

Download Ros Operating System

For this we need to install three programs in order which is:

- 1.Virtual box
2. Ubuntu 20.04.4 LTS (Focal Fossa)
- 3.Ros melodic

Steps:

- 1.install Virtual box from
<https://www.virtualbox.org/wiki/Downloads>

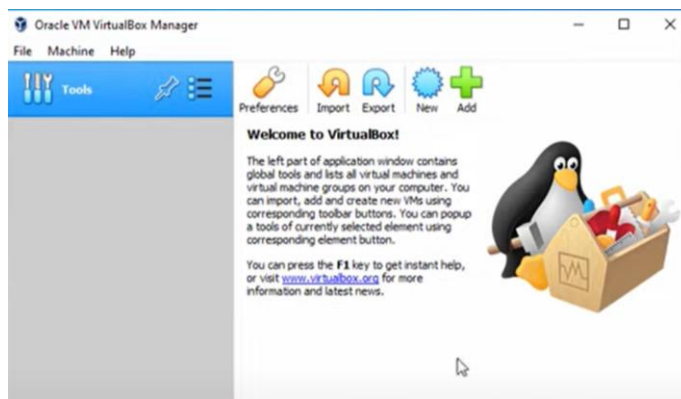
Then select your os to start download

VirtualBox 6.1.36 platform packages

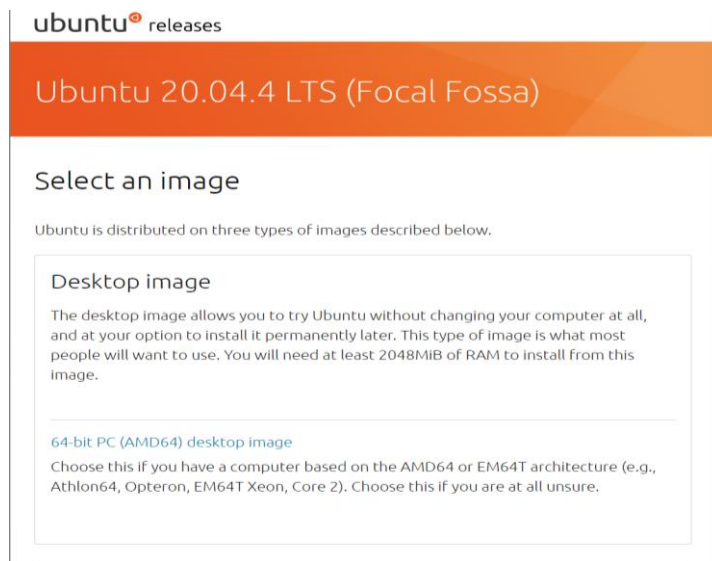
- [Windows hosts](#)
- [OS X hosts](#)
- [Linux distributions](#)
- [Solaris hosts](#)
- [Solaris 11 IPS hosts](#)

Then after finish download start install on your computer press next until start install.

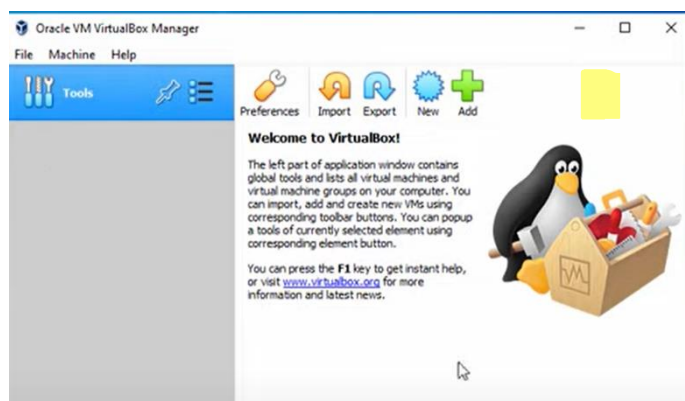
The home page will be like this



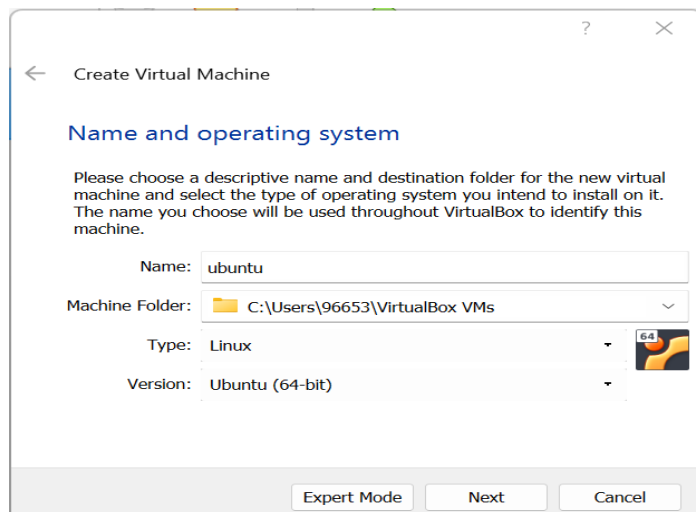
2.download and install Ubuntu 20.04.4 LTS (Focal Fossa) from <https://releases.ubuntu.com/20.04/>



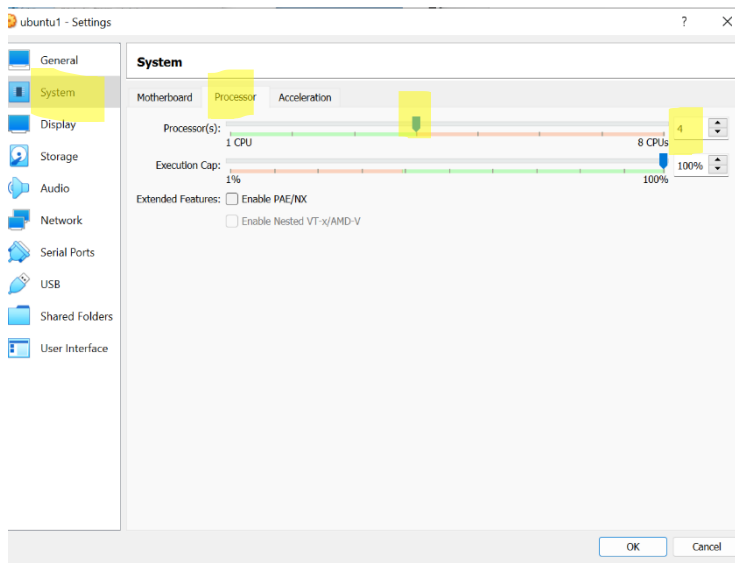
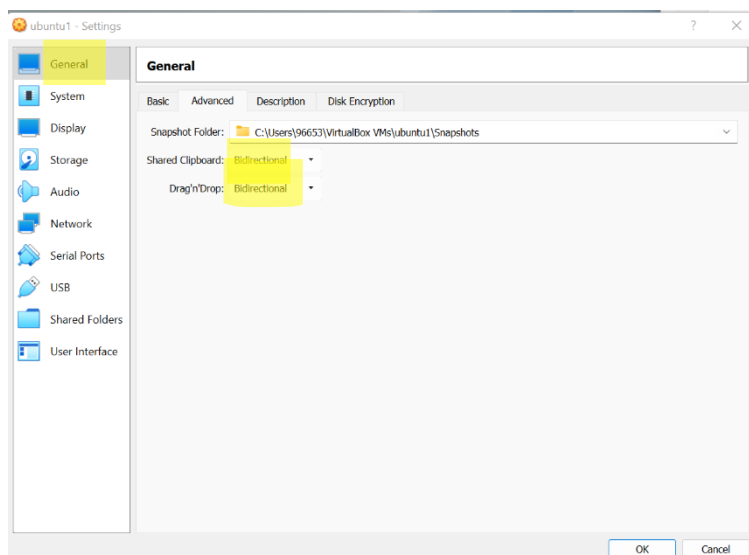
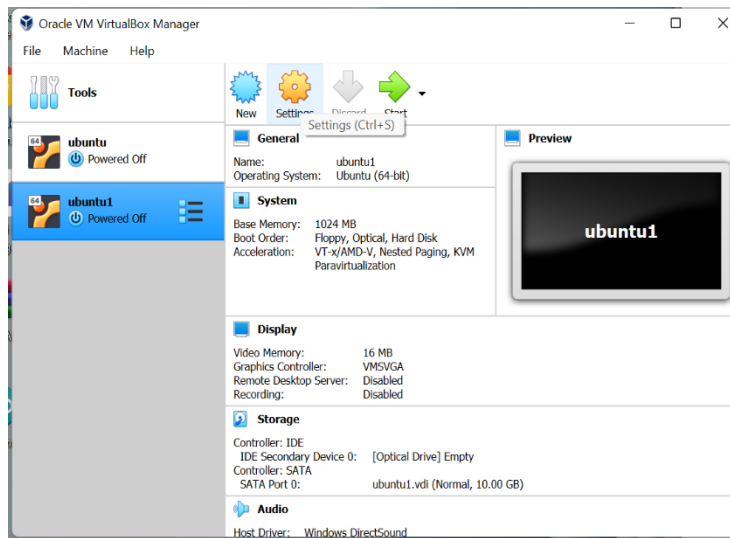
Click on the link to start download, then go to virtual box click new

