

DEVOPS TRAINING

DAY 1 TASK

Step 1:

(Installing Ubuntu)

Installing and Setting Up WSL with Ubuntu

on Windows 10 Step 1: Enable WSL

Before installing Ubuntu, ensure that WSL is enabled on your Windows system.

Enable WSL Feature

1. Open **PowerShell** as Administrator and run:
2. `wsl --install`

This installs the default Linux distribution and enables necessary components.

3. If WSL is already installed but not enabled, use:
 4. `dism.exe/online/enable-feature/featurename:Microsoft-Windows-Subsystem-Linux /all /norestart`
 5. Enable the `VirtualMachinePlatform` feature (required for WSL2):
 6. `dism.exe/online/enable-feature/featurename:VirtualMachinePlatform/all/norestart`
 7. Restart your computer to apply changes.

Step 2: Install Ubuntu

1. Open **Command Prompt** or **PowerShell** and run:
2. `wsl --install -d Ubuntu`

If the installation fails due to timeout issues, retry the command after shutting down WSL: `wsl --shutdown`

```
wsl--install -dUbuntu
```

3. Once installed, start Ubuntu:
4. wsl.exe -dUbuntu

Step3: Set Up Ubuntu

When Ubuntu runs for the first time, it will ask you to create a new user account.

1. **Enter a username** (must start with a lowercase letter or underscore, and contain only lowercase letters, digits, underscores, and dashes).
2. **Set a password** (enter and confirm the password). If passwords do not match, you will need to retry.
3. Once successful, Ubuntu will be set up and ready to use.

Step4: Verify Installation

To check the installed distributions and their versions:

```
wsl -l -v
```

To verify Ubuntu is running:

```
wsl -d Ubuntu
```

Step5: Configure Ubuntu Update

System Packages

After logging in, update the package list and upgrade installed packages:

```
sudo apt update && sudo apt upgrade -y
```

Set Default WSL Version

To use WSL2 as the default version for future installations:

```
wsl --set-default-version2
```

To check the current WSL version:

```
wsl -l -v
```

To convert an existing installation to WSL2:

```
wsl --set-version Ubuntu2
```

Step6:Enable.hushlogintoSuppressLoginMessage

To disable the daily login message, create a .hushlogin file in your home directory: touch `~/.hushlogin`

Additional Commands

Commands to Restart WSL:

```
wsl--shutdown
```

Uninstall a Distribution:

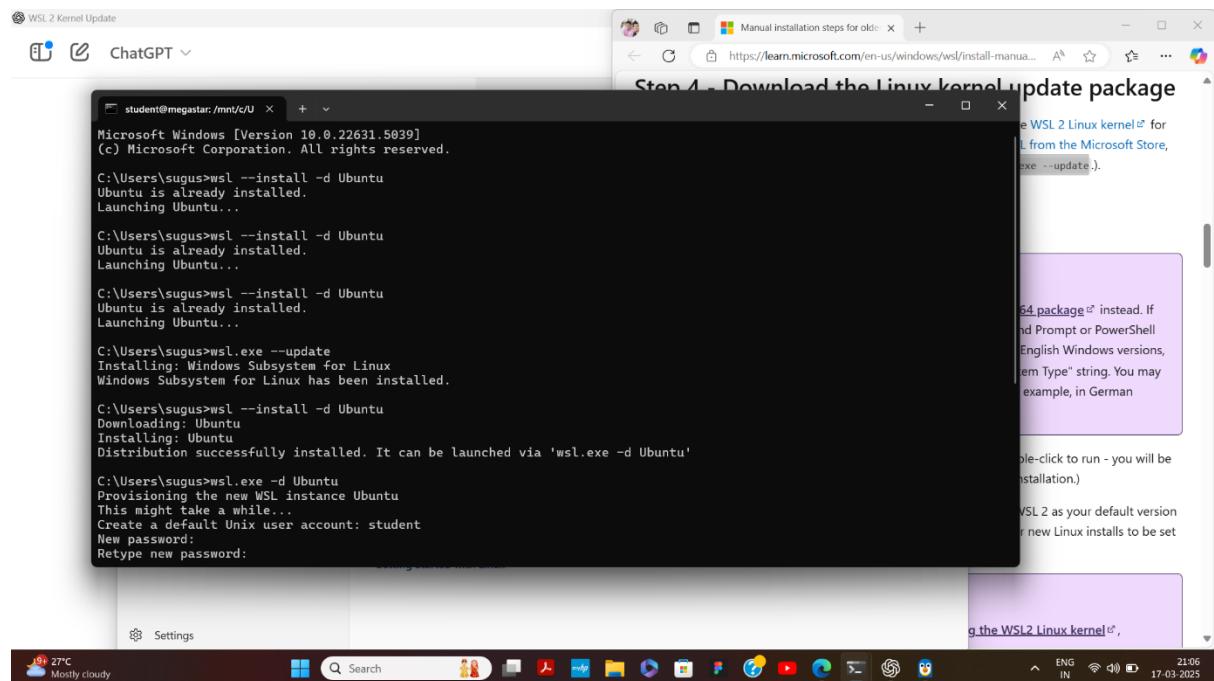
```
wsl--unregisterUbuntu
```

Access Windows Files in WSL:

