Virtual Linux OS development environment

I have pre-configured/build Ubuntu (Linux) OS VM image running on VirtualBox (you can also use the image with VMware player) which include all the necessary software for the class.

I included the following software in the prebuild image:

- ✓ **C++**, Python, **Perl**, Java
- ✓ MySQL
- ✓ IDE: QT, Geany
- ✓ LibreOffice (Microsoft office compatible)
- ✓ And some sample C++ codes: Classes, MySQL connector; etc.
- ✓ You can also add any other software/packages to the images since the OS has a persistent state.

By using the pre-build virtual Linux OS image for our software development environment, we can save a lot of time at the beginning of class so we don't spend a time trying to figure out computer related issues for the software development environment since I have tested and grouped all the necessary software we will use for the class.

If you want to install the software yourself without using the prebuild OS image (it is ok to use your own software development environment such Microsoft Visual Studio and Xcode), please make sure you install C++, MySQL server as we will be using them in our class.

Unfortunately, Ohlone lab computer has tight security so you can't install/run the VM (require system administrator privilege to install) on our CNET lab computer. So, you need to install the Ubuntu OS image on your home computer/laptop.

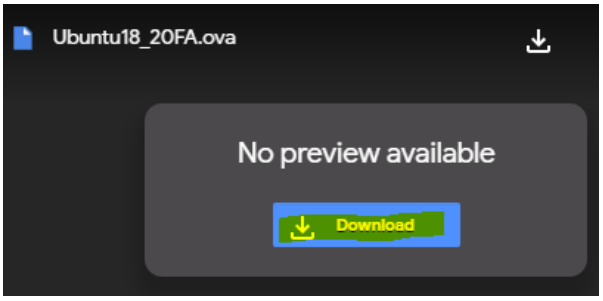
Below is the documentation (how to download & install the image), feel free to use it if you want.

Download software:

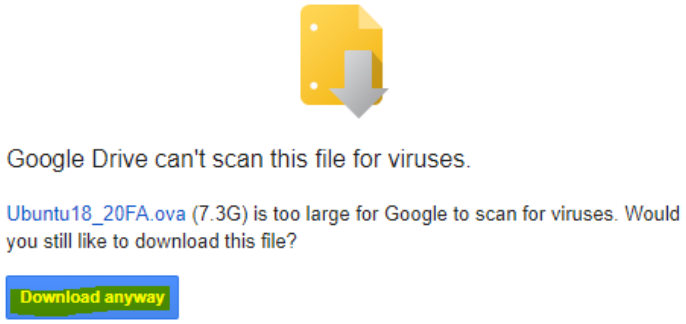
Download and install Oracle VM VirtualBox: https://www.virtualbox.org/ (You can install the VirtualBox on your window PC, MacOS, Linux.)

Prebuild Ubuntu OS Image file: Ubuntu18_20FA.ova. This image file is 64bits based as most computer for the past 3 years are 64bits. If you have an older computer that is 32bits based, please let me so make sure it is 64bits based (not 32bits). You can download the prebuild Ubuntu OS image file for this class here.

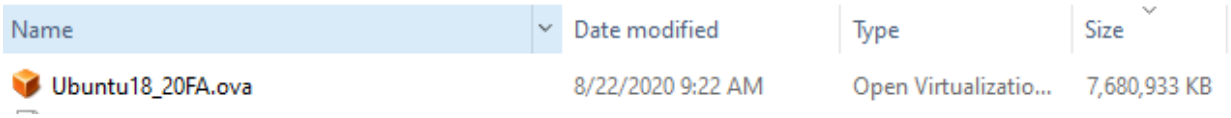
https://drive.google.com/file/d/1wpr4veHaV5gKxX0RmxFgxFQN3VZ0Zu6D/view?usp=sharing



The file is over 7GB in size, so, please download it when you have a good internet connection before you download the file.



Don't worry about warning can't scan for virus, just " download anyway" button. Depends on your internet speed, it may take between 5-10 minutes to download the file. After it is successfully downloaded, by default, the file will be saved in the Download folder.



(The .ova file can be imported to VirtualBox or VMware player)

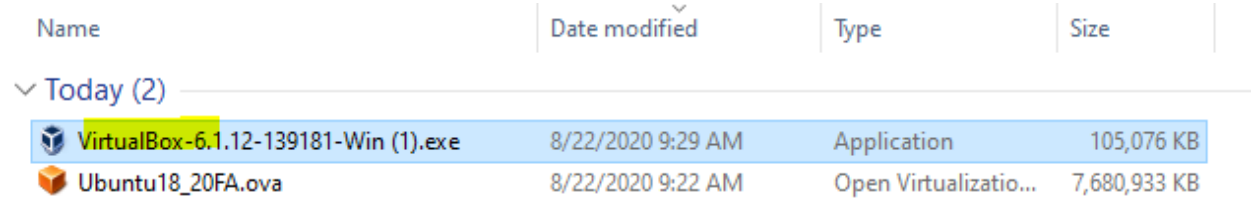
The prebuild Image OS is based on Ubuntu, username/password:

Username: ***ohlone***
Password: ***ohlone***

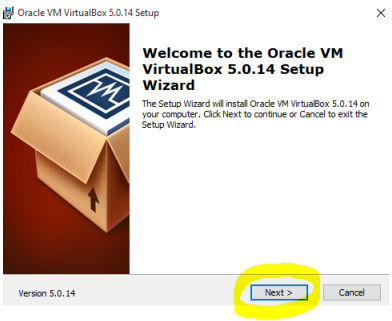
Installing Virtualization Software – VirtualBox

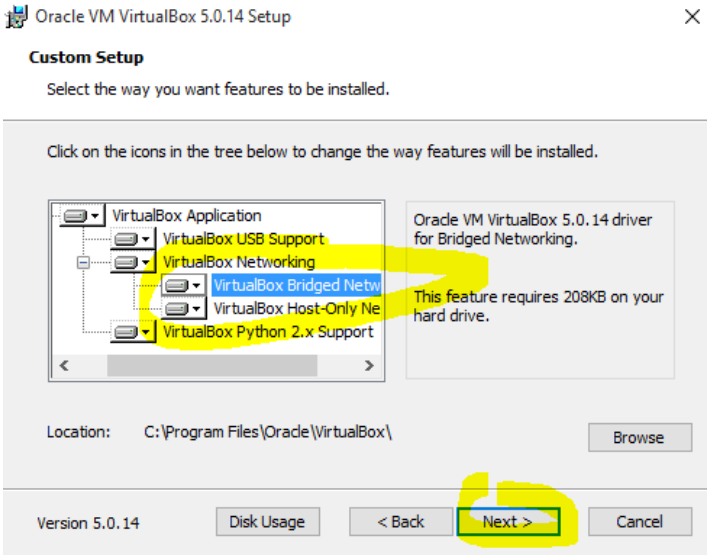
- Go to https://www.virtualbox.org/ to download the VirtualBox software (https://www.virtualbox.org/wiki/Downloads)
- Select the right platform packages for your OS client. If you are using Windows OS, then select Windows hosts. Apple Mac uses OS X hosts.

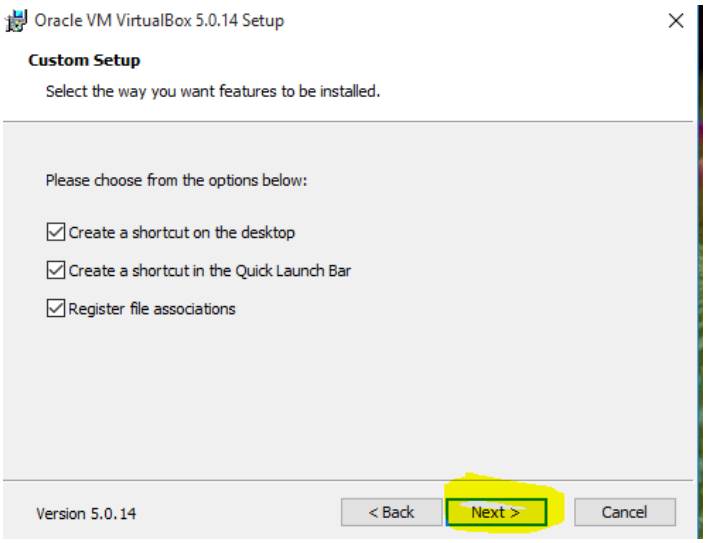
After the VirtualBox is successfully downloaded, by default, it should be at the Download folder.