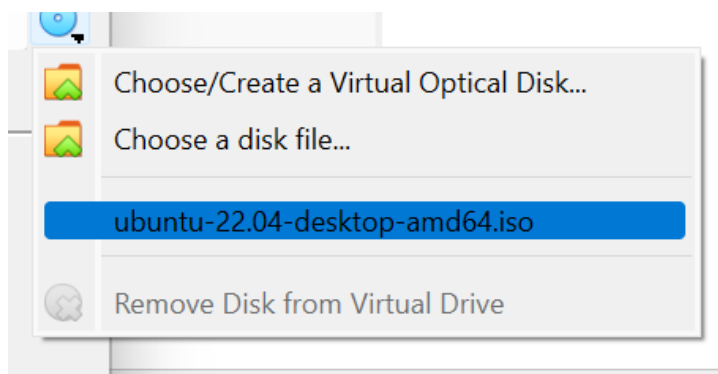
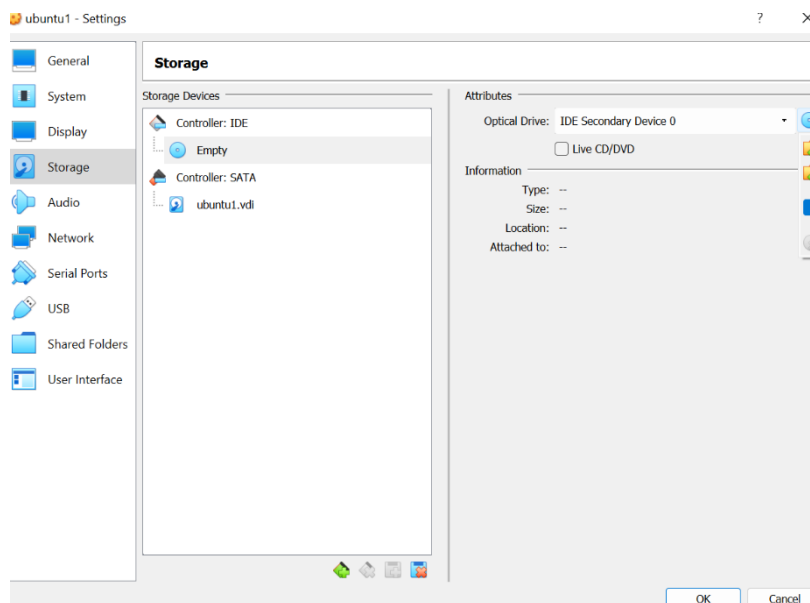
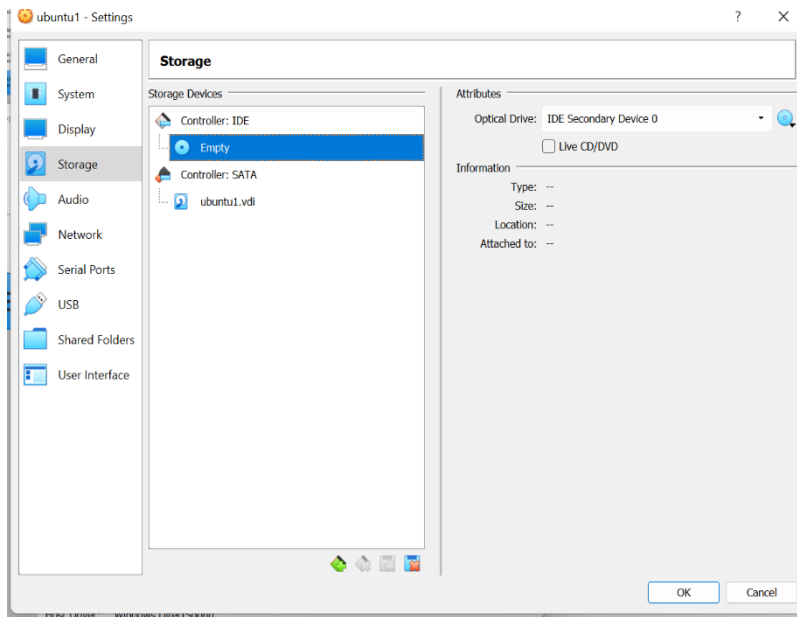


