Lab 1 Installation of ROS, Ubuntu and VirtualBox on your computer

In this lab, you will learn how to install Ubuntu 18.04 and ROS melodic on VirtualBox 6.1.14.

1.1 Preparation

Download Ubuntu 18.04.5 Desktop Image from here:

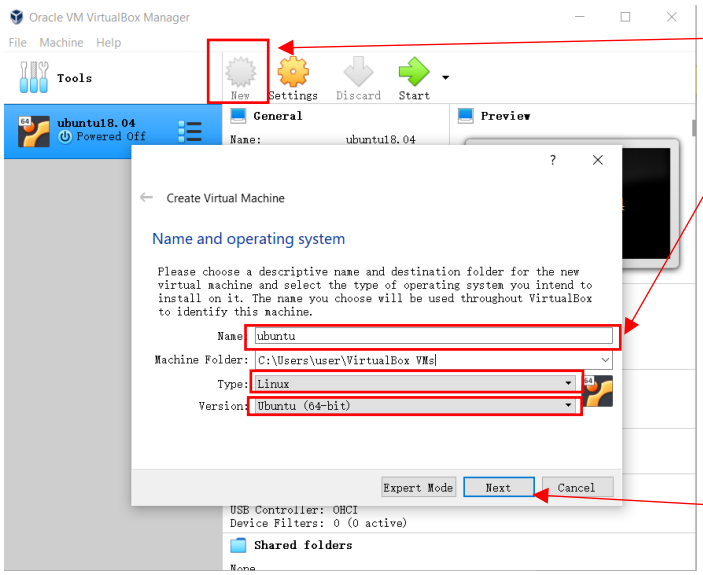
https://releases.ubuntu.com/18.04/ubuntu-18.04.5-desktop-amd64.iso

Download VirtualBox and install, with default settings:

https://www.virtualbox.org/wiki/Downloads

1.2 **Create the Virtual Machine (VM)**

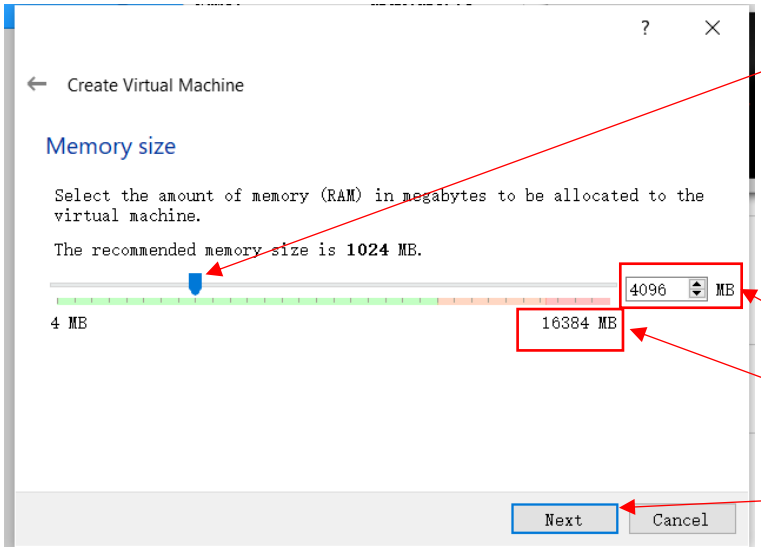
The first step is to create a virtual machine with the resources required.



Click new.

Type in a name for the VM. If Ubuntu18.04 is typed, the Type and Version fields will change automatically to reflect the OS. If a name is typed not reflecting the OS, say RobotDev, then ensure that the Type and Version fields are set correctly, Linux and Ubuntu (64-bit) respectively

Then click Next

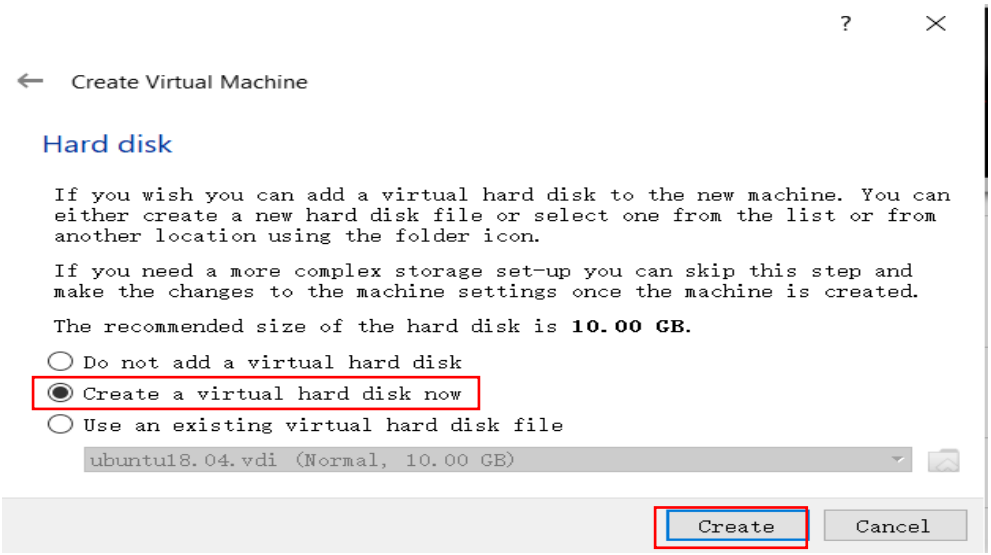


Use the slider to allocate a suitable amount of system RAM to the VM, as a rule of thumb, make it 25% of the system memory, with a minimum of 1GB. This setting can easily be changed once the VM has been created.

