

DAY-1

DEVOPS

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`

Step 2: Install Ubuntu

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

Step 3: 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.

Install Jenkins on Ubuntu

Update package lists

`sudo apt update -y`

Install Java (Required for Jenkins) `sudo apt install -y openjdk-17-jdk`

```
#verify java
version java -
version
```

Add Jenkins GPG Key

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

Add the Jenkins Repository

```
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
```

Install Jenkins

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

Start and enable Jenkins

```
service sudo systemctl start
jenkins
```

Step 4: Add Jenkins Repository Key

Step 4.1: Add Jenkins GPG Key

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

Step 4.2: Add Jenkins Repository

```
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
```

Step 5: Install Jenkins

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

Step 6: Start and Enable Jenkins Service

```
sudo systemctl start jenkins
```

```
sudo systemctl enable jenkins
```

Step 7: Check Jenkins Status

```
sudo systemctl status jenkins
```

cat copy the localhost:8080 admin path

display the password

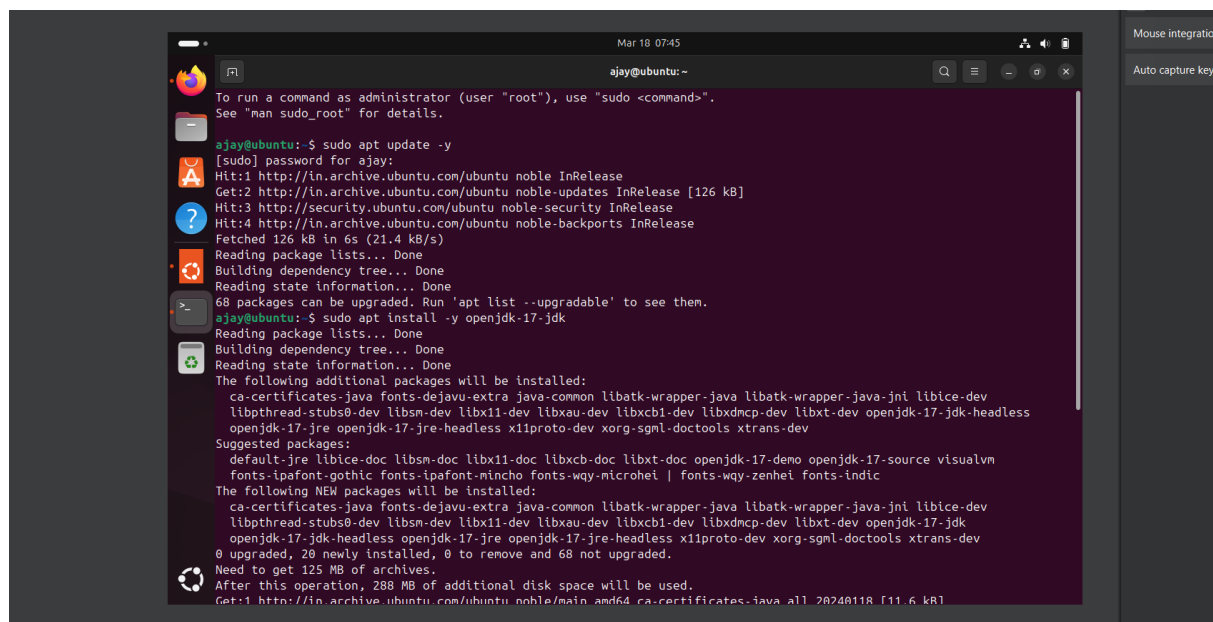
=>By default, Jenkins runs as a system user (jenkins). If your script requires sudo, you must allow the Jenkins user to run commands without a password.

```
sudo visudo
```

Add the following line at the end:

```
jenkins ALL=(ALL) NOPASSWD: ALL
```

Save and exit.



```
Mar 18 07:45
ajay@ubuntu: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ajay@ubuntu:~$ sudo apt update -y
[sudo] password for ajay:
Hit:1 http://in.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:3 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu noble-backports InRelease
Fetched 126 kB in 6s (21.4 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
68 packages can be upgraded. Run 'apt list --upgradable' to see them.

ajay@ubuntu:~$ sudo apt install -y openjdk-17-jdk
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java libatk-wrapper-java-jni libice-dev
  libpthread-stubs0-dev libsm-dev libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-17-jdk-headless
  openjdk-17-jre openjdk-17-jre-headless x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  default-jre libice-doc libsm-doc libx11-doc libxcb-doc libxt-dev openjdk-17-demo openjdk-17-source visualvm
  fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei fonts-indic
The following NEW packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java libatk-wrapper-java-jni libice-dev
  libpthread-stubs0-dev libsm-dev libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-17-jdk
  openjdk-17-jdk-headless openjdk-17-jre openjdk-17-jre-headless x11proto-dev xorg-sgml-doctools xtrans-dev
0 upgraded, 20 newly installed, 0 to remove and 68 not upgraded.
Need to get 125 MB of archives.
After this operation, 288 MB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu noble/main amd64 ca-certificates-java all 20240118 [11.6 kB]
```

```

Reading state information... Done
jenkins is already the newest version (2.492.2).
0 upgraded, 0 newly installed, 0 to remove and 56 not upgraded.
ajay@Sparkajay:~$ sudo systemctl start jenkins
ajay@Sparkajay:~$ sudo systemctl enable jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
ajay@Sparkajay:~$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; pres>
   Active: active (running) since Tue 2025-03-18 13:33:23 UTC; 1min 21s a>
   Main PID: 168 (java)
   Tasks: 55 (limit: 4363)
   Memory: 516.7M (C)
   CGroup: /system.slice/jenkins.service
           └─168 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/j>
Mar 18 13:33:22 Sparkajay jenkins[168]: 2025-03-18 13:33:22.004+0000 [id=47>
Mar 18 13:33:22 Sparkajay jenkins[168]: 2025-03-18 13:33:22.010+0000 [id=38>
Mar 18 13:33:22 Sparkajay jenkins[168]: 2025-03-18 13:33:22.425+0000 [id=47>
Mar 18 13:33:23 Sparkajay jenkins[168]: 2025-03-18 13:33:23.778+0000 [id=41>
Mar 18 13:33:23 Sparkajay jenkins[168]: 2025-03-18 13:33:23.781+0000 [id=38>
Mar 18 13:33:23 Sparkajay jenkins[168]: 2025-03-18 13:33:23.833+0000 [id=33>
Mar 18 13:33:23 Sparkajay jenkins[168]: 2025-03-18 13:33:23.847+0000 [id=41>
Mar 18 13:33:23 Sparkajay jenkins[168]: 2025-03-18 13:33:23.892+0000 [id=44>
Mar 18 13:33:23 Sparkajay jenkins[168]: 2025-03-18 13:33:23.938+0000 [id=25>
Mar 18 13:33:23 Sparkajay systemd[1]: Started jenkins.service - Jenkins Con>

```

```

jeeva@Jeeva:~$ sudo more /var/lib/jenkins/secrets/initialAdminPassword
0e507d6b0f14097ba040a5e1dd67f6d
jeeva@Jeeva:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Tue 2025-03-18 05:55:19 UTC; 17s ago
     Docs: man:nginx(8)
   Main PID: 11168 (nginx)
   Tasks: 13 (limit: 4585)
   Memory: 9.0M (C)
   CGroup: /system.slice/nginx.service
           └─11168 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─11169 "nginx: worker process"
               └─11170 "nginx: worker process"
                 └─11171 "nginx: worker process"
                   └─11173 "nginx: worker process"
                     └─11174 "nginx: worker process"
                       └─11175 "nginx: worker process"
                         └─11176 "nginx: worker process"
                           └─11177 "nginx: worker process"
                             └─11178 "nginx: worker process"
                               └─11179 "nginx: worker process"
                                 └─11180 "nginx: worker process"
                                   └─11181 "nginx: worker process"
Mar 18 05:55:19 Jeeva systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Mar 18 05:55:19 Jeeva systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.

```

2. Access Jenkins Web Interface

Jenkins will be available at <http://localhost: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>

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:

```
echo "Updating package lists..."  
  
sudo apt update -y  
  
echo "Installing Nginx..."  
  
sudo apt install -y nginx  
  
echo "Starting Nginx service..."  
  
sudo systemctl start nginx  
  
echo "Enabling Nginx to start on boot..."  
  
sudo systemctl enable nginx  
  
echo "Nginx Installation Completed!"
```

Step 3: Save and Run the Job

1. Click **Save**.
2. Click **Build Now**.
3. Check the **Console Output** to verify the installation.

Step 4: Verify the Installation

1. Check Nginx Status

```
systemctl status nginx
```

If running, you should see output like *"active (running)"*.

2. Open Nginx in Browser

```
http://localhost:80
```

You should see the default Nginx welcome page

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Jenkins 2.501

Getting Started

Getting Started

✓ Folders	✓ OWASP Markup Formatter	🔄 Build Timeout	🔄 Credentials Binding	** Ionicons API
🔄 Timestampper	🔄 Workspace Cleanup	🔄 Ant	🔄 Gradle	Folders
🔄 Pipeline	🔄 GitHub Branch Source	🔄 Pipeline: GitHub Groovy Libraries	🔄 Pipeline Graph View	OWASP Markup Formatter
🔄 Git	🔄 SSH Build Agents	🔄 Matrix Authorization Strategy	○ PAM Authentication	** ASM API
○ LDAP	🔄 Email Extension	○ Mailer	○ Dark Theme	** JSON Path API
				** Struts
				** Pipeline: Step API
				** Token Macro
				** - required dependency

Jenkins 2.501

Jenkins is ready!

You have skipped the **setup of an admin user**.

To log in, use the username: "admin" and the administrator password you used to access the setup wizard.


You have skipped the configuration of the Jenkins URL.






To configure the Jenkins URL, go to "Manage Jenkins" page.

Your Jenkins setup is complete.


Start using Jenkins


Jenkins 2.501


 **Jenkins**


    admin  log out


Dashboard >

 New Item


 Build History


 Manage Jenkins

 My Views

Build Queue 

No builds in the queue.


Build Executor Status  0/2

 Add description


Welcome to Jenkins!


This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.


Start building your software project

Create a job 

Set up a distributed build

Set up an agent 

Configure a cloud 

Learn more about distributed builds 

REST API Jenkins 2.501

Dashboard > Install-Nginx > Configuration

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Configure

- General
- Source Code Management
- Triggers
- Environment
- Build Steps**
- Post-build Actions

Execute shell ?

Command

See the list of available environment variables

```
#!/bin/  
echo "Updating package lists..."  
sudo apt update -y  
echo "Installing Nginx..."  
sudo apt install -y nginx
```

Advanced ▾

Add build step ▾

Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

Add post-build action ▾

Save Apply

REST API Jenkins 2.492.2

Dashboard > Install-Nginx

Install-Nginx

Permalinks

Add description

- Status
- </> Changes
- Workspace
- Build Now
- Configure
- Delete Project
- Rename

Builds *** ↕

Today

- #1 5:55 AM ▾

REST API Jenkins 2.492.2