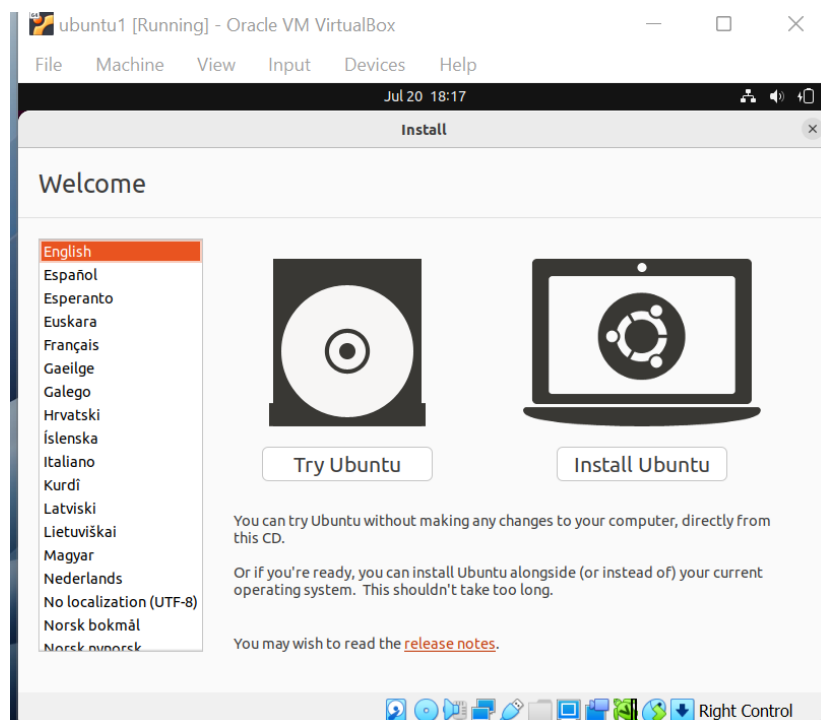
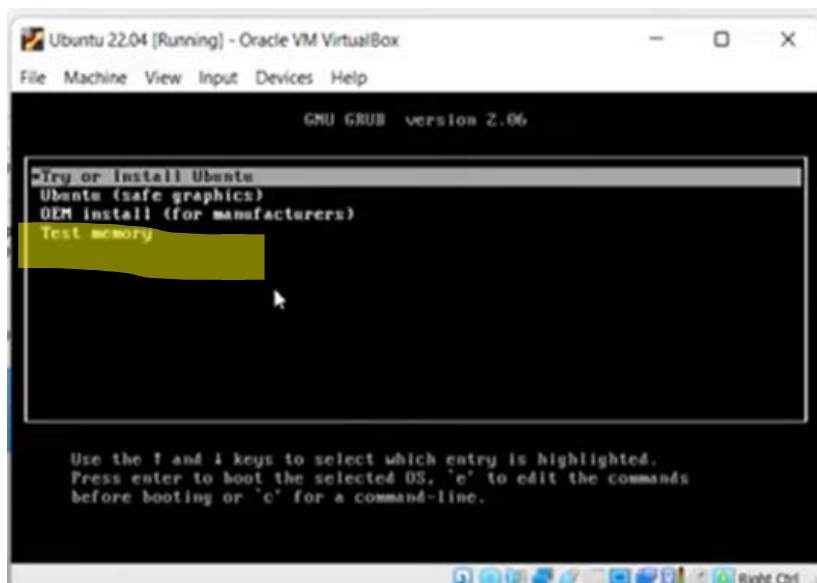
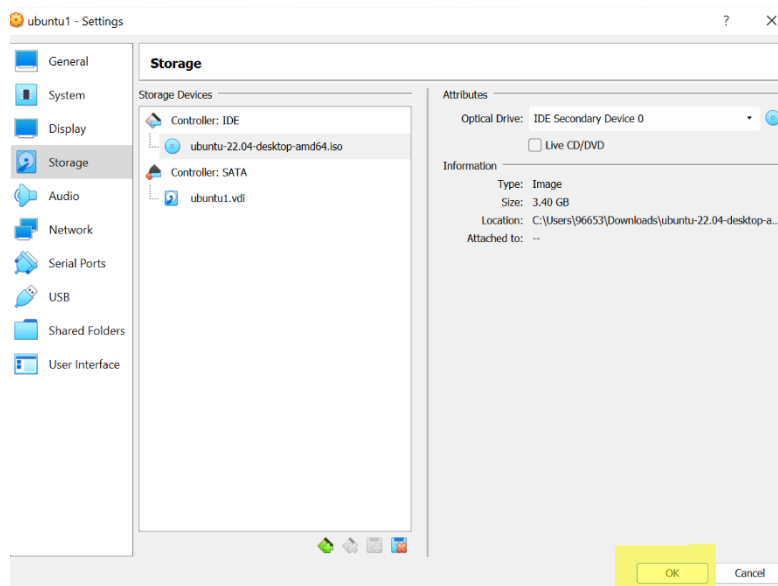
Then name it and click next if you want you can edit the setting like size of memory but it's not necessary

