

# Operating Systems, Virtual Machine and Cloud computing

Dumrong Mairiang, PhD

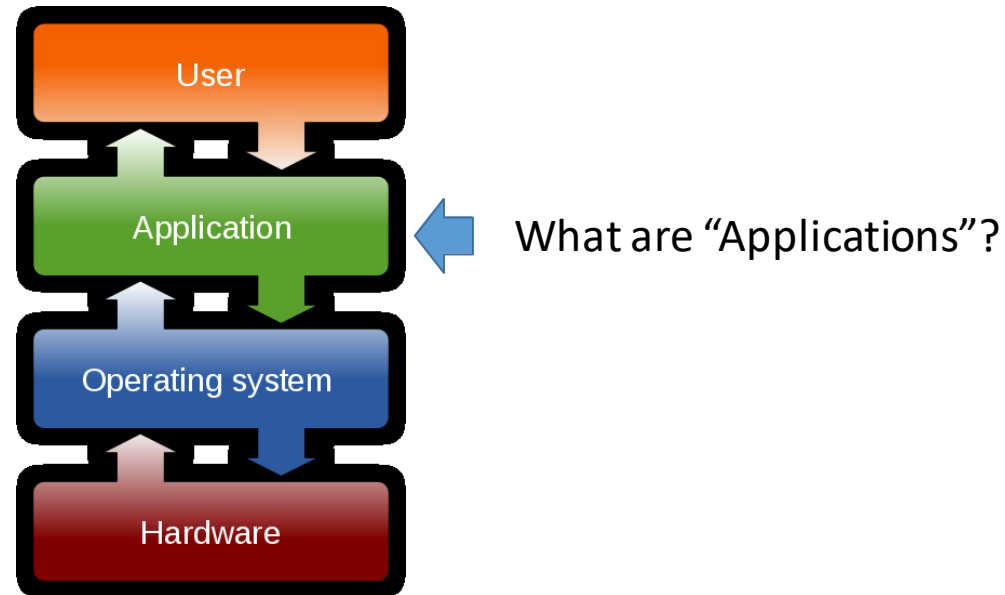
SIRE507: FUNDAMENTAL COMPUTER SCIENCE FOR BIOLOGIST

# Operating system (OS)

- What is it?
- Why is it important?

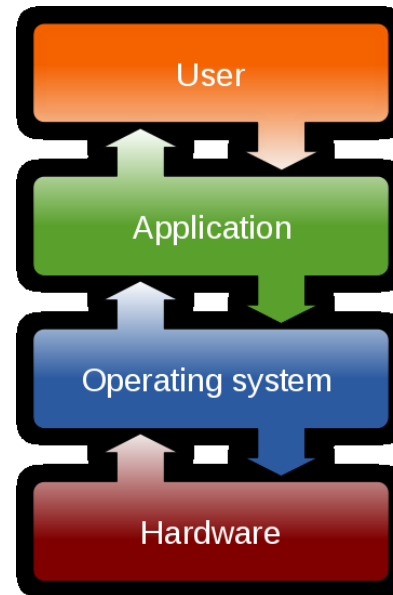
# Operating system (OS)

- What is it?
  - The basic software that manages a computer
- Why is it important?



# Operating system (OS)

- What is it?
  - The basic software that manages a computer
- Why is it important?



[Golftheman](#)



What are “Applications”?

Self-contained programs that perform a specific function



# Operating system (OS)

- Do you know any OS?

# Operating system (OS)

- Do you know any OS?



**macOS**

**ubuntu** 

The Ubuntu logo, which is a red circle containing a white stylized gear or flower-like shape.

# Operating system (OS)

- Do you know any OS?



Mobile OS

**iOS**



# Operating system (OS)

- Do you know any OS?

Microsoft Windows family



Other



MS-DOS



RISC OS

Unix/Unix-like family



Macintosh/Darwin



Linux





Why do you need to know about Oses  
other than MS Windows or MacOS?

# Why do you need to know about Oses other than MS Windows or MacOS?

- Many of bioinformatic applications were developed for Unix-like or Linux OS
- If you want to develop a web application, web servers usually run with Linux
- Many free and open-source applications must be run on Unix-like or Linux OS

# Operating system (OS)

- Old machine, legacy application and IoT:
  - 32-bit vs 64-bit (32-bit apps on Windows?)
  - Lite version/distribution of OS (e.g. Alpine, Lubuntu)
  - AMD/intel vs ARM/Apple silicon
- Users of your application
  - Windows
    - General users with probably no or little bioinformatic background
    - Graphical user interface is likely to be expected
  - Unix-like/Linux
    - Bioinformaticians
    - Command lines are acceptable

# Distinct features of each OS

- OS-specific file extensions
  - Microsoft Windows
    - File.exe
    - File.msi
  - Mac OS
    - File.dmg
  - Linux (Debian/Ubuntu)
    - File.deb

