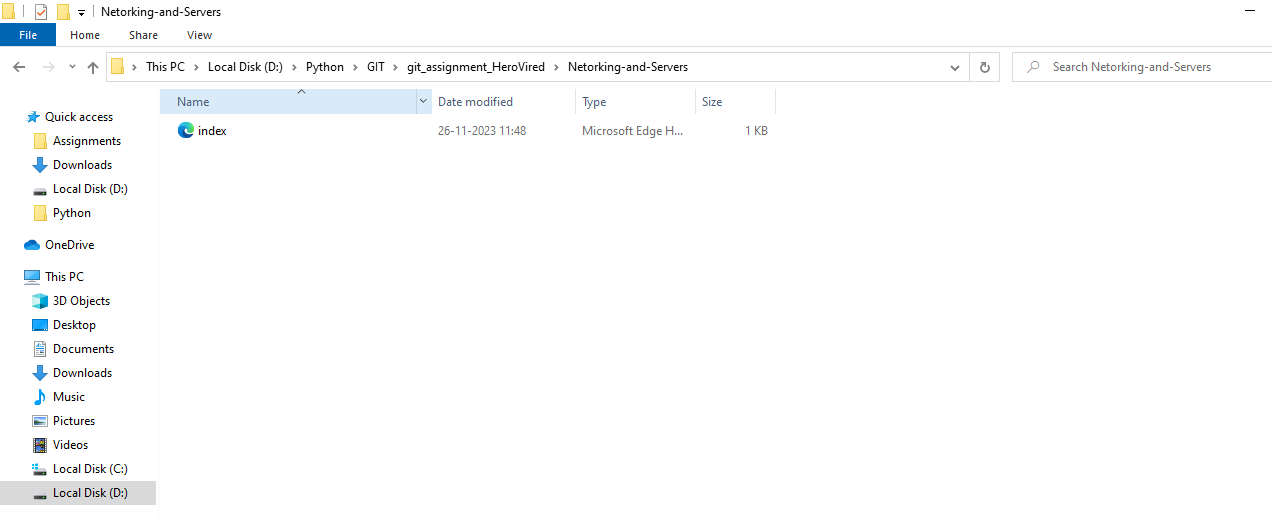
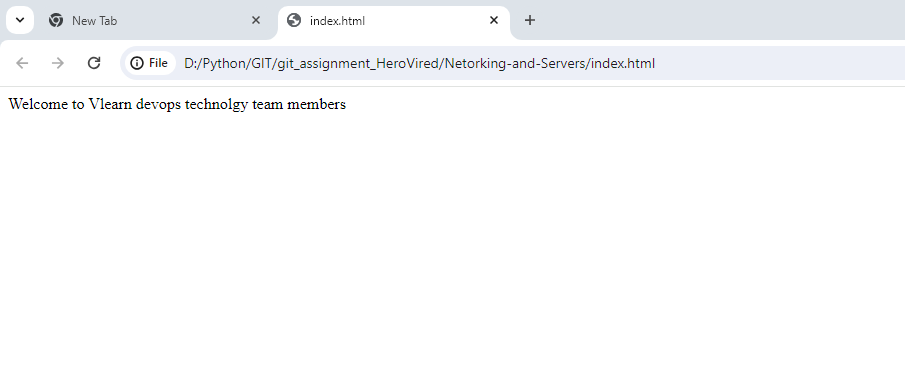
Assignment Name: Networking and servers

Git link:

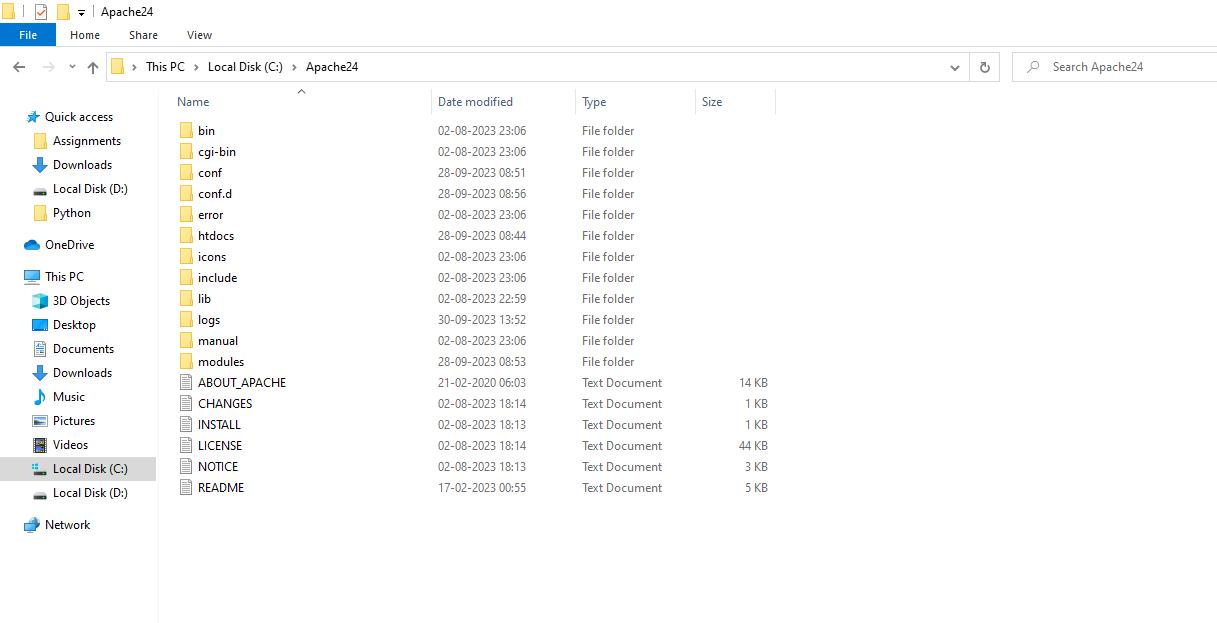
Q1. Deploy a website on localhost using either apache2 or Nginx. Create a DNS name for this website as ‘awesomeweb’. You can use any web template you want or can write your own simple HTML code.

