Daizy Dhiman

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

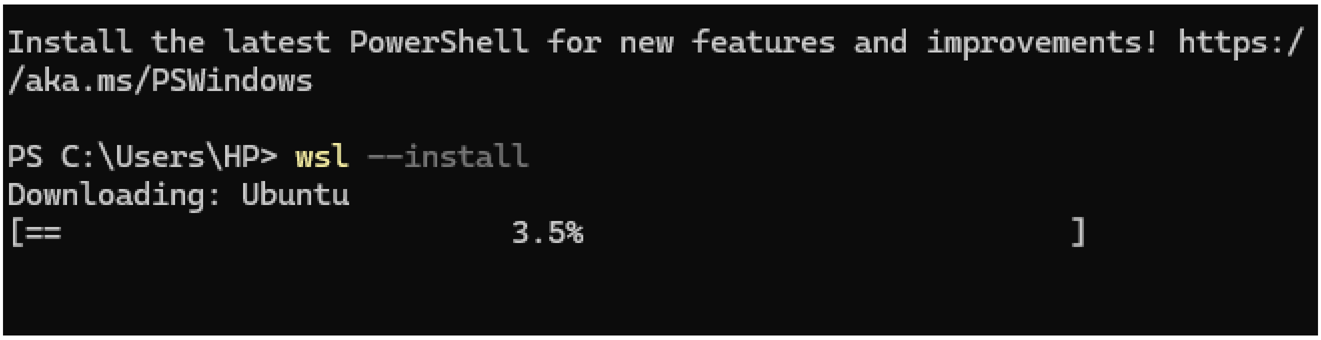
■ This guide uses WSL (Windows Subsystem for Linux) to simulate a Linux environment directly on Windows.

Step 1: Enable WSL

1. Open PowerShell as Administrator

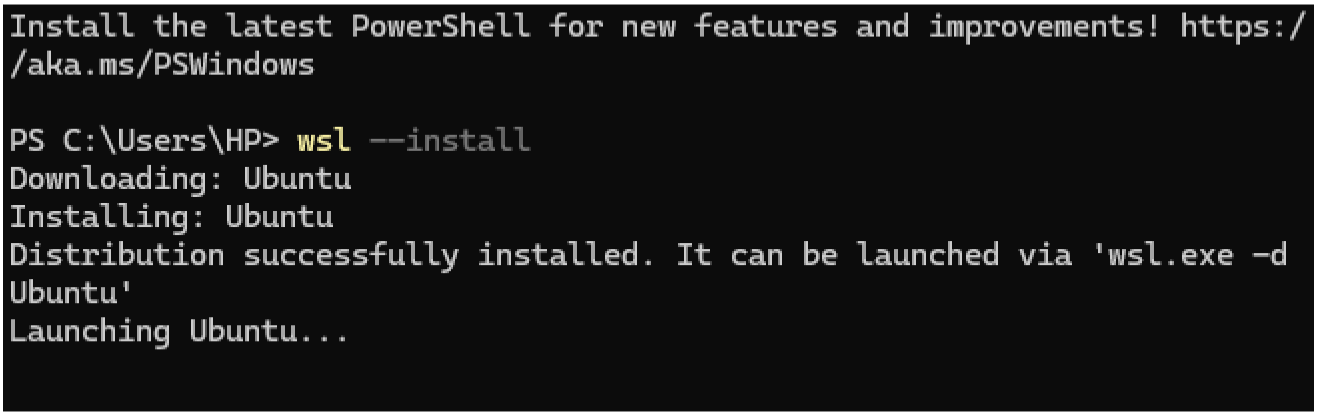
2. Run the following command:

wsl –install



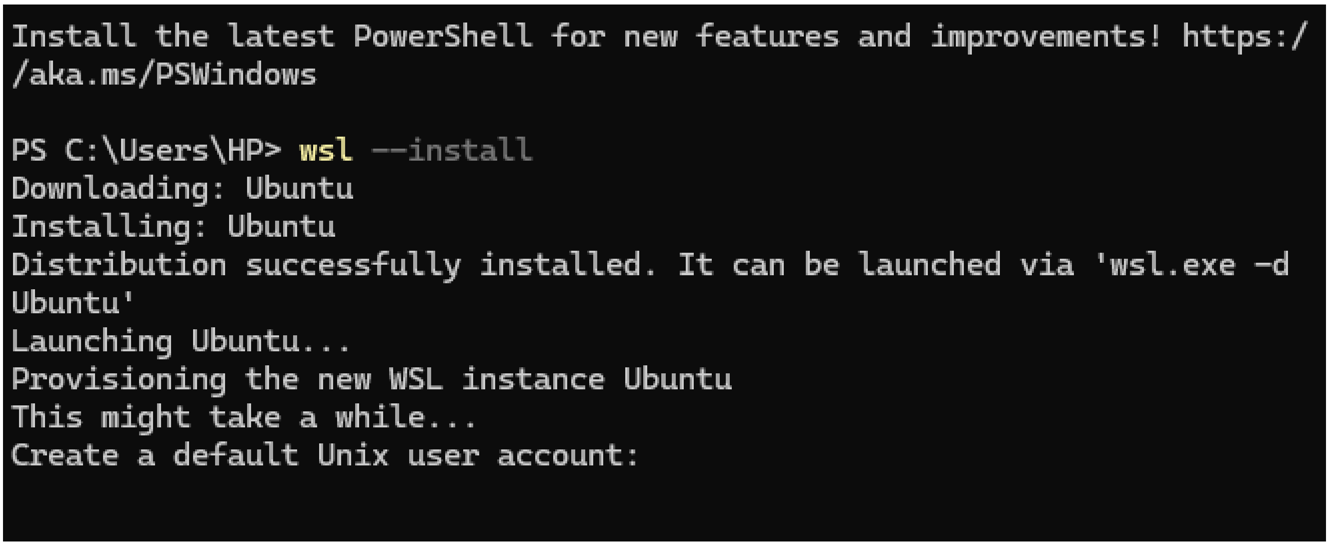
```
Install the latest PowerShell for new features and improvements! https:/
/aka.ms/PSWindows

PS C:\Users\HP> wsl --install
Downloading: Ubuntu
[==                        3.5%                          ]
```



```
Install the latest PowerShell for new features and improvements! https:/
/aka.ms/PSWindows

PS C:\Users\HP> wsl --install
Downloading: Ubuntu
Installing: Ubuntu
Distribution successfully installed. It can be launched via 'wsl.exe -d
Ubuntu'
Launching Ubuntu...
```



```
Install the latest PowerShell for new features and improvements! https:/
/aka.ms/PSWindows

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

3. Restart your computer when prompted.

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

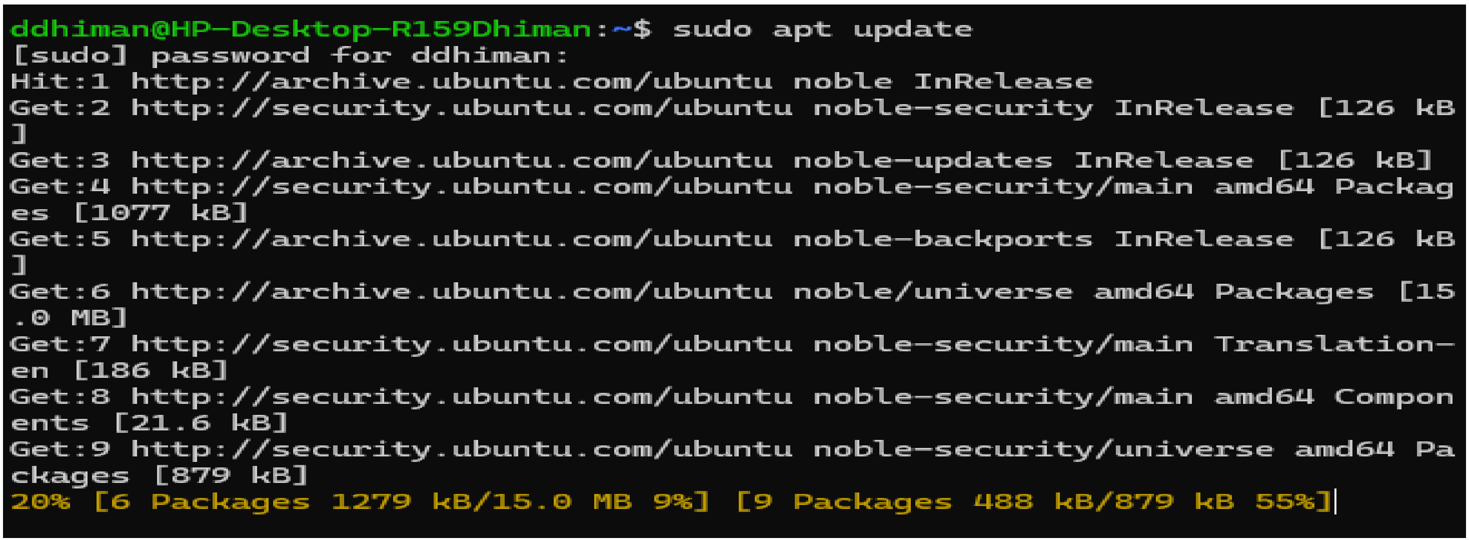
Step 2: Open Ubuntu (WSL)

- Search for 'Ubuntu' in Start Menu and open it.
- It will initialize and ask for a username and password.

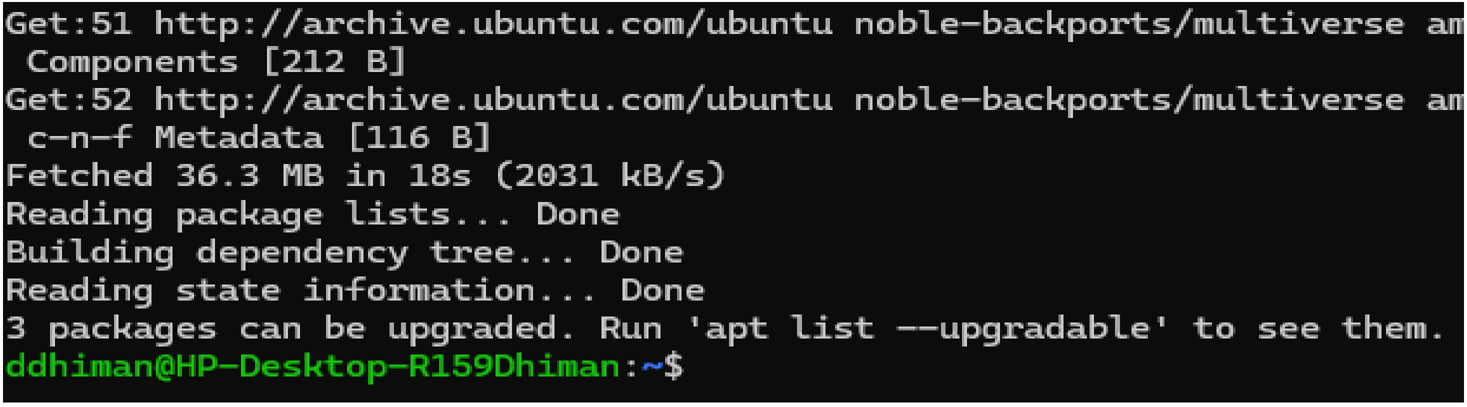
Step 3: Update the Package List

Run the following command:

sudo apt update



```
ddhiman@HP-Desktop-R159Dhiman:~$ sudo apt update
[sudo] password for ddhiman:
Hit:1 http://archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB
]
Get:3 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packag
es [1077 kB]
Get:5 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB
]
Get:6 http://archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15
.0 MB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/main Translation-
en [186 kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 Compon
ents [21.6 kB]
Get:9 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Pa
ckages [879 kB]
20% [6 Packages 1279 kB/15.0 MB 9%] [9 Packages 488 kB/879 kB 55%]
```



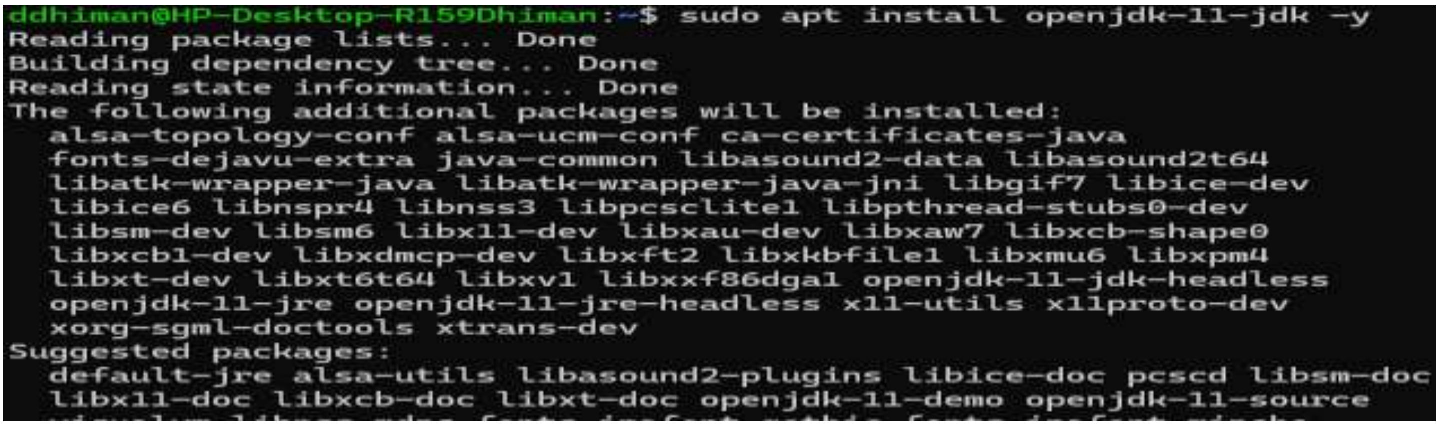
```
Get:51 http://archive.ubuntu.com/ubuntu noble-backports/multiverse am
 Components [212 B]
Get:52 http://archive.ubuntu.com/ubuntu noble-backports/multiverse am
 c-n-f Metadata [116 B]
Fetched 36.3 MB in 18s (2031 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
3 packages can be upgraded. Run 'apt list --upgradable' to see them.
ddhiman@HP-Desktop-R159Dhiman:~$
```

Step 4: Install Java (OpenJDK 8 or 11)

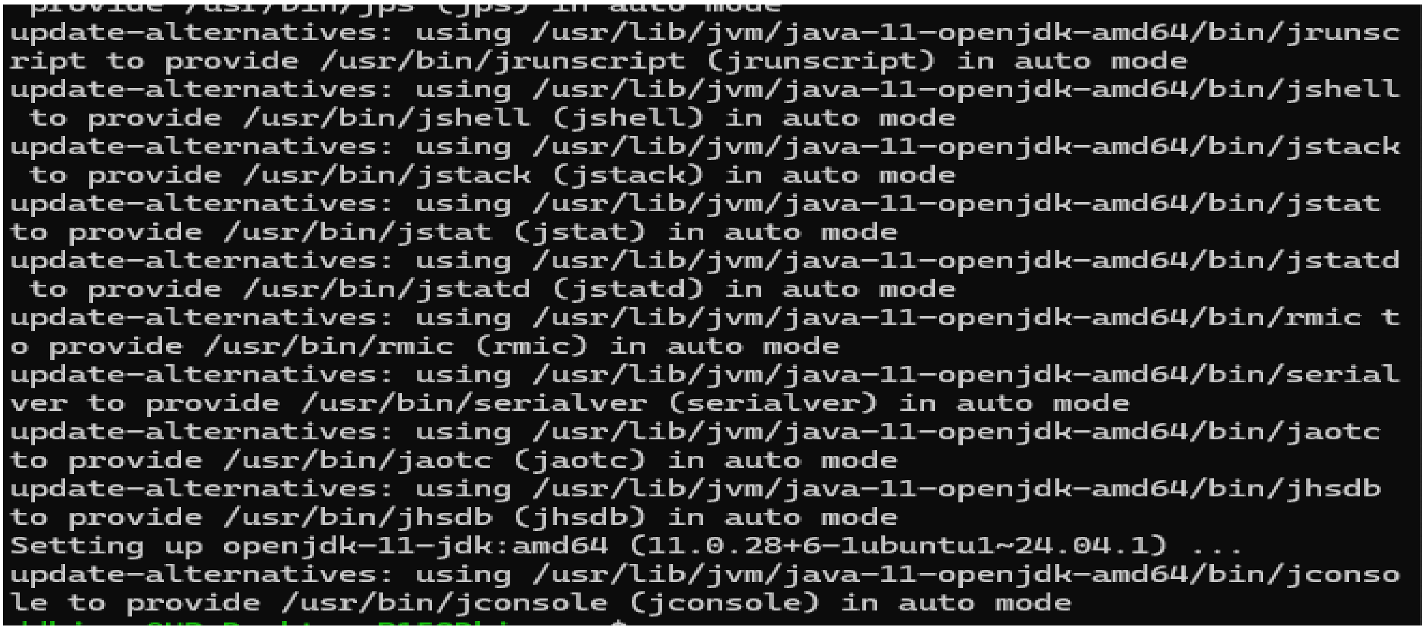
- For Java 8:

sudo apt install openjdk-8-jdk -y - For Java 11:

sudo apt install openjdk-11-jdk –y