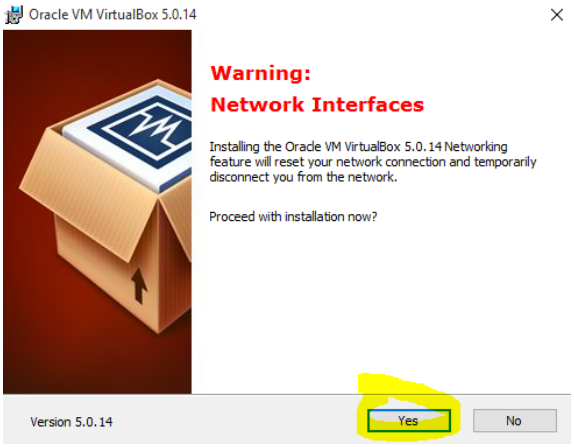


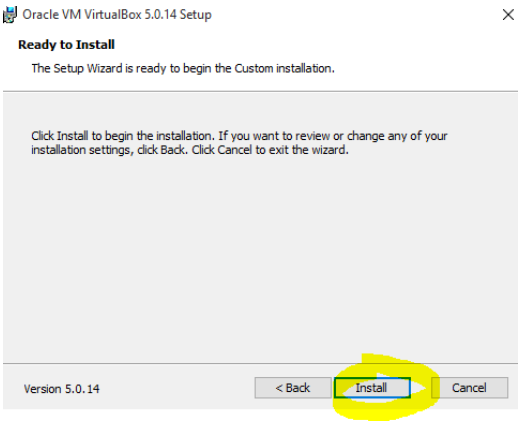
Double click on the VirtualBox software you just downloaded to install the software.



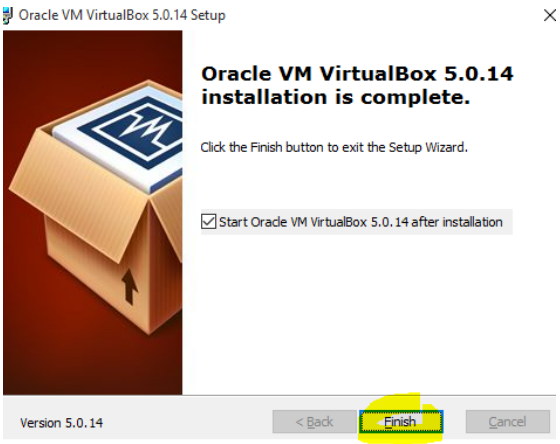




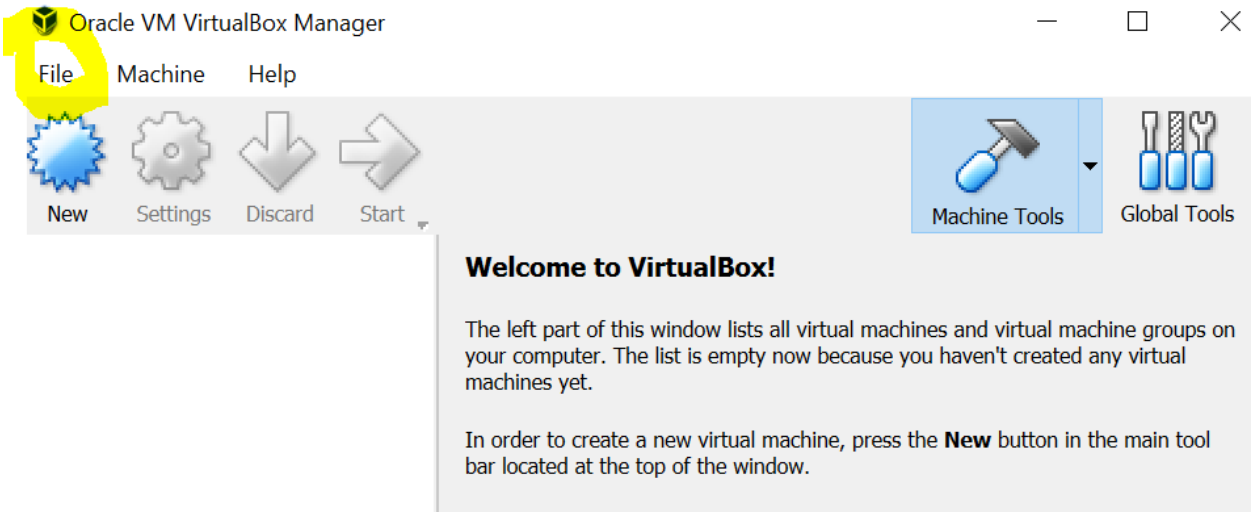




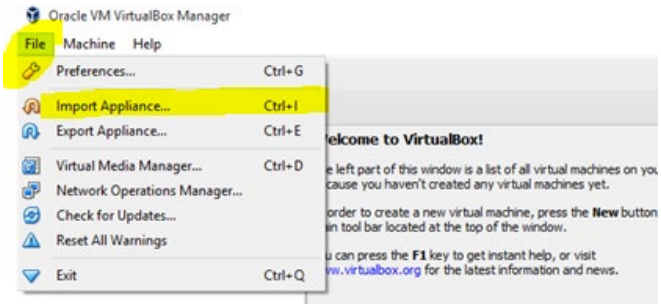
Answer yes, if prompt to install any device driver

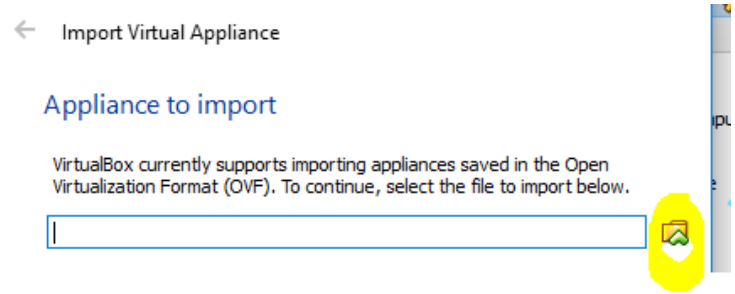


After the VirtualBox software is successfully installed, you need to configure and load the OS to VirtualBox.

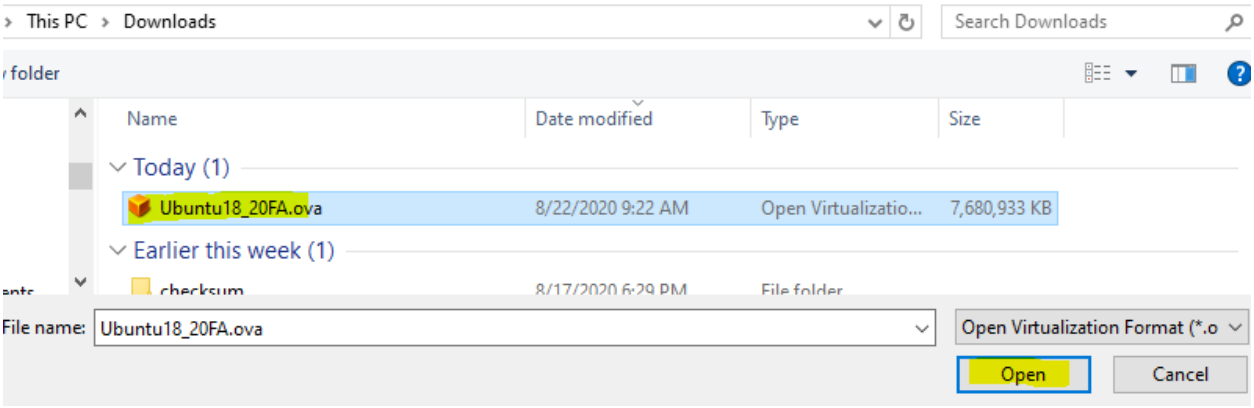


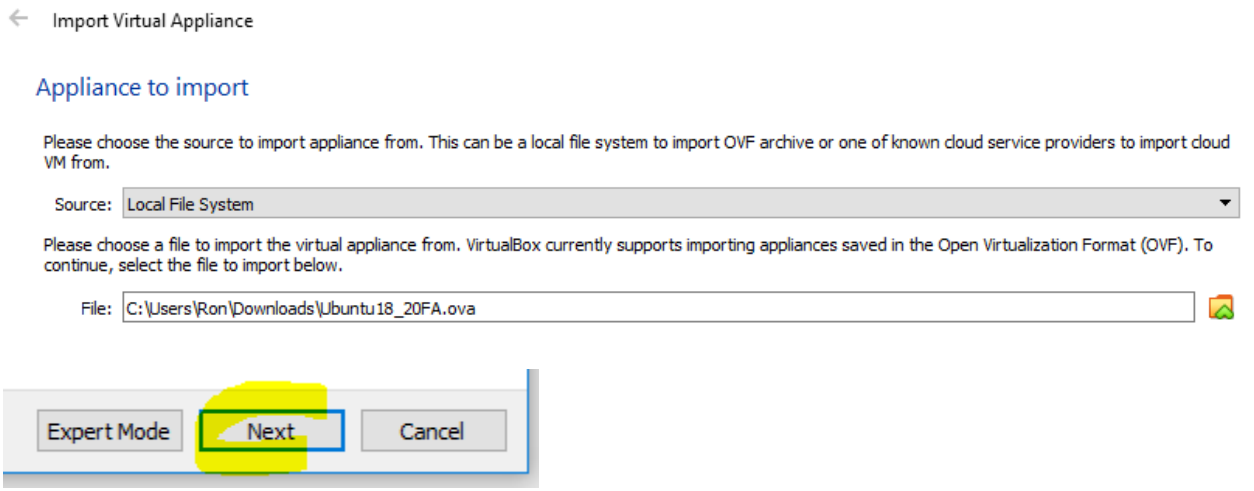
File-> Import appliance and select the *Ubuntu18_20FA.ova* file you just downloaded. Usually the file is in your Downloads folder.