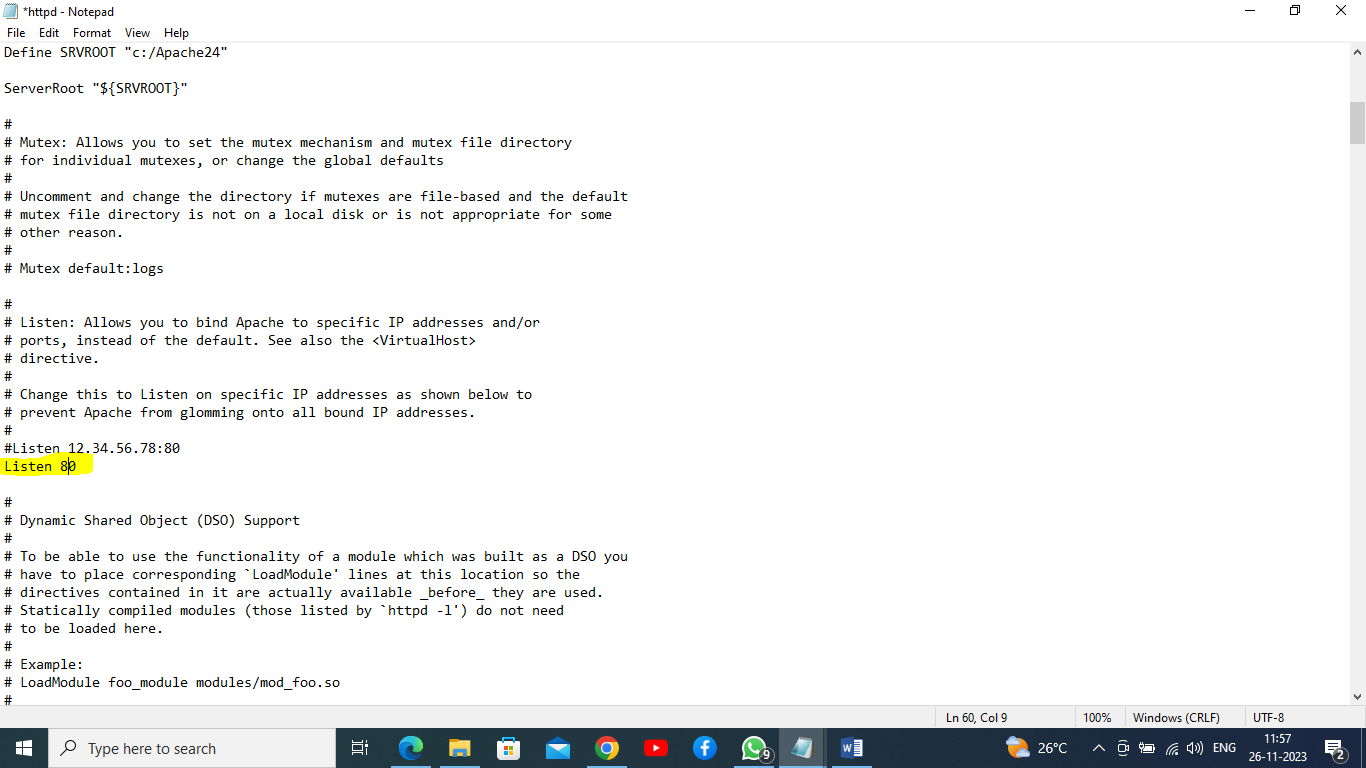
Website index file

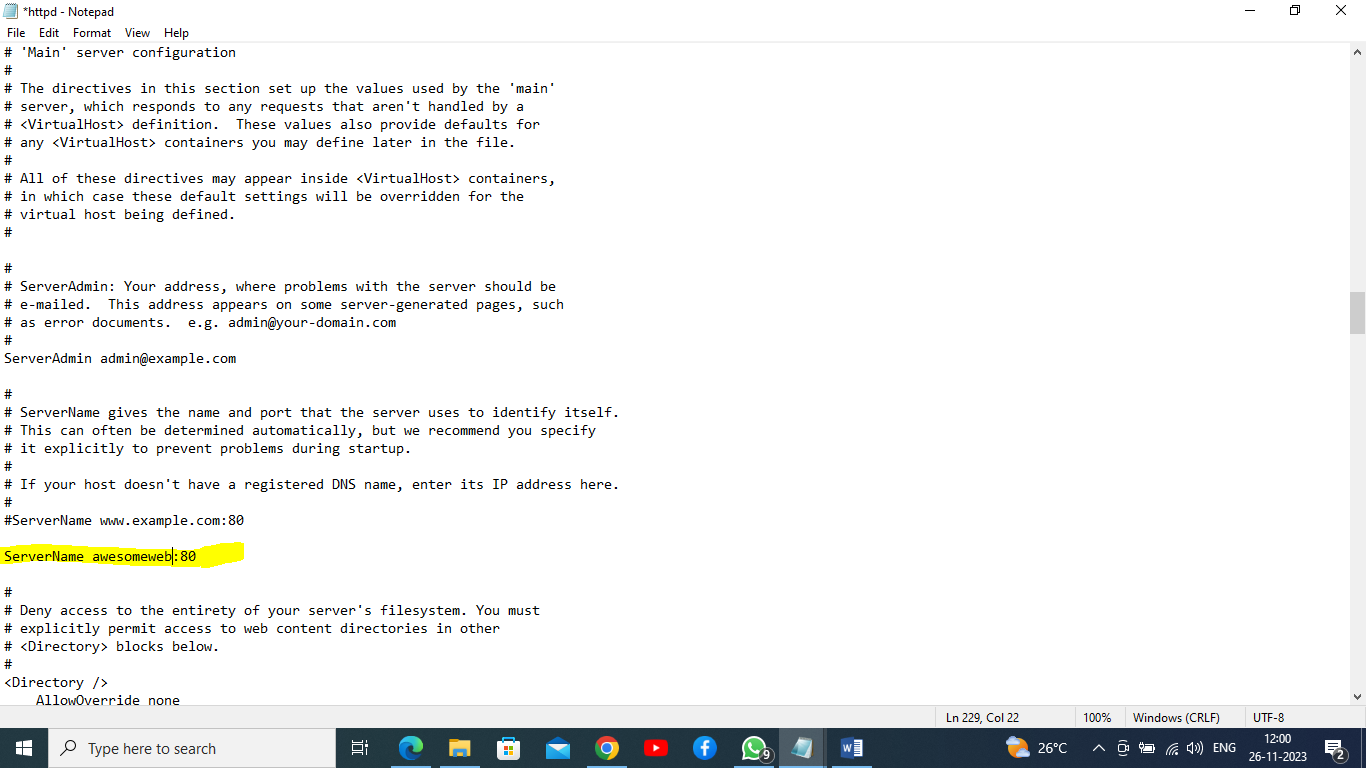


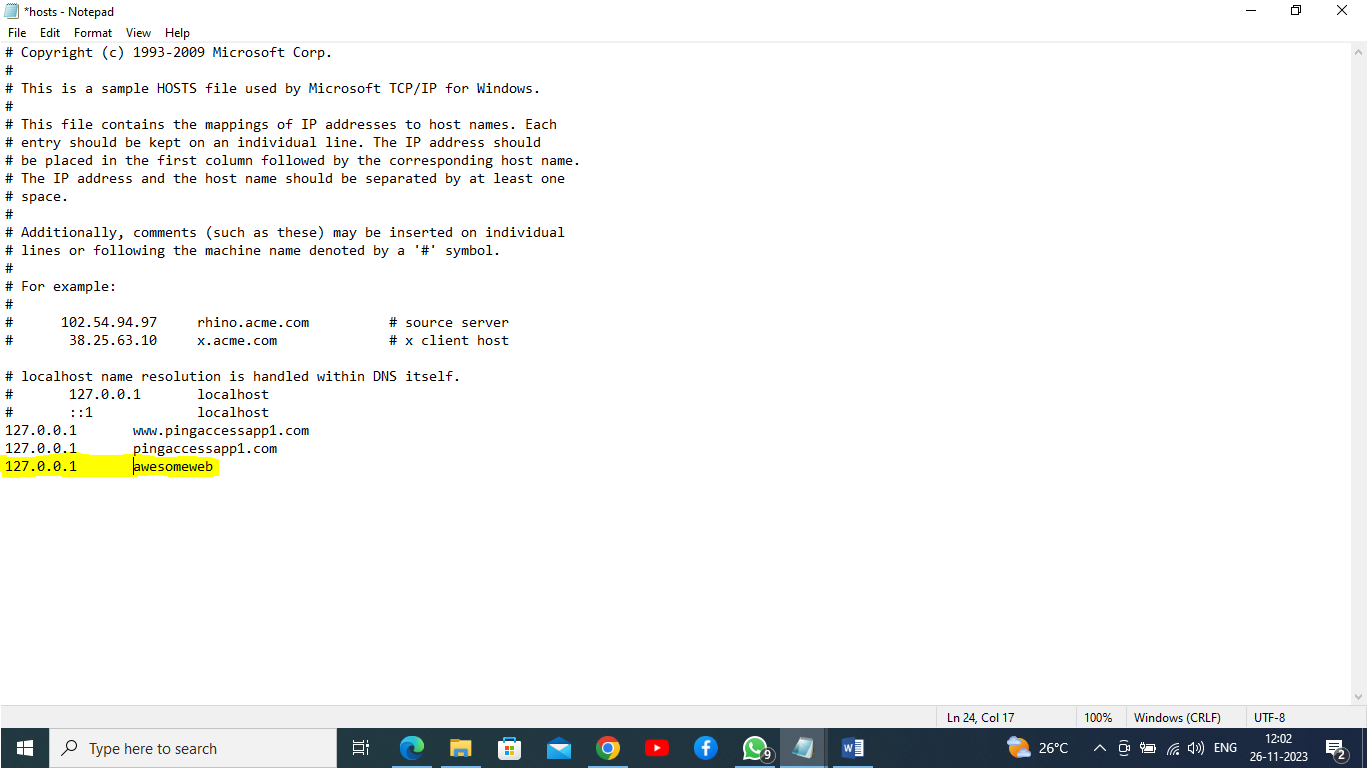


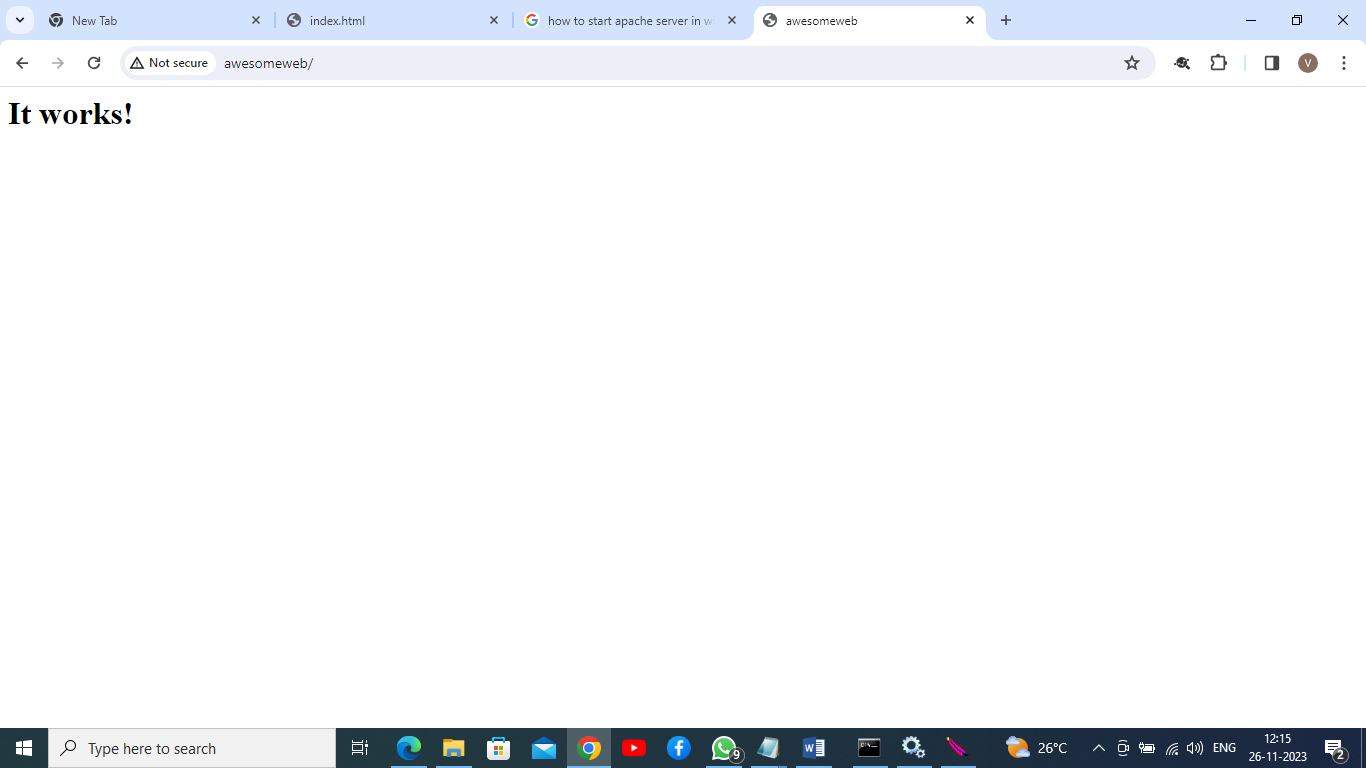
Apache 24 downloaded and extracted fom apache web site