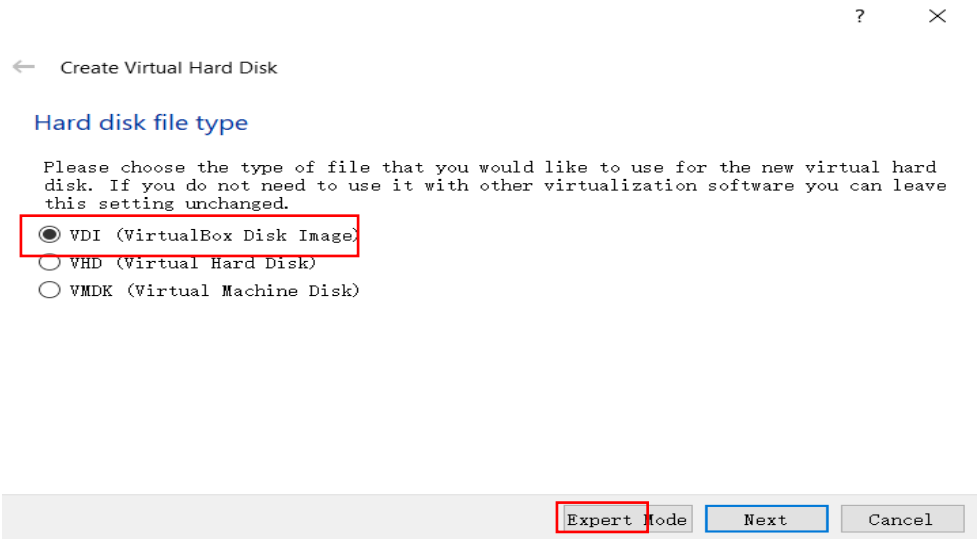
Here 4GB of RAM has been allocated from a system of total 16GB. A maximum 4GB would be fine.

Then click Next.

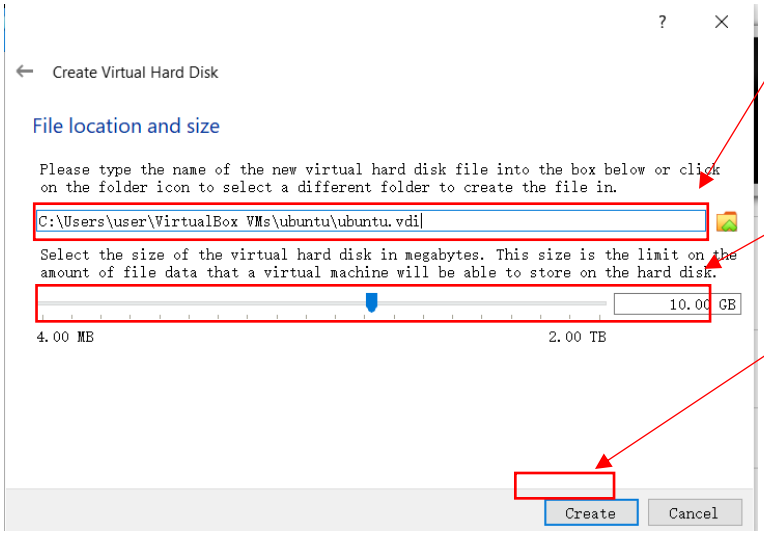
Ensure Create a virtual hard disk now is checked then click Create.



Ensure VDI (VirtualBox Disk Image) is checked and click Next.



On the next panel**,** Confirm the default **Dynamically allocated** option is checked, click **Next**.



Now ensure the default folder location is fine for your system, (you may wish to move it to another partition or drive).

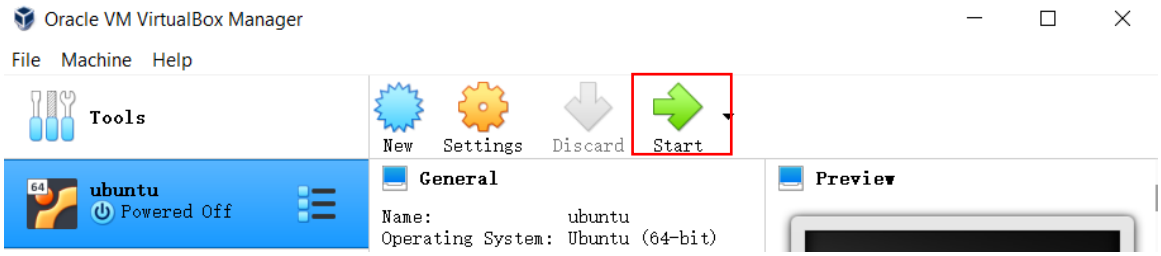
Using the slider or typing the value into the box, set the size of the virtual disk to a minimum 12GB, or as your needs require.

Click Create.

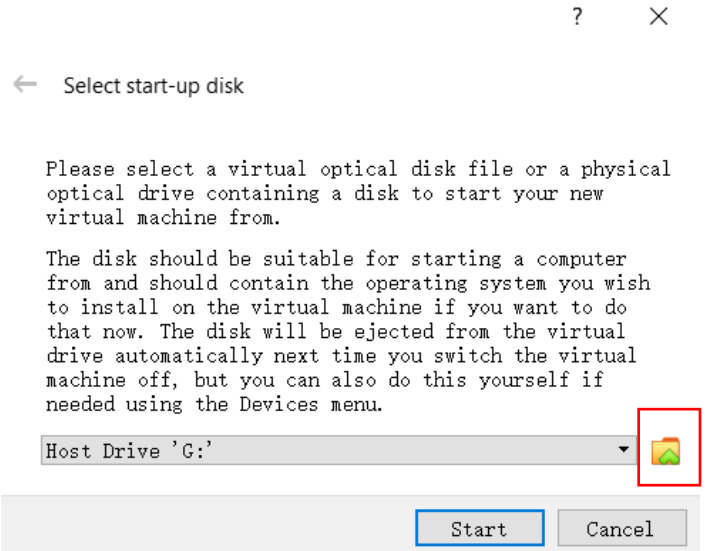
The environment for the Virtual Machine has now been created and the Ubuntu 18.04 image needs to be installed.