```
ddhiman@HP-Desktop-R159Dhiman:~$ 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
```



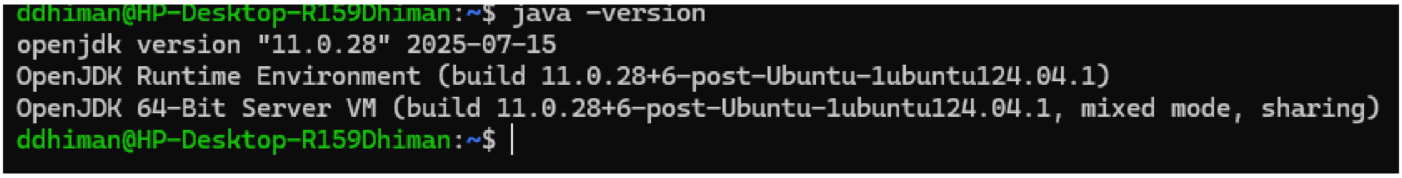
```
 provide /usr/bin/jps (jps) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jrunsc
ript to provide /usr/bin/jrunscript (jrunscript) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jshell
 to provide /usr/bin/jshell (jshell) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jstack
 to provide /usr/bin/jstack (jstack) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jstat
to provide /usr/bin/jstat (jstat) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jstatd
 to provide /usr/bin/jstatd (jstatd) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/rmic t
o provide /usr/bin/rmic (rmic) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/serial
ver to provide /usr/bin/serialver (serialver) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jaotc
to provide /usr/bin/jaotc (jaotc) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jhsdb
to provide /usr/bin/jhsdb (jhsdb) in auto mode
Setting up openjdk-11-jdk:amd64 (11.0.28+6-1ubuntu1~24.04.1) ...
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jconso
le to provide /usr/bin/jconsole (jconsole) in auto mode
```

Step 5: Verify Java Installation

Run:

java -version Expected output:

openjdk version "1.8.0_xxx" ...

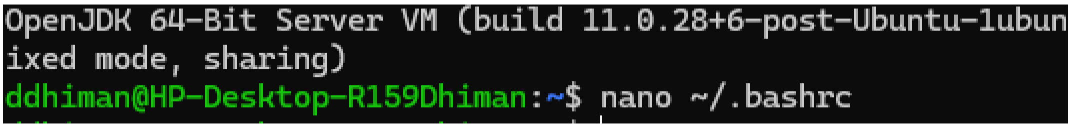