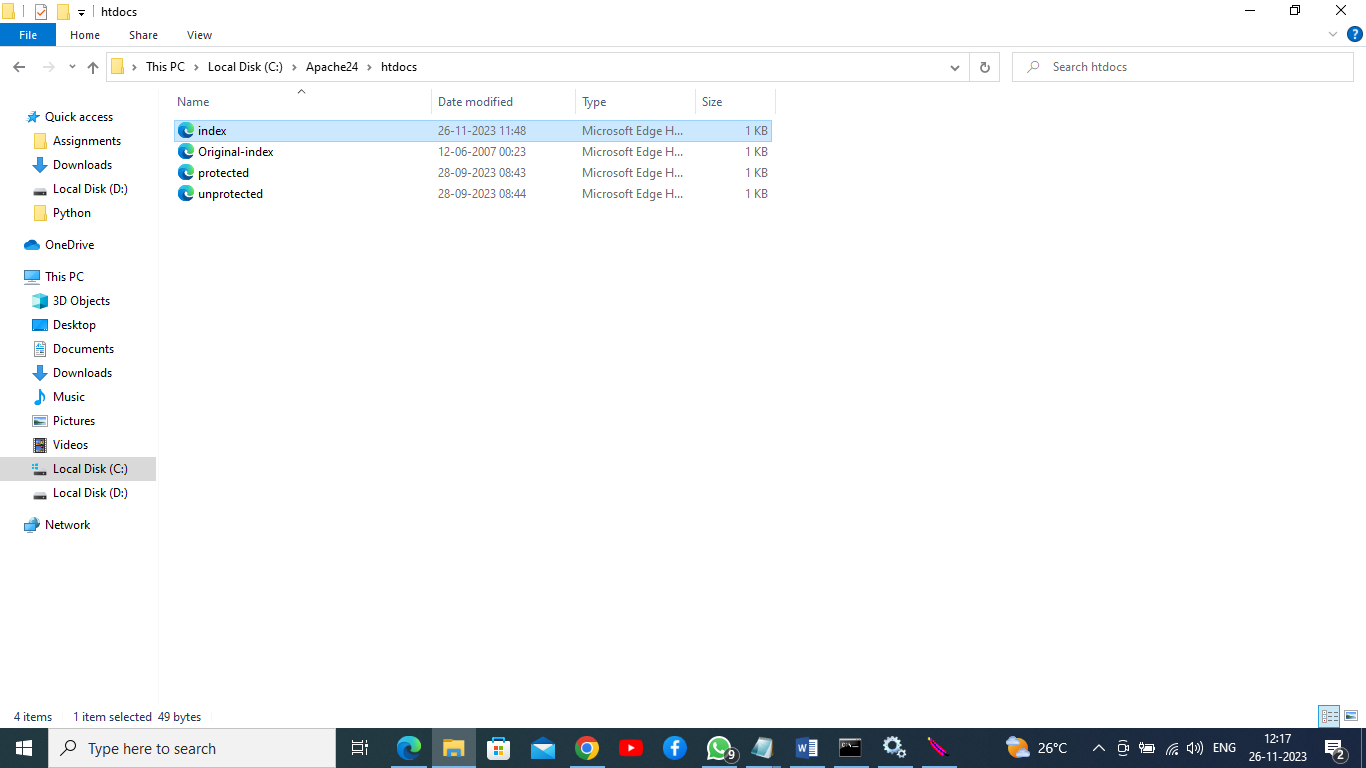






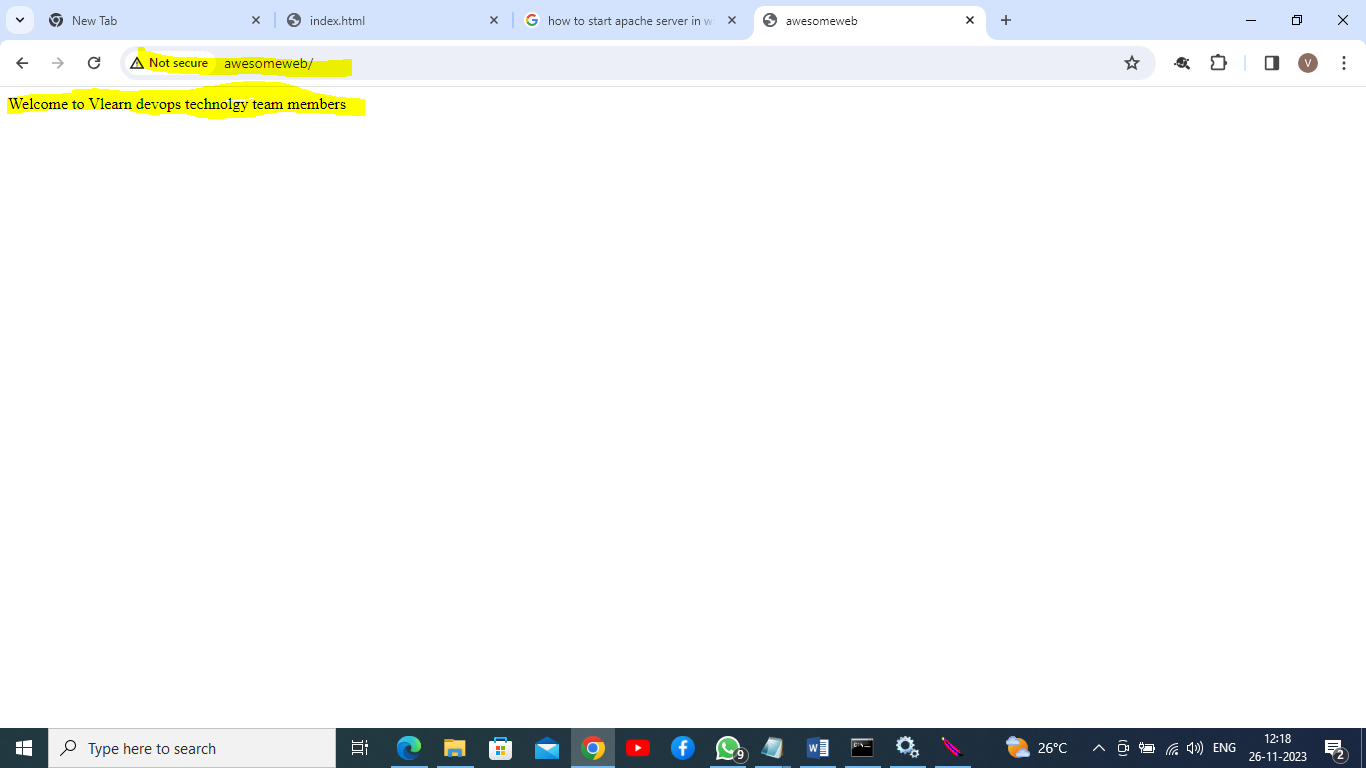
Apache web server started and getting html pages, this page default page.

Now going to deploy our custom web pages in apache server.



Hitting url in browser like: http:// awesomeweb

Result:



Q1 assignment completed.

Q2. A website can have many subdomains and different services are running on them. Write a Python script to check the status of the subdomains which are up or down. The script should automatically check the status every minute and should update it in tabular format on the screen.

Python code:

import requests

import time

from tabulate import tabulate

def check\_subdomain\_status(subdomains):

results = []

for subdomain in subdomains:

url = f"http://{subdomain}"

try:

response = requests.get(url, timeout=5)

status = response.status\_code

except requests.ConnectionError:

status = "Down"

results.append((subdomain, status))

return results

def display\_results(results):

headers = ["Subdomain", "Status"]

table = tabulate(results, headers, tablefmt="grid")

print(table)

def main():

subdomains = ["subdomain1.example.com", "subdomain2.example.com", "subdomain3.example.com"]

while True:

results = check\_subdomain\_status(subdomains)

display\_results(results)

time.sleep(60) # Check every minute

if \_\_name\_\_ == "\_\_main\_\_":

main()

Out put:

PS D:\Python\GIT\git\_assignment\_HeroVired> & "C:/Program Files/Python311/python.exe" d:/Python/GIT/git\_assignment\_HeroVired/Netorking-and-Servers/domain-and-subdomain-health.py

+------------------------+----------+

| Subdomain | Status |

+========================+==========+

| subdomain1.example.com | Down |

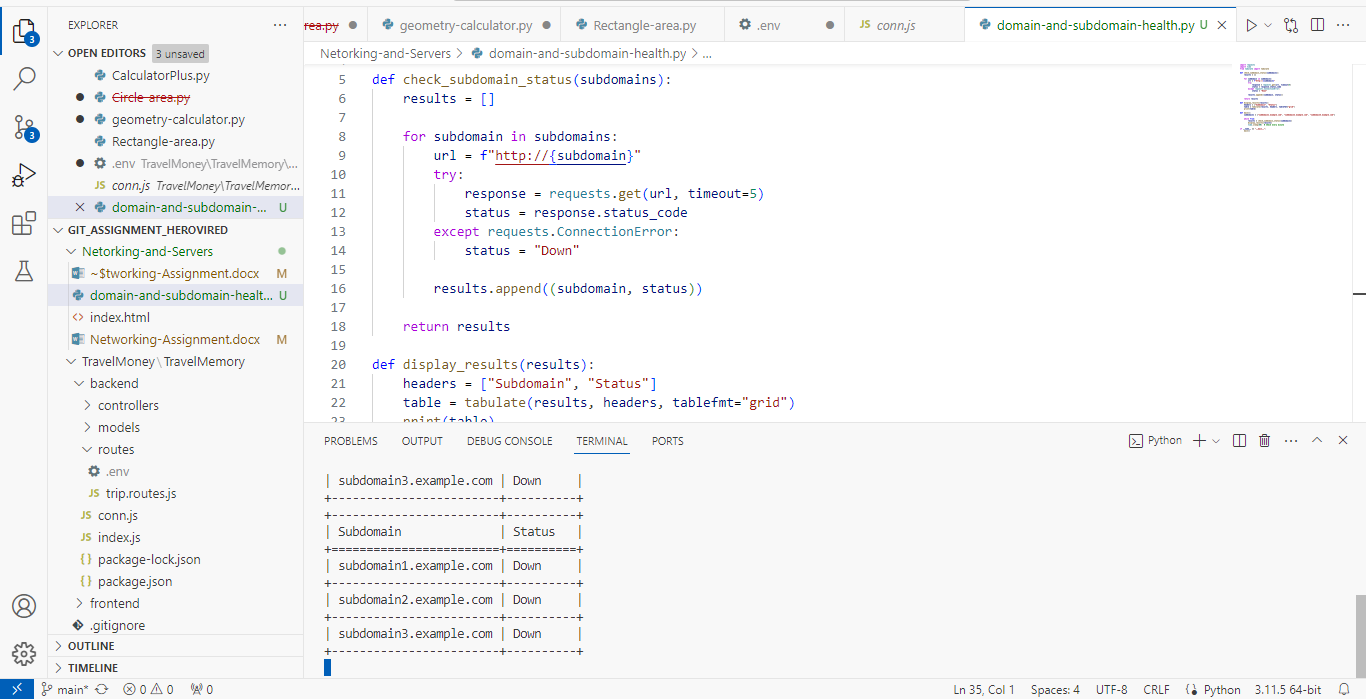
+------------------------+----------+

| subdomain2.example.com | Down |

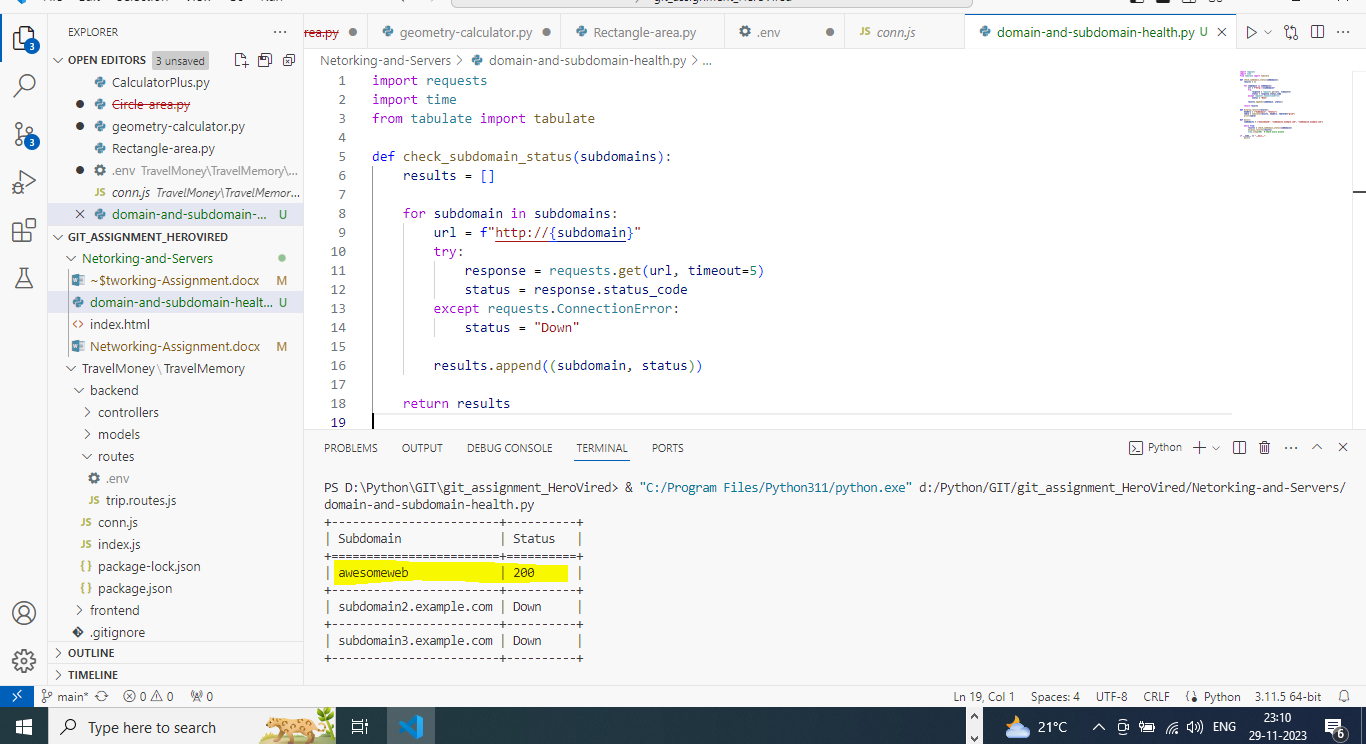
+------------------------+----------+

| subdomain3.example.com | Down |

+------------------------+----------+



Awesomeweb web site hosted in local and its loading in browser and tested from python code , its showing 200 status code in below screen shot.



Q2 assignment completed.

