**Exp - 2- Installing and Creating Virtual Machine**

1. Download the Oracle Virtual Box

Select the respective components and path for installation

Continue the installation wizard, install and finish it

2. Download the Ubuntu and Fedora Guest Os

3. In Oracle Virtual Box

Common for both Ubuntu and Fedora

Click New

Enter the name and select the respective ISO file

Choose the username, respective RAM and Hard disk. finish

Start the VM

Ubuntu

Install and skip login to account

Continue the Installation procedure and finish it

Fedora

Start the Fedora Workstation Live 37

Install to Hard Drive

Continue and In Installation Destination Click done

Install the Fedora and Finish it

su root

nano /etc/sudoers

user ALL=(ALL) ALL

**Exp-3 Working in VMs**

**Ubuntu**

vboxuser@Ubuntu:~$ sudo apt update

[sudo] password for vboxuser:

vboxuser is not in the sudoers file. This incident will be reported.

vboxuser@Ubuntu:~$ su

Password:

root@Ubuntu:/home/vboxuser# sudo apt update

root@Ubuntu:/home/vboxuser# cc

cc: fatal error: no input files

compilation terminated.

root@Ubuntu:/home/vboxuser# sudo apt install gcc

root@Ubuntu:/home/vboxuser# gcc --version

root@Ubuntu:/home/vboxuser# sudo apt install python3

root@Ubuntu:/home/vboxuser# python3 --version

root@Ubuntu:/home/vboxuser# java

root@Ubuntu:/home/vboxuser# sudo apt install default-jre

root@Ubuntu:/home/vboxuser# java -version

root@Ubuntu:/home/vboxuser# sudo apt install default-jdk

root@Ubuntu:/home/vboxuser# javac -version

root@Ubuntu:/home/vboxuser# ls

root@Ubuntu:/home/vboxuser# cd Documents

root@Ubuntu:/home/vboxuser/Documents# touch file.c

root@Ubuntu:/home/vboxuser/Documents#

root@Ubuntu:/home/vboxuser/Documents# vi file.c

root@Ubuntu:/home/vboxuser/Documents#

root@Ubuntu:/home/vboxuser/Documents# touch python.py

root@Ubuntu:/home/vboxuser/Documents# vi python.py

root@Ubuntu:/home/vboxuser/Documents#

root@Ubuntu:/home/vboxuser/Documents# touch javafile.java

root@Ubuntu:/home/vboxuser/Documents# vi javafile.java

root@Ubuntu:/home/vboxuser/Documents#

root@Ubuntu:/home/vboxuser/Documents# cc file.c

root@Ubuntu:/home/vboxuser/Documents# ./a.out

Hello Worldroot@Ubuntu:/home/vboxuser/Documents#

root@Ubuntu:/home/vboxuser/Documents# python3 python.py

Hello Python

root@Ubuntu:/home/vboxuser/Documents# javac javafile.java

root@Ubuntu:/home/vboxuser/Documents# java javafile

Hello Java

**For fedora**

Similar above command instead of apt dnf

sudo dnf update

sudo dnf groupinstall "Development Tools"

[liveuser@localhost-live ~]$ sudo dnf install gcc  
  
[liveuser@localhost-live ~]$ gcc --version  
  
  
[liveuser@localhost-live ~]$ sudo dnf  install python3  
  
[liveuser@localhost-live ~]$ python3 --version  
Python 3.11.0  
  
[liveuser@localhost-live ~]$ sudo dnf install java-devel  
  
[liveuser@localhost-live ~]$ java -version  
  
[liveuser@localhost-live ~]$ cd Documents  
[liveuser@localhost-live Documents]$  
[liveuser@localhost-live Documents]$ touch file.c  
[liveuser@localhost-live Documents]$ touch file2.py  
[liveuser@localhost-live Documents]$ touch file3.java  
  
[liveuser@localhost-live Documents]$  
[liveuser@localhost-live Documents]$ cc file.c  
[liveuser@localhost-live Documents]$ ./a.out  
hello C  
  
[liveuser@localhost-live Documents]$ python3 file2.py  
Hello World  
  
  
[liveuser@localhost-live Documents]$ javac file3.java  
[liveuser@localhost-live Documents]$ java file3.java  
Hello java

file.c  
  
#include<stdio.h>  
void main(){  
printf("hello C");  
}  
  
file2.py  
  
print("Hello World")  
  
file3.java  
  
class file3{  
public static void main(String args[]){  
System.out.println("Hello java");  
}  
}

**Exp-4- Communication between VMs**

1. Open Two VMs (Ubuntu and Fedora)

2. Go to Settings and Select the Network.

3. In Adapter 2 Select the Host-only Adapter in the Attached to section and

select Allow All in the Promiscuous Mode in Advanced section

4. Save it. Incorporate the above changes in both the VMS and Start both the

VMs

Simultaneous execution on two VMs

**1. Ubuntu to Fedora**

**Ubuntu**

vboxuser@Ubuntu:~$ su

Password:

root@Ubuntu:/home/vboxuser# sudo apt install openssh-server

root@Ubuntu:/home/vboxuser#

root@Ubuntu:/home/vboxuser# sudo apt install openssh-client

root@Ubuntu:/home/vboxuser#

root@Ubuntu:/home/vboxuser# scp

**Fedora**

Set password for user under users in settings

[liveuser@localhost-live ~]$ sudo dnf install openssh-server

[liveuser@localhost-live ~]$ sudo dnf install openssh-client

[liveuser@localhost-live ~]$ ifconfig

192.168.56.102

[liveuser@localhost-live ~]$ pwd

/home/liveuser

[liveuser@localhost-live ~]$ whoami

liveuser

[liveuser@localhost-live ~]$ sudo systemctl start sshd

[liveuser@localhost-live ~]$ ls

Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos

[liveuser@localhost-live ~]$ cd Desktop

[liveuser@localhost-live Desktop]$ pwd

/home/liveuser/Desktop

[liveuser@localhost-live Desktop]$

**Ubuntu**

root@Ubuntu:/home/vboxuser# ping 192.168.56.102

root@Ubuntu:/home/vboxuser# cd Desktop

root@Ubuntu:/home/vboxuser/Desktop# ls

root@Ubuntu:/home/vboxuser/Desktop# touch hello.txt

root@Ubuntu:/home/vboxuser/Desktop# vi hello.txt

root@Ubuntu:/home/vboxuser/Desktop# cat hello.txt

Ubuntu to Fedora

root@Ubuntu:/home/vboxuser/Desktop# ls

root@Ubuntu:/home/vboxuser/Desktop#

root@Ubuntu:/home/vboxuser/Desktop#

scp hello.txt [liveuser@192.168.56.102:/home/liveuser/Desktop](mailto:liveuser@192.168.56.102:/home/liveuser/Desktop)

Enter Fedora password

root@Ubuntu:/home/vboxuser/Desktop#

**Fedora**

[liveuser@localhost-live Desktop]$ ls

hello.txt

[liveuser@localhost-live Desktop]$ cat hello.txt

Ubuntu to Fedora

**2. Fedora to Ubuntu**

**Fedora**

[liveuser@localhost-live ~]$ su

[root@localhost-live liveuser]# sudo dnf install openssh-server

[root@localhost-live liveuser]# sudo dnf install openssh-client

[root@localhost-live liveuser]# sudo systemctl start sshd

**Ubuntu**

vboxuser@Ubuntu:~$ sudo apt install openssh-server

vboxuser@Ubuntu:~$ sudo apt install openssh-client

vboxuser@Ubuntu:~$ ifconfig

192.168.56.101

vboxuser@Ubuntu:~$ pwd

/home/vboxuser

vboxuser@Ubuntu:~$ whoami

vboxuser

vboxuser@Ubuntu:~$ cd Desktop

vboxuser@Ubuntu:~/Desktop$ ls

Docker Docker1 hello.txt webapp webapplication

vboxuser@Ubuntu:~/Desktop$

**Fedora**

[root@localhost-live liveuser]# ping 192.168.56.101

[root@localhost-live liveuser]# ls

Desktop Documents Downloads Music Pictures Public Templates Videos

[root@localhost-live liveuser]# cd Desktop

[root@localhost-live Desktop]# ls

hello.txt

[root@localhost-live Desktop]# touch file2.txt

[root@localhost-live Desktop]# vi file2.txt

[root@localhost-live Desktop]# cat file2.txt

Fedora to Ubuntu

[root@localhost-live Desktop]#

scp file2.txt vboxuser@192.168.56.101:/home/vboxuser/Desktop

Enter Ubuntu password

**Ubuntu**

vboxuser@Ubuntu:~/Desktop$ ls

Docker Docker1 file2.txt hello.txt webapp webapplication

vboxuser@Ubuntu:~/Desktop$ cat file2.txt

Fedora to Ubuntu

vboxuser@Ubuntu:~/Desktop$

**Exercise 5 – Running a Web application in VMs**

**Ubuntu Terminal 1**

vboxuser@Ubuntu:~$ sudo apt update

vboxuser@Ubuntu:~$ sudo apt install tomcat9 tomacat9-admin

vboxuser@Ubuntu:~$ sudo systemctl start tomcat9

vboxuser@Ubuntu:~$ sudo systemctl status tomcat9

vboxuser@Ubuntu:~$ sudo apt install npm

vboxuser@Ubuntu:~$ npm init -y

vboxuser@Ubuntu:~$ npm install express

In home location move the files generated like package.json , package-lock.json and node\_modules to a new folder called webapp.(created by us)

Download visual studio from Ubuntu Software and open the above webapp folder.

