Ex.No.3 Implementation of RMI in VMs

**Aim:**

To run RMI using java in Ubuntu on VMWare and Oracle VM Virtual Box and

compare their performance.

**Procedure:**

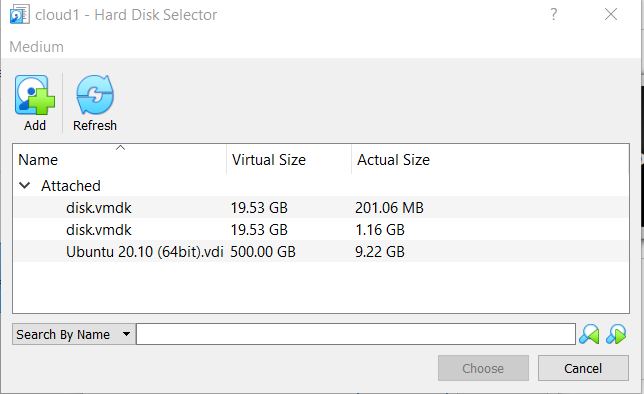
**Oracle VM Virtual Box**

1.Install Oracle VM VirtualBox in your machine.

2.Download ubuntu VirtualBox file from the website. <https://www.osboxes.org/ubuntu/>

3.Extract the vdi file.