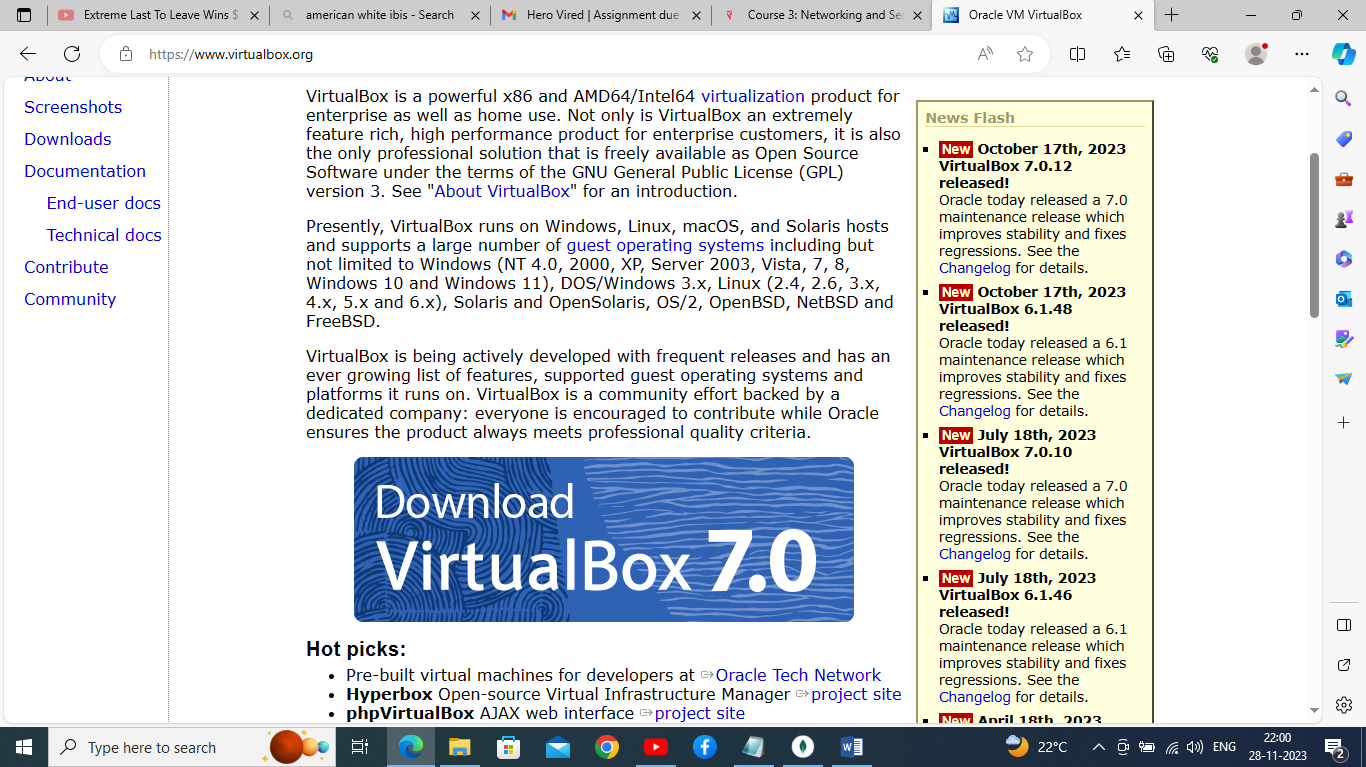
**Q3.Bonus Task:**

**Virtual Machine (VM) -**

Step 1: Download VirtualBox

1. Go to the official VirtualBox website: https://www.virtualbox.org/

2. Click on the "Downloads" link in the top navigation menu.



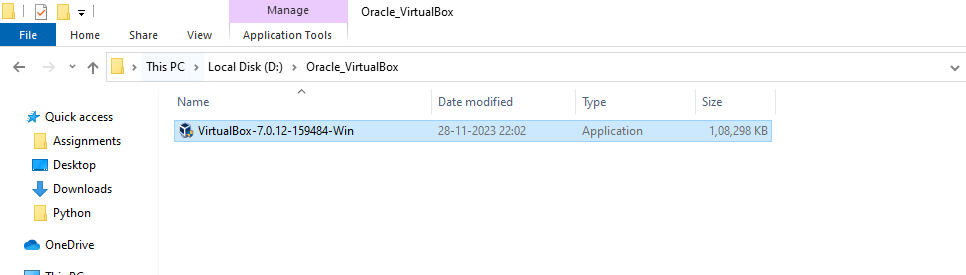


Step 3: Install VirtualBox

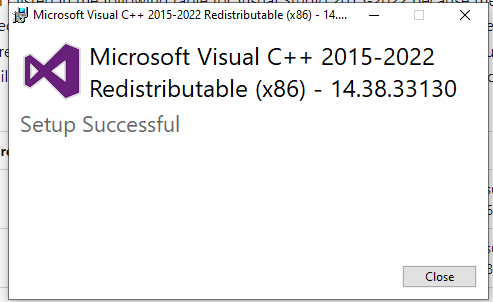
1. For Windows:

- Download the installer for Windows and double-click on the downloaded file to start the installation.

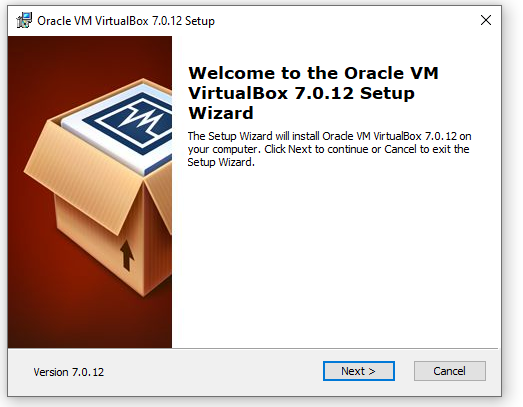
- Follow the on-screen instructions and accept the license agreement. - Choose the components you want to install and the installation path. - Complete the installation process.

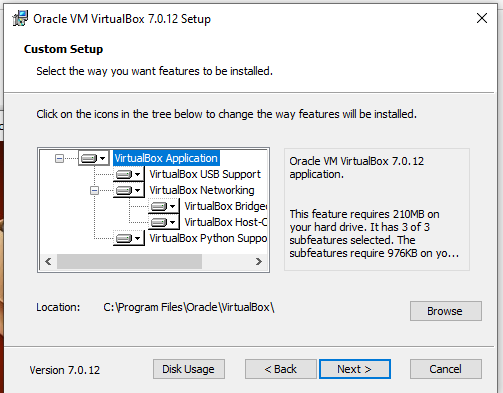


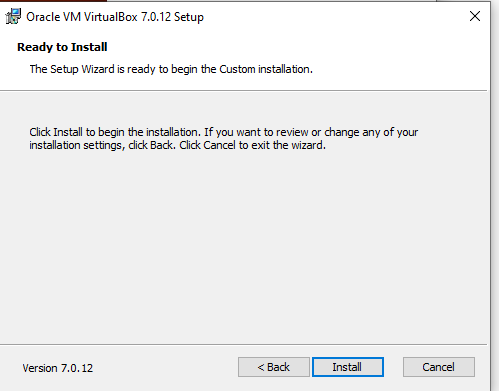
Dependency package installed.