1.3 Install Ubuntu 18.04 Desktop

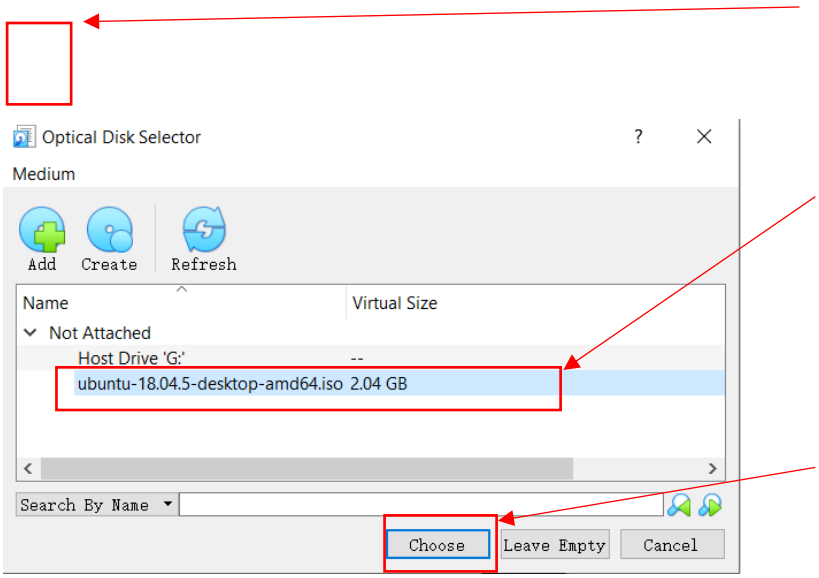
In the VirtualBox Manager click Start.



Now click on folder icon



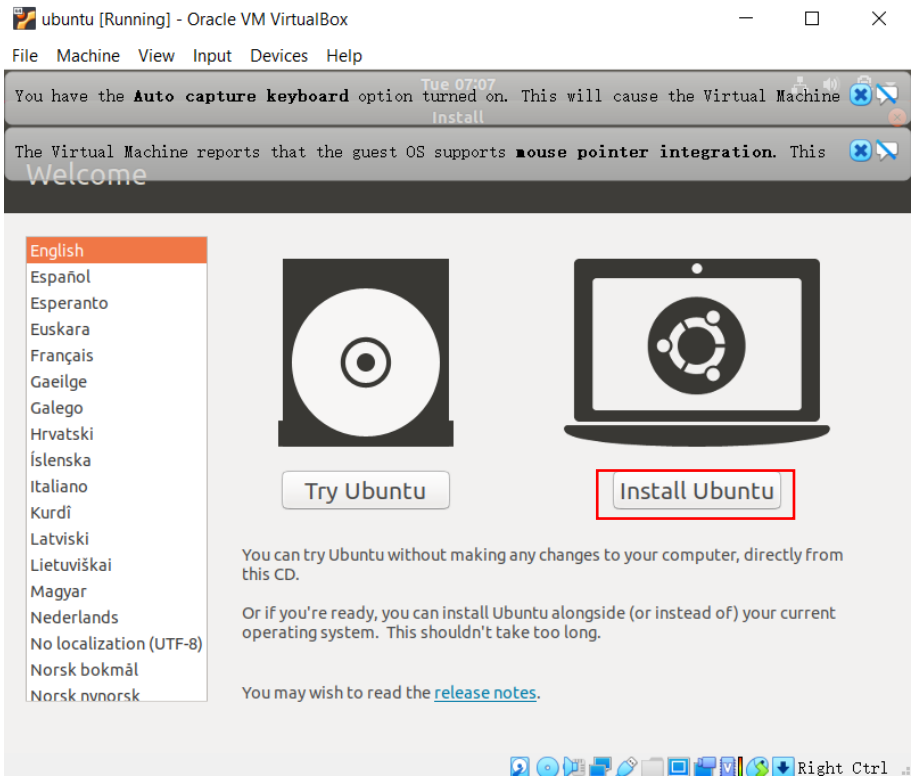
In the Optical Disk Selector pane Click Add. Then navigate to the location of the previously downloaded Ubuntu 18.04 image, select it and click Open.



Once the image has been selected and is highlighted, (there may be multiple images here from previous Virtual Machines, ensure the one required for this Virtual Machine is highlighted).

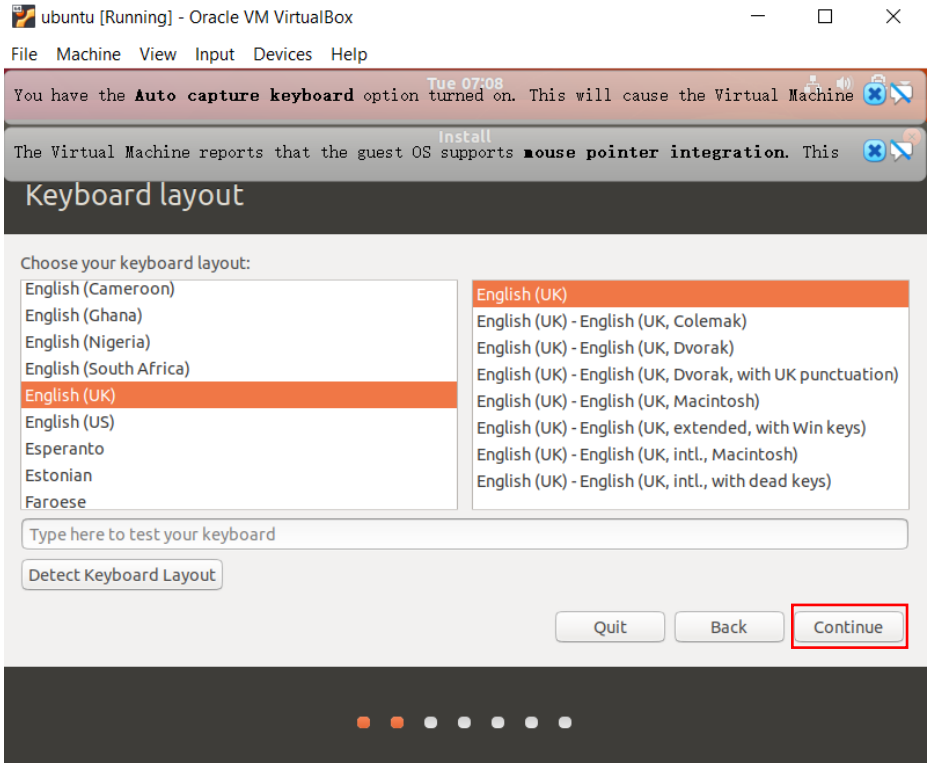
Then click Choose.