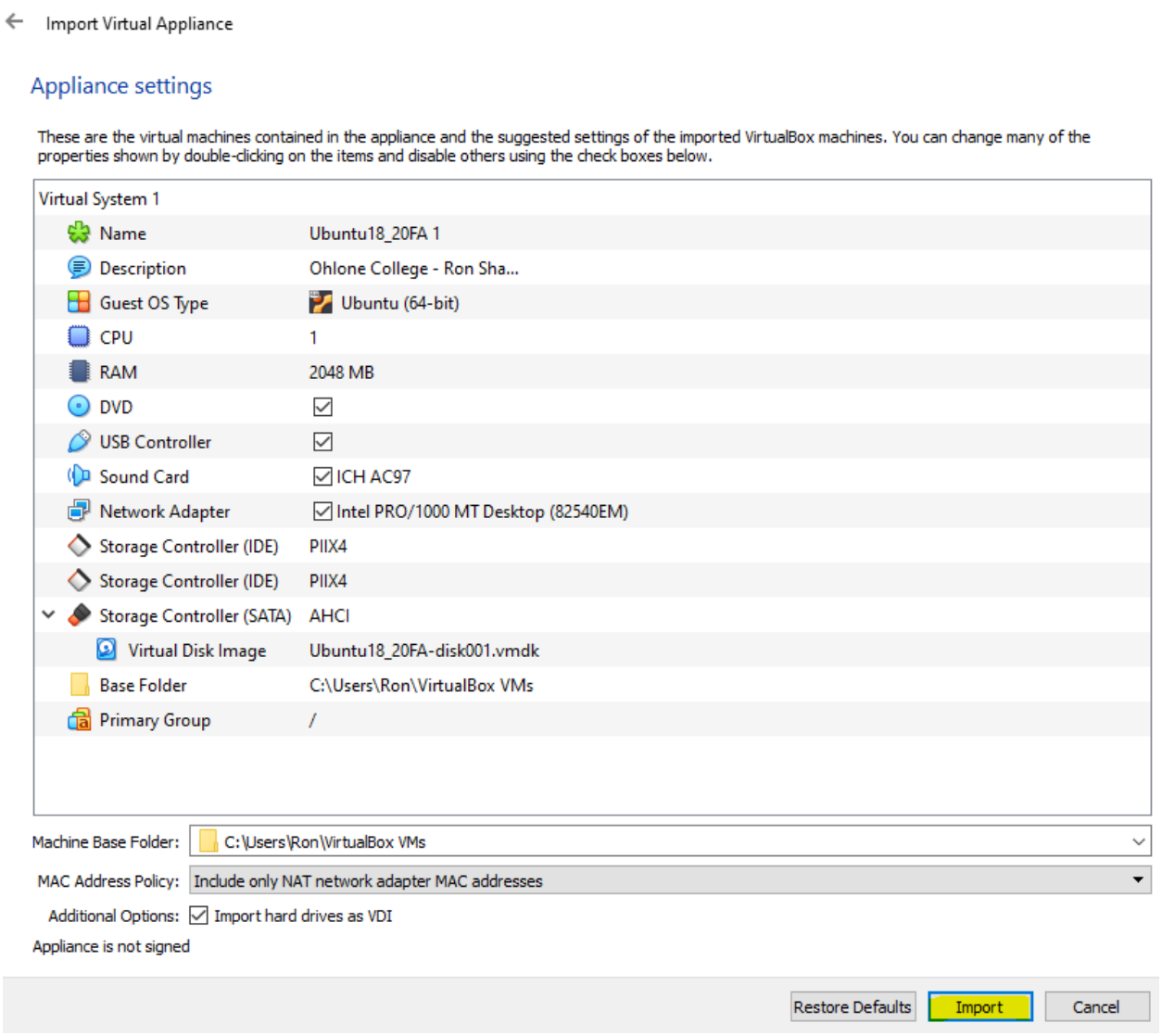


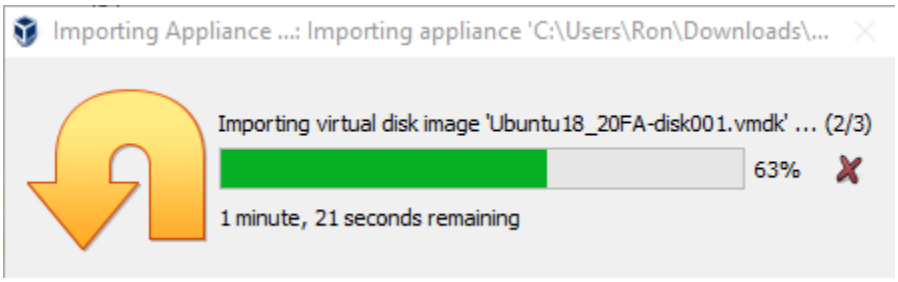


This is where you save the OVF file. Go to that directory where the *Ubuntu18_20FA.ova* file is at and select that file to be imported. (The file name is the name of the file you downloaded, it usually is in the Download folder or the folder where you saved the file.).

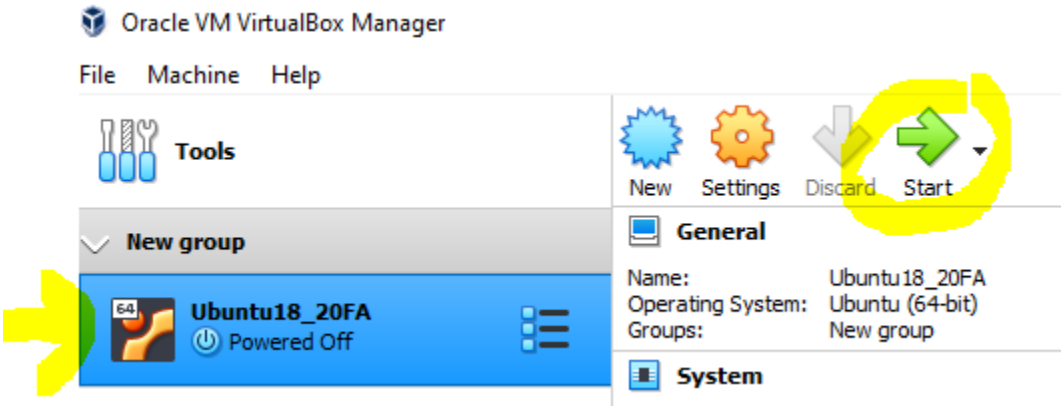






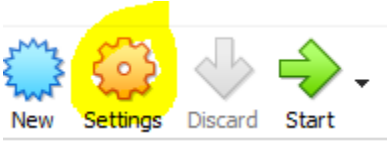


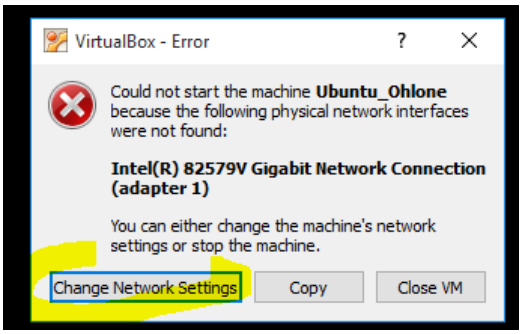
After finished importing, select the image you just imported (the name you assigned for the image you just imported, and click "Start"



The ubuntu OS will be loaded.

