Practical: Installation of Java on Unix/Linux Machine (Without VirtualBox)

■ This guide uses WSL (Windows Subsystem for Linux) to simulate a Linux environment directly on Windows.

Step 1: Enable WSL

1. Open PowerShell as Administrator

2. Run the following command:

wsl –install

```
PS C:\Users\Himanshi> wsl --install
Downloading: Ubuntu
Installing: Ubuntu
Distribution successfully installed. It can be launched via 'wsl.exe -d Ub
Launching Ubuntu...
Provisioning the new WSL instance Ubuntu
This might take a while...
```

3. Restart your computer when prompted.

4. After restart, choose Ubuntu or install it from the Microsoft Store.

Step 2: Open Ubuntu (WSL)

- Search for 'Ubuntu' in Start Menu and open it.

- It will initialize and ask for a username and password.

```
Create a default Unix user account: himanshi2005
New password:
Retype new password:
passwd: password updated successfully
```

Step 3: Update the Package List

Run the following command:

sudo apt update

```
himanshi2005@LAPTOP-BUL13UAT:/mnt/c/Users/Himanshi$  sudo apt update
[sudo] password for himanshi2005:
Get:1 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Hit:2 http://archive.ubuntu.com/ubuntu noble InRelease
Get:3 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [1075 kB]
Get:5 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Ign:6 http://archive.ubuntu.com/ubuntu noble/universe amd64 Packages
Get:7 http://archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:8 http://archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:9 http://archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:10 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:11 http://archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:12 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:13 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:14 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1337 kB]
Get:15 http://archive.ubuntu.com/ubuntu noble-updates/main Translation-en [267 kB]
42% [15 Translation-en 205 kB/267 kB 77%] [4 Packages 534 kB/1075 kB 50%]
Get:16 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [175 kB]
Get:17 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1121 kB]
43% [17 Packages 143 kB/1121 kB 13%] [4 Packages 534 kB/1075 kB 50%]
```

Step 4: Install Java (OpenJDK 8 or 11)

- For Java 8:

sudo apt install openjdk-8-jdk -y

- For Java 11:

sudo apt install openjdk-11-jdk -y

```
himanshi2005@LAPTOP-BUL13UAT:/mnt/c/Users/Himanshi$  sudo apt install openjdk-11-jdk -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  alsa-topology-conf alsa-ucm-conf ca-certificates-java fonts-dejavu-extra java-common libasound2-data libasound2t64
  libatk-wrapper-java libatk-wrapper-java-jni libgif7 libice-dev libice6 libnspr4 libnss3 libpcsclite1
  libpthread-stubs0-dev libsm-dev libsm6 libx11-dev libxau-dev libxaw7 libxcb-shape0 libxcb1-dev libxdmcp-dev libxft2
  libxkbfile1 libxmu6 libxpm4 libxt-dev libxt6t64 libxv1 libxxf86dga1 openjdk-11-jdk-headless openjdk-11-jre
  openjdk-11-jre-headless x11-utils x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  default-jre alsa-utils libasound2-plugins libice-doc pcscd libsm-doc libx11-doc libxcb-doc libxt-doc openjdk-11-demo
  openjdk-11-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 alsa-ucm-conf ca-certificates-java fonts-dejavu-extra java-common libasound2-data libasound2t64
```

Step 5: Verify Java Installation

Run:

java -version

```
himanshi2005@LAPTOP-BUL13UAT:/mnt/c/Users/Himanshi$  java -version
openjdk version "11.0.28" 2025-07-15
OpenJDK Runtime Environment (build 11.0.28+6-post-Ubuntu-1ubuntu124.04.1)
OpenJDK 64-Bit Server VM (build 11.0.28+6-post-Ubuntu-1ubuntu124.04.1, mixed mode, sharing)
himanshi2005@LAPTOP-BUL13UAT:/mnt/c/Users/Himanshi$
```

Step 6: Set JAVA_HOME Environment Variable (Optional)

1. Open .bashrc file:

nano ~/.bashrc

```
himanshi2005@LAPTOP-BUL13UAT:/mnt/c/Users/Himanshi$ nano ~/.bashrc
himanshi2005@LAPTOP-BUL13UAT:/mnt/c/Users/Himanshi$
```

2. Add the following lines at the end:

export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64

export PATH=$JAVA_HOME/bin:$PATH

```
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
shopt -s checkwinsize

# If set, the pattern "**" used in a pathname expansion context will
 export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
 export PATH=$JAVA_HOME/bin:$PATH
```

3. Save and exit (Ctrl+X, then Y, then Enter)

4. Reload bashrc: source ~/.bashrc

5. Verify JAVA_HOME is set:

echo $JAVA_HOME

■ You now have Java installed on a Linux system simulated within Windows without using VirtualBox!