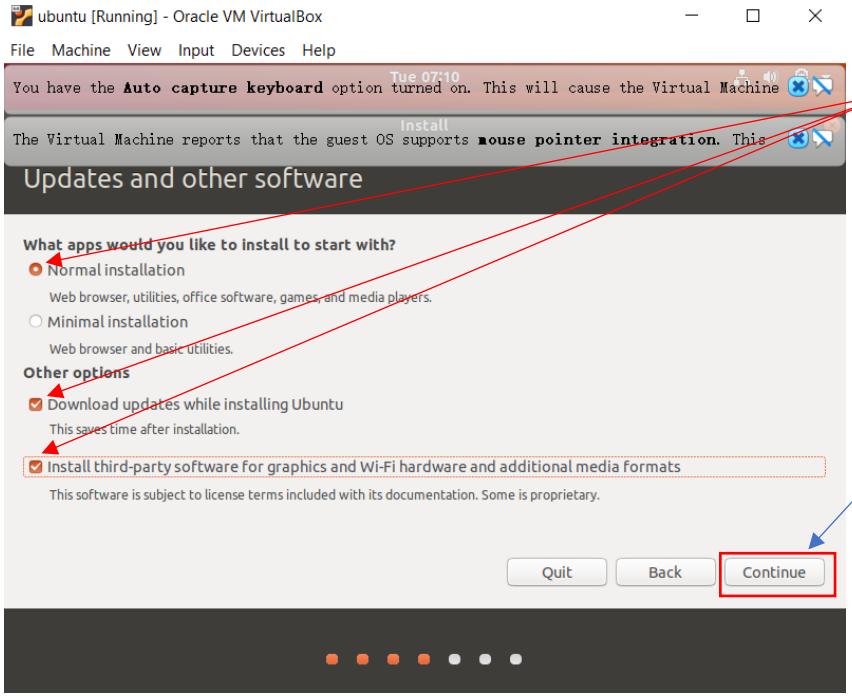
Again, check image selected, Now click Start.



Click Install Ubuntu

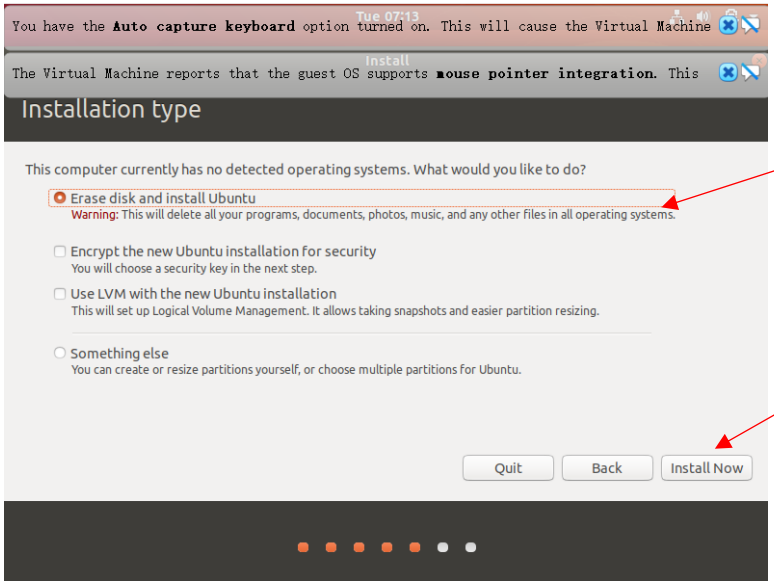
For labs it is recommended English (UK) and English (UK), but it is easy to change later within Ubuntu. Then click "Continue"





Ensure: Normal Installation, Download Updates and Install third party software are checked.

Click Continue

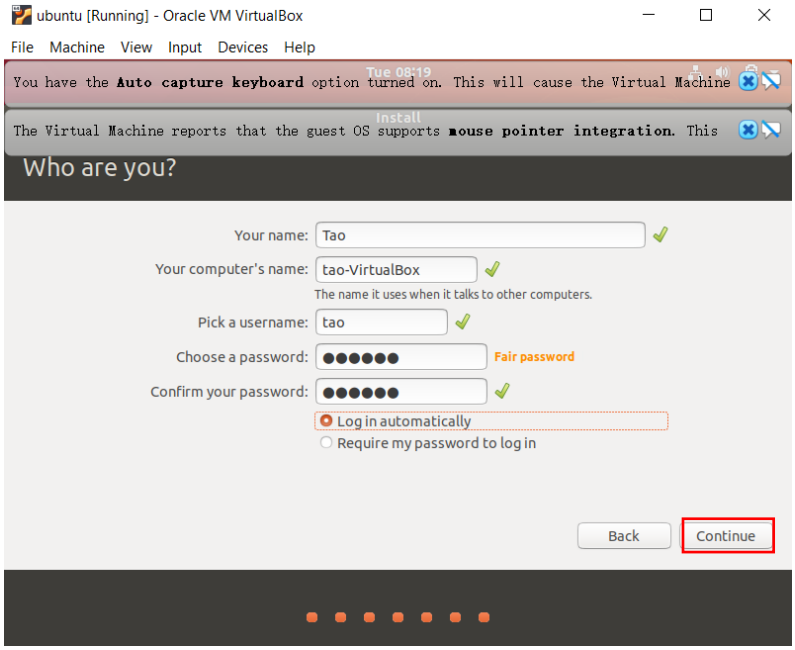


Check Erase disk and install Ubuntu. This is erasing the virtual disk created earlier, it is not erasing your hard drive.