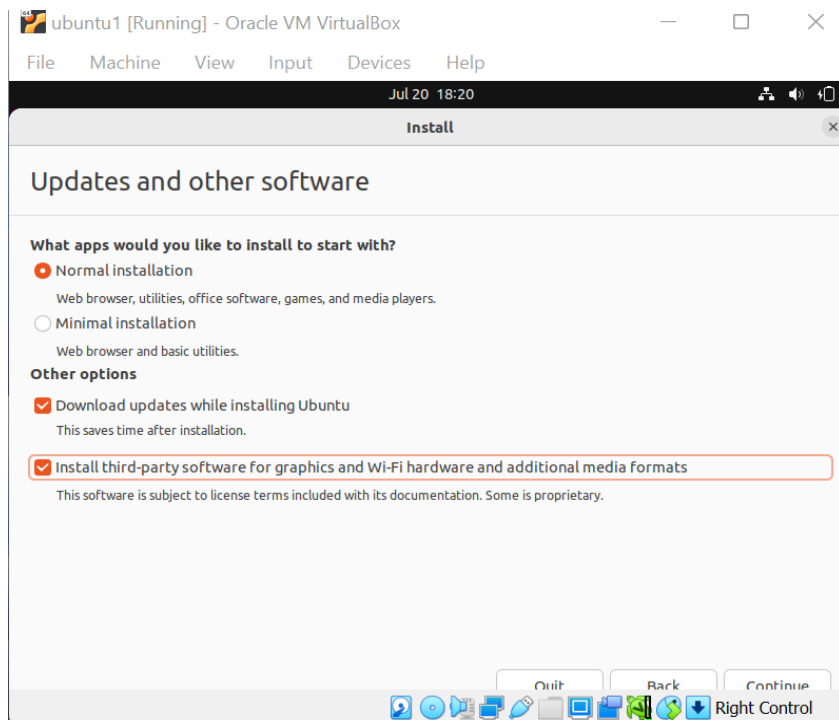
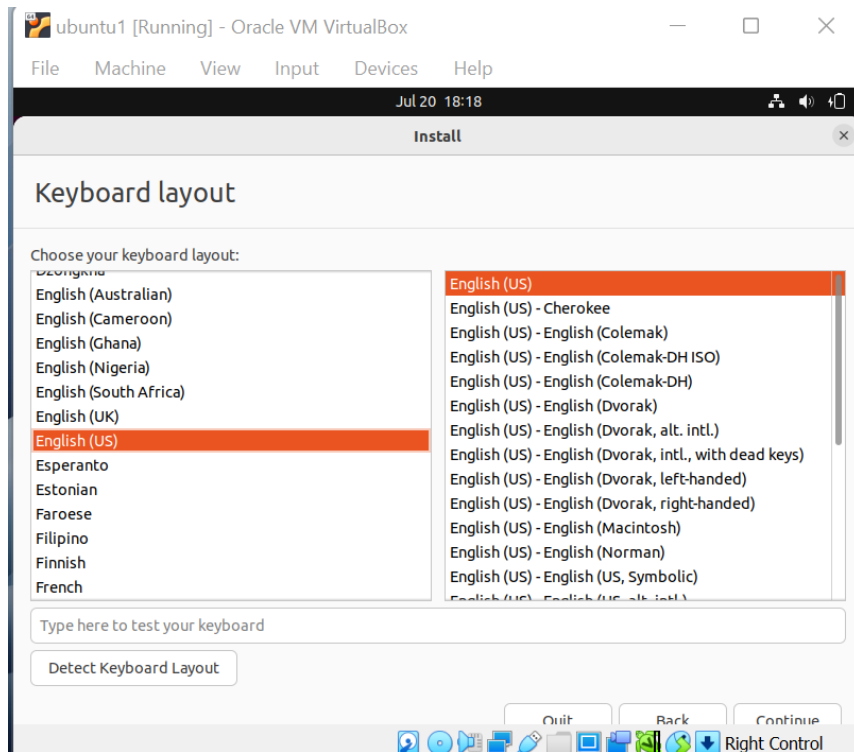


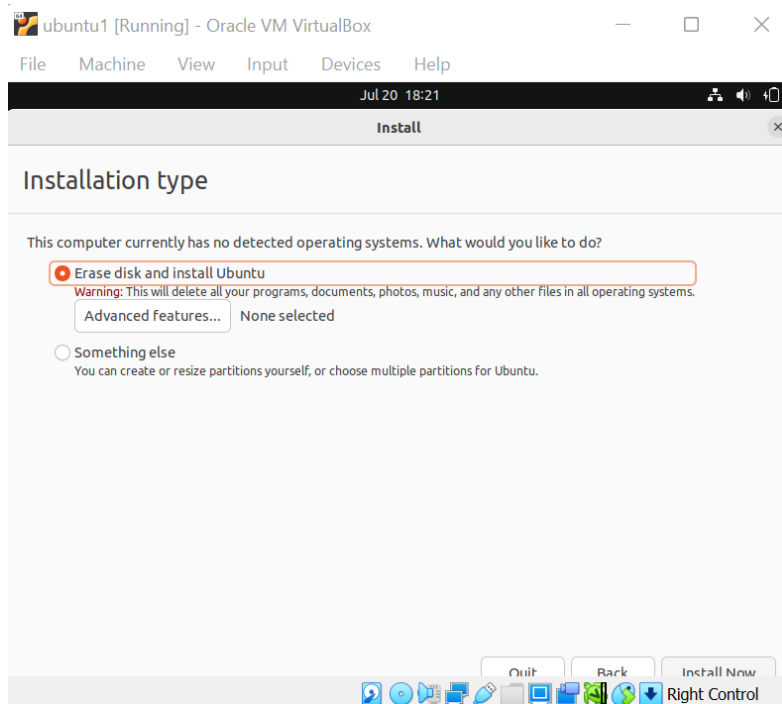
click on setting to add ubuntu



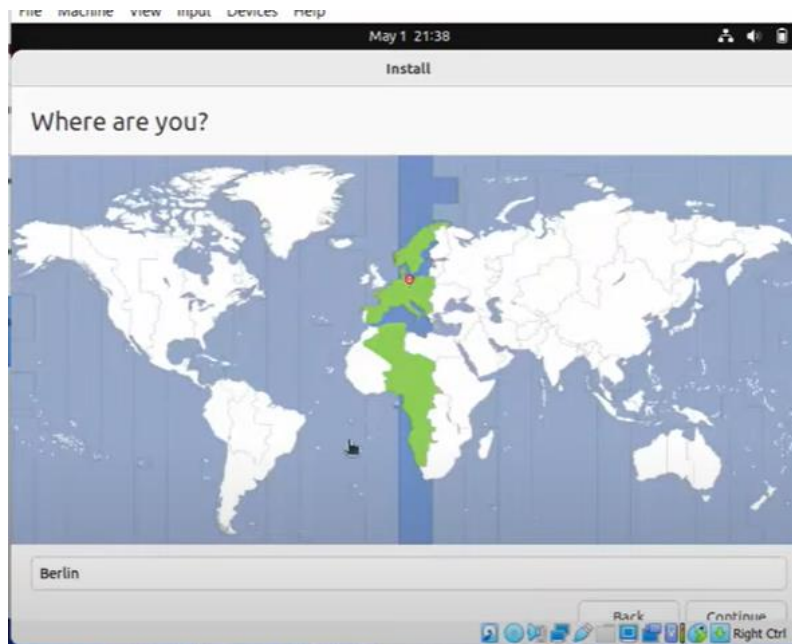




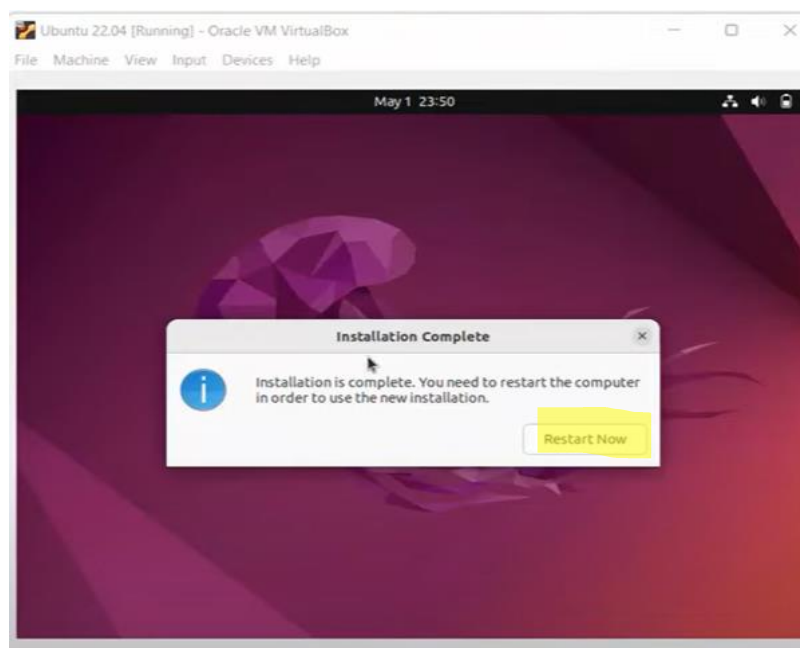
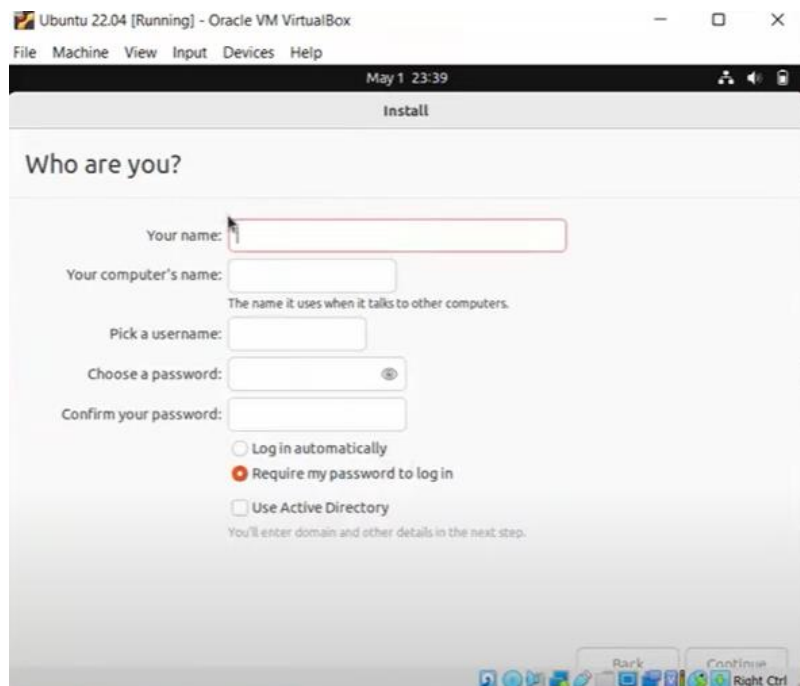