```
ddhiman@HP-Desktop-R159Dhiman:~$ java -version
openjdk version "11.0.28" 2025-07-15
OpenJDK Runtime Environment (build 11.0.28+6-post-Ubuntu-1ubuntu124.04.1)
OpenJDK 64-Bit Server VM (build 11.0.28+6-post-Ubuntu-1ubuntu124.04.1, mixed mode, sharing)
ddhiman@HP-Desktop-R159Dhiman:~$ |
```

Step 6: Set JAVA_HOME Environment Variable (Optional)

1. Open .bashrc file:

nano ~/.bashrc

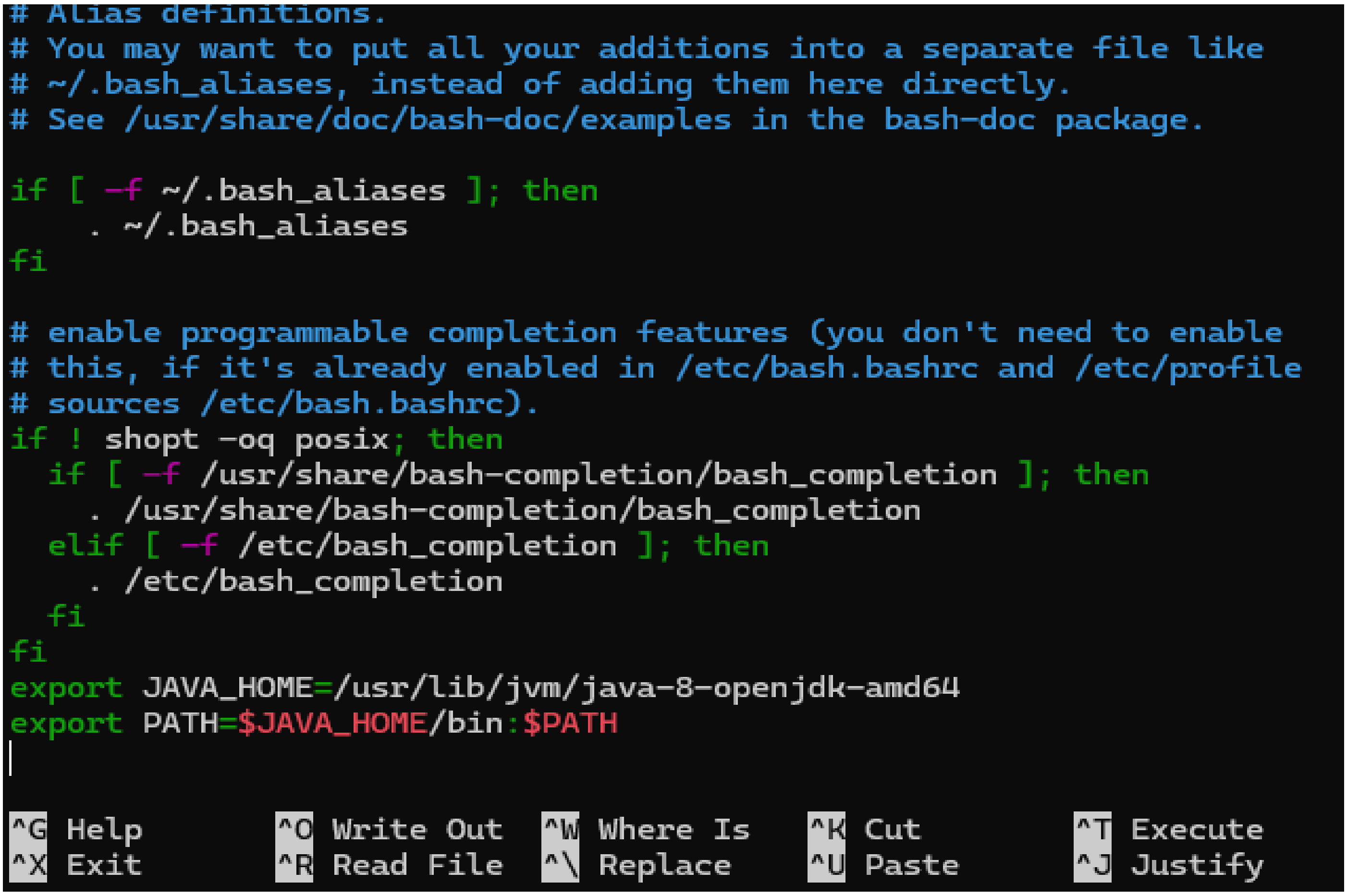