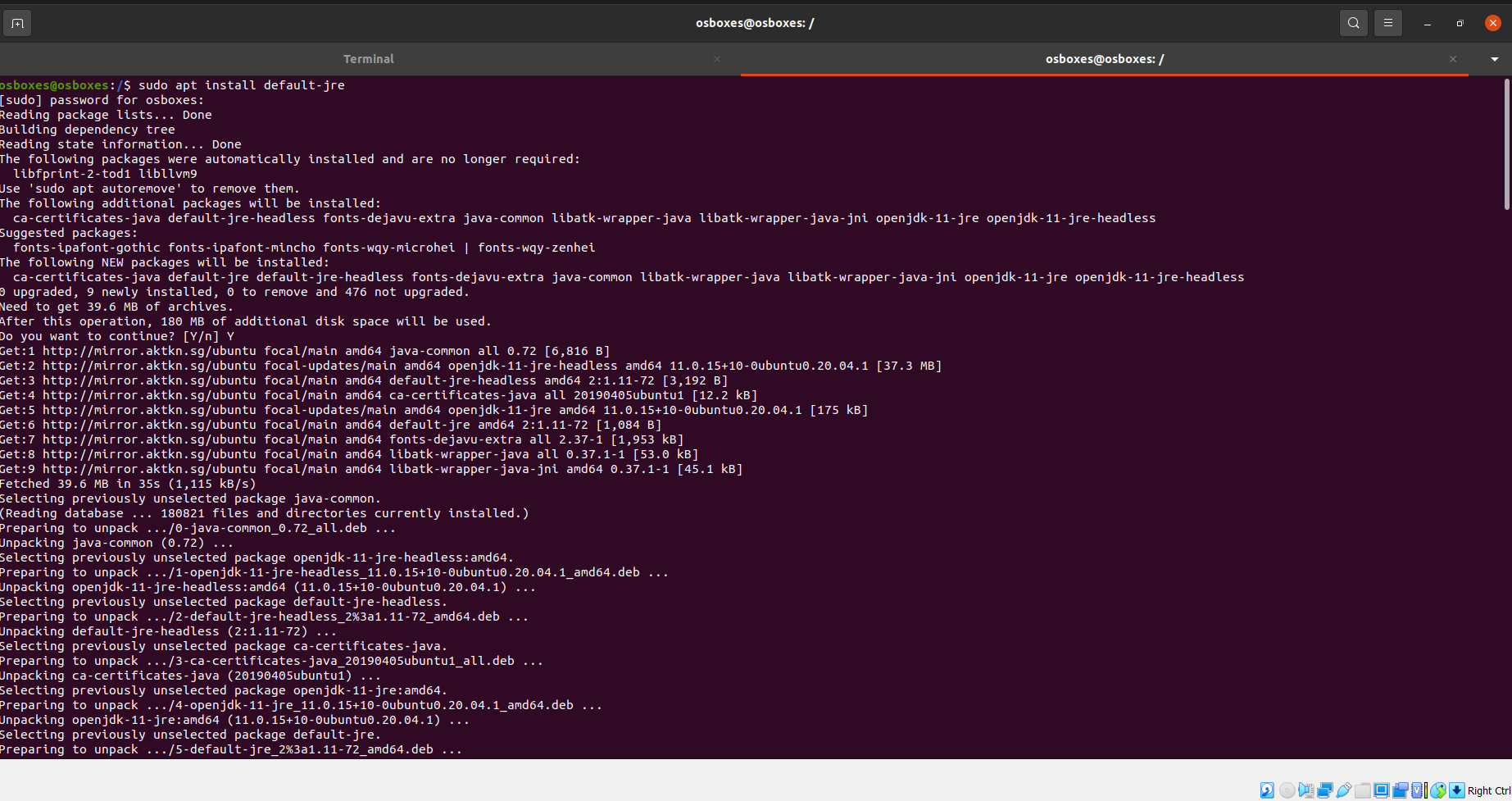
4.Open the image through the VirtualBox

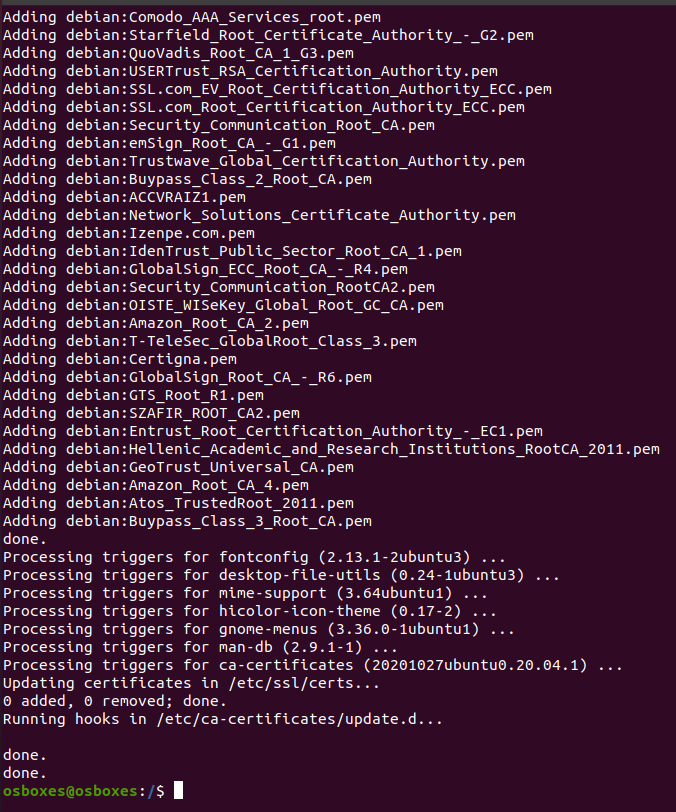


Installing JDK in Ubuntu Linux:

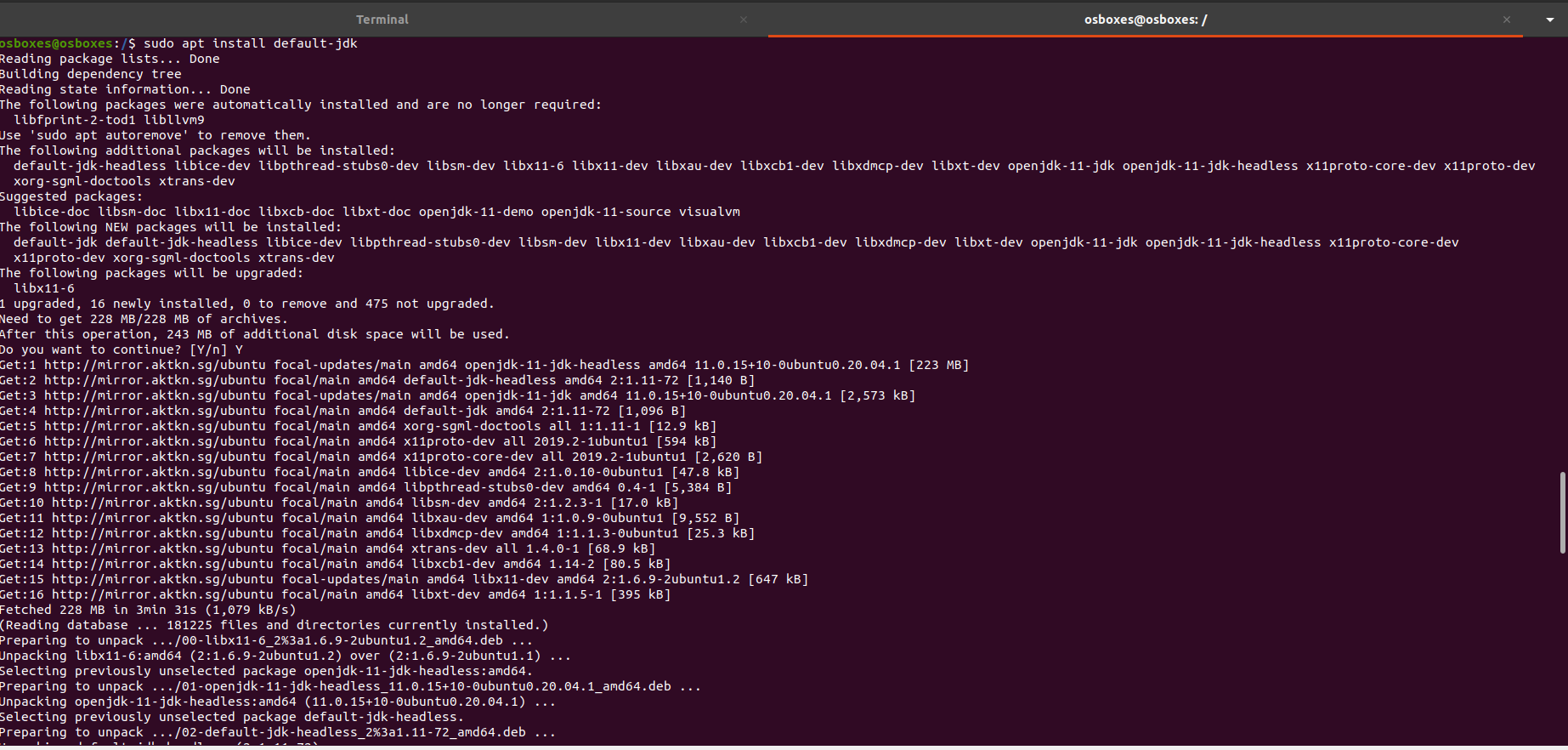
sudo apt install default-jre (3 mins 10 sec)

sudo apt install default-jdk (17 mins 40 secs)

INSTALLING JRE IN UBUNTU:  




JDK:



**LOAD ALL YOUR JAVA AND CLASS FILES INTO YOUR UBUNTU OS:**

**CODE:**

**App.java**

import java.rmi.\*;

public interface App extends Remote{

public String mathe(int x, int y)throws RemoteException;

}

**AppRemote.java**

import java.rmi.\*; import java.rmi.server.\*; import java.util.\*; import java.io.\*;

public class AppRemote extends UnicastRemoteObject implements App{ AppRemote()throws RemoteException{

super();}

public String mathe(int x, int y){

int temp,a; int c = 0;

String ans1 = "Armstrong Number";

String ans2 = "Not an Armstrong Number";

String ans3 = "Neon Number";

String ans4 = "Not an Neon Number"; if(y==1){

temp = x; while(x>0)

{ a=x%10;

x=x/10; c=c+(a\*a\*a);}

if(temp==c) return ans1; else

return ans2;}

else if(y==2){

int square=x\*x; int sum=0; while(square!=0){

int b=square%10; sum=sum+b; square=square/10;}

if(sum==x){

return ans3;}

else{

return ans4;}}

else if(y==3){

int i, last = 0 ;

Scanner sc = new Scanner(System.in); System.out.println("\n Please Enter any Number: "); int Number = sc.nextInt();

for(i = 1 ; i < x ; i++) { if(x % i == 0) {

last = last + i;}}

if (last == x) {

return "Perfect Number";}

else {

return "Not a Perfect Number";}}

else{

return "Sorry, Wrong Choice";}}}

**MyServer.java**

import java.rmi.\*;

import java.rmi.registry.\*; public class MyServer{

public static void main(String args[]){ try{

App stub=new AppRemote(); Naming.rebind("rmi://localhost:5000/sonoo",stub);

} catch(Exception e){System.out.println(e);}}}

**MyClient.java**

import java.rmi.\*; import java.io.\*; import java.util.\*;

public class MyClient{

public static void main(String args[]){ try{

int a, b;

App stub=(App)Naming.lookup("rmi://localhost:5000/sonoo");

Scanner sc = new Scanner(System.in); System.out.println("1.Armstrong \n 2. Neon");

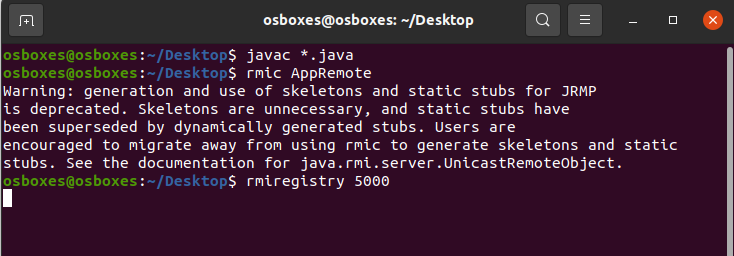
System.out.println("Enter choice");

b =sc.nextInt(); System.out.println("Enter number"); a = sc.nextInt(); System.out.println(stub.mathe(a, b));

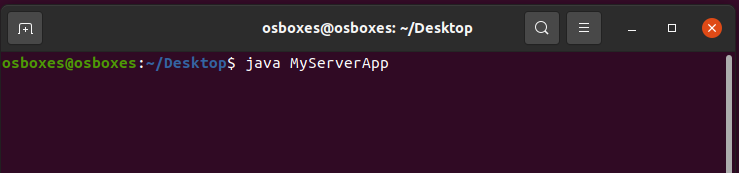
}catch(Exception e){}}}

**After loading compile all the files**

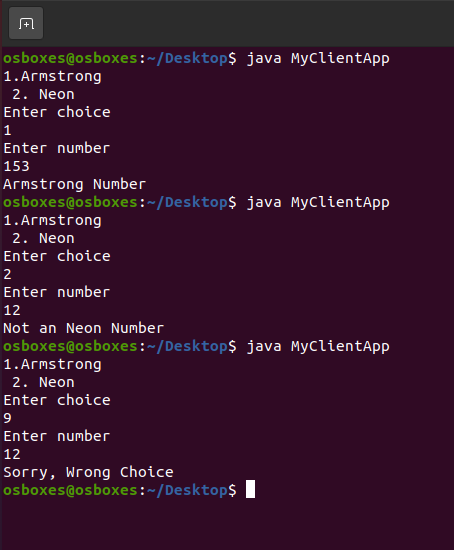
Run the command **rmic AppRemote and then rmiregistry 5000**