If you get error Networking Connection error, then "Change Network Settings" to NAT in





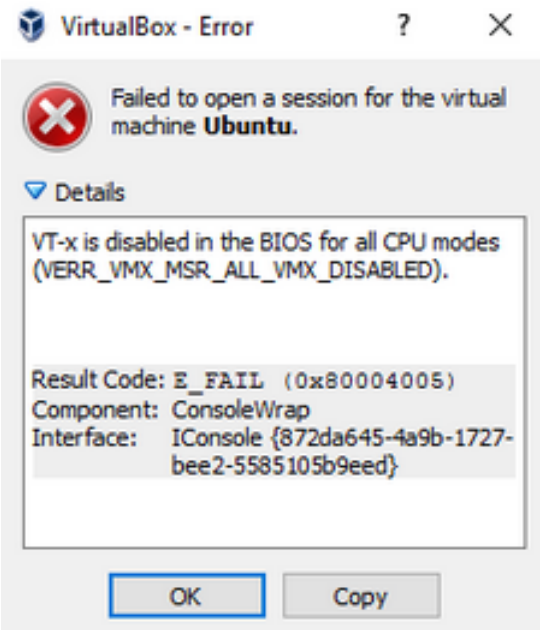
If you get this error below, please follow the instruction to fix the issue:

```
You are in emergency mode. After logging in, type "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, "systemctl default" or "exit"
to boot into default mode.
Press Enter for maintenance
```

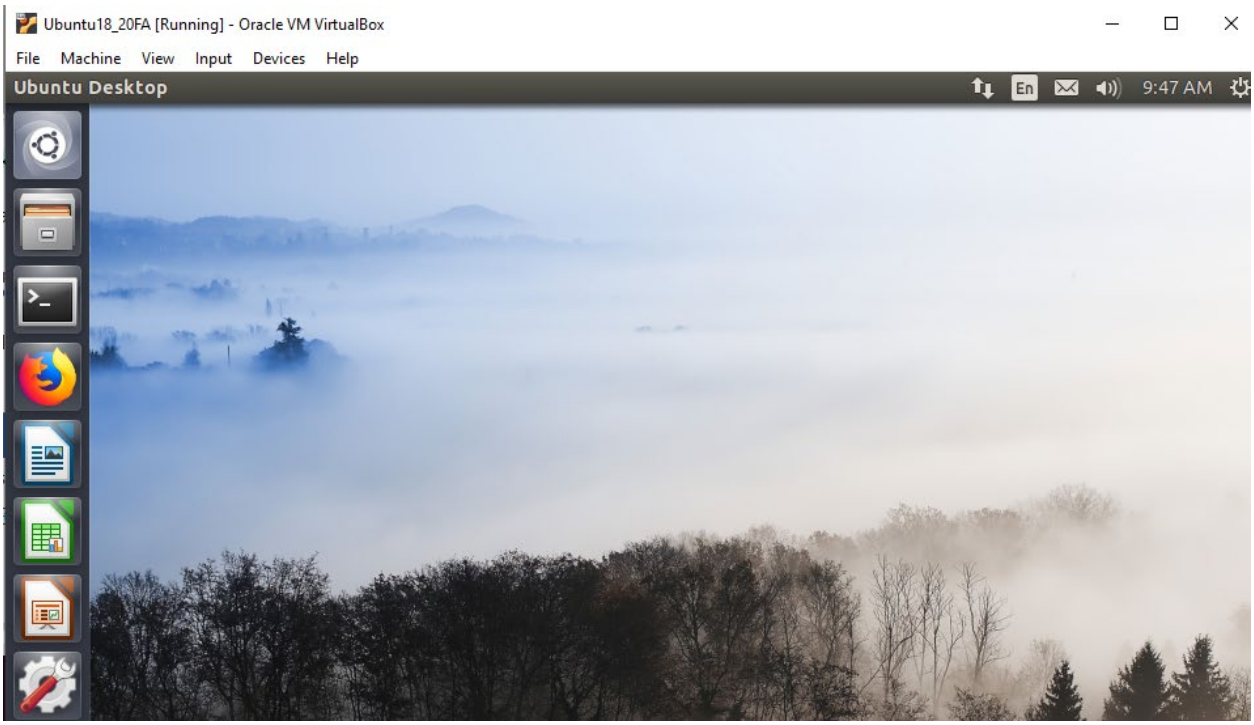
Virtualization BIOs

If you get this error below, that means that your computer/laptop virtualization is not enabled in bios. Since Access Commander runs in a virtual environment, your computer's BIOS settings must be configured to allow virtualization. Accessing a computer's BIOS will differ from manufacturer to manufacturer. Additionally, the same manufacturer may implement different BIOS versions depending on the age or model of your computer.

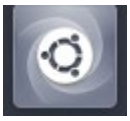
https://helpdeskgeek.com/how-to/enable-virtualization-in-the-bios/

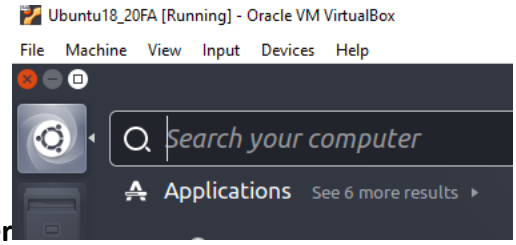


If all goes well, the import should be successful and get this screen

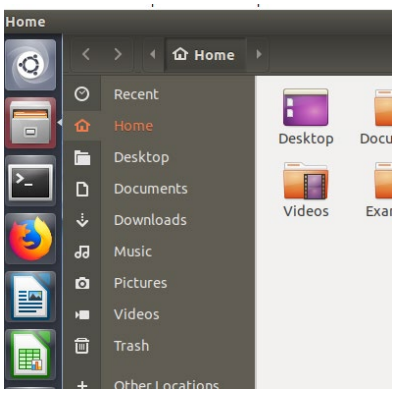


There are set of icons on the left side of the screen.