# Distinct features of each OS

- File naming rules
  - Microsoft Windows
    - Reserved characters: \, /, :, ?, \*, >, <, |, “
  - Linux and Mac OS
    - Reserved characters: /, >, <, |, &, (, ), “, ‘
    - Reserved characters but will be “automatically escaped”: \, \*, :, space
- As bioinformatician: Please AVOID using “space” in the file name
  - FileName.txt (Camel case)
  - File\_Name.txt (Underscore case)

# Distinct features of each OS

- File paths to YourFile.txt in your “home” directory
  - Microsoft Windows
    - C:\Users\YourName\YourFile.txt
  - Mac OS
    - /Users/YourName/YourFile.txt
  - Linux
    - /home/YourName/YourFile.txt
- Linux (root access)
  - /root/YourFile.txt

# Distinct features of each OS

- Application for accessing command line
  - Microsoft Windows
    - Command Prompt
  - Mac OS
    - Terminal
  - Linux
    - Terminal

# Distinct features of each OS

- Application for Back-up
  - Microsoft Windows
    - System Restore and Restore Point
  - Mac OS
    - Time Machine
  - Linux
    - Ubuntu backups
    - Backups application (e.g. cronopete)



# Distinct features of each OS

- Application for Hardware Management/List
  - Microsoft Windows
    - Device Manager
  - Mac OS
    - System Reports (“About this Mac”)
  - Linux (Ubuntu)
    - hardinfo, lspci, lsusb

# Distinct features of each OS

- Checking or interrupting programs
  - Microsoft Windows
    - Task Manager
  - Mac OS
    - top (basic) and htop (to be installed)
  - Linux (Ubuntu)
    - top (basic) and htop (to be installed)

# Distinct features of each OS

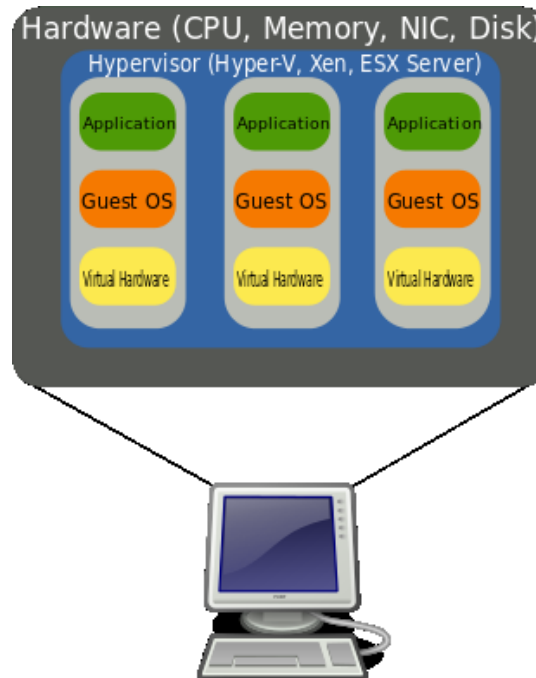
- Any other features?

# Virtual Machine (VM)

- What is it?

# Virtual Machine (VM)

- What is it?
  - A complete environment for a guest operating system to function as though that operating system were installed on its own computer



# Virtual Machine (VM)

- What VM is not:
  - Virtual machine  $\neq$  Emulator
  - Emulator converts commands to and from a host machine to an entirely different platform
  - Emulator: DosBOX (CPU), PuTTY (Terminal), ZSNES (Gaming), PCSX2 (Gaming), N64 Emulator (Gaming), Android Virtual Device (Application Development)

Why do you need to know about VMs?

# Why do you need to know about VMs?

- SENARIO 1 (Flexibility): The OS of your machine is not compatible with the application you need to run/test:
  - No spare machine for installing a new OS
  - No space or resources to create a dual boot



# Why do you need to know about VMs?

- SENARIO 2 (Simulation): You want to simulate network connection to your web application in your own machine
  - No spare machine
  - Do not want to deploy in web server yet

# Why do you need to know about VMs?

- SENARIO 3 (Security): Controlled/Quarantined environment for developing, testing or running applications
  - Potentially harmful applications
  - Secured VM

# Why do you need to know about VMs?

- Any other scenario?

# Let's create the first VM

- Applications for virtualization:
  - VM ware (Commercial)
  - **Oracle VM VirtualBox (Free and open-source)**
  - Vagrant (Free and open-source, but no GUI)
  - Other...

# Let's create the first VM

- Check your machine
  - CPU: at least 2 cores
  - RAM: Host OS minimal requirement + Guest OS minimal requirement
  - Hard disk: Guest OS minimal requirement or External HDD or USB Flash drive

# Let's create the first VM



The screenshot shows the VirtualBox website's Downloads page. The browser's address bar shows the URL <https://www.virtualbox.org/wiki/Downloads>. The page features the VirtualBox logo and a navigation menu on the left with links to About, Screenshots, Downloads, Documentation, End-user docs, Technical docs, