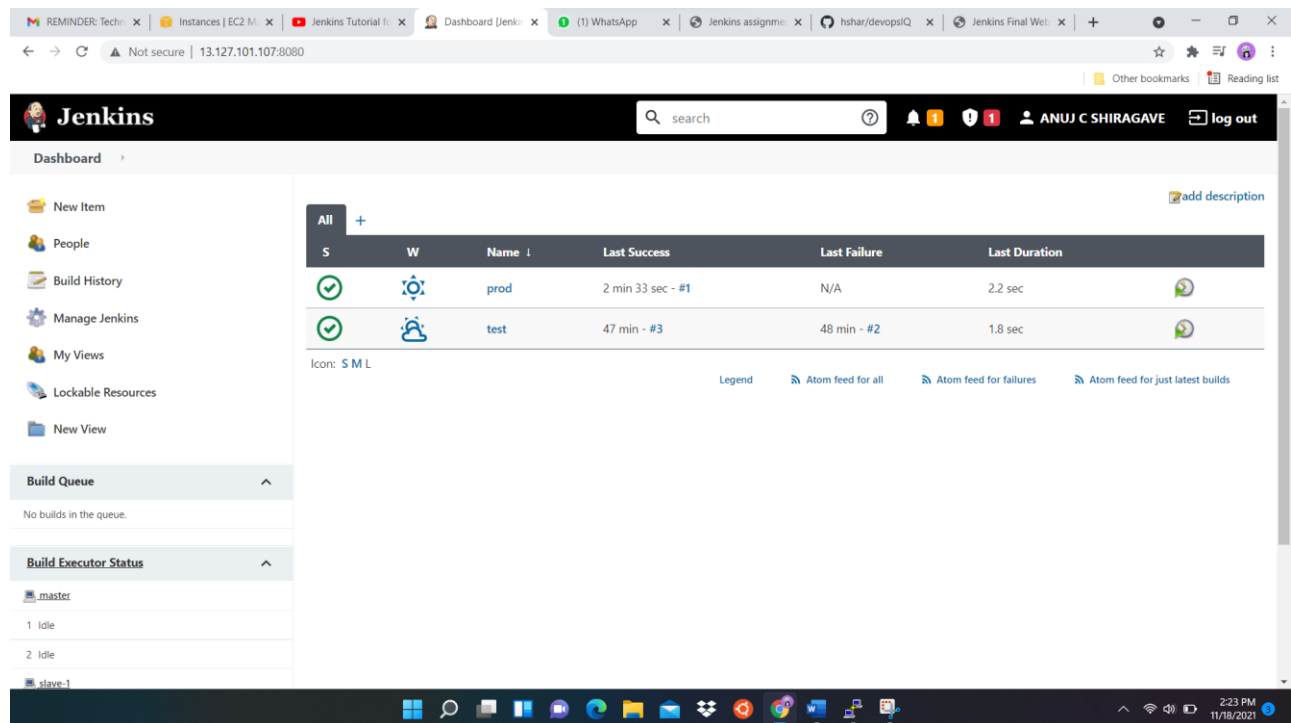
```
ubuntu@ip-172-31-15-239:~$ sudo apt-get install docker.io
Preparing to unpack .../3-containerd_1.5.5-0ubuntu3-20.04.1_amd64.deb ...
Unpacking containerd (1.5.5-0ubuntu3-20.04.1) ...
Selecting previously unselected package dns-root-data.
Preparing to unpack .../4-dns-root-data_2019052802_all.deb ...
Unpacking dns-root-data (2019052802) ...
Selecting previously unselected package libidn11:amd64.
Preparing to unpack .../5-libidn11_1.33-2.2ubuntu2_amd64.deb ...
Unpacking libidn11:amd64 (1.33-2.2ubuntu2) ...
Selecting previously unselected package dnsmasq-base.
Preparing to unpack .../6-dnsmasq-base_2.80-1.1ubuntu1.4_amd64.deb ...
Unpacking dnsmasq-base (2.80-1.1ubuntu1.4) ...
Selecting previously unselected package docker.io.
Preparing to unpack .../7-docker.io_20.10.7-0ubuntu5-20.04.2_amd64.deb ...
Unpacking docker.io (20.10.7-0ubuntu5-20.04.2) ...
Selecting previously unselected package ubuntu-fan.
Preparing to unpack .../8-ubuntu-fan_0.12.13_all.deb ...
Unpacking ubuntu-fan (0.12.13) ...
Setting up runc (1.0.1-0ubuntu2-20.04.1) ...
Setting up dns-root-data (2019052802) ...
Setting up libidn11:amd64 (1.33-2.2ubuntu2) ...
Setting up bridge-utils (1.6-2ubuntu1) ...
Setting up pigz (2.4-1) ...
Setting up containerd (1.5.5-0ubuntu3-20.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service -> /lib/systemd/system/containerd.service.
Setting up docker.io (20.10.7-0ubuntu5-20.04.2) ...
Adding group 'docker' (GID 119) ...
Done.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service -> /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket -> /lib/systemd/system/docker.socket.
Setting up dnsmasq-base (2.80-1.1ubuntu1.4) ...
Setting up ubuntu-fan (0.12.13) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service -> /lib/systemd/system/ubuntu-fan.service.
Processing triggers for systemd (245.4-4ubuntu3.13) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for dbus (1.12.16-2ubuntu2.1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
ubuntu@ip-172-31-15-239:~$ docker --version
Docker version 20.10.7, build 20.10.7-0ubuntu5-20.04.2