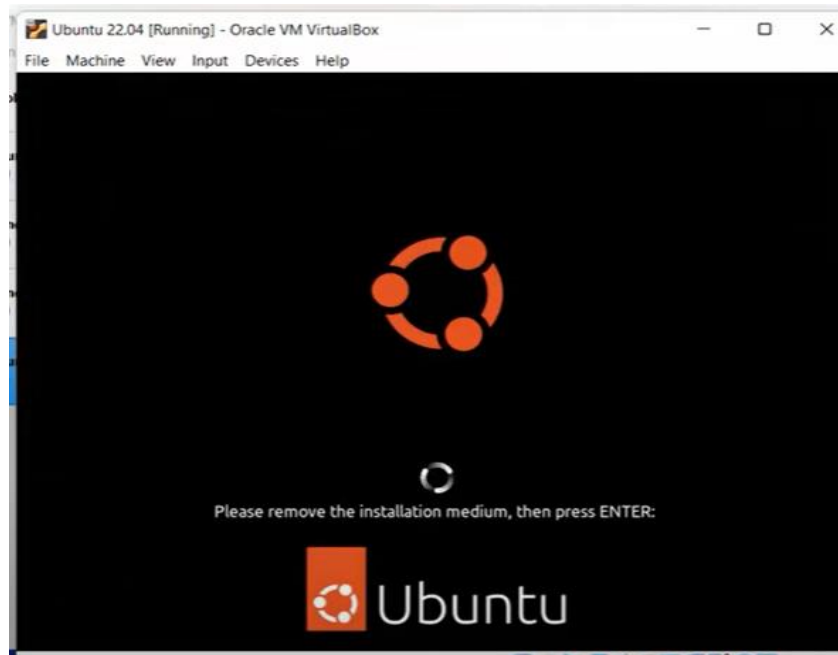
Enter your region



Enter your info



Press enter



It will be started and allow you to use it.

3. install Ros on ubuntu, First visit the site from the firefox in ubuntu as you see the site has a set of commands with explain

<https://docs.ros.org/en/foxy/Installation/Ubuntu-Install-Debian.html>

- Jenkins Instance
- Repositories

Set locale

Make sure you have a locale which supports `UTF-8`. If you are in a minimal environment (such as a docker container), the locale may be something minimal like `POSIX`. We test with the following settings. However, it should be fine if you're using a different UTF-8 supported locale.

```
locale # check for UTF-8

sudo apt update && sudo apt install locales
sudo locale-gen en_US en_US.UTF-8
sudo update-locale LC_ALL=en_US.UTF-8 LANG=en_US.UTF-8
export LANG=en_US.UTF-8

locale # verify settings
```

Setup Sources

You will need to add the ROS 2 apt repositories to your system. First, make sure that the Ubuntu Universe repository is enabled by checking the output of this command.

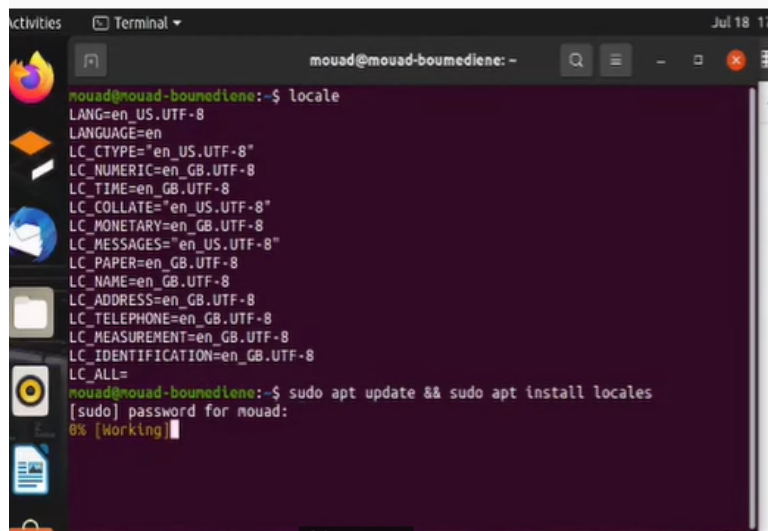
```
apt-cache policy | grep universe
```