Click Install Now

In the pop-up window, click continue.

In the next pane Where are you? accept the default London and click Continue.



Fill in your name and password.

Click Continue.

Ubuntu takes a few minutes to install showing various informative panels and a progress bar at the bottom of the pane.

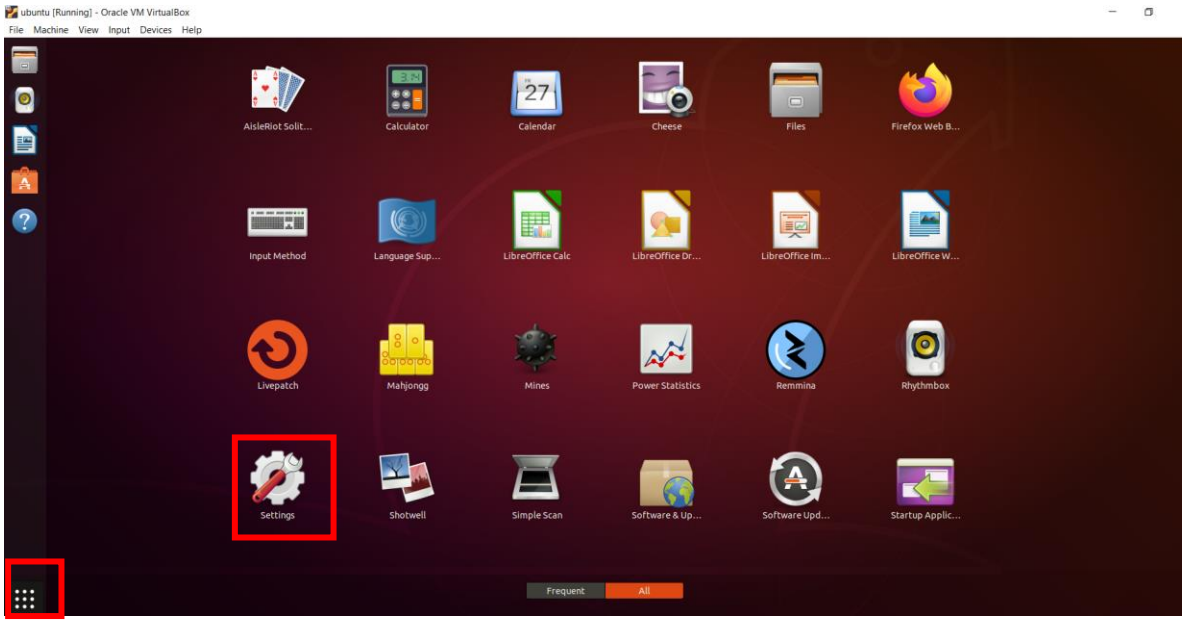
1.4 Configure Ubuntu

Once installation has completed, click Restart Now.

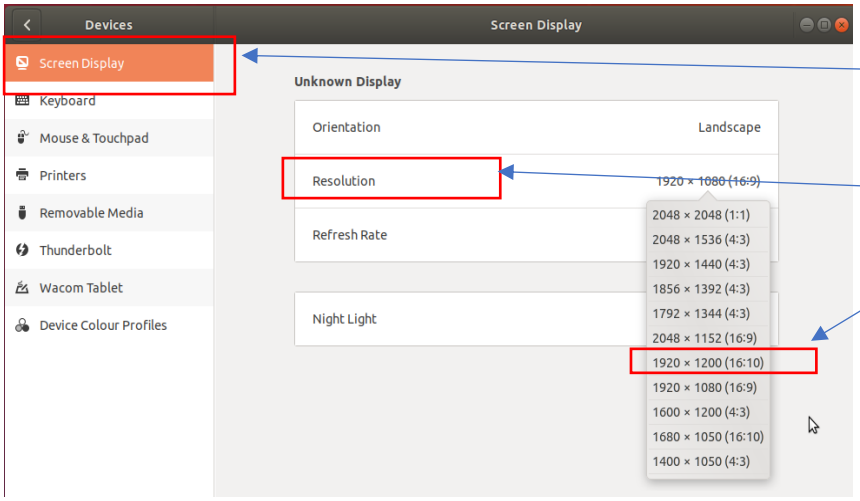
At the next pane press **ENTER**; there is no installation medium to remove.

There may also be a pop-up to Upgrade to Ubuntu 20.04, click **the Don't Upgrade** button and then Confirm.

To resize VM window, click **show applications,** select **Settings.**



Then in the Settings pane scroll down to Devices, located at the bottom of the list and click on it.



Select Screen Display.