**Create a new file app.js and paste the code**

const express = require('express');

const app = express()

const port = 3002;

app.get("/",(request,response)=>{

response.send("Hello World!");

});

app.listen(port,()=>{

console.log('Server Started at port');

});

In case if error generated change the port number in all the places

**Open a new terminal 2 type the following code**

vboxuser@Ubuntu:~$ ls

Desktop Downloads Pictures snap Videos

Documents Music Public Templates webapp

vboxuser@Ubuntu:~$ cd webapp

vboxuser@Ubuntu:~/webapp$ node app.js

Server Started at port

**Terminal 1**

vboxuser@Ubuntu:~$ sudo apt install apache2

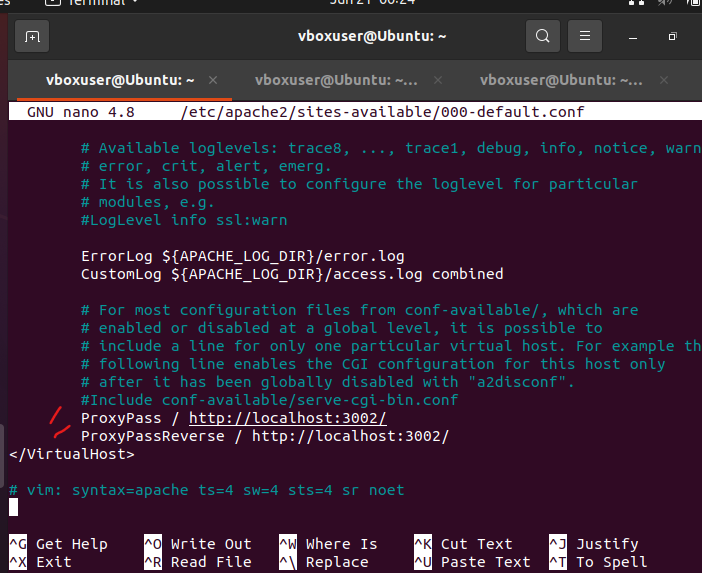
vboxuser@Ubuntu:~$ sudo a2enmod proxy

vboxuser@Ubuntu:~$ sudo a2enmod proxy\_http

vboxuser@Ubuntu:~$ sudo nano /etc/apache2/sites-available/000-default.conf

Write the two code heighted below

Press Ctrl+O Enter Ctrl+X to exit



vboxuser@Ubuntu:~$ sudo systemctl restart apache2

vboxuser@Ubuntu:~$ ifconfig

Note the ipadress of Ubuntu (192.168.56.101)

In fedora open a browser and type

192.168.56.101:3002/

The output is displayed

**Ex-6- Install Docker and create Images, Containers**

su

Change location to Desktop

root@Ubuntu:/home/vboxuser/Desktop# sudo apt update  
  
root@Ubuntu:/home/vboxuser/Desktop# sudo apt install -y [docker.io](http://docker.io/)  
  
root@Ubuntu:/home/vboxuser/Desktop# sudo systemctl start docker  
  
root@Ubuntu:/home/vboxuser/Desktop# sudo systemctl enable docker  
  
root@Ubuntu:/home/vboxuser/Desktop# docker --version  
  
root@Ubuntu:/home/vboxuser/Desktop# nano Dockerfile

Write the code in Dockerfile  
                               
FROM python:3.9  
  
WORKDIR /app  
  
COPY . /app  
  
CMD ["python","app.py"]

Press Ctrl+O Enter and followed by Ctrl+X for Exit

root@Ubuntu:/home/vboxuser/Desktop# nano app.py  
  
Enter the python code of your wish

root@Ubuntu:/home/vboxuser/Desktop# docker build -t my-app .  
  
root@Ubuntu:/home/vboxuser/Desktop# docker run  my-app  
  
root@Ubuntu:/home/vboxuser/Desktop# sudo docker images  
  
root@Ubuntu:/home/vboxuser/Desktop# docker build -t my-app2 .  
  
root@Ubuntu:/home/vboxuser/Desktop# docker run  my-app2

root@Ubuntu:/home/vboxuser/Desktop# sudo docker images

DOCKER SWARM

Sudo docker swarm leave --force

INSTALL DOCKER IN BOTH VMS

In which vm your are running suppose vm1[Ubuntu]

sudo docker pull mysql

sudo docker run -d -p0.0.0.0:80:80 mysql:latest

if port is in use use this command

sudo lsof -i :80

sudo service apache2 stop [again run]

sudo docker run -d -p 80:80 mysql:latest

to list containers

sudo docker ps –a

sudo docker Swarm init --advertise-addr 192.168.2.151[ip address of vm1 manager]