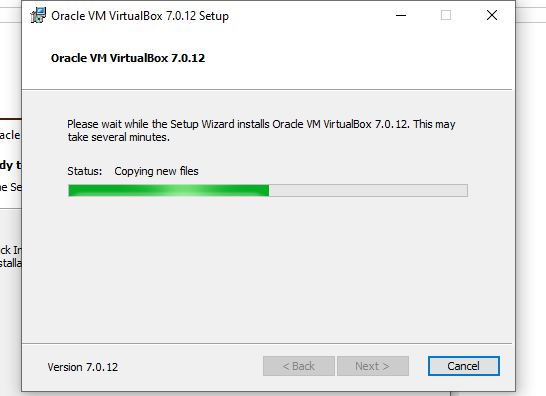


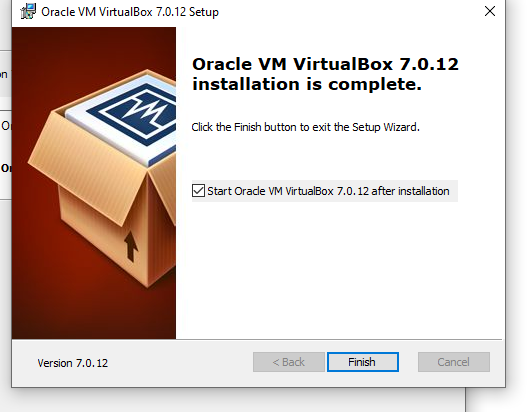
Now installing VM BOX

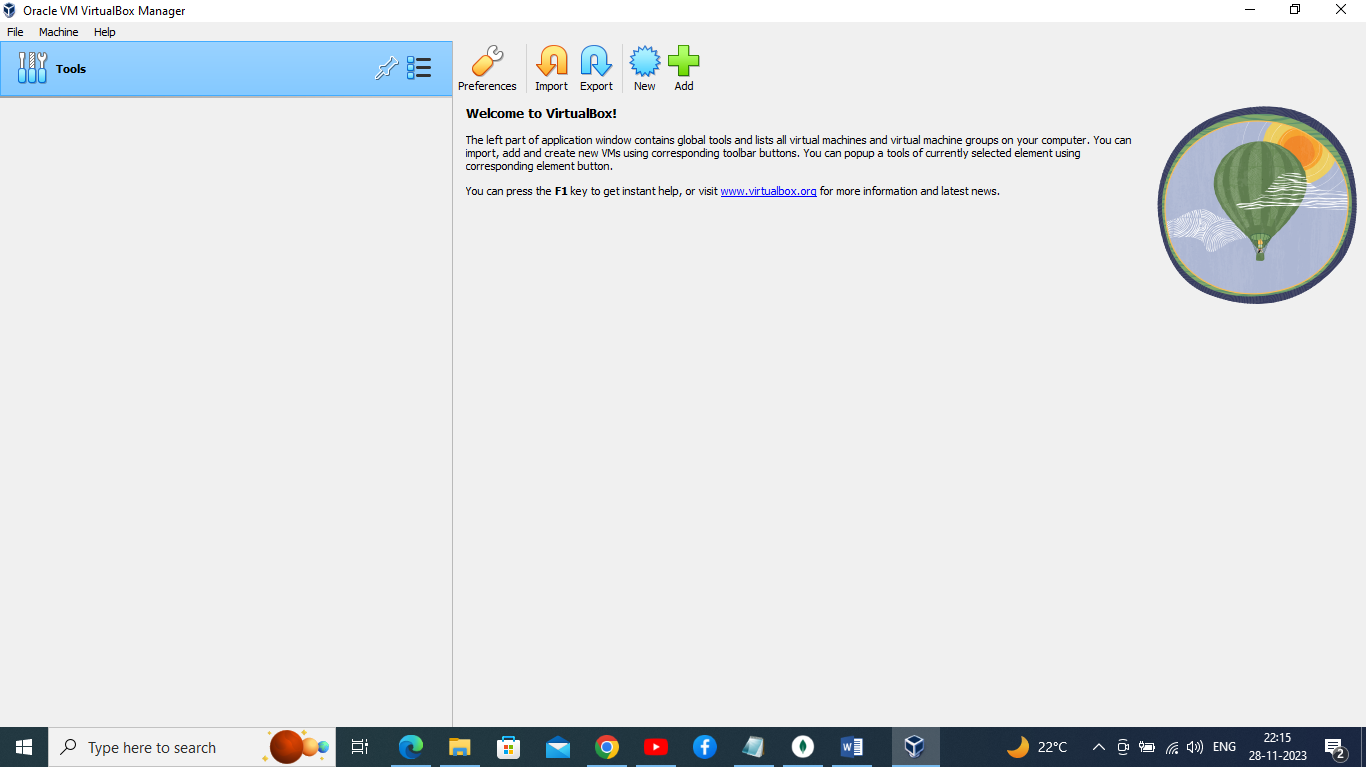


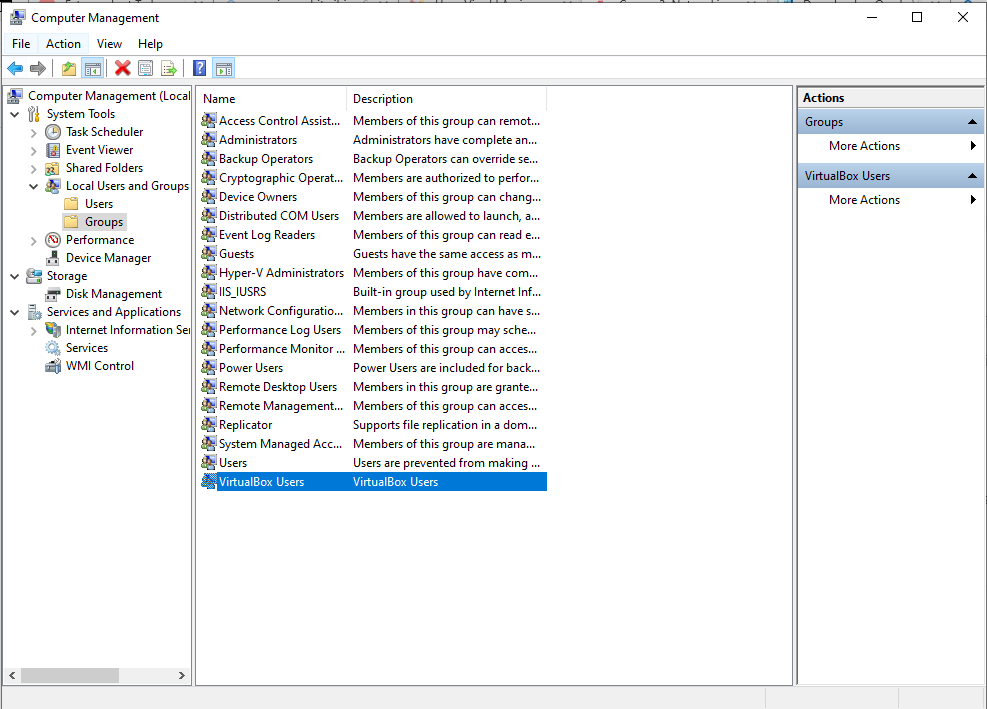




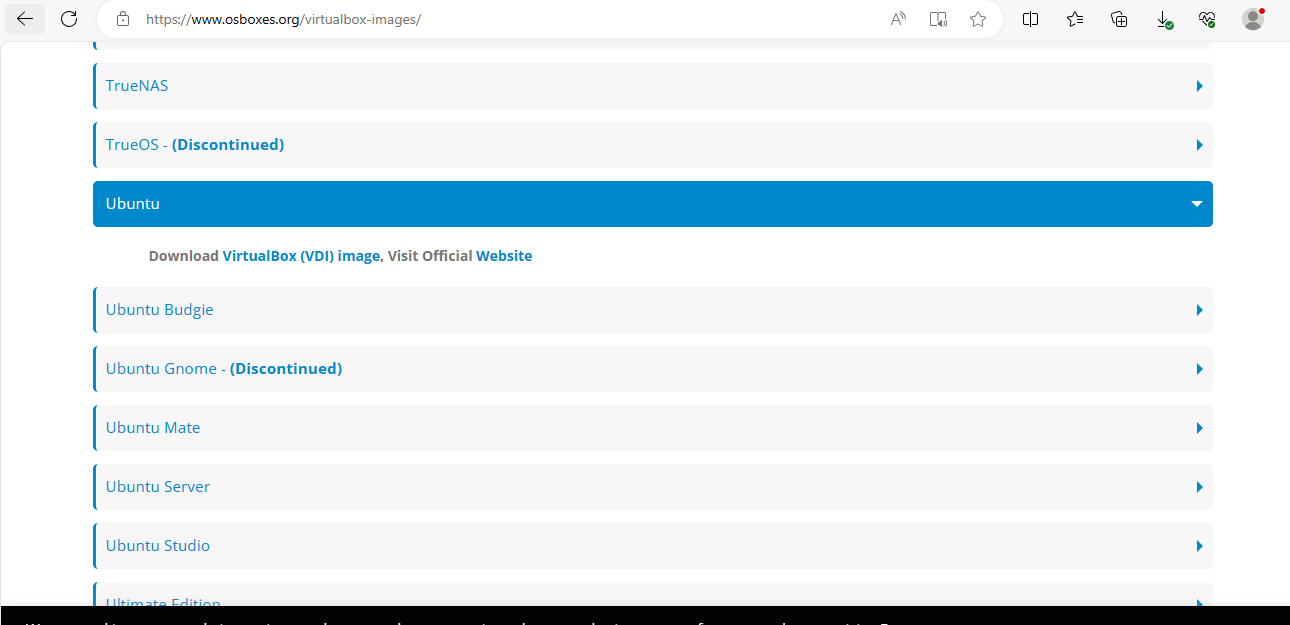




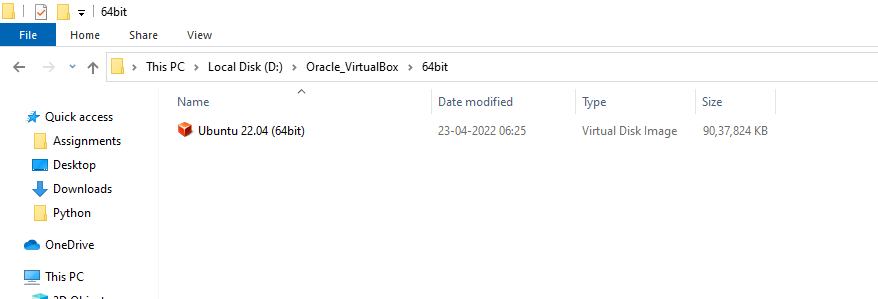


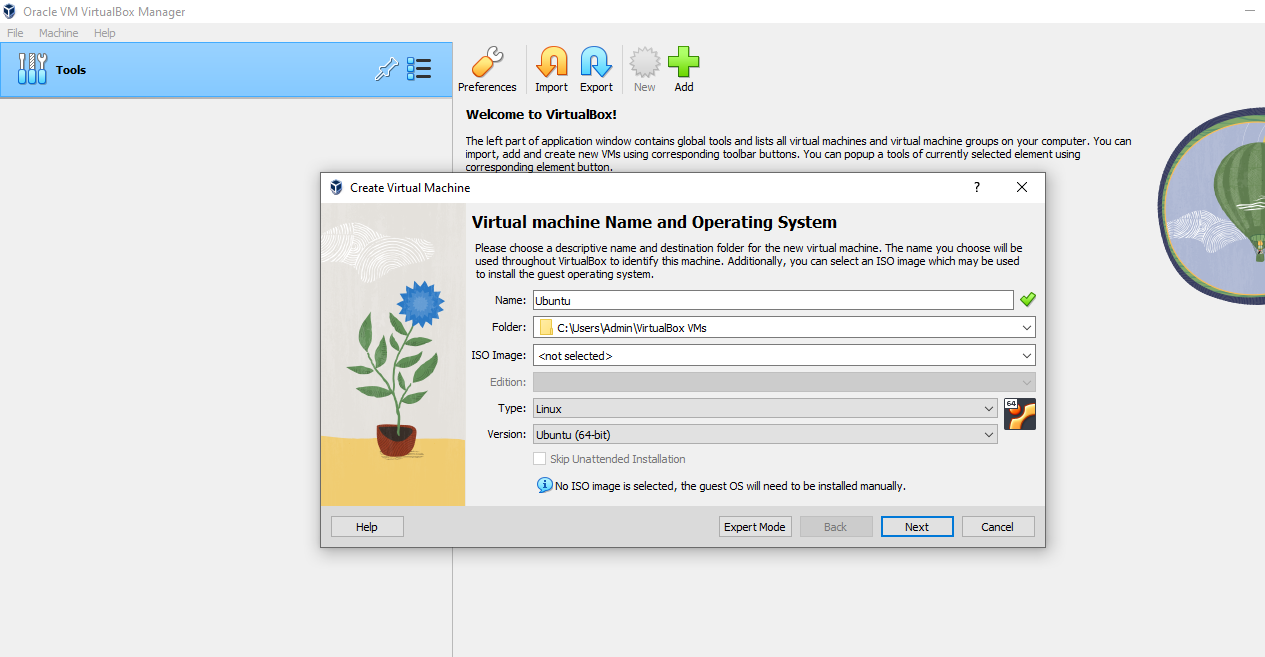


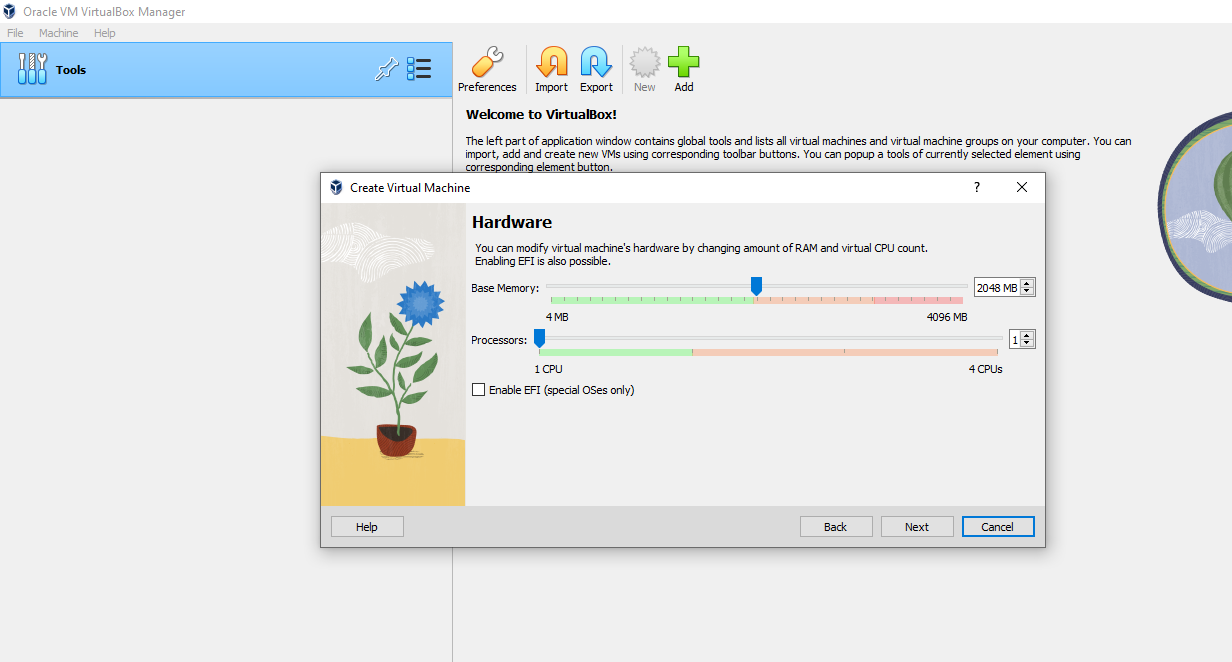
Once the VM has been installed, visit <https://www.osboxes.org/> download a Ubuntu 22.04 image and start it through your VirtualBox.