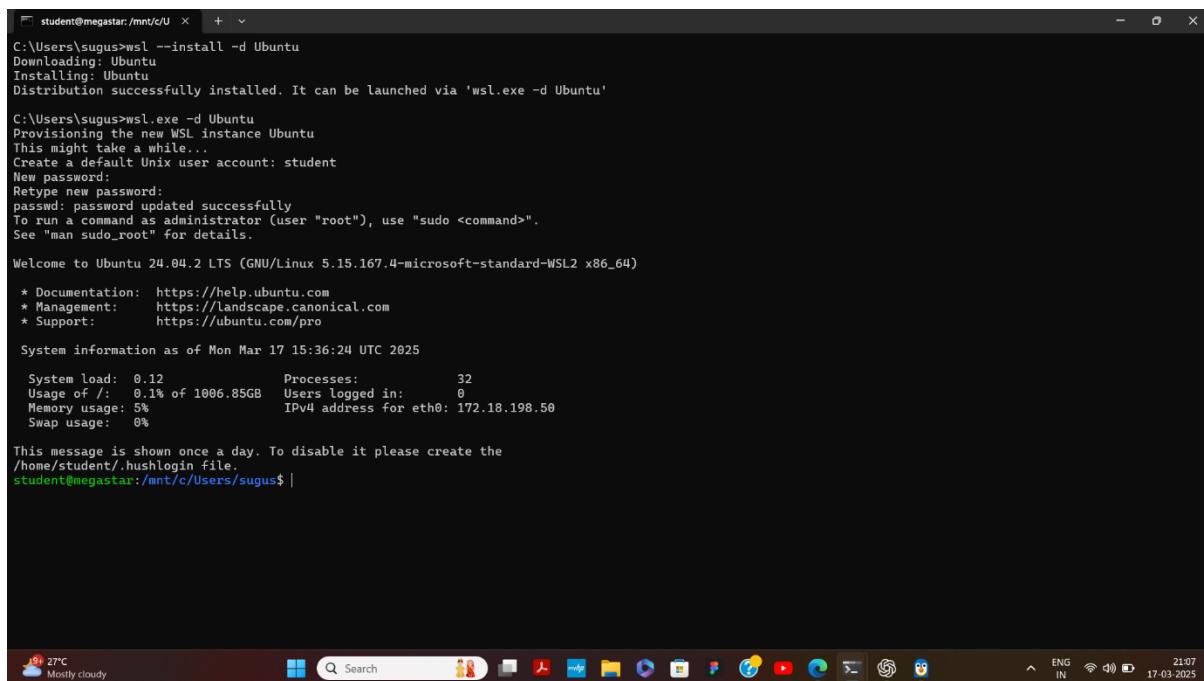
```
cd/mnt/c
```

Conclusion

You have successfully installed and set up WSL with Ubuntu on Windows 10. You can now use the Ubuntu terminal to run Linux commands and manage your system efficiently.



Step 2: (Set New Password)



```
student@megastar:/mnt/c/U x + v
C:\Users\sugus>wsl --install -d Ubuntu
Downloading: Ubuntu
Installing: Ubuntu
Distribution successfully installed. It can be launched via 'wsl.exe -d Ubuntu'

C:\Users\sugus>wsl -d Ubuntu
Provisioning the new WSL instance Ubuntu
This might take a while...
Create a default Unix user account: student
New password:
Retype new password:
passwd: password updated successfully
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Mon Mar 17 15:36:24 UTC 2025

System load: 0.12      Processes: 32
Usage of /: 0.1% of 1006.85GB  Users logged in: 0
Memory usage: 5%          IPv4 address for eth0: 172.18.198.50
Swap usage: 0%

This message is shown once a day. To disable it please create the
/home/student/.hushlogin file.
student@megastar:/mnt/c/Users/sugus$ |
```

Step-by- Step Guide to Creating a Freestyle Job in Jenkins to Install Nginx on a Local Ubuntu VM

Prerequisites for Setting Up a Freestyle Job to Install Nginx in Jenkins

Before creating the Freestyle Job, ensure that the following prerequisites are met:

1. Install Jenkins on Ubuntu (If Not Installed)

If Jenkins is not installed on your Ubuntu VM, follow these steps:

Step 1: Update Package Lists

```
sudo apt update -y
```

Step 2: Install Java (Required for Jenkins)

```
sudo apt install -y openjdk-17-jdk
```

Step3:VerifyJavaVersion

```
java-version
```

Step4:AddJenkinsRepositoryKey

(Note: The apt-key add command is deprecated in newer Ubuntu versions. Use the correct method below.)

Correct Way to Add Jenkins Repository (Without apt-key)

Step4.1:Add JenkinsGPGKey

```
wget -q -O- https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee /usr/share/keyrings/jenkinskeyring.asc > /dev/null
```

Step4.2:Add JenkinsRepository

```
echo "deb[signed-by=/usr/share/keyrings/jenkins-keyring.asc]https://pkg.jenkins.io/debian-stable binary/" | sudo tee /etc/apt/sources.list.d/jenkins.list > /dev/null
```

Step5:InstallJenkins

```
sudo apt update -y
```

```
sudo apt install -y jenkins
```

Step6:Start and Enable Jenkins Service

```
sudo systemctl start jenkins
```

```
sudo systemctl enable jenkins
```

jenkinsStep7:Check Jenkins Status

```
sudo systemctl status jenkins
```

Jenkins

2. Access Jenkins Web Interface

Jenkins will be available at `http://<VM_IP>:8080`

To get the Jenkins Server URL, Follow

These Steps: Method 1: Check the

Default URL

By default, Jenkins runs on port 8080. Open in a browser:

`http://<your-server-ip>:8080`

If you're on the same machine as Jenkins, use: `http://localhost:8080`

Method 2: Get Server IP Address

`hostname -I`

or

`ipa|grepinet`

Method 3: Check Jenkins Logs (If Unable to Access)

`sudo journalctl -u jenkins --no-pager --lines=50`

Look for lines mentioning "Jenkins is fully up and running" and the URL.

3. Access Jenkins Web Interface and Login

1. Open a browser and go to `http://<JENKINS_SERVER_IP>:8080`
2. Enter the username (admin) and the admin password retrieved from the following command:

`sudo cat /var/lib/jenkins/secrets/initialAdminPassword`

3. Choose *Install Suggested Plugins* (recommended) or manually select plugins.

4. Ensure Sudo Access for Jenkins User

Jenkins runs as a system user (jenkins). If your script requires sudo, allow Jenkins to execute commands without a password:

sudovisudo

Add the following line at the end of the file:

jenkins ALL=(ALL) NOPASSWD:ALL

Save and exit.