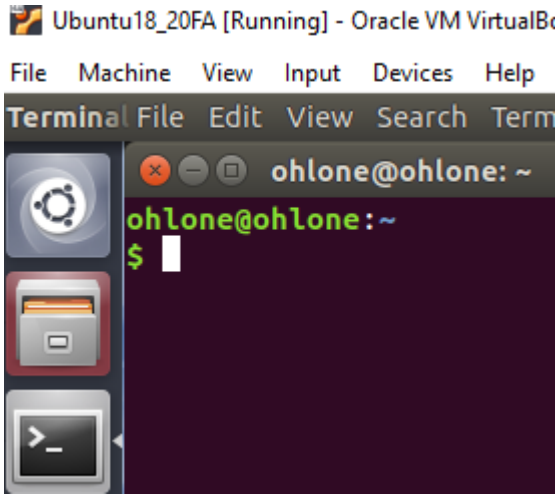
 **Search anything in the computer**



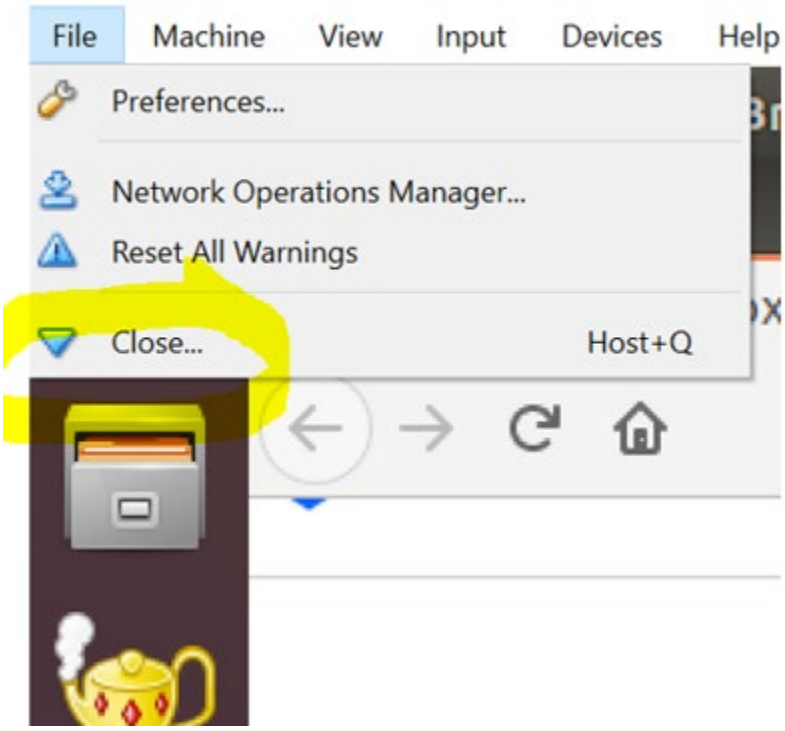
 **File Manager (File Explorer)**

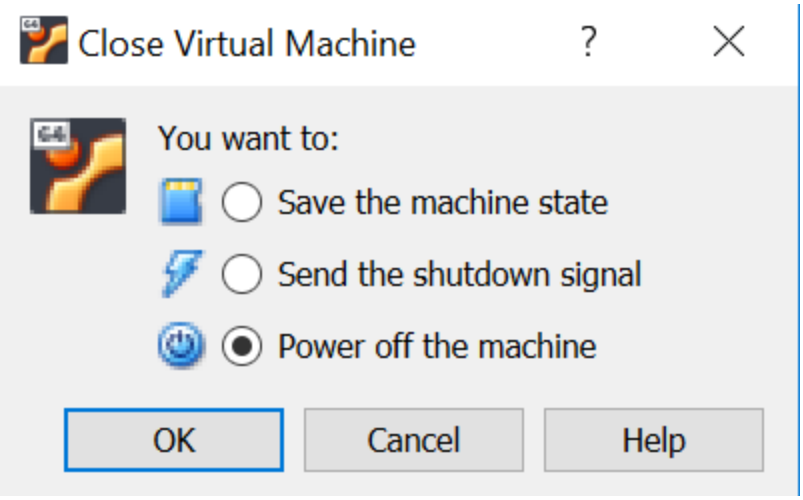


 **This is command line tool.**



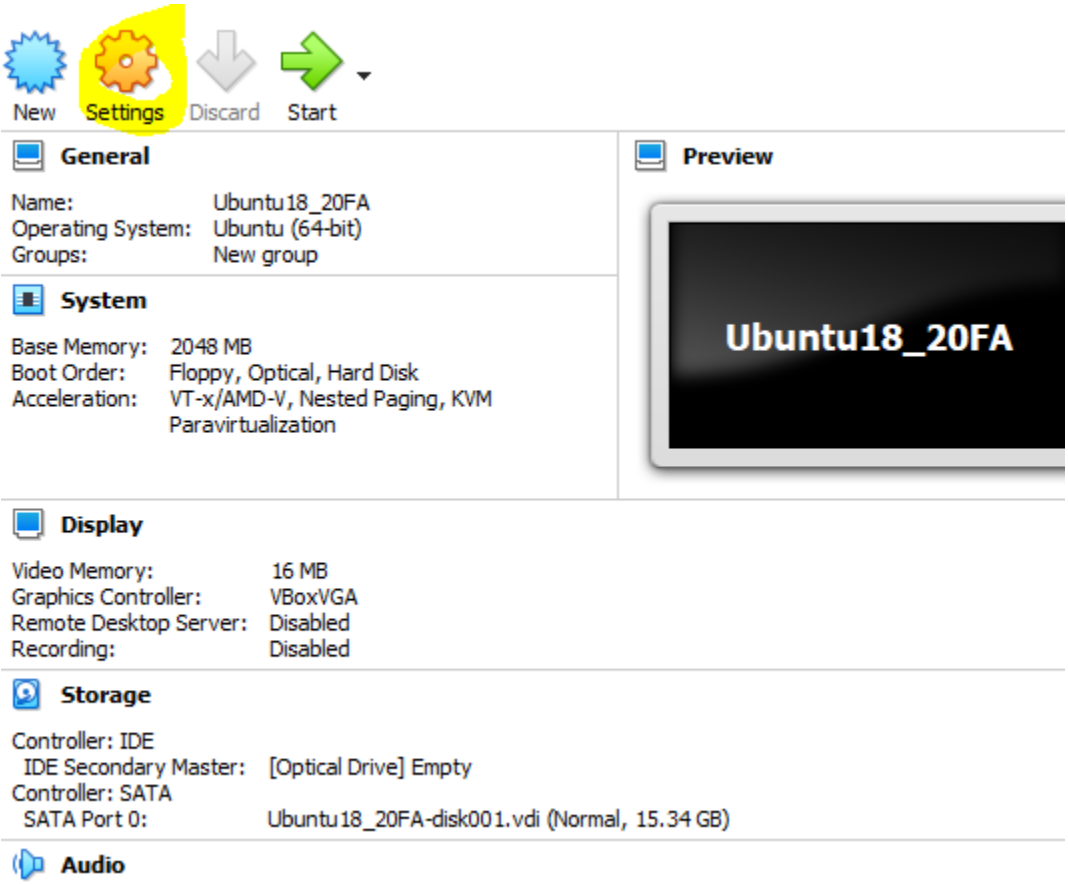
To shut down the VM,



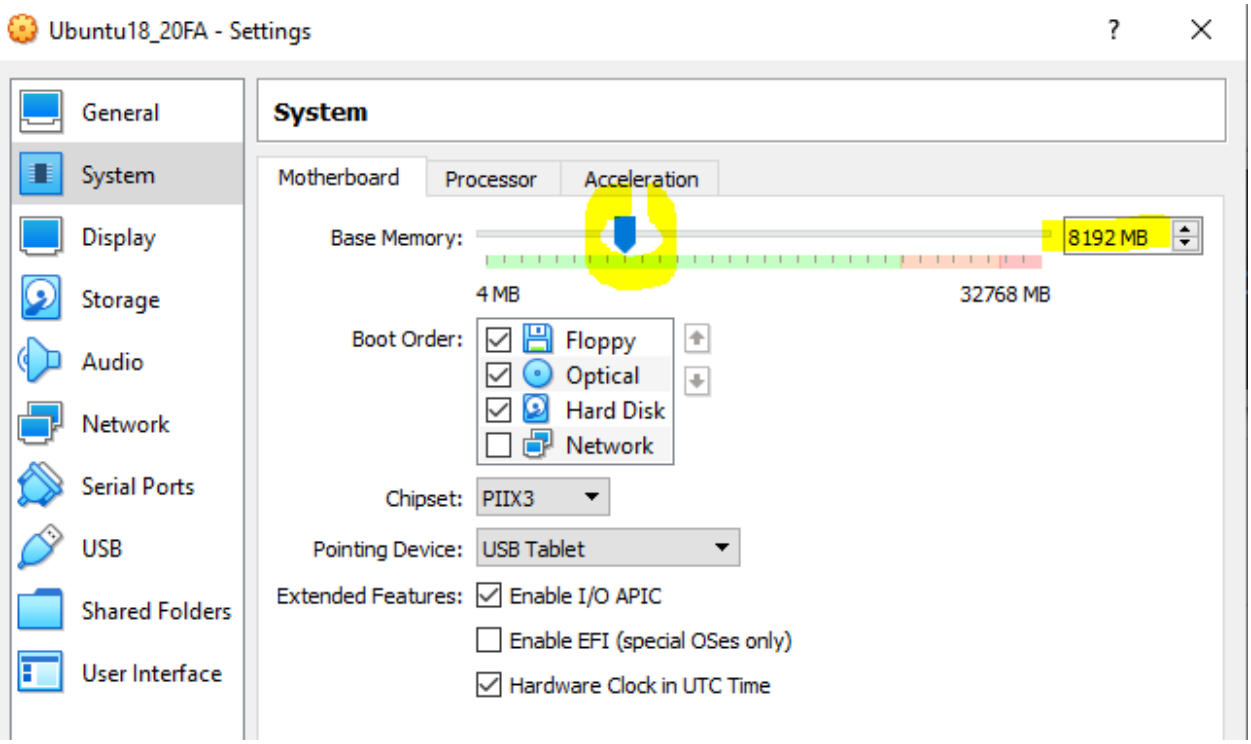


- **Save the machine state – this will save the current state and will resume what it left off when you restart the VM**
- **Power off the machine – this will shutdown the system**

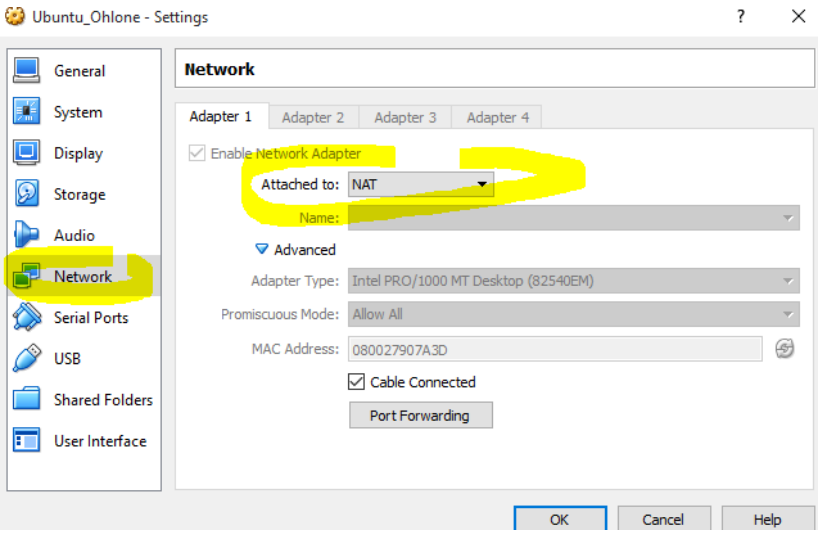
If you want to make any changes to the settings (don't make any changes if it works)



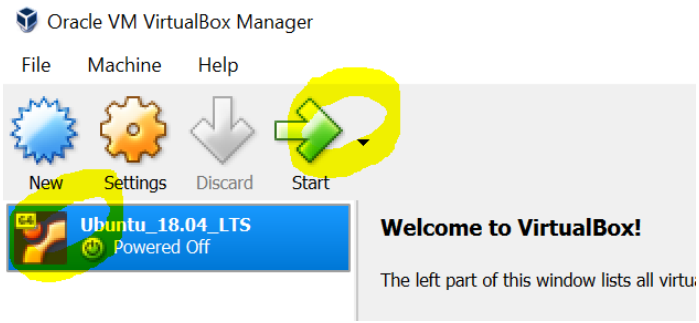
You should set the base memory to a minimum 4GB (4-8GB) so the system performance will be acceptable.