This should output a line like the one below:

```
500 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 Packages
release v=20.04,o=Ubuntu,a=focal,n=focal,l=Ubuntu,c=universe,b=amd64
```

If you don't see an output line like the one above, then enable the Universe repository with these instructions.

Then open terminal and write the first command make

A terminal window titled 'Terminal' with the user 'mouad@nouad-boumediene'. The prompt is 'nouad@nouad-boumediene:~\$'. The user has entered the command 'locale', which outputs the following settings: LANG=en_US.UTF-8, LANGUAGE=en, LC_CTYPE=en_US.UTF-8, LC_NUMERIC=en_GB.UTF-8, LC_TIME=en_GB.UTF-8, LC_COLLATE=en_US.UTF-8, LC_MONETARY=en_GB.UTF-8, LC_MESSAGES=en_US.UTF-8, LC_PAPER=en_GB.UTF-8, LC_NAME=en_GB.UTF-8, LC_ADDRESS=en_GB.UTF-8, LC_TELEPHONE=en_GB.UTF-8, LC_MEASUREMENT=en_GB.UTF-8, LC_IDENTIFICATION=en_GB.UTF-8, and LC_ALL=. The user then enters 'sudo apt update && sudo apt install locales'. The terminal shows '[sudo] password for mouad:' followed by '0% [Working]'.

sure it is given you the same result, and then continue one by one in the same way like in the link <https://youtu.be/POhKxEhT3LA>.

Download Ros Operating System on jetson nano

Installation

Open a new terminal by pressing Ctrl + Alt + t or executing the “Terminal” application using the Ubuntu 18 launch system.

Set up the Jetson Nano to accept software from *packages.ros.org*:

```
$ sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'
```

Add a new apt key:

```
$ sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --recv-key C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
```

[Note: the ROS GPG key has changed due to a [security issue on the ROS build farm server](#). If you configured your Jetson Nano for ROS following this guide before the 24th June 2019, please follow [this guide](#) to replace the old key in the correct way]

Update the Debian packages index:

```
$ sudo apt update
```

Install the ROS Desktop package, including support for `rqt`, `rviz` and other useful robotics packages:

```
$ sudo apt install ros-melodic-desktop
```

Note: “ROS Desktop Full” is a more complete package, however, it is not recommended for an embedded platform; 2D/3D simulators will be installed with it and they take too much space on ROM, and are too computationally hungry to be used on the Nano.

It is recommended to load the ROS environment variables automatically when you execute a new shell session. Update your `.bashrc` script:

```
$ echo "source /opt/ros/melodic/setup.bash" >> ~/.bashrc
$ source ~/.bashrc
```

Install and initialize `rosdep`. `rosdep` enables you to easily install system dependencies for source code you want to compile and is required to run some core components in ROS:

```
$ sudo apt install python-rosdep python-rosinstall python-rosinstall-generator python-wstool build-essential

$ sudo rosdep init

$ rosdep update
```

Now the Jetson Nano is ready to execute ROS packages and become the brain of your autonomous robot.

Configure a catkin workspace

To start running your own ROS packages or install other packages from the source (such as the [ZED ROS wrapper](#) for example), you must create and configure a `catkin` workspace.

Install the following dependencies:

```
$ sudo apt-get install cmake python-catkin-pkg python-empy  
python-nose python-setuptools libgtest-dev python-  
rosinstall python-rosinstall-generator python-wstool  
build-essential git
```

Create the catkin root and source folders:

```
$ mkdir -p ~/catkin_ws/src  
$ cd ~/catkin_ws/
```

Configure the catkin workspace by issuing a first “empty” build command:

```
$ catkin_make
```

Finally, update your `.bashrc` script with the information about the new workspace:

```
$ echo "source ~/catkin_ws/devel/setup.bash" >> ~/.bashrc  
$ source ~/.bashrc
```

Your catkin workspace is now ready to compile your ROS packages from source directly onto the Jetson Nano.

Resources

1. <https://youtu.be/z3FFnIO1Q9M>
2. <https://youtu.be/v1JVqd8M3Yc>
3. <https://www.stereolabs.com/blog/ros-and-nvidia-jetson-nano/>