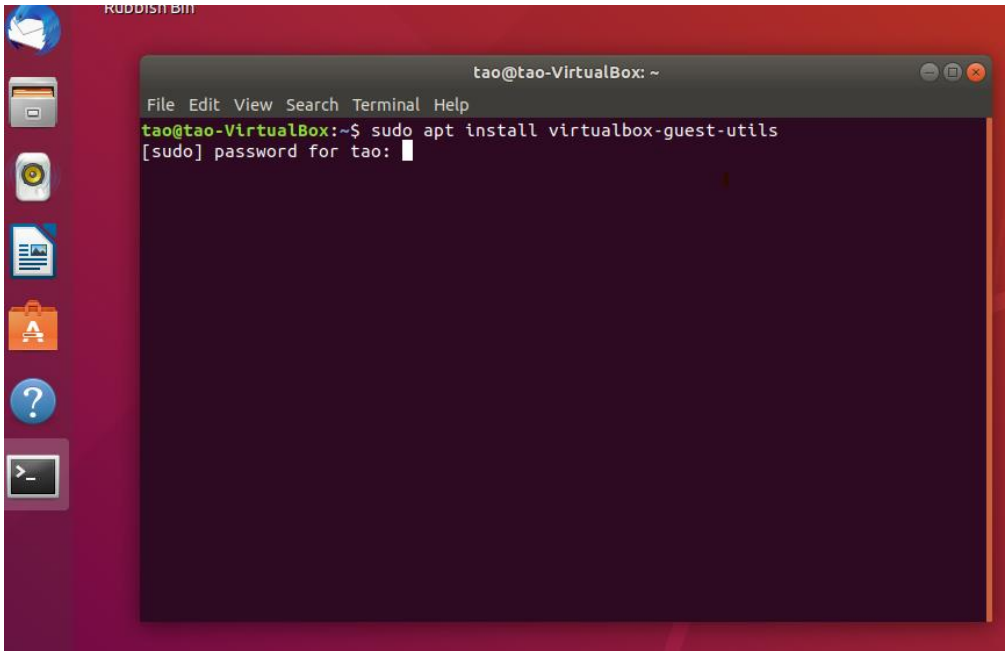
Select Resolution.

Select a best resolutionbased that can fit in VM window. please ensure your computer Computer display ratio

Then click **Apply** button inthe top right conner.

Now within the Ubuntu VM, press **CTRL+ALT+T**, to open a terminal**.**

Into the terminal type: **sudo apt install virtualbox-guest-utils** Followed by Enter your user password.



1.5 Install ROS melodic

Full instructions for installing ROS Melodic are found here:

http://wiki.ros.org/melodic/Installation/Ubuntu

The following are the relevant commands to be executed in the terminal and are extracted from the above link.

1) set up source （Copying the following instruction to terminal may result in incorrect format. It is recommended to copy the instruction from the above URL）

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

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

2) set up key （there is a space behind the word "key"）

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

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

3) Install ROS melodic

sudo apt update

sudo apt install ros-melodic-desktop-full （If it cannot be executed successfully, try it again.The download time is about 30 minutes）

4) Setup environment

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

source ~/.bashrc

5) Install dependencies

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

```
tao@tao-VirtualBox:~$ sudo apt install python-rosdep python-rosinstall python-rosinstall-generator python-wstool build-essential
```

sudo rosdep init (**Students in China Attention!!!**Please try several times. If you still fail, please see the next page)

rosdep update (**this command should not output any error information, otherwise you need to run it again until no error shown**)

```
tao@tao-VirtualBox:~$ rosdep update
reading in sources list data from /etc/ros/rosdep/sources.list.d
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/osx-homebrew.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/base.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/python.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/ruby.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/releases/fuerte.yaml
Query rosdistro index https://raw.githubusercontent.com/ros/rosdistro/master/index-v4.yaml
Skip end-of-life distro "ardent"
Skip end-of-life distro "bouncy"
Skip end-of-life distro "crystal"
Skip end-of-life distro "dashing"
Skip end-of-life distro "eloquent"
Add distro "foxy"
Add distro "galactic"
Skip end-of-life distro "groovy"
Skip end-of-life distro "hydro"
Skip end-of-life distro "indigo"
Skip end-of-life distro "jade"
Skip end-of-life distro "kinetic"
Skip end-of-life distro "lunar"
Add distro "melodic"
Add distro "noetic"
Add distro "rolling"
updated cache in /home/tao/.ros/rosdep/sources.cache
```

This is the correct information you should got

Until this, ROS is installed.

To solve a problem of launching Gazebo simulation, in terminal, type

gedit ~/.ignition/fuel/config.yaml

```
tao@tao-VirtualBox:~$ gedit ~/.ignition/fuel/config.yaml
```

```
---
# The list of servers.
servers:
  -
    name: osrf
    url: https://api.ignitionfuel.org

  # -
    # name: another_server
    # url: https://myserver

# Where are the assets stored in disk.
# cache:
#   path: /tmp/ignition/fuel
```

Change the url to https://api.ignitionrobotics.org

Save it.

➢

```
sudo rosdep init
```

(for students live in china,there will be several errors at this step.Like:”website may be down”and what shown in picture)

```
Traceback (most recent call last):
  File "/usr/lib/python2.7/dist-packages/rosdep2/main.py", line 146, in rosdep_main
    exit_code = _rosdep_main(args)
  File "/usr/lib/python2.7/dist-packages/rosdep2/main.py", line 441, in _rosdep_main
    return _no_args_handler(command, parser, options, args)
  File "/usr/lib/python2.7/dist-packages/rosdep2/main.py", line 450, in _no_args_handler
    return command_handlers[command](options)
  File "/usr/lib/python2.7/dist-packages/rosdep2/main.py", line 594, in command_init
    data = download_default_sources_list()
  File "/usr/lib/python2.7/dist-packages/rosdep2/sources_list.py", line 340, in download_default_sources_list
    f = urlopen(url, timeout=DOWNLOAD_TIMEOUT)
  File "/usr/lib/python2.7/urllib2.py", line 154, in urlopen
    return opener.open(url, data, timeout)
  File "/usr/lib/python2.7/urllib2.py", line 429, in open
    response = self._open(req, data)
  File "/usr/lib/python2.7/urllib2.py", line 447, in _open
    '_open', req)
  File "/usr/lib/python2.7/urllib2.py", line 407, in _call_chain
    result = func(*args)
  File "/usr/lib/python2.7/urllib2.py", line 1248, in https_open
    context=self._context)
  File "/usr/lib/python2.7/urllib2.py", line 1208, in do_open
    r = h.getresponse(buffering=True)
  File "/usr/lib/python2.7/httplib.py", line 1178, in getresponse
    response.begin()
  File "/usr/lib/python2.7/httplib.py", line 452, in begin
    version, status, reason = self._read_status()
  File "/usr/lib/python2.7/httplib.py", line 408, in _read_status
    line = self.fp.readline(_MAXLINE + 1)
  File "/usr/lib/python2.7/socket.py", line 480, in readline
    data = self._sock.recv(self._rbufsize)
  File "/usr/lib/python2.7/ssl.py", line 754, in recv
    return self.read(buflen)
  File "/usr/lib/python2.7/ssl.py", line 641, in read
    v = self._sslobj.read(len)
SSLError: ('The read operation timed out',)
```

So,

1)

```
sudo gedit /etc/hosts
```

2)add IP address

```
199.232.28.133 raw.githubusercontent.com
```

Or

```
151.101.84.133 raw.githubusercontent.com
```

```
127.0.0.1       localhost
127.0.1.1       txz-laptop

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
151.101.84.133 raw.githubusercontent.com
```

Save and quit

3)

```
sudo rosdep init
```

```
wyu@wyu-VirtualBox:~$ sudo rosdep init
Wrote /etc/ros/rosdep/sources.list.d/20-default.list
Recommended: please run

        rosdep update
```

➢ Then

```
rosdep update
```

generally you will meet this error

```
wyu@wyu-VirtualBox:~$ rosdep update
reading in sources list data from /etc/ros/rosdep/sources.list.d
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/osx-homebrew.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/base.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/python.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/ruby.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/releases/fuerte.yaml
Query rosdistro index https://raw.githubusercontent.com/ros/rosdistro/master/index-v4.ya
Skip end-of-life distro "ardent"
Skip end-of-life distro "bouncy"
Skip end-of-life distro "crystal"
Skip end-of-life distro "dashing"
Skip end-of-life distro "eloquent"
Add distro "foxy"
Add distro "galactic"
ERROR: error loading sources list:
        ('The read operation timed out',)
```

1)

```
sudo passwd root
```

Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
(your password won't be display on the the screen,just do it and ENTER)

```
wyu@wyu-VirtualBox:~$ sudo passwd root
[sudo] password for wyu:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
wyu@wyu-VirtualBox:~$ su -
```

2)

```
su -
```

3)

```
xdg-open /usr/lib/python2.7/dist-packages/rosdep2/sources_list.py
```

add

```
url="https://ghproxy.com/"+url
```

to line 311 in this file(Warning!!:DO NOT change the format and location of the code in the original file, the same is true below)

```python
def download_rosdep_data(url):
    """
    :raises: :exc:`DownloadFailure` If data cannot be
        retrieved (e.g. 404, bad YAML format, server down).
    """
    try:
        # http/https URLs need custom requests to specify the user-agent, since some repositories reject
        # requests from the default user-agent.
        url="https://ghproxy.com/"+url
        if url.startswith("http://") or url.startswith("https://"):
            url_request = request.Request(url, headers={'User-Agent': 'rosdep/{version}'.format(version=__version__)})
        else:
            url_request = url
        f = urlopen(url_request, timeout=DOWNLOAD_TIMEOUT)
        text = f.read()
        f.close()
        data = yaml.safe_load(text)
        if type(data) != dict:
            raise DownloadFailure('rosdep data from [%s] is not a YAML dictionary' % (url))
        return data
    except (URLError, httplib.HTTPException) as e:
        raise DownloadFailure(str(e) + ' (%s)' % url)
    except yaml.YAMLError as e:
        raise DownloadFailure(str(e))
```

4)

```
xdg-open /usr/lib/python2.7/dist-packages/rosdistro/__init__.py
```

change `DEFAULT_INDEX_URL` to

```
DEFAULT_INDEX_URL='https://ghproxy.com/https://raw.githubusercontent.com/ros/rosdistro/master/index-v4.yaml'
```

5)

```
xdg-open /usr/lib/python2.7/dist-packages/rosdep2/gbpdistro_support.py
```

Add“ `https://ghproxy.com/` ” to the front of the URL (line 36):

```python
21 from .rosdistrohelper import get_targets, get_release_file, PreRep137Warning
22
23 from .rep3 import download_targets_data  # deprecated, will output warning
24
25 import warnings
26
27 create_default_installer_context = None
28
29 # py3k
30 try:
31     unicode
32 except NameError:
33     basestring = unicode = str
34
35 # location of an example gbpdistro file for reference and testing
36 FUERTE_GBPDISTRO_URL = 'https://ghproxy.com/https://raw.githubusercontent.com/ros/rosdistro/' \
37     'master/releases/fuerte.yaml'
38
39 # seconds to wait before aborting download of gbpdistro data
40 DOWNLOAD_TIMEOUT = 15.0
41
42
43 def get_owner_name(url):
44     """
45     Given a gbpdistro url, returns the name of the github user in the url.
46
47     If the url is not a valid github url it returns the default `ros`.
48
49     This information is used to set the homebrew tap name, see:
50     https://github.com/ros-infrastructure/rosdep/pull/17
51
52     :returns: The github account in the given gbpdistro url
53     """
54     result = 'ros'
55     try:
56         parsed = urlparse.urlparse(url)
57         if parsed.netloc == 'github.com':
```

Display line numbers ☑
Display right margin ☐
Highlight current line ☐
Text wrapping ☑

Python Tab Width: 8 Ln 206, Col 24 INS

6)

In the same way,add “ `https://ghproxy.com/` ”to the front of the corresponding URL in these files:

`/usr/lib/python2.7/dist-packages/rosdep2/sources_list.py` line 72
`/usr/lib/python2.7/dist-packages/rosdep2/rep3.py` line 39
`/usr/lib/python2.7/dist-packages/rosdistro/manifest_provider/github.py` line 68 and line 119