Network configuration so you can access your base system networking connection. You SHOULD NOT have to do this if all is working. For now, use the following as the sample settings. BUT DON'T CHANGE ANYTHING IF IT IS WORKING.

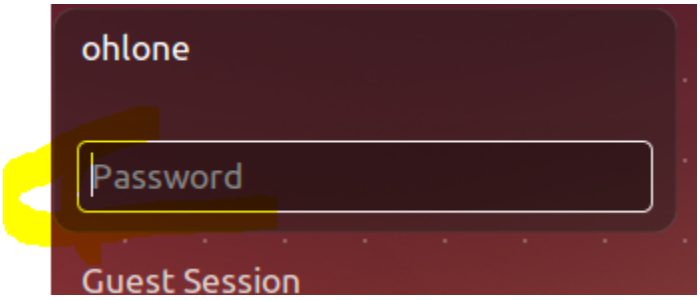


To start the VM:



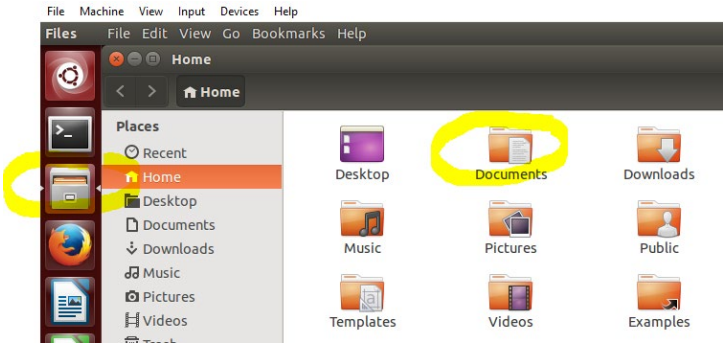
Select the Image and click "Start"

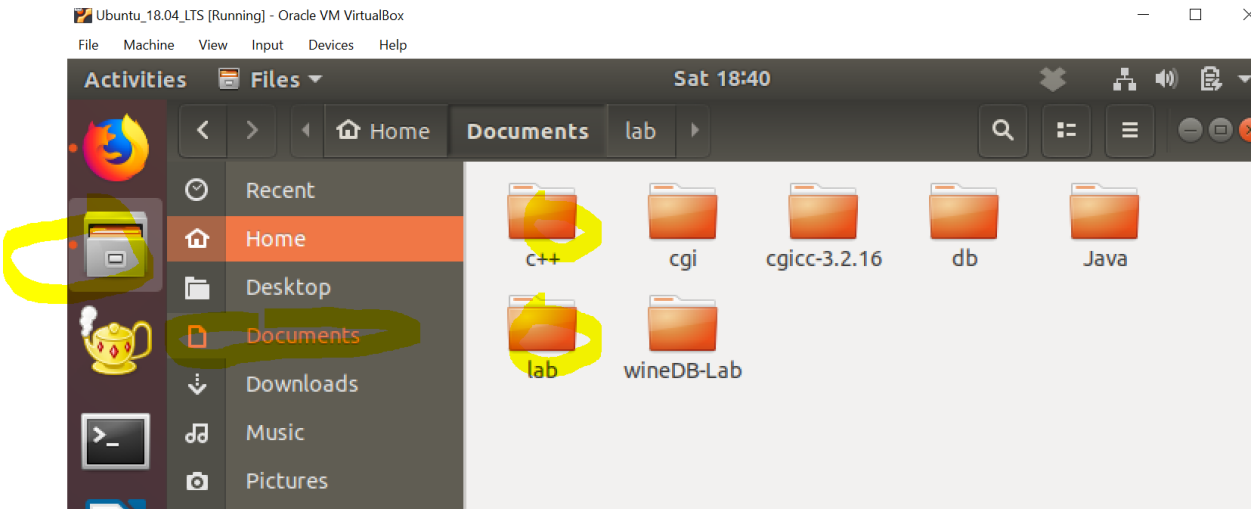
It should go directly to Ubuntu system without the login screen. However, if you get the login screen, use the username *ohlone* and password *ohlone* to login



User name: *ohlone*
Password: *ohlone*

Once you login, you can access File Folders, Firefox web browser, etc.



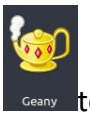


The "Documents" directory contains many C++ example programs. Go into C++ directory and start compile any of the .cpp file and it should compile and able to run the program.

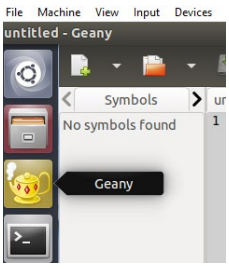
For this class, you should place all the lab assignments in the lab folder. If the ***lab*** folder doesn't exist, you need to **create** the ***lab*** folder.

Geany: C++ IDE - https://www.geany.org/manual/current/index.html

Geany is a C++ IDE where you can write your C++ and compile your C++ program. You can search 

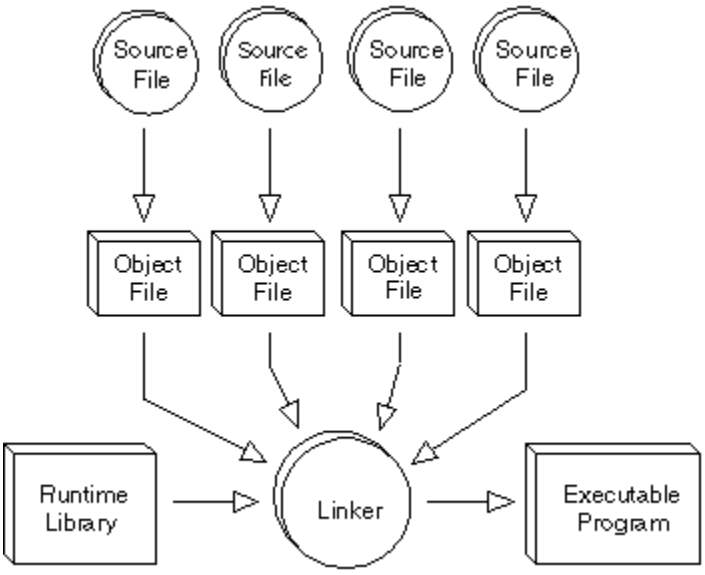
the Geany application by using the search and then place the Geany IDE application to the left side of the shortcut by dragging the Geany icon 

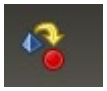
to the left.



How to use Geany to compile and run your programs - 

What is compile a program means?



To **compile** the current program, use  icon.

- Compile is to transform a program written in a high-level programming language from *source code* into *object code*. Programmers write programs in a form called source code. Source code must go through several steps before it becomes an executable program. The first step is to pass the source code through a *compiler*, which translates the high-level language instructions into object code.

To **Build/Link** (create executable) program, use 

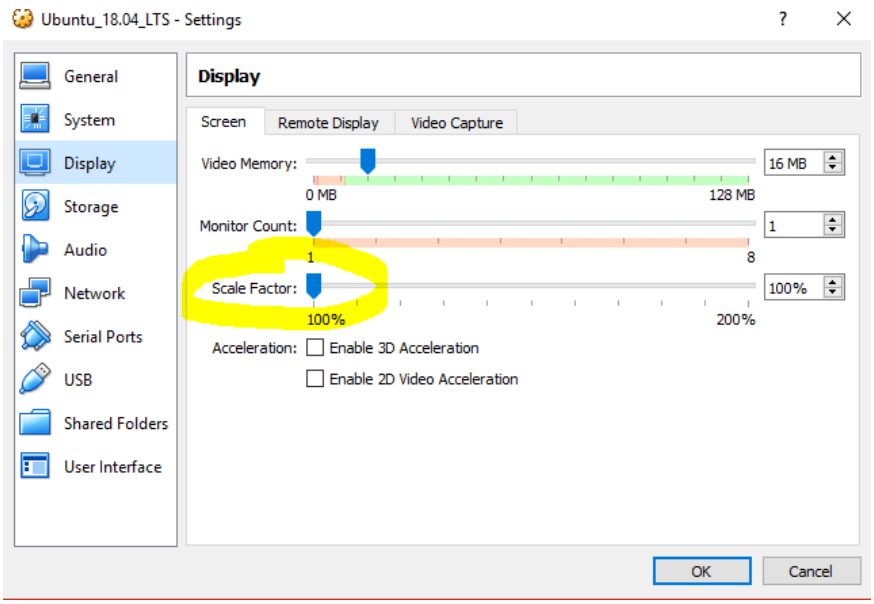
- The final step in producing an executable program -- after the compiler has produced object code -- is to pass the object code through a *linker*. The linker combines modules and gives real values to all symbolic addresses, thereby producing machine code

To **run** the program you just build, use 

- Run the program

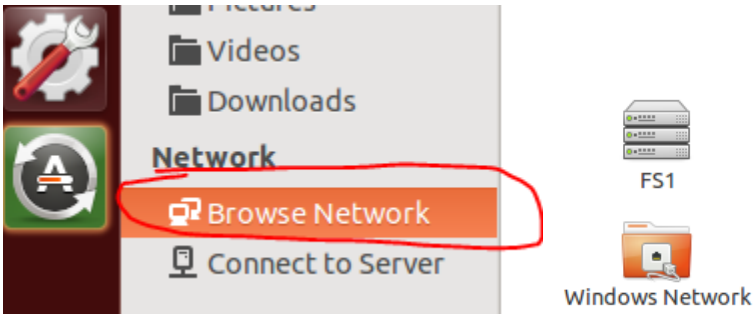
Display

If your screen display is too small, you can adjust the scale of the display by change the scale factor



Copy files between VM host and VM:

In VM Ubuntu, select "**Browse Network**". If you home network is Open, you will be able to see other servers on your network. But, if the network is closed, then you would not see any thing.