Step-by-

Step Guide to Creating a Freestyle Job in Jenkins to install Nginx

Step 1: Create a New Freestyle Job

1. Click on **New Item** from the Jenkins Dashboard.
2. Enter a name for the job, e.g., *Install-Nginx*.
3. Select **Freestyle project**.
4. Click **OK**.

Step 2: Configure the Job Add Build

Step

1. Scroll down to **Build** → Click **Add build step** → Select **Execute shell**.
2. Paste the following script in the command box:

```
#!/bin/bash
echo "Updating package lists..."
sudo apt update -y

echo "Installing Nginx..."
sudo apt install -y nginx
```

```
echo"StartingNginxservice..."
```

```
sudosystemctl start nginx
```

```
echo"EnablingNginxtostartonboot..."
```

```
sudosystemctl enable nginx
```

```
echo"NginxInstallationCompleted!"
```

Step3:SaveandRuntheJob

1. Click**Save**.
2. Click**BuildNow**.
3. Checkthe**ConsoleOutput**toverifythe installation.

Step4:VerifytheInstallation

1. CheckNginx Status

```
systemctlstatusnginx
```

Ifrunning, you should seeoutput like "*active(running)*".

2. OpenNginxin Browser

```
http://<VM_IP>
```

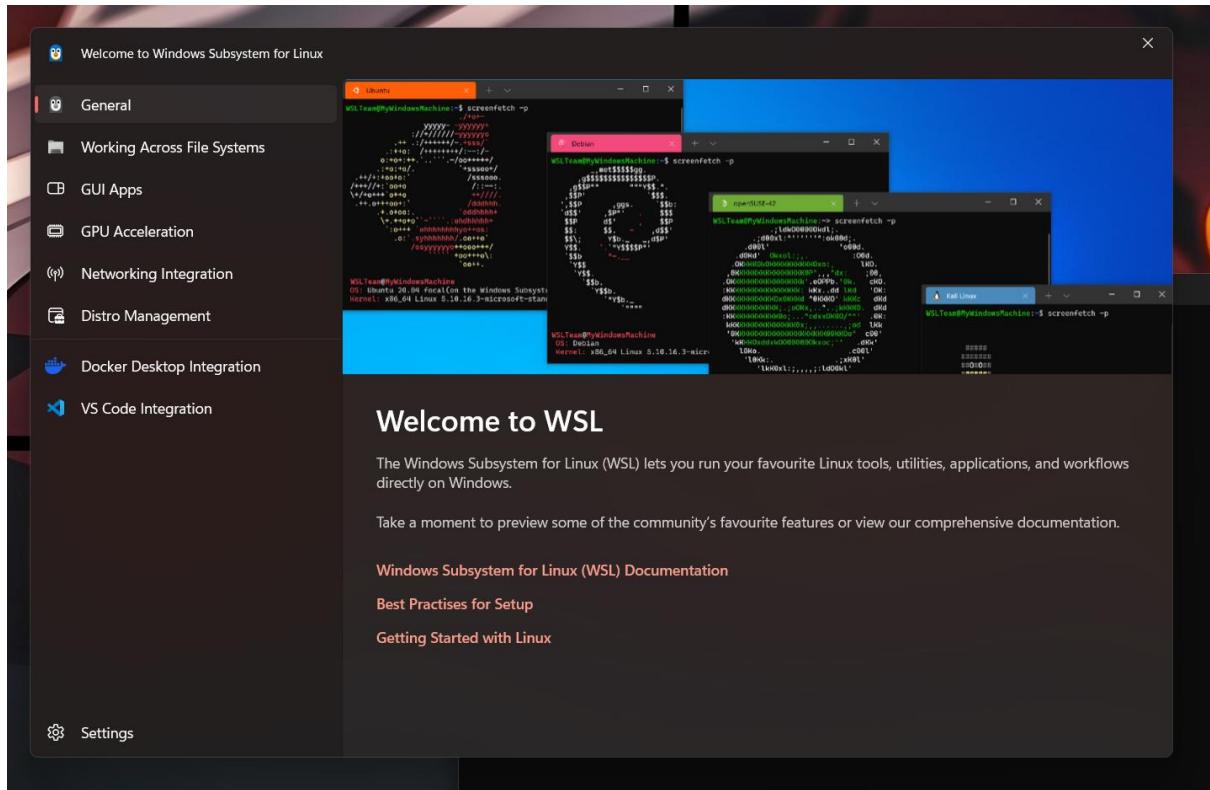
You should see the default Nginx welcome page.

Conclusion

You have successfully set up a Jenkins Freestyle Job to install Nginx on a local Ubuntu VM. This guide covers everything from Jenkins installation, configuration, and running the job to verify that Nginx is installed and running correctly.