ubuntu@ip-172-31-15-239:~$
```

```
ubuntu@ip-172-31-12-71:~$ sudo apt-get install docker.io
Preparing to unpack .../3-containerd_1.5.5-0ubuntu3-20.04.1_amd64.deb ...
Unpacking containerd (1.5.5-0ubuntu3-20.04.1) ...
Selecting previously unselected package dns-root-data.
Preparing to unpack .../4-dns-root-data_2019052802_all.deb ...
Unpacking dns-root-data (2019052802) ...
Selecting previously unselected package libidn11:amd64.
Preparing to unpack .../5-libidn11_1.33-2.2ubuntu2_amd64.deb ...
Unpacking libidn11:amd64 (1.33-2.2ubuntu2) ...
Selecting previously unselected package dnsmasq-base.
Preparing to unpack .../6-dnsmasq-base_2.80-1.1ubuntu1.4_amd64.deb ...
Unpacking dnsmasq-base (2.80-1.1ubuntu1.4) ...
Selecting previously unselected package docker.io.
Preparing to unpack .../7-docker.io_20.10.7-0ubuntu5-20.04.2_amd64.deb ...
Unpacking docker.io (20.10.7-0ubuntu5-20.04.2) ...
Selecting previously unselected package ubuntu-fan.
Preparing to unpack .../8-ubuntu-fan_0.12.13_all.deb ...
Unpacking ubuntu-fan (0.12.13) ...
Setting up runc (1.0.1-0ubuntu2-20.04.1) ...
Setting up dns-root-data (2019052802) ...
Setting up libidn11:amd64 (1.33-2.2ubuntu2) ...
Setting up bridge-utils (1.6-2ubuntu1) ...
Setting up pigz (2.4-1) ...
Setting up containerd (1.5.5-0ubuntu3-20.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service -> /lib/systemd/system/containerd.service.
Setting up docker.io (20.10.7-0ubuntu5-20.04.2) ...
Adding group 'docker' (GID 119) ...
Done.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service -> /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket -> /lib/systemd/system/docker.socket.
Setting up dnsmasq-base (2.80-1.1ubuntu1.4) ...
Setting up ubuntu-fan (0.12.13) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service -> /lib/systemd/system/ubuntu-fan.service.
Processing triggers for systemd (245.4-4ubuntu3.13) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for dbus (1.12.16-2ubuntu2.1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
ubuntu@ip-172-31-12-71:~$ docker --version
Docker version 20.10.7, build 20.10.7-0ubuntu5-20.04.2
ubuntu@ip-172-31-12-71:~$
```

## Step 8:

Create two jobs (test for Slave-1 and prod for Slave-2)

In configure, we are setting source code management as git and passing our Github repo link, in build we are selecting execute shell and writing some commands to run.



The screenshot shows the Jenkins Dashboard interface. The top navigation bar includes the Jenkins logo, a search bar, and user information for ANUJ C SHIRAGAVE. The left sidebar contains links to various dashboard sections: New Item, People, Build History, Manage Jenkins, My Views, Lockable Resources, and New View. The main content area displays a table of jobs with columns for Status (S), Webhook (W), Name, Last Success, Last Failure, and Last Duration. Two jobs are listed: 'prod' and 'test'. The 'prod' job has a status of 'Success' (green checkmark) and a duration of 2.2 sec. The 'test' job has a status of 'Success' (green checkmark) and a duration of 1.8 sec. Below the table, there are links for 'Icon: S M L' and 'Legend'. The bottom status bar shows the time as 2:23 PM on 11/18/2021.

S	W	Name	Last Success	Last Failure	Last Duration
✓	⚙️	prod	2 min 33 sec - #1	N/A	2.2 sec
✓	⚙️	test	47 min - #3	48 min - #2	1.8 sec

## Step 9: Build Test job.

The image shows two screenshots of the Jenkins web interface. The top screenshot displays the 'Console Output' for a build job named 'test'. The output shows the build process starting with cloning a repository from GitHub, fetching upstream changes, and checking out a specific revision. The build is successful, and the console output is visible in a scrollable area. The bottom screenshot shows the 'Project test' page in Jenkins. The left sidebar contains navigation links such as 'Back to Project', 'Status', 'Changes', 'Console Output', 'View as plain text', 'Edit Build Information', 'Delete build #1', and 'Git Build Data'. The main content area displays the 'Project test' details, including a 'Workspace' section, 'Recent Changes', and 'Permalinks' for various builds. The 'Build History' section shows a list of builds with their status and timestamps.

**Console Output**

```
Started by user ANUJ C SHIRAGAVE
Running as SYSTEM
Building remotely on slave-1 in workspace /home/ubuntu/workspace/test
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/hshar/devopsIQ.git
> git init /home/ubuntu/workspace/test # timeout=10
Fetching upstream changes from https://github.com/hshar/devopsIQ.git
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git fetch --tags --force --progress -- https://github.com/hshar/devopsIQ.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/hshar/devopsIQ.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision cc26380b7f45bfcf31831fc4f4f6e438fd6e2ab2 (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f cc26380b7f45bfcf31831fc4f4f6e438fd6e2ab2 # timeout=10
Commit message: "Set up CI with Azure Pipelines"
First time build. Skipping changelog.
Finished: SUCCESS
```

**Project test**

add description  
Disable Project

Workspace  
Recent Changes

**Permalinks**

- Last build (#2), 1 min 18 sec ago
- Last stable build (#1), 19 min ago
- Last successful build (#1), 19 min ago
- Last failed build (#2), 1 min 18 sec ago
- Last unsuccessful build (#2), 1 min 18 sec ago
- Last completed build (#2), 1 min 18 sec ago

**Build History** trend ^

Build	Status	Timestamp
#3	Success	Nov 18, 2021 8:06 AM
#2	Failure	Nov 18, 2021 8:05 AM
#1	Success	Nov 18, 2021 7:47 AM

## Step 10: Build prod job.

The image shows two screenshots of the Jenkins web interface. The top screenshot displays the 'Console Output' for the 'prod' job, showing the build process initiated by user 'ANUJ C SHIRAGAVE'. The output includes commands for fetching changes from a remote Git repository, checking out a specific revision, and building the Docker image 'production'. The bottom screenshot shows the 'Project prod' overview page, which includes a sidebar with navigation options like 'Status', 'Changes', 'Workspace', 'Build Now', 'Configure', 'Delete Project', 'GitHub', and 'Rename'. The main content area shows 'Permalinks' for the latest build and a 'Build History' table with one entry for build #1 on Nov 18, 2021 at 8:51 AM.

**Jenkins Console Output**

```
Started by user ANUJ C SHIRAGAVE
Running as SYSTEM
Building remotely on slave-2 in workspace /home/ubuntu/jenkins/workspace/prod
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /home/ubuntu/jenkins/workspace/prod/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/hshar/devopsIQ.git # timeout=10
Fetching upstream changes from https://github.com/hshar/devopsIQ.git
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git fetch --tags --force --progress -- https://github.com/hshar/devopsIQ.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision cc26380b7f45bfcf31831fc4f4f6e438fd6e2ab2 (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f cc26380b7f45bfcf31831fc4f4f6e438fd6e2ab2 # timeout=10
Commit message: "Set up CI with Azure Pipelines"
First time build. Skipping changelog.
[prod] $ /bin/sh -xe /tmp/jenkins5802493792001261148.sh
+ sudo docker ps -a -q
+ sudo docker rm -f 0bb7da7df03e
0bb7da7df03e
+ sudo docker build /home/ubuntu/jenkins/workspace/prod -t production
Sending build context to Docker daemon 24.35MB

Step 1/2 : FROM hshar/webapp
----> 0c1f535ed8
```

**Project prod**

add description  
Disable Project

Workspace  
Recent Changes

**Permalinks**

- Last build (#1), 3 min 26 sec ago
- Last stable build (#1), 3 min 26 sec ago
- Last successful build (#1), 3 min 26 sec ago
- Last completed build (#1), 3 min 26 sec ago

**Build History** trend ^

#	Build	Time
1	#1	Nov 18, 2021 8:51 AM

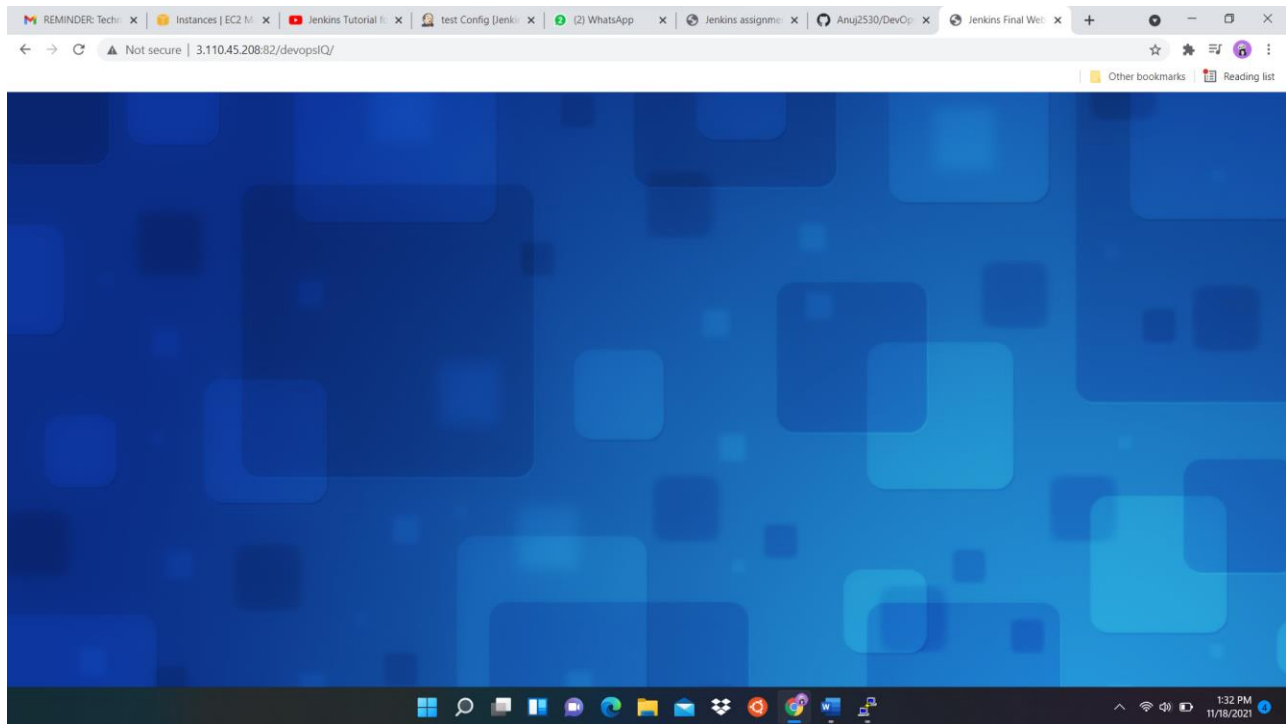
Atom feed for all  
Atom feed for failures

13.127.101.107:8080/job/prod/#

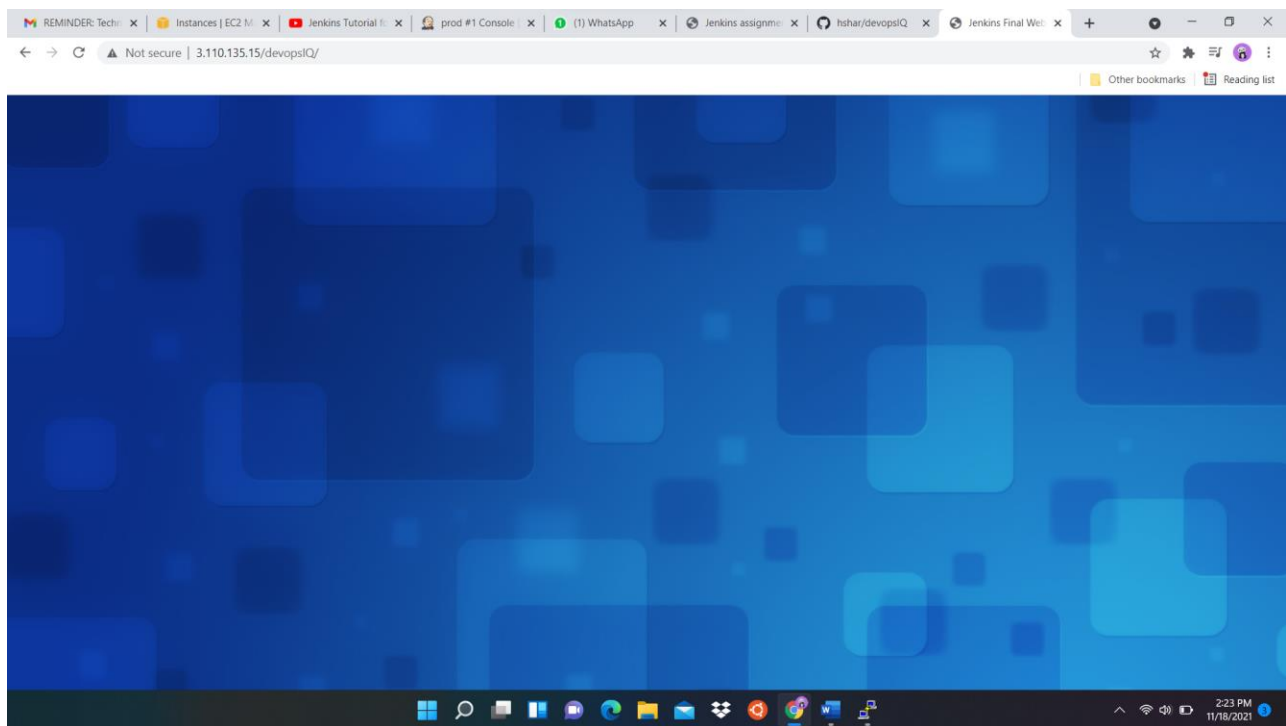
## Step-11: Creating the pipeline

The screenshot displays the Jenkins 'Build Pipeline: CICD' dashboard. At the top, there's a breadcrumb navigation: 'Dashboard > CICD'. Below this, the title 'Build Pipeline: CICD' is centered. A row of icons for 'Run', 'History', 'Configure', 'Add Step', 'Delete', and 'Manage' is visible. The main area shows a pipeline graph with two stages: '#4 Test' and '#4 Prod'. Each stage is represented by a green box with a light green header. The '#4 Test' stage shows a build log entry: 'Nov 4, 2021 7:40:43 AM', '3.5 sec', and 'jenkins@10'. The '#4 Prod' stage shows a build log entry: 'Nov 4, 2021 7:42:14 AM', '1.8 sec', and 'jenkins@10'. A green arrow points from the '#4 Test' stage to the '#4 Prod' stage. On the left, a 'Pipeline' sidebar shows a list of pipelines, with '#4' selected. At the bottom, a Windows taskbar is visible with the search bar, task view, and various application icons. The system tray shows the date and time: '11/4/2021 1:14 PM'. The Jenkins version '2.303.2' is displayed in the top right corner.

Step-12: After successfully building our project we can see our website using slave-1 IP at port 82.



We can see our website using slave-2 IP at port 82.



THANK YOU