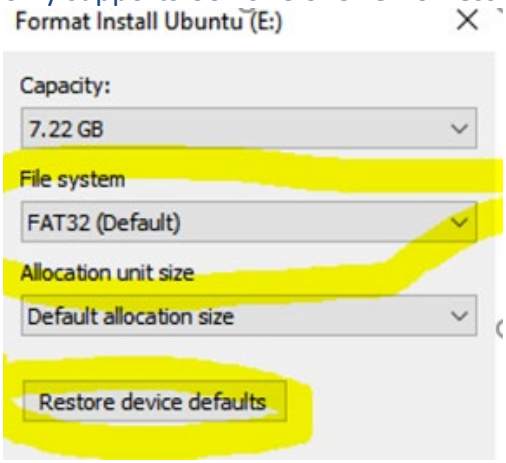


A list of hosts from your network will appear,

You can then access the network host. You may be asked to authenticate if your network has security setup.

USB – transfer files

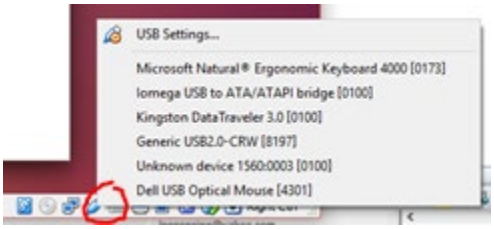
File USB drive must be formatted as FAT32 in order for both window, Mac, Linux to see it. It only supports USB size of 32GB or less



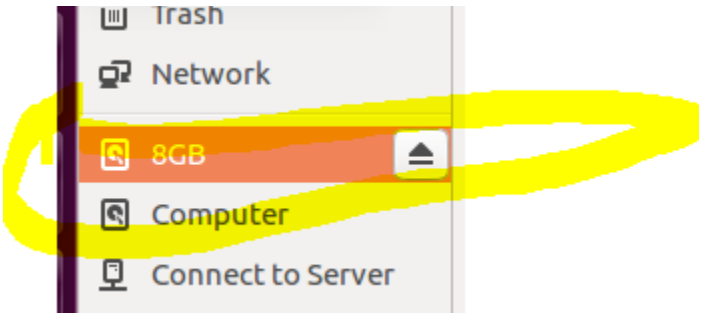
Another way to copy files is via USB:

- copy the files to your USB
- insert the USB to your local host where the VM is hosted
- make sure in your window environment you are NOT accessing any files/folders on the USB drive before you do the steps below.
-
- On the bottom of Ubutu host, select USB icon

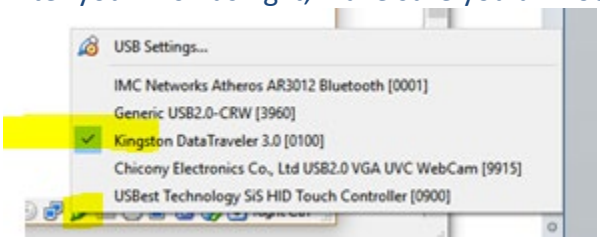
Right click the USB icon, and popup screen below will appear. Select the USB drive you want to attach to the VM Ubutu



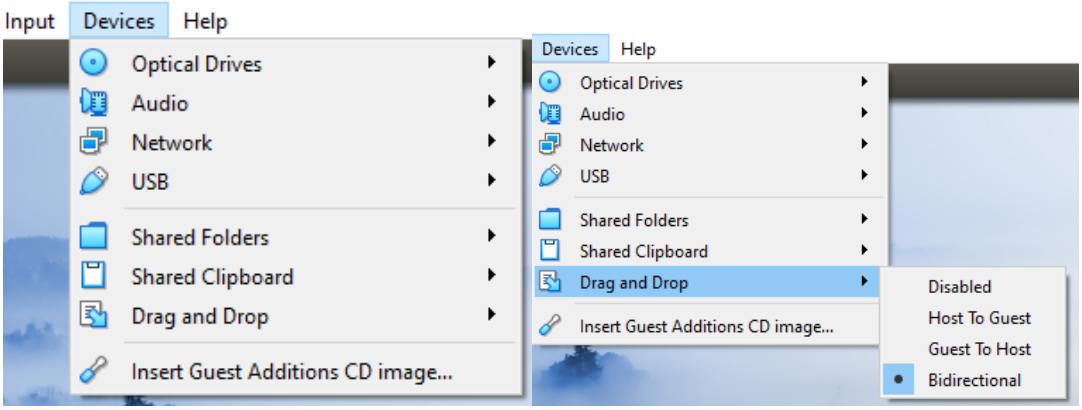
Once it is successfully attached, a USB icon will appear on the left side of the window.

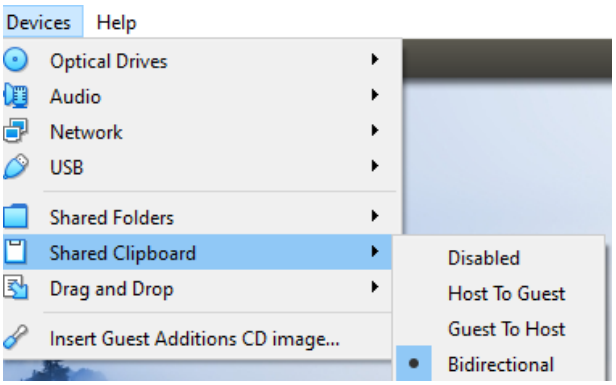


After you finish using it, make sure you unmount (unselect the USB)