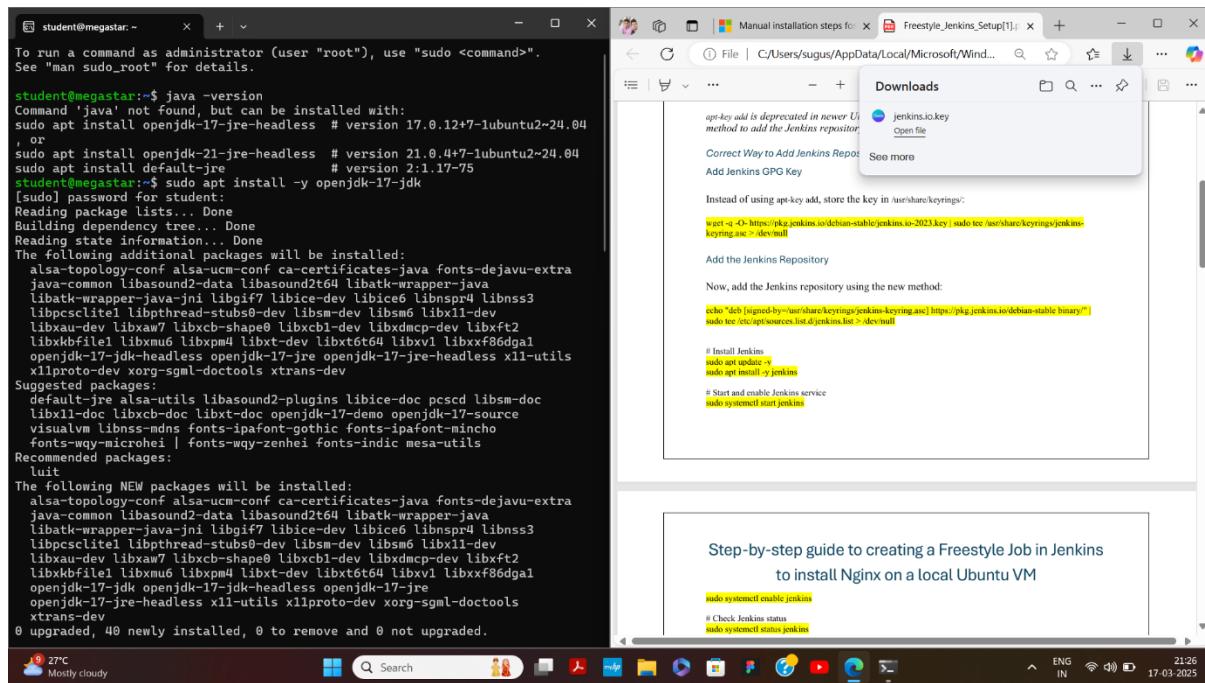
Now, your Jenkins automation is ready to deploy Nginx effortlessly!

Step 3: (New WSL Window Will Open)



Step 4:

(Check JDK Version) If not available install it .



```
student@megastar:~$ java -version
To run a command as administrator (user "root"), use "sudo <command>". See "man sudo_root" for details.

student@megastar:~$ java -version
Command 'java' not found, but can be installed with:
sudo apt install openjdk-17-jre-headless # version 17.0.12+7-1ubuntu2~24.04
sudo apt install default-jre # version 2:1.17-75
student@megastar:~$ sudo apt install -y openjdk-17-jdk
[sudo] password for student:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
alsa-topology-conf alsu-ucm-conf ca-certificates-java fonts-dejavu-extra
java-common libasound2-data libasound2t64 libatk-wrapper-java
libatk-wrapper-java-jni libgi7 libice-dev libice6 libnspr4 libnss3
libpcslite1 libpthread-stubs0-dev libsm-dev libsm6 libx11-dev
libxau-dev libxaw7 libxcb-shape0 libxcb1-dev libxdmcp-dev libxft2
libxbffile1 libxmu6 libxpm4 libxt-dev libxt6t64 libxv1 libxxf86dgal
openjdk-17-jdk-headless openjdk-17-jre openjdk-17-jre-headless x11-utils
x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
default-jre alsu-ucm libasound2-plugins libice-doc pcscd libsm-doc
libx11-doc libxcb-doc libxt-doc openjdk-17-demo openjdk-17-source
visualvm libnss-mdns fonts-ipafont-gothic fonts-ipafont-mincho
fonts-wqy-microhei | fonts-wqy-zenhei fonts-indic mesa-utils
Recommended packages:
luit
The following NEW packages will be installed:
alsa-topology-conf alsu-ucm-conf ca-certificates-java fonts-dejavu-extra
java-common libasound2-data libasound2t64 libatk-wrapper-java
libatk-wrapper-java-jni libgi7 libice-dev libice6 libnspr4 libnss3
libpcslite1 libpthread-stubs0-dev libsm-dev libsm6 libx11-dev
libxau-dev libxaw7 libxcb-shape0 libxcb1-dev libxdmcp-dev libxft2
libxbffile1 libxmu6 libxpm4 libxt-dev libxt6t64 libxv1 libxxf86dgal
openjdk-17-jdk openjdk-17-jdk-headless openjdk-17-jre
openjdk-17-jre-headless x11-utils x11proto-dev xorg-sgml-doctools
xtrans-dev
0 upgraded, 40 newly installed, 0 to remove and 0 not upgraded.

student@megastar:~$
```

Downloads

apt-key add is deprecated in newer Ubuntu releases. Use the new method to add the Jenkins repository.

Correct Way to Add Jenkins Repository

Add Jenkins GPG Key

Instead of using apt-key add, store the key in /usr/share/keyrings:

```
 wget -qO- https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee /usr/share/keyrings/jenkins-keyring.gpg
```

Add the Jenkins Repository

Now, add the Jenkins repository using the new method:

```
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.gpg] https://pkg.jenkins.io/debian-stable binary" | sudo tee /etc/apt/sources.list.d/jenkins.list
```

Install Jenkins

```
sudo apt update
```

Start and enable Jenkins service

```
sudo systemctl start jenkins
```

Step-by-step guide to creating a Freestyle Job in Jenkins to install Nginx on a local Ubuntu VM

```
sudo systemctl enable jenkins
sudo apt update
# Check Jenkins status
sudo systemctl status jenkins
```