Open a new terminal and run the command **java MyServerApp**

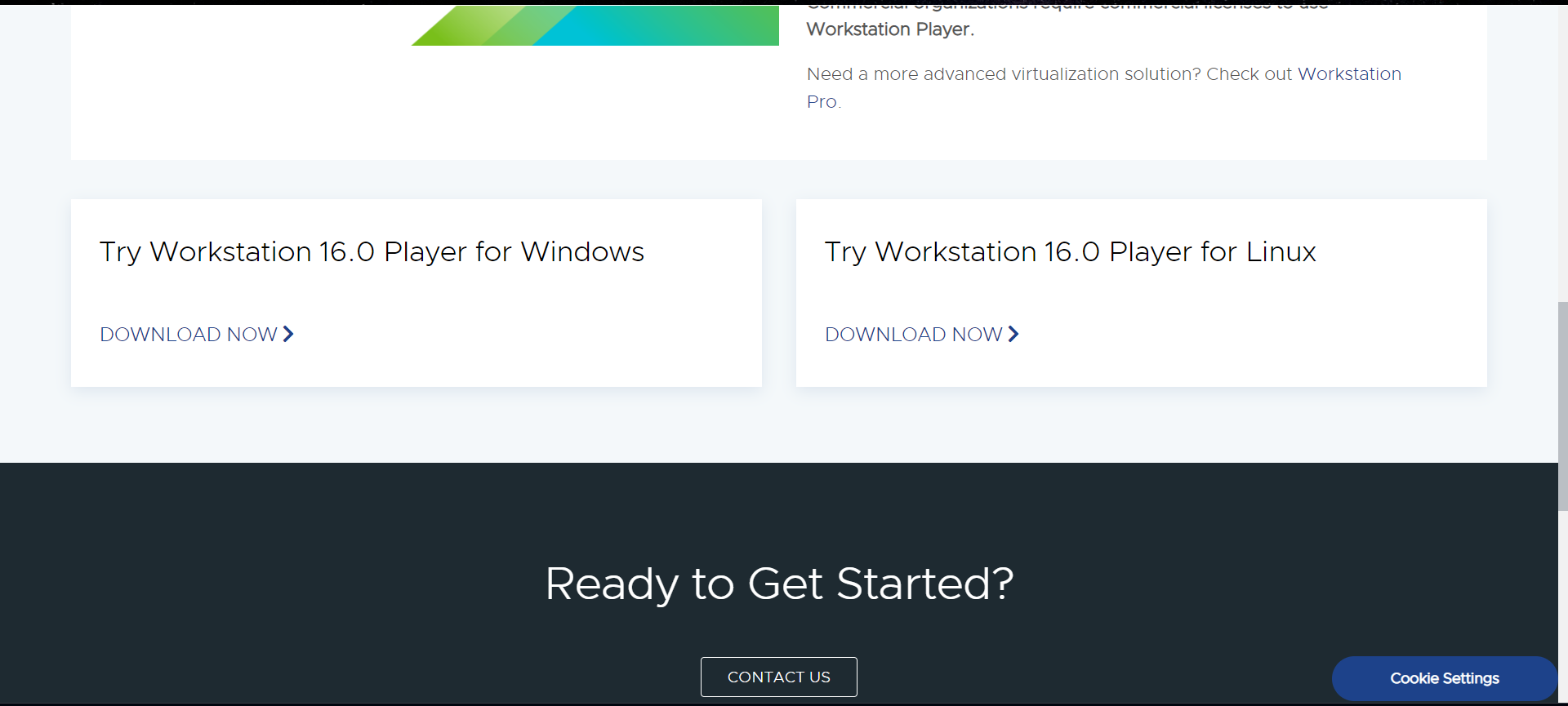


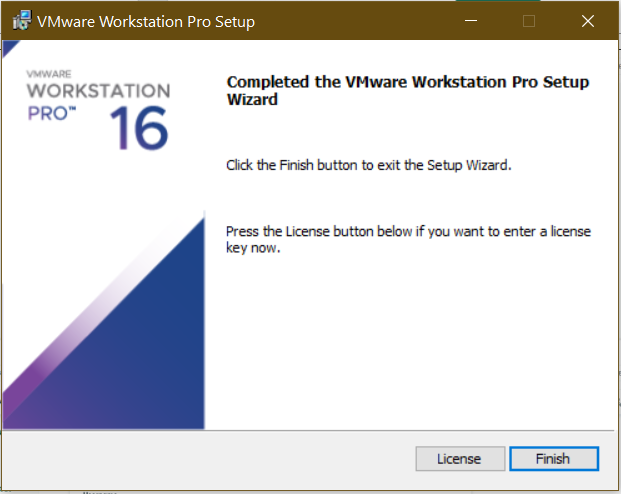
In another terminal run the command **java MyClientApp.**

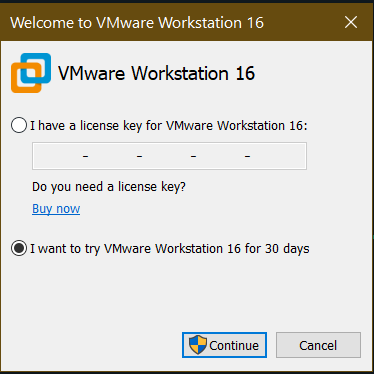


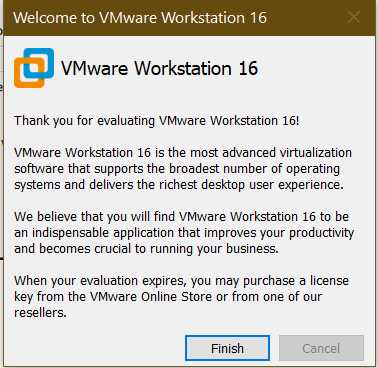
**VMWARE:**

1.Install VMWare in your machine.









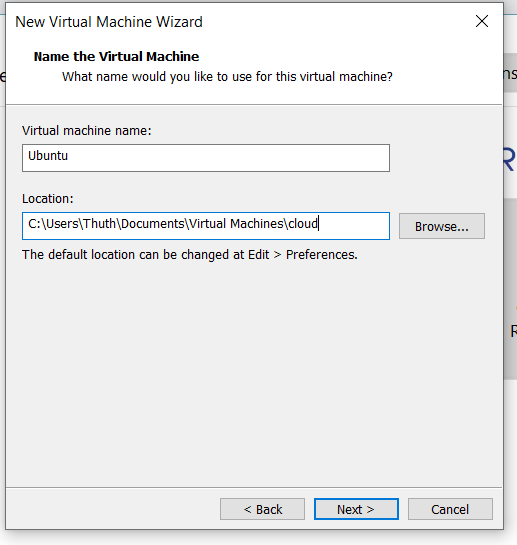
2.Download ubuntu VMware image file from the website. <https://www.osboxes.org/ubuntu/>

3.After installation choose “create new Virtual Machine”

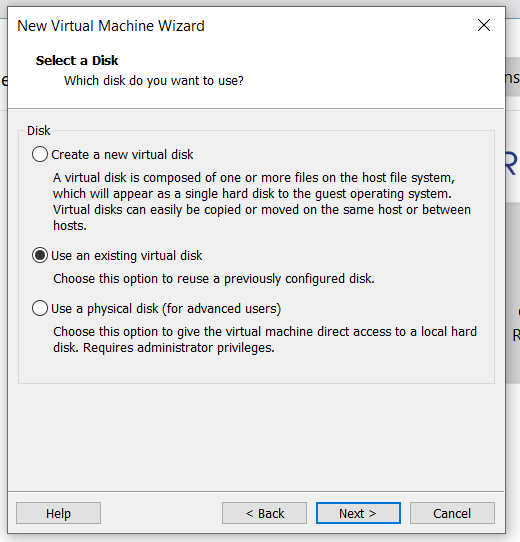
4.Click custom, select hardware requirements, and then click next.

5.Choose “I will install OS later”.

6.Select the guest operating system that is installed, including the version. Click Next.

7. Provide a file name and choose the location where you want to save the virtual machine and click next.

8.Click next and keep moving on. When prompted to select a disk choose “use an existing virtual disk”.



9.Once done open the VM for usage.

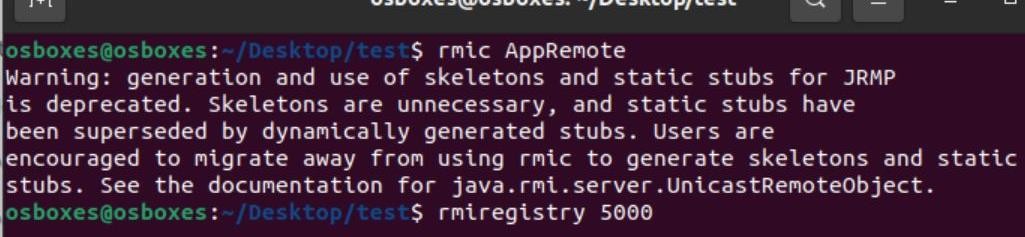
10.Install OpenJDK in ubuntu.

sudo apt install default-jre (3 mins 10 sec)

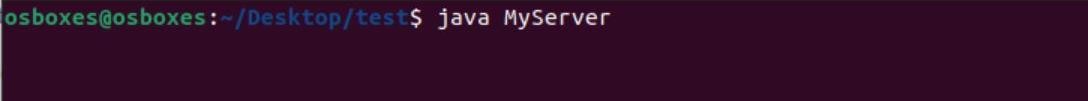
sudo apt install default-jdk (17 mins 40 secs)

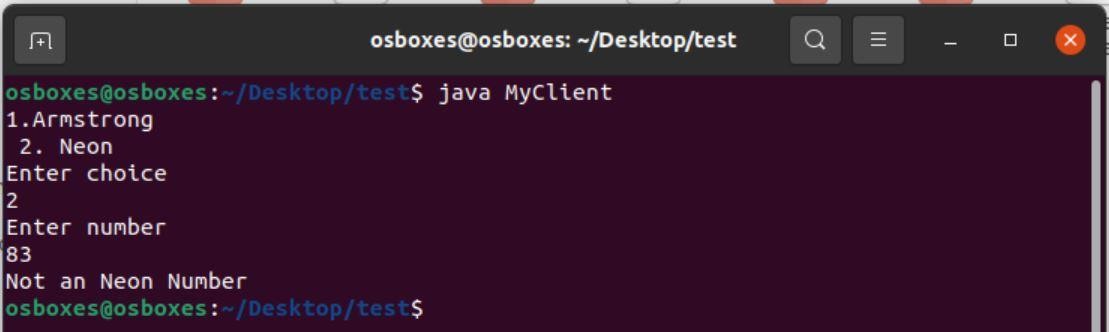
11.Copy the java and Class files to ubuntu OS.

12.Run rmic AppRemote

13. Run rmi registry with the port number.

14. Run java MyServer



15. Run java MyClient

**Result:**

The RMI using java in Ubuntu on VMWare and Oracle VM Virtual Box is done and their performance is compared.