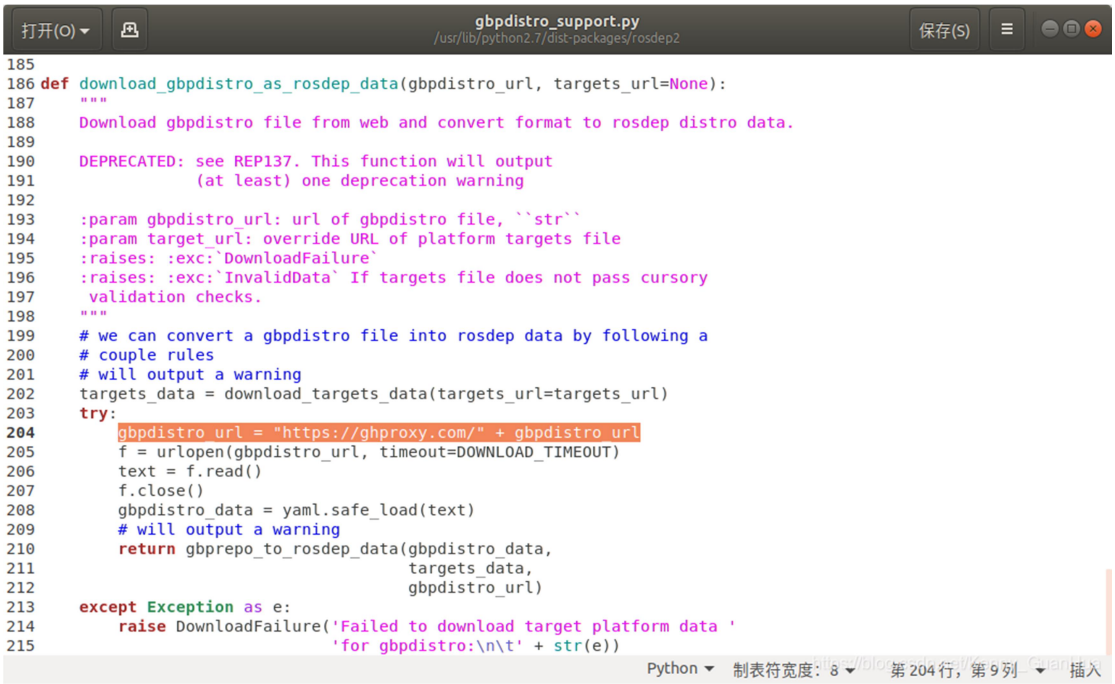
7)

```
xdg-open /usr/lib/python2.7/dist-packages/rosdep2/gbpdistro_support.py
```

add

```
gbpdistro_url = "https://ghproxy.com/" + gbpdistro_url
```

to line204




```
185
186 def download_gbpdistro_as_rosdep_data(gbpdistro_url, targets_url=None):
187     """
188     Download gbpdistro file from web and convert format to rosdep distro data.
189
190     DEPRECATED: see REP137. This function will output
191                 (at least) one deprecation warning
192
193     :param gbpdistro_url: url of gbpdistro file, ``str``
194     :param target_url: override URL of platform targets file
195     :raises: :exc:`DownloadFailure`
196     :raises: :exc:`InvalidData` If targets file does not pass cursory
197      validation checks.
198     """
199     # we can convert a gbpdistro file into rosdep data by following a
200     # couple rules
201     # will output a warning
202     targets_data = download_targets_data(targets_url=targets_url)
203     try:
204         gbpdistro_url = "https://ghproxy.com/" + gbpdistro_url
205         f = urlopen(gbpdistro_url, timeout=DOWNLOAD_TIMEOUT)
206         text = f.read()
207         f.close()
208         gbpdistro_data = yaml.safe_load(text)
209         # will output a warning
210         return gbprepo_to_rosdep_data(gbpdistro_data,
211                                       targets_data,
212                                       gbpdistro_url)
213     except Exception as e:
214         raise DownloadFailure('Failed to download target platform data '
215                               'for gbpdistro:\n\t' + str(e))
```

8)finally,

```
Exit
```

9)

```
rosdep update
```

```
root@wyu-VirtualBox:~# xdg-open /usr/lib/python2.7/dist-packages/rosdep2/sources_list.py
root@wyu-VirtualBox:~# xdg-open /usr/lib/python2.7/dist-packages/rosdistro/__init__.py
root@wyu-VirtualBox:~# xdg-open /usr/lib/python2.7/dist-packages/rosdep2/gbpdistro_support.py
root@wyu-VirtualBox:~# xdg-open /usr/lib/python2.7/dist-packages/rosdep2/sources_list.py
root@wyu-VirtualBox:~# xdg-open /usr/lib/python2.7/dist-packages/rosdep2/rep3.py
root@wyu-VirtualBox:~# xdg-open /usr/lib/python2.7/dist-packages/rosdistro/manifest_provider/github.py
root@wyu-VirtualBox:~# xdg-open /usr/lib/python2.7/dist-packages/rosdep2/gbpdistro_support.py
root@wyu-VirtualBox:~# rosdep update
reading in sources list data from /etc/ros/rosdep/sources.list.d
Warning: running 'rosdep update' as root is not recommended.
  You should run 'sudo rosdep fix-permissions' and invoke 'rosdep update' again without sudo.
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/osx-homebrew.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/base.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/python.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/ruby.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/releases/fuerte.yaml
Query rosdistro index https://ghproxy.com/https://raw.githubusercontent.com/ros/rosdistro/master/index-v4.yaml
Skip end-of-life distro "ardent"
Skip end-of-life distro "bouncy"
Skip end-of-life distro "crystal"
Skip end-of-life distro "dashing"
Skip end-of-life distro "eloquent"
Add distro "foxy"
Add distro "galactic"
Skip end-of-life distro "groovy"
Skip end-of-life distro "hydro"
Skip end-of-life distro "indigo"
Skip end-of-life distro "jade"
Skip end-of-life distro "kinetic"
Skip end-of-life distro "lunar"
Add distro "melodic"
Add distro "noetic"
Add distro "rolling"
Updated cache in /root/.ros/rosdep/sources.cache
root@wyu-VirtualBox:~# exit
logout
```