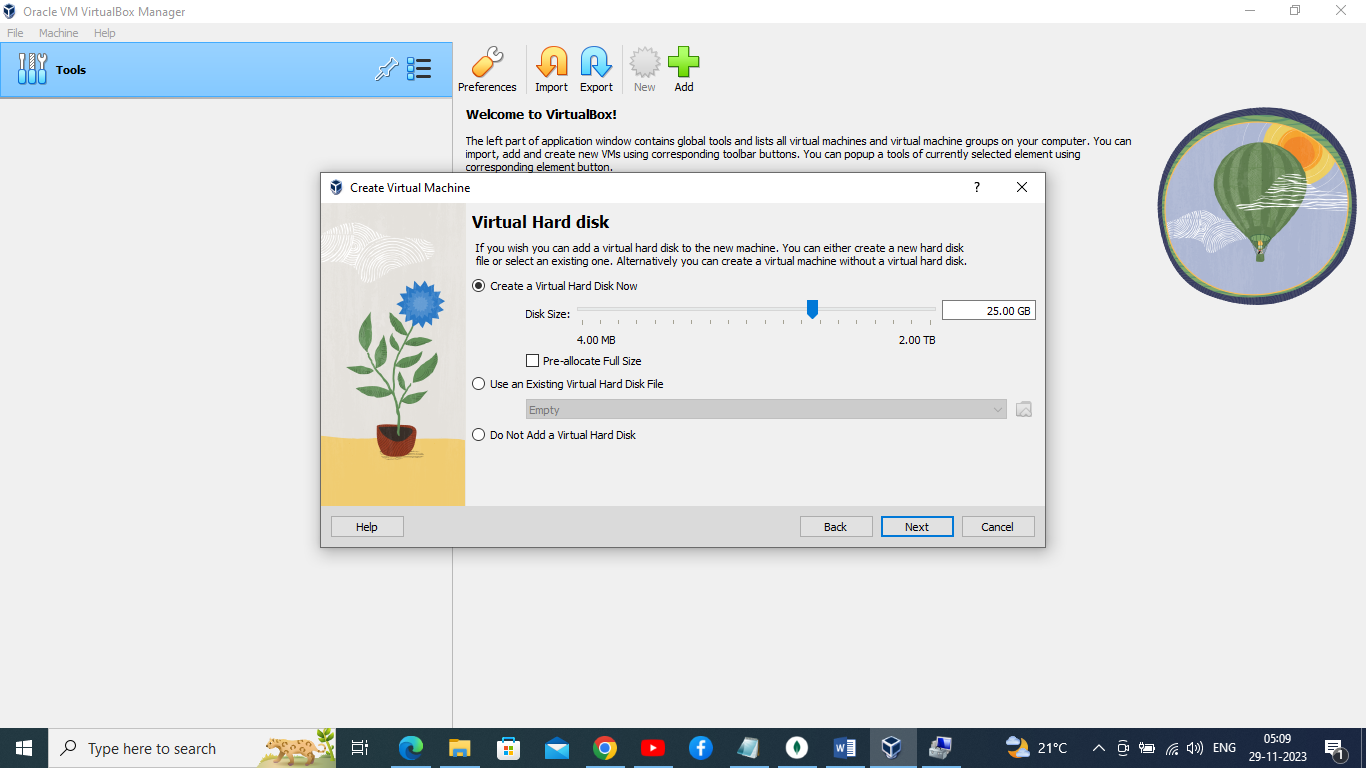


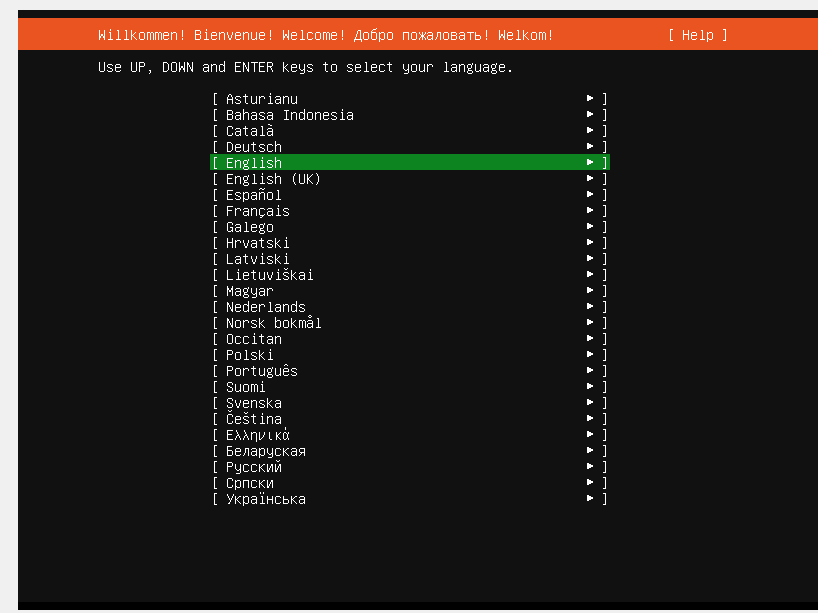
Ubuntu downloaded

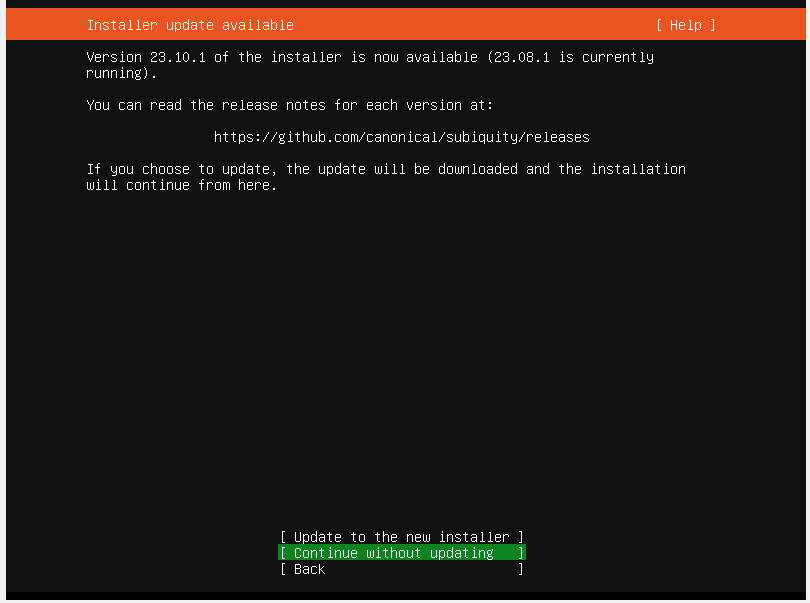


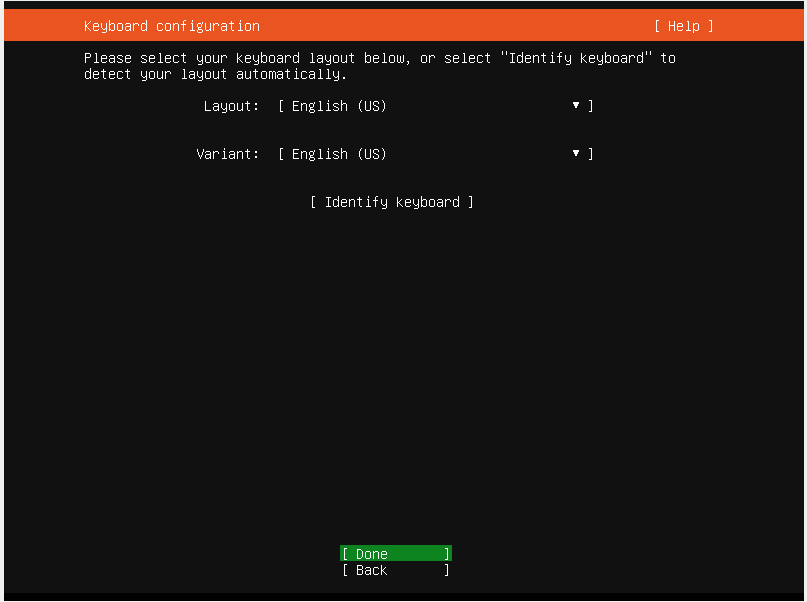


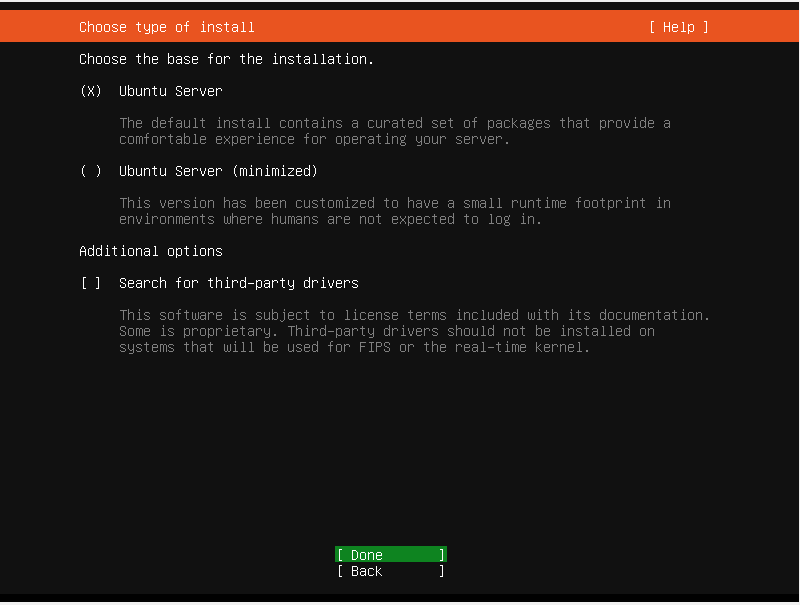


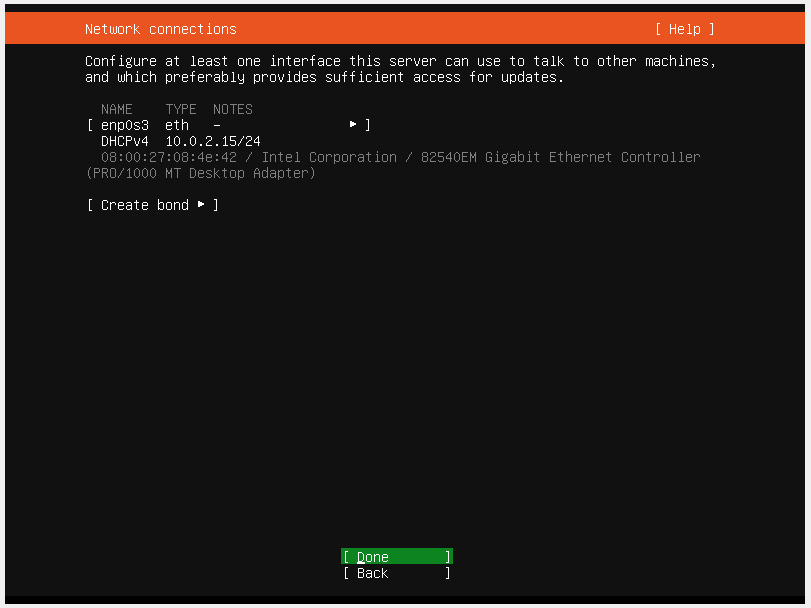


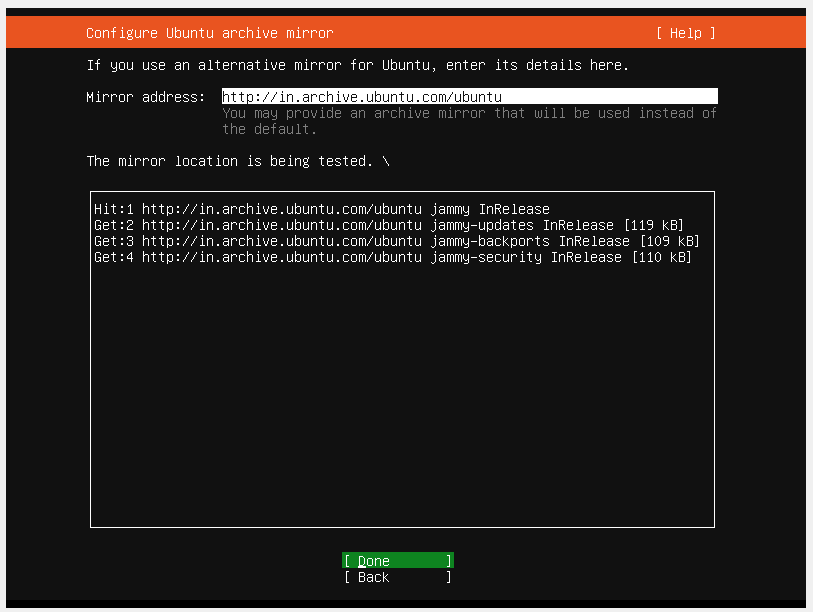


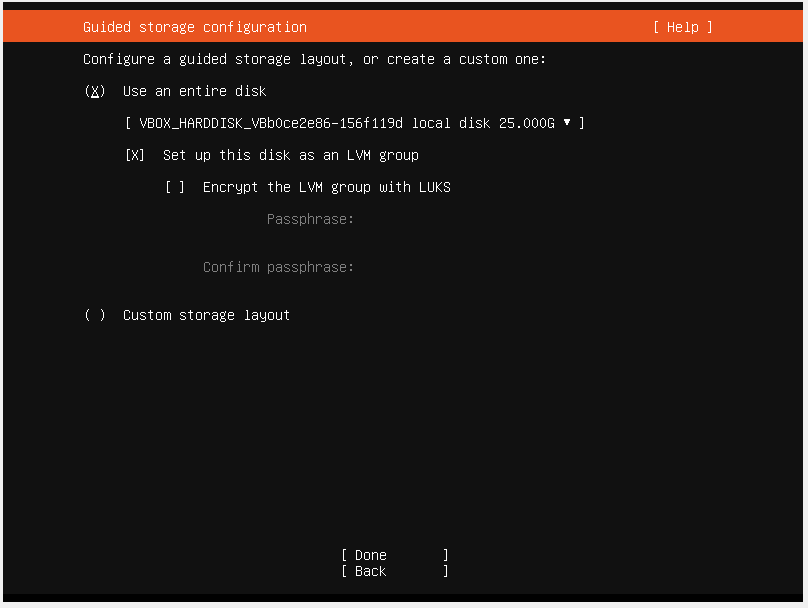


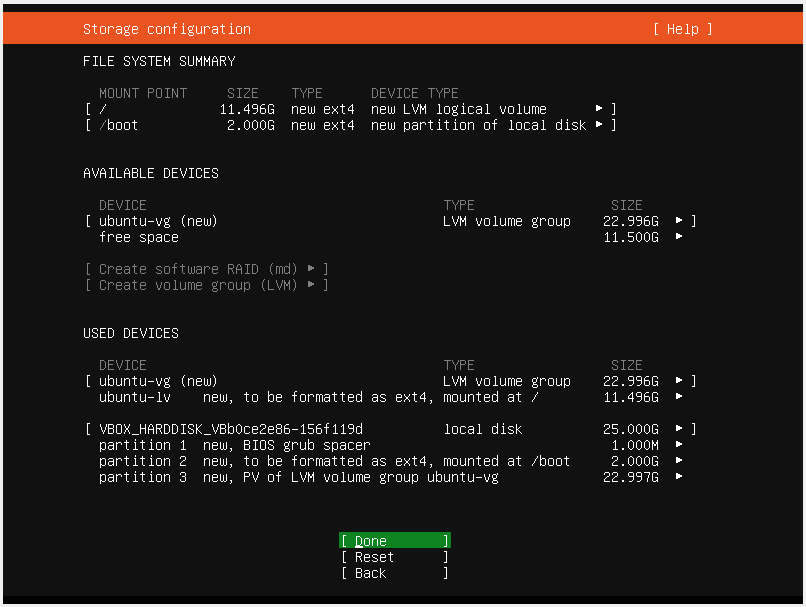


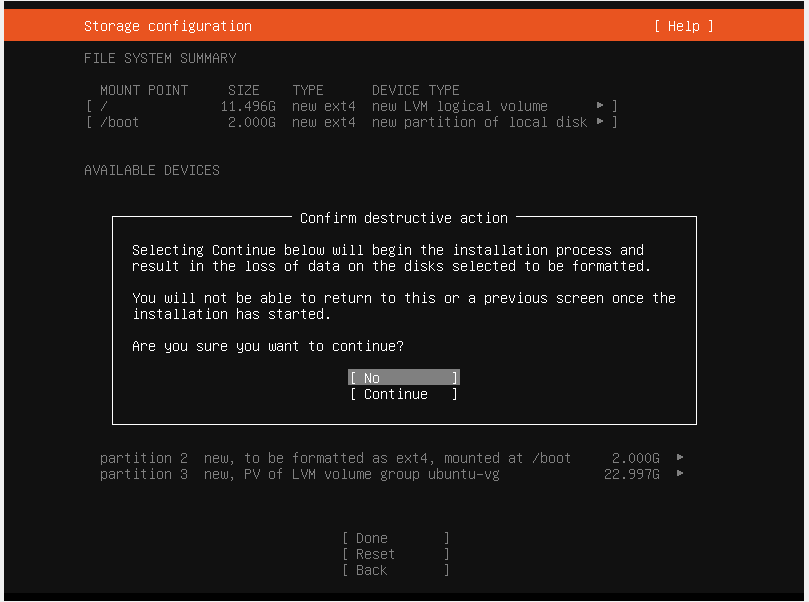


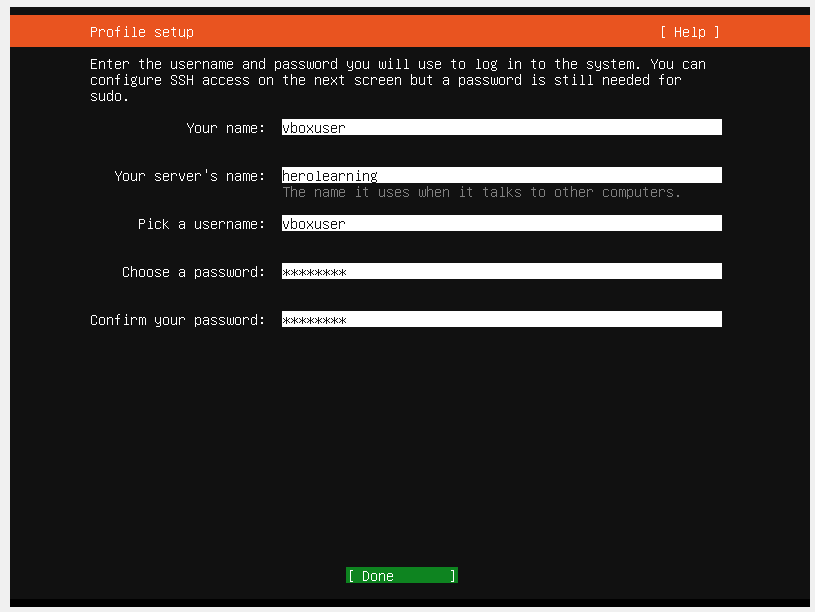


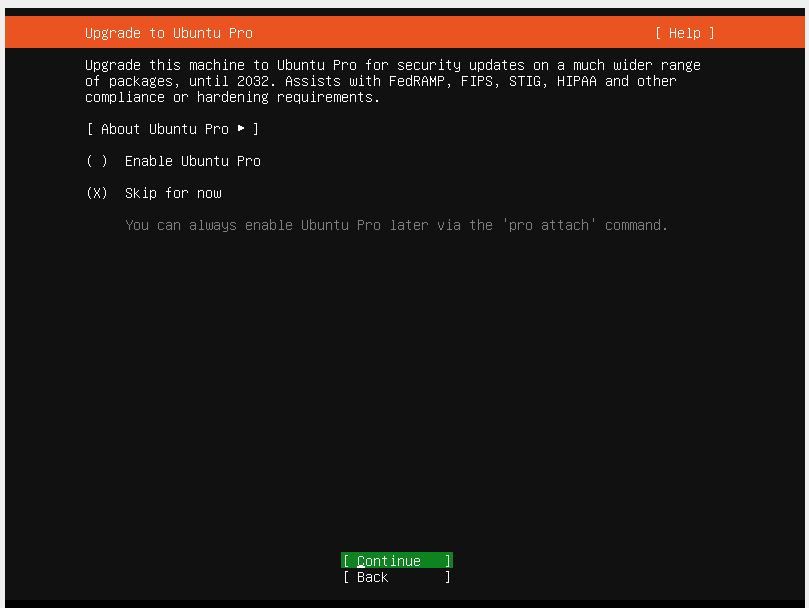


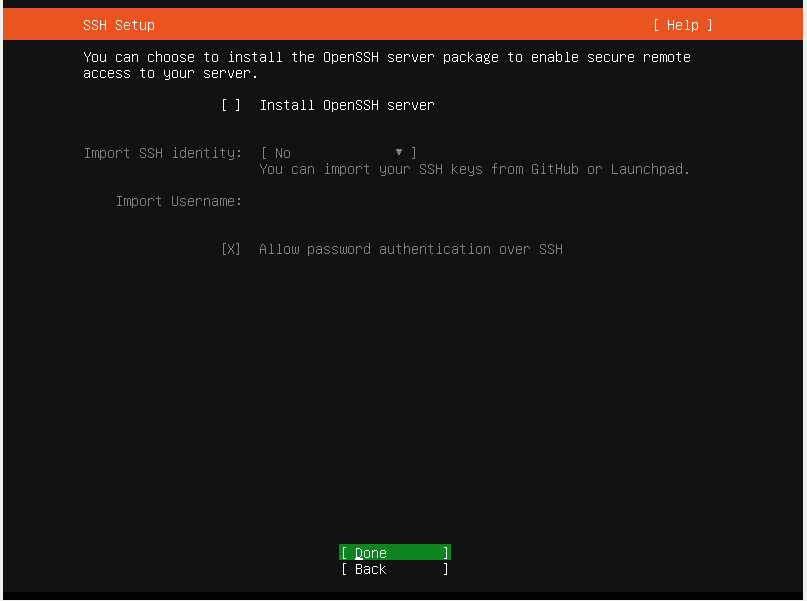


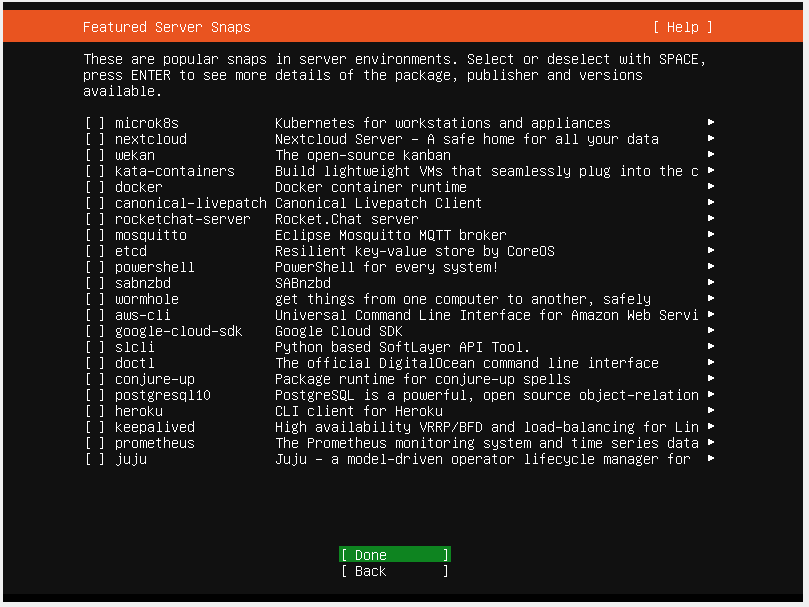


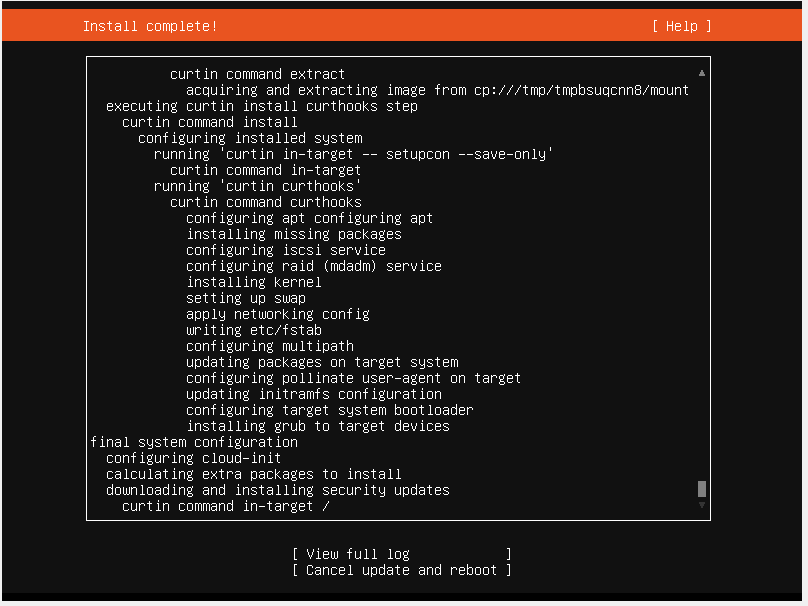


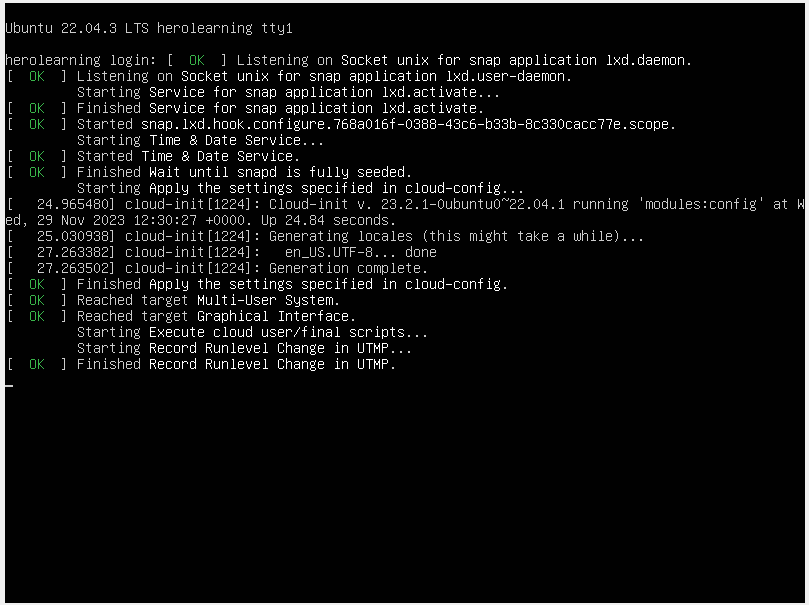


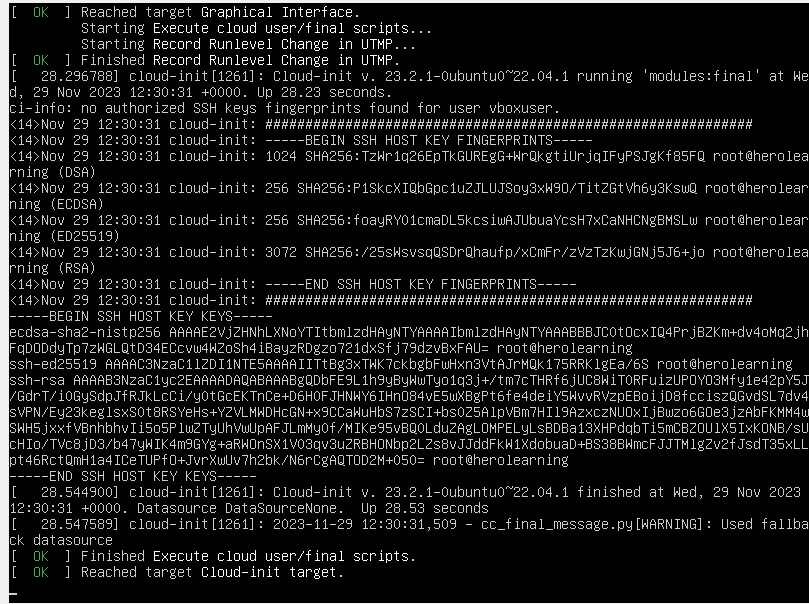


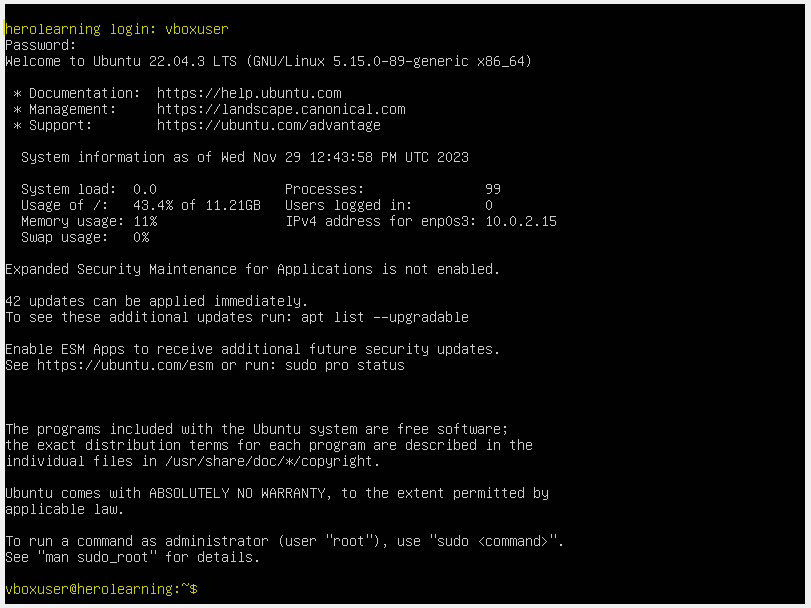






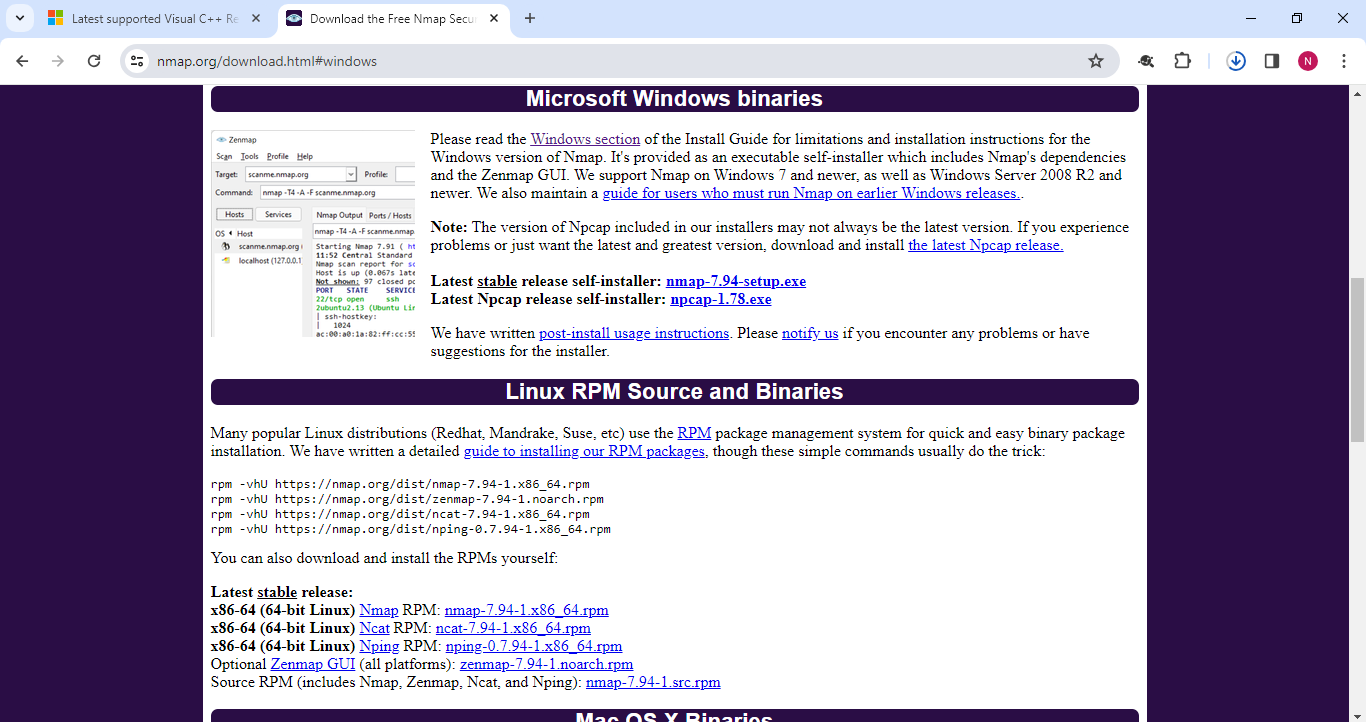


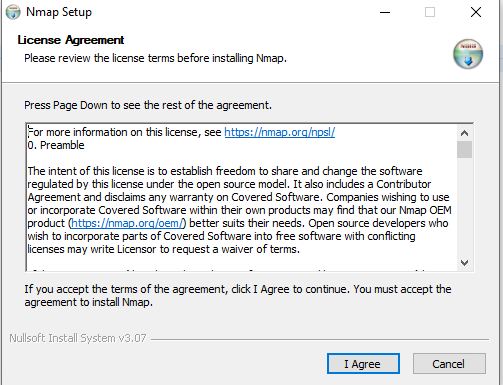


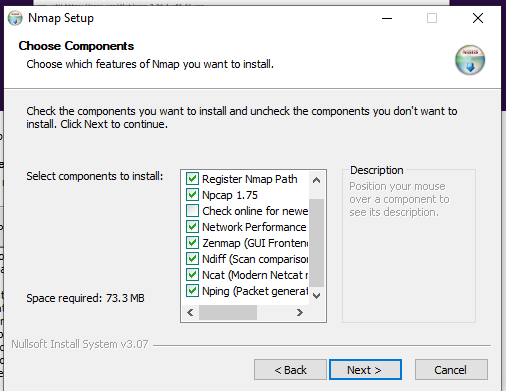


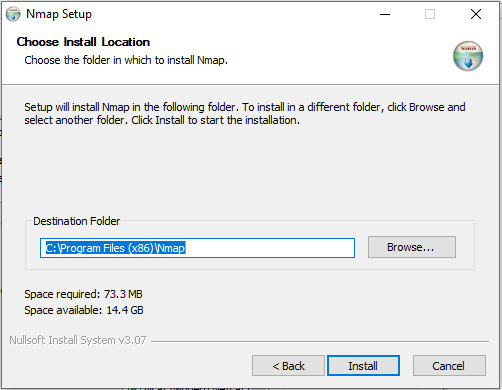
Installation completed and logged in with vboxuser id.

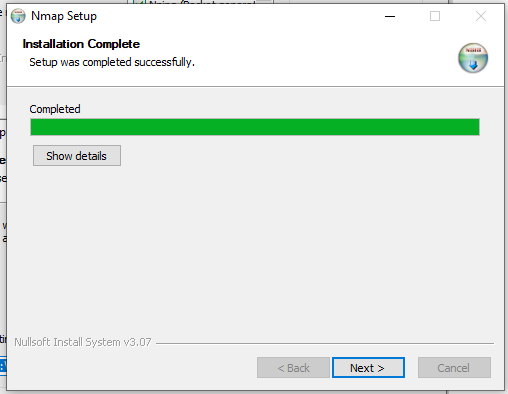
nmap tool downloading to install in local host

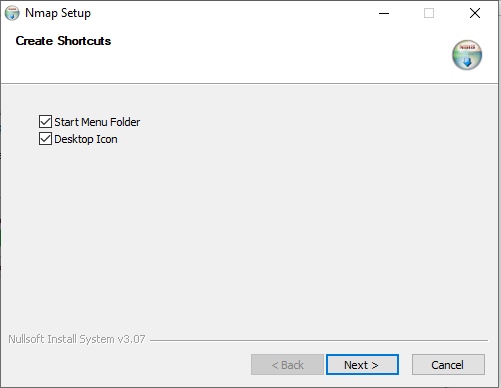


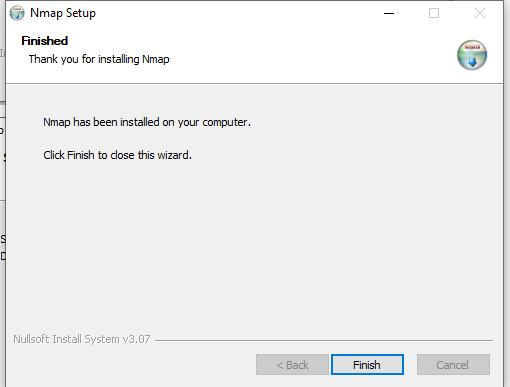




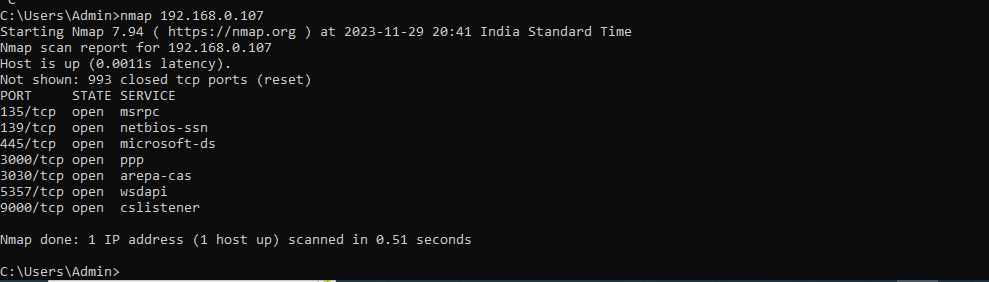








NMAP command out put:



Bonus VMware task completed.