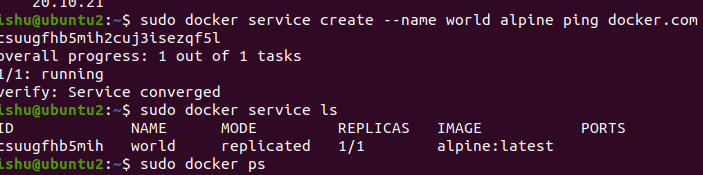
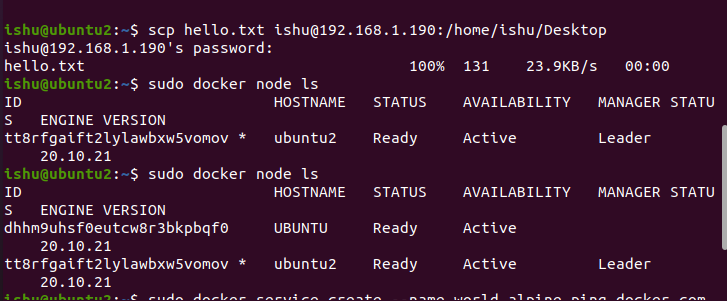
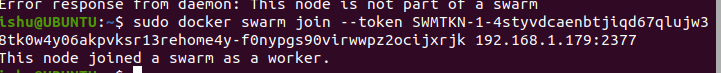
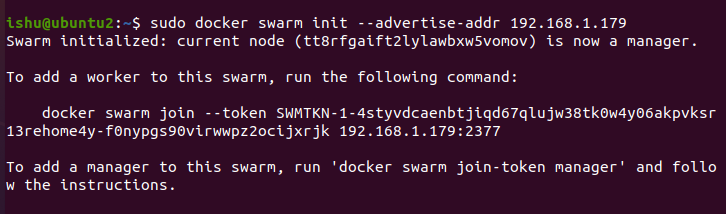
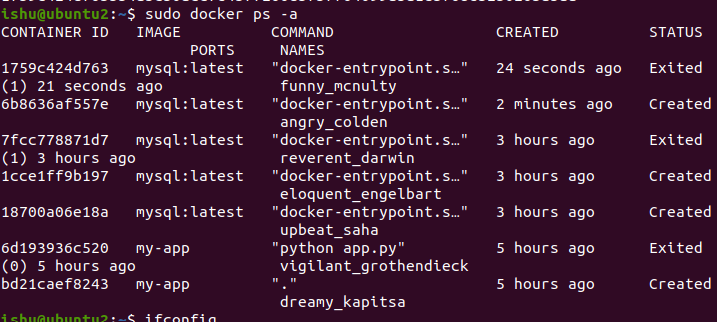
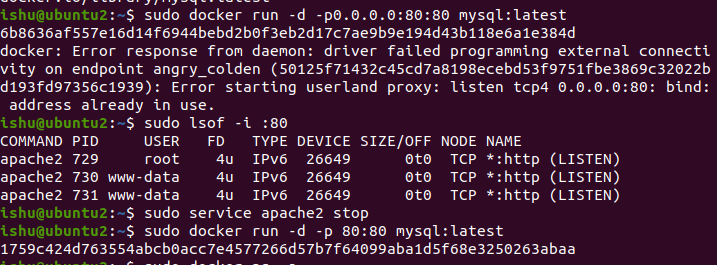
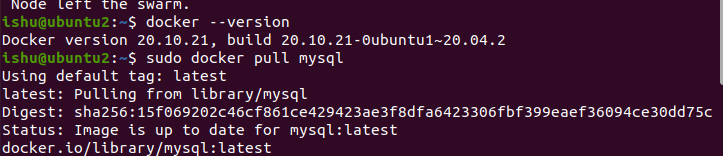
sudo docker [copy the output of previous command] and paste it in vm2 the worker node

IN VM1

sudo docker node ls

sudo docker service create --name HelloWorld alpine ping docker.com

sudo docker service ls



import numpy as np

from sklearn.datasets import load\_iris

from sklearn.model\_selection import train\_test\_split

from sklearn.neighbors import KNeighborsClassifier

from sklearn.metrics import accuracy\_score

# Load the Iris dataset

iris = load\_iris()

X = iris.data

y = iris.target

# Split the dataset into training and test sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Create a KNN classifier

k = 3

knn = KNeighborsClassifier(n\_neighbors=k)

# Train the classifier

knn.fit(X\_train, y\_train)

# Make predictions on the test set

y\_pred = knn.predict(X\_test)

# Calculate the accuracy of the classifier

accuracy = accuracy\_score(y\_test, y\_pred)

print("Accuracy:", accuracy)

pip install scikit-learn

python knn\_algorithm.py