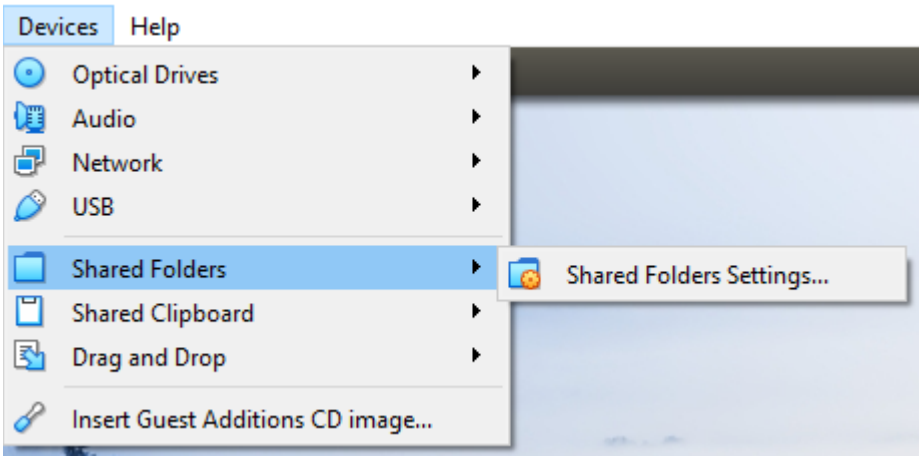


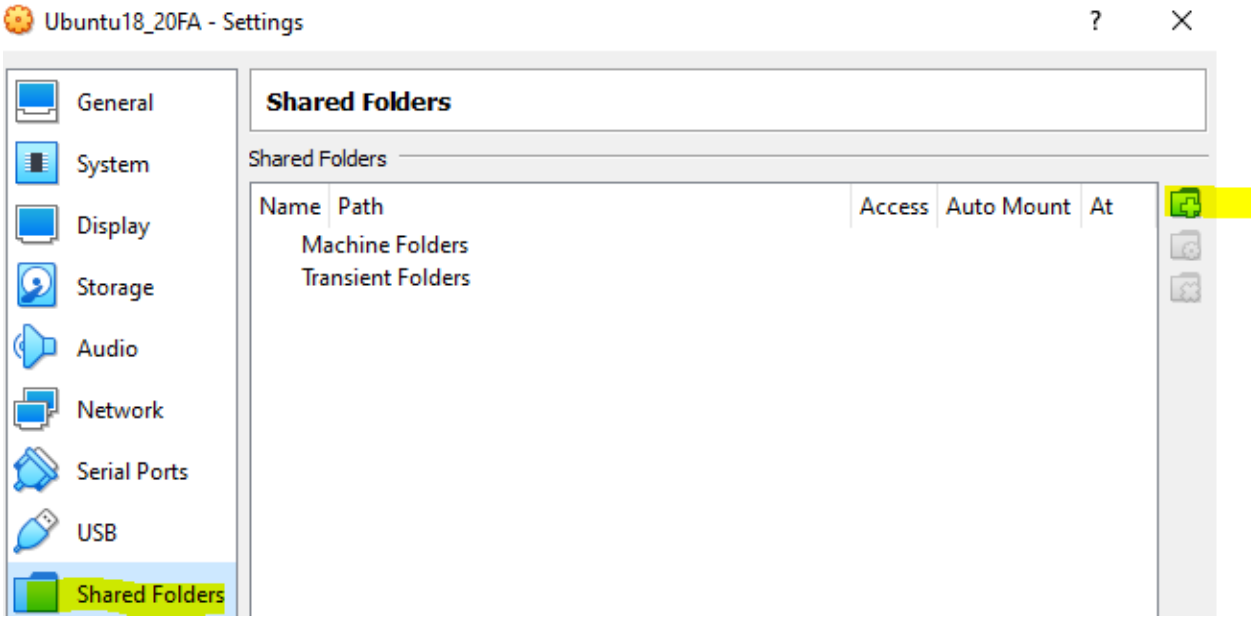
You can set sharing so you can copy-paste between the virtualization environment with the host computer by enable bi-directional sharing.

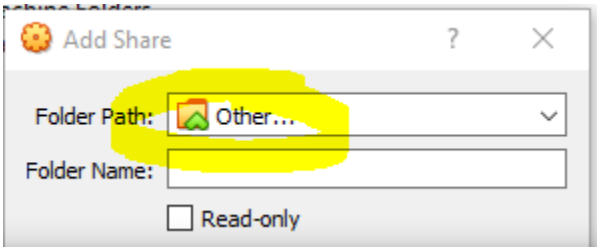




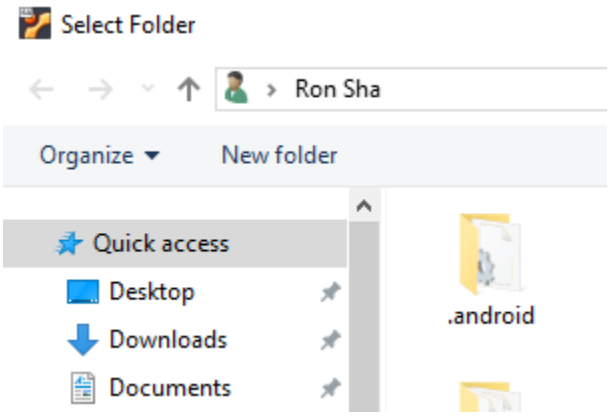
To share folders between the host computer the VM environment, use the share folder feature.



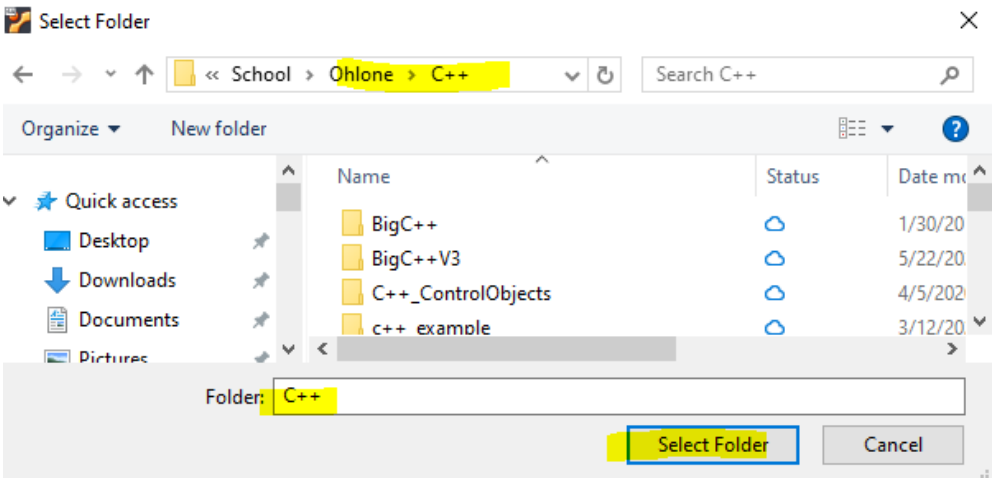


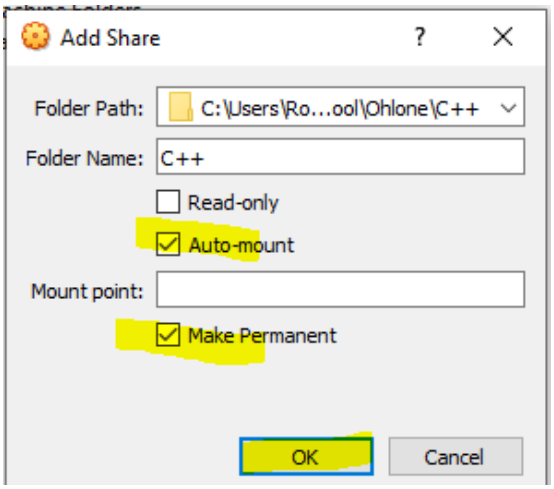


Then you local (host) folder window should appear:



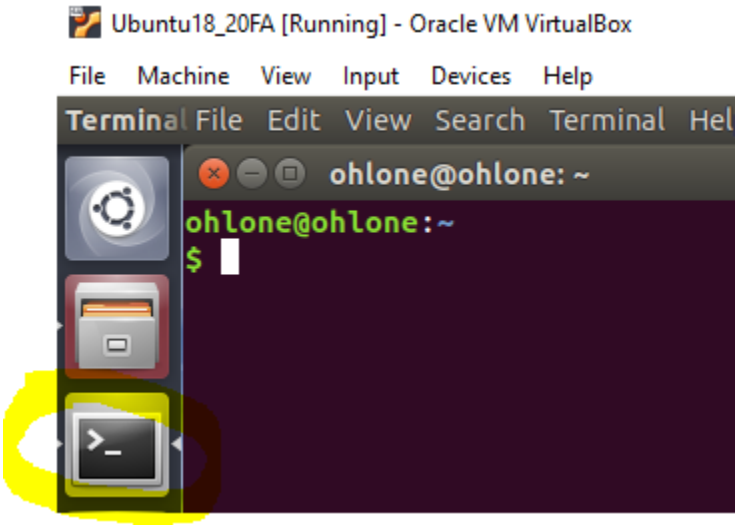
Then select the folder you want to share, for example, if I want to share my C++ folder with my VM environment, I select the folder, and then press the "Select Folder" button.





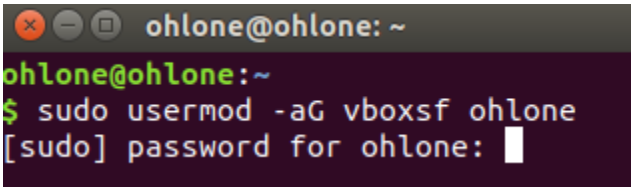
Select OK

Now add permission by using the command line terminal.

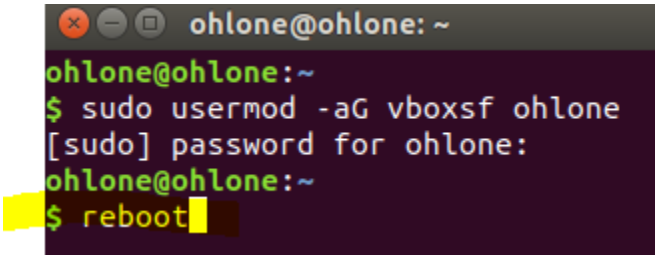


In the command line terminal, type the following command to allow permission for username *ohlone*. When prompted for password, use the password for username *ohlone* which if you haven't change it, ***ohlone*** is the default password.

sudo usermod -aG vboxsf ohlone



After this command, you need to restart the Ubuntu OS by using the command ***reboot***



Now, after you login to the Ubuntu OS from now on, your host folder will be automounted and accessible within the VM Ubuntu system. The automounted folder name should be something similar to “sf_xx”. Once you select the folder, now you can access the folders on your host computer.