27°C Mostly cloudy

ENG IN 21:26 17-03-2025

Step 5: (Installing Packages)

```
student@LAPTOP-INTAVHGL: ~
Get:13 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [16.9 kB]
Get:14 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [726 kB]
Get:15 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [146 kB]
Get:16 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [212 kB]
Get:17 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 c-n-f Metadata [432 B]
Get:18 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [26.2 kB]
Get:19 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [4892 kB]
Get:20 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [212 kB]
Get:21 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [448 B]
Get:22 http://security.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:23 http://archive.ubuntu.com/ubuntu noble/universe amd64 Components [3071 kB]
  5% [23 Components-amd64] 2155 kB/3871 kB 50%
Get:24 http://archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:25 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:26 http://archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:27 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:28 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:29 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [921 kB]
Get:30 http://archive.ubuntu.com/ubuntu noble-updates/main Translation-en [208 kB]
Get:31 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [151 kB]
Get:32 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [13.4 kB]
Get:33 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1040 kB]
Get:34 http://archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [262 kB]
Get:35 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [364 kB]
Get:36 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [25.8 kB]
Get:37 http://archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [759 kB]
Get:38 http://archive.ubuntu.com/ubuntu noble-updates/restricted Translation-en [153 kB]
Get:39 http://archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 kB]
Get:40 http://archive.ubuntu.com/ubuntu noble-updates/restricted amd64 c-n-f Metadata [464 B]
Get:41 http://archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [50.1 kB]
Get:42 http://archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [5884 kB]
Get:43 http://archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [948 B]
Get:44 http://archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [656 B]
Get:45 http://archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [288 B]
Get:46 http://archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [112 B]
Get:47 http://archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [14.2 kB]
Get:48 http://archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [12.1 kB]
Get:49 http://archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [28.0 kB]
Get:50 http://archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [256 B]
Get:51 http://archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]
Get:52 http://archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Get:53 http://archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]
Fetched 32.8 MB in 19s (1707 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
56 packages can be upgraded. Run 'apt list --upgradable' to see them.
student@LAPTOP-INTAVHGL:~$
```

```

student@LAPTOP-INTAVHGL:~ update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/serialver to provide /usr/bin/serialver (serialver) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jhsdb to provide /usr/bin/jhsdb (jhsdb) in auto mode
Setting up openjdk-17-jdk-amd64 (17.0.14-7-1:24.04) ...
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jconsole to provide /usr/bin/jconsole (jconsole) in auto mode
student@LAPTOP-INTAVHGL:~ jenkins version
openjdk version "17.0.14" 2025-01-21
OpenJDK Runtime Environment (build 17.0.14+7-Ubuntu-124.04)
student@LAPTOP-INTAVHGL:~$ wget -q -O https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -
wget: missing URL
Usage: wget [OPTION]... [URL]...
Try `wget --help` for more options.
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
gpg: valid OpenPGP data found.
student@LAPTOP-INTAVHGL:~$ wget -q -O https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee /usr/share/keyrings/jenkins-keyring.asc>/dev/null
sudo: tee/usr/share/keyrings/jenkins-keyring.asc: command not found
student@LAPTOP-INTAVHGL:~$ wget -q -O https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee /usr/share/keyrings/jenkins-keyring.asc>/dev/null
wget: missing URL
Usage: wget [OPTION]... [URL]...
Try `wget --help` for more options.
sudo: tee/usr/share/keyrings/jenkins-keyring.asc: command not found
student@LAPTOP-INTAVHGL:~$ wget -q -O https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee /etc/apt/sources.list.d/jenkins.list>/dev/null
student@LAPTOP-INTAVHGL:~$ echo "deb [signed-by/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian-stable binary/" | sudo tee /etc/apt/sources.list.d/jenkins.list>/dev/null
student@LAPTOP-INTAVHGL:~$ sudo apt update -y
Ign1: https://pkg.jenkins.io/debian-stable binary/ InRelease
get:1 https://pkg.jenkins.io/debian-stable binary/ Release [2044 B]
get:3 https://pkg.jenkins.io/debian-stable binary/ Release.gpg [833 B]
get:4 https://pkg.jenkins.io/debian-stable binary/ Packages [28.7 kB]
Hit:5 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:6 http://archive.ubuntu.com/ubuntu noble InRelease
Hit:7 http://archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:8 http://archive.ubuntu.com/ubuntu noble-backports InRelease
Fetching 31.6 kB in 2s (17.6 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
56 packages can be upgraded. Run 'apt list --upgradable' to see them.
student@LAPTOP-INTAVHGL:~$
```

Step 6:

(Installing Jenkins)

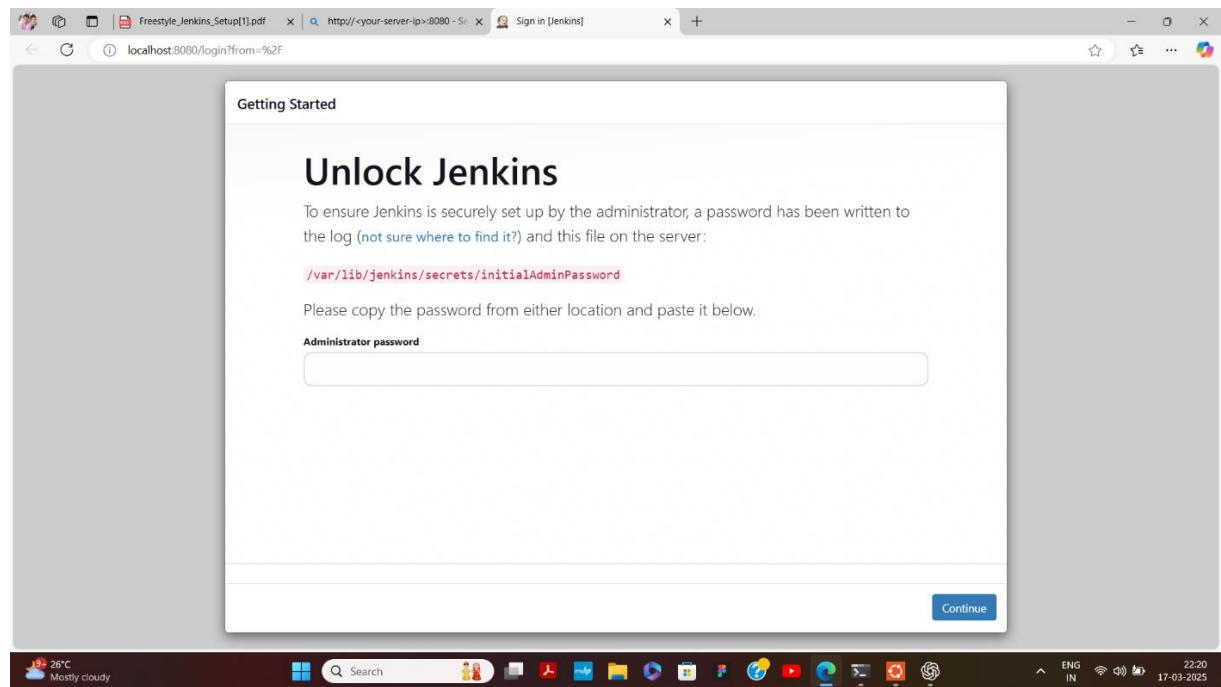
```

student@LAPTOP-INTAVHGL:~ wget: missing URL
Usage: wget [OPTION]... [URL]...
Try `wget --help` for more options.
sudo: tee/usr/share/keyrings/jenkins-keyring.asc: command not found
student@LAPTOP-INTAVHGL:~$ wget -q -O https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee /etc/apt/sources.list.d/jenkins.list>/dev/null
student@LAPTOP-INTAVHGL:~$ echo "deb [signed-by/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian-stable binary/" | sudo tee /etc/apt/sources.list.d/jenkins.list>/dev/null
student@LAPTOP-INTAVHGL:~$ sudo apt update -y
Ign1: https://pkg.jenkins.io/debian-stable binary/ InRelease
get:2 https://pkg.jenkins.io/debian-stable binary/ Release [2044 B]
get:3 https://pkg.jenkins.io/debian-stable binary/ Release.gpg [833 B]
get:4 https://pkg.jenkins.io/debian-stable binary/ Packages [28.7 kB]
Hit:5 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:6 http://archive.ubuntu.com/ubuntu noble InRelease
Hit:7 http://archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:8 http://archive.ubuntu.com/ubuntu noble-backports InRelease
Fetching 31.6 kB in 2s (17.6 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
56 packages can be upgraded. Run 'apt list --upgradable' to see them.
student@LAPTOP-INTAVHGL:~$ sudo apt install -y jenkins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  net-tools
The following NEW packages will be installed:
  jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 56 not upgraded.
Need to get 95.0 kB of archives.
After this operation, 97.6 MB of additional disk space will be used.
get:1 http://archive.ubuntu.com/ubuntu/noble/main amd64 net-tools amd64 2.20-0.1ubuntu4 [204 kB]
get:2 https://pkg.jenkins.io/debian-stable binary/ jenkins 2.492.2 [94.8 MB]
548 [2 jenkins 28.7 MB/94.8 MB 30%]
2852 kB/s 23s
```

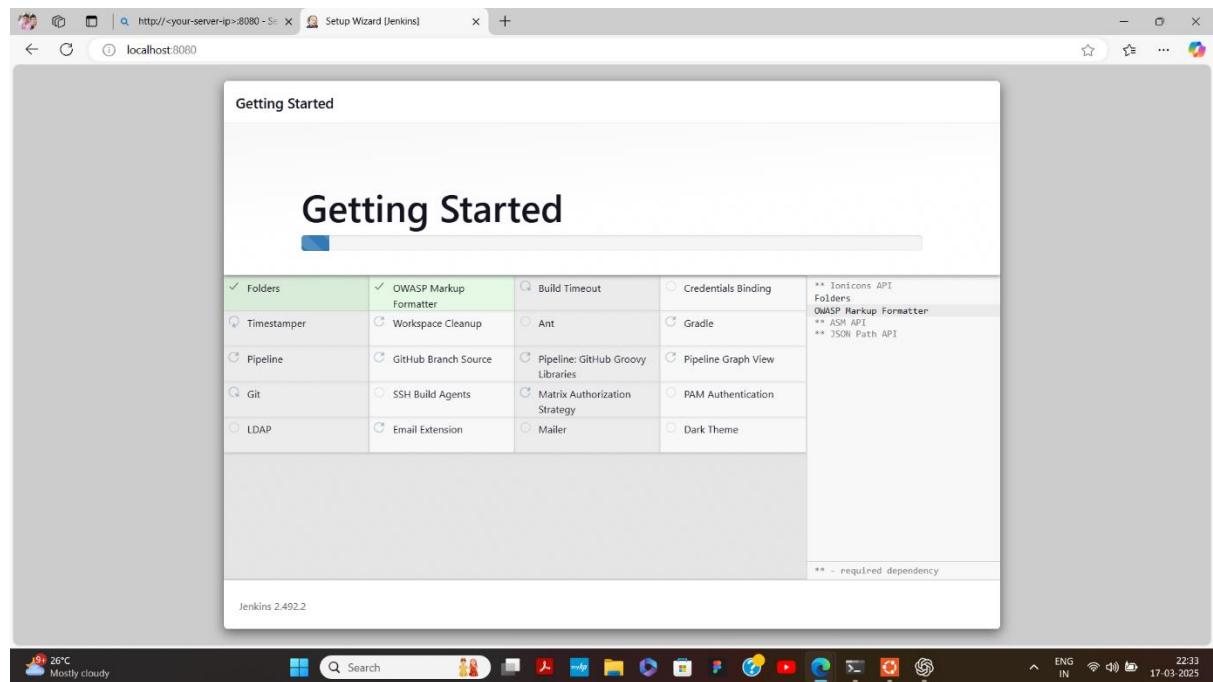
Step 7:

```
student@LAPTOP-INTAWHGL:~  
student@LAPTOP-INTAWHGL:~$ sudo systemctl status jenkins  
● jenkins.service - Jenkins Continuous Integration Server  
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)  
   Active: active (running) since Mon 2025-03-17 17:37:50 UTC; 3min 55s ago  
     Main PID: 2392 (java)  
       Tasks: 58 (limit: 9090)  
      Memory: 949.5M  
        CGroup: /system.slice/jenkins.service  
           └─2392 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/  
  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: 6513675f53d02427bfec2b0e9121d6e  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: ****  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: ****  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: ****  
Mar 17 17:37:50 LAPTOP-INTAWHGL jenkins[2392]: 2025-03-17 17:37:50.706+0000 [id=55] INFO >  
Mar 17 17:37:50 LAPTOP-INTAWHGL jenkins[2392]: 2025-03-17 17:37:50.736+0000 [id=33] INFO >  
Mar 17 17:37:50 LAPTOP-INTAWHGL systemd[1]: Started jenkins.service - Jenkins Continuous Integration Se  
Mar 17 17:37:53 LAPTOP-INTAWHGL jenkins[2392]: 2025-03-17 17:37:53.053+0000 [id=87] INFO >  
Mar 17 17:37:53 LAPTOP-INTAWHGL jenkins[2392]: 2025-03-17 17:37:53.055+0000 [id=87] INFO >  
lines 1-5 of 1999 (use -l to show more)  
● jenkins.service - Jenkins Continuous Integration Server  
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)  
   Active: active (running) since Mon 2025-03-17 17:37:50 UTC; 3min 55s ago  
     Main PID: 2392 (java)  
       Tasks: 58 (limit: 9090)  
      Memory: 949.5M  
        CGroup: /system.slice/jenkins.service  
           └─2392 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080  
  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: 6513675f53d02427bfec2b0e9121d6e  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: ****  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: ****  
Mar 17 17:37:41 LAPTOP-INTAWHGL jenkins[2392]: ****  
Mar 17 17:37:50 LAPTOP-INTAWHGL jenkins[2392]: 2025-03-17 17:37:50.706+0000 [id=55] INFO jenkins.InitReactorRunner$1@#0nAttained: Completed initialization  
Mar 17 17:37:50 LAPTOP-INTAWHGL jenkins[2392]: 2025-03-17 17:37:50.736+0000 [id=33] INFO hudson.lifecycle.Lifecycle$onReady: Jenkins is fully up and running  
Mar 17 17:37:50 LAPTOP-INTAWHGL systemd[1]: Started jenkins.service - Jenkins Continuous Integration Se  
Mar 17 17:37:53 LAPTOP-INTAWHGL jenkins[2392]: 2025-03-17 17:37:53.053+0000 [id=87] INFO h.m.DownloadService$Downloadable#load: Obtained the updated data file for hudson.tasks.MavenInstaller  
Mar 17 17:37:53 LAPTOP-INTAWHGL jenkins[2392]: 2025-03-17 17:37:53.055+0000 [id=87] INFO hudson.util.Retryer$start: Performed the action check updates server successfully at the attempt #1  
-  
-  
-
```

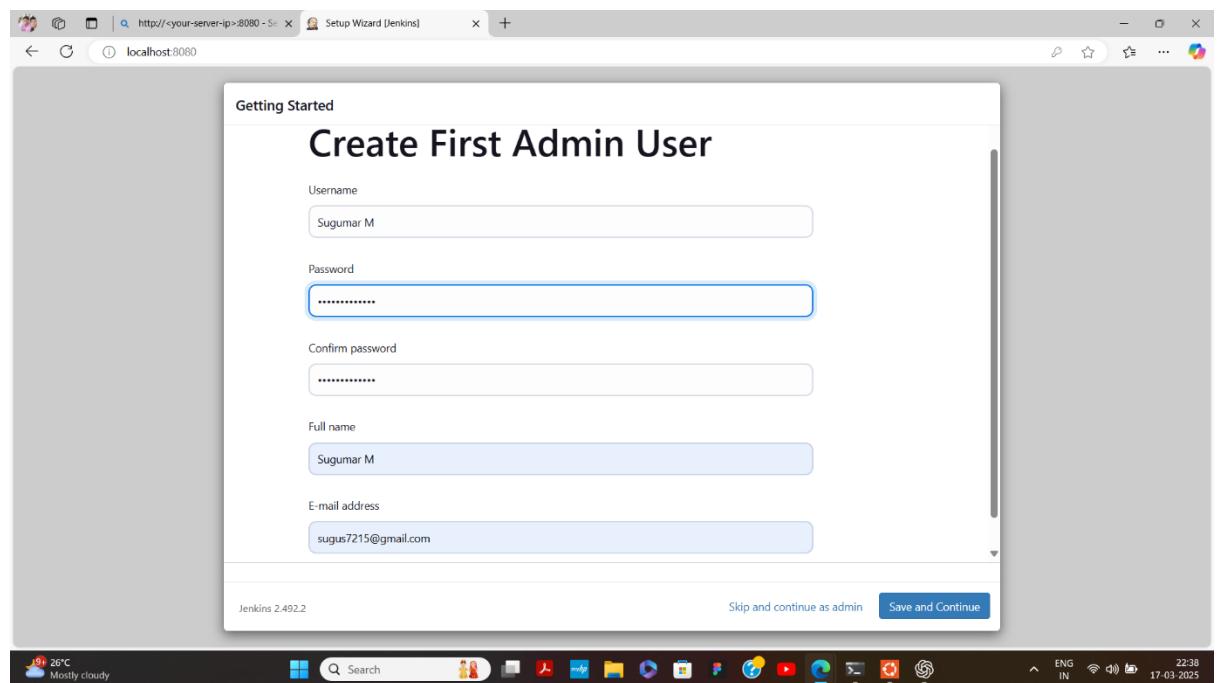
Step 8:
(Enter the Administrator Password)