```
OpenJDK 64-Bit Server VM (build 11.0.28+6-post-Ubuntu-1ubun
ixed mode, sharing)
ddhiman@HP-Desktop-R159Dhiman:~$ nano ~/.bashrc
```

2. Add the following lines at the end:

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

export PATH=$JAVA_HOME/bin:$PATH



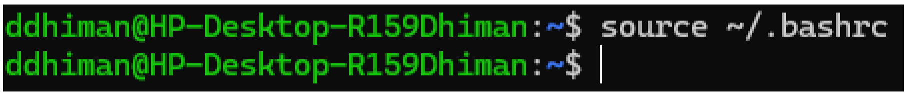
```
# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi
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)

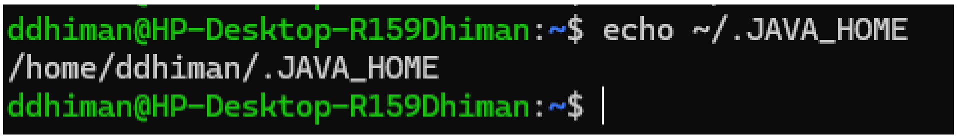
4. Reload bashrc: source ~/.bashrc



```
ddhiman@HP-Desktop-R159Dhiman:~$ source ~/.bashrc
ddhiman@HP-Desktop-R159Dhiman:~$
```

5. Verify JAVA_HOME is set:

echo $JAVA_HOME



```
ddhiman@HP-Desktop-R159Dhiman:~$ echo ~/.JAVA_HOME
/home/ddhiman/.JAVA_HOME
ddhiman@HP-Desktop-R159Dhiman:~$
```

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