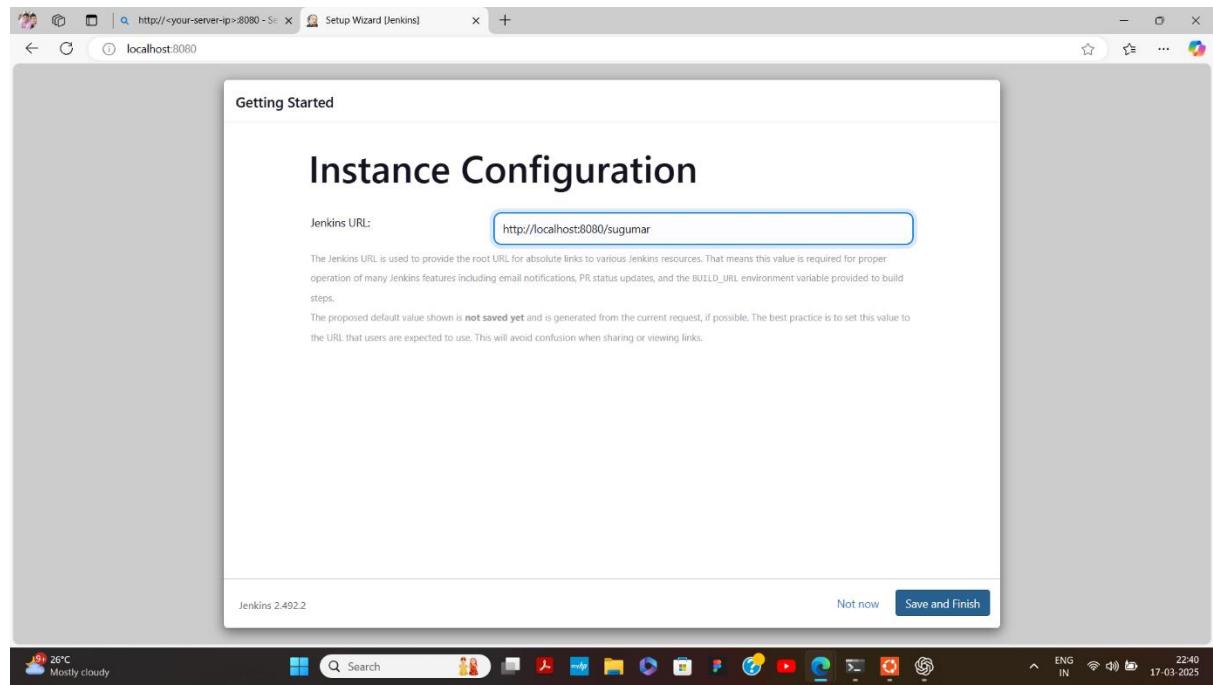
Step 9:



Step 10:
(Create First Admin User)



Step 11:
(Enter the URL)



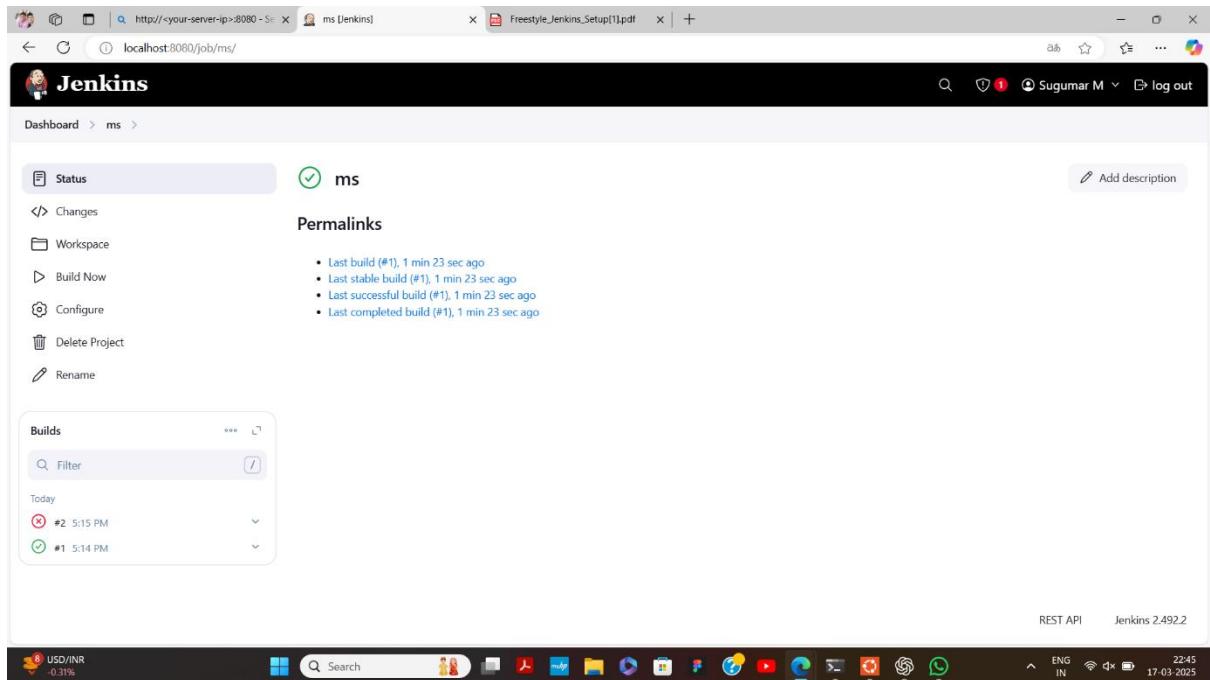
Step 12:

(Create an Project ex:ms)

(In Configure build and Select Execute shell)

(Build the Project)

SELECT BUILD NOW



Step 13:

(We can check the output in console output)



Started by user [Sugumar M](#)
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/ms
[ms] \$ /bin/bash /tmp/jenkins2746058650598173611.sh
Updating package lists...
sudo: a terminal is required to read the password; either use the -s option to read from standard input or configure an askpass helper
sudo: a password is required
Installing Nginx...
sudo: a terminal is required to read the password; either use the -s option to read from standard input or configure an askpass helper
sudo: a password is required
Starting Nginx service...
sudo: a terminal is required to read the password; either use the -s option to read from standard input or configure an askpass helper
sudo: a password is required
Enabling Nginx to start on boot...
sudo: a terminal is required to read the password; either use the -s option to read from standard input or configure an askpass helper
sudo: a password is required
Nginx Installation Completed!
Finished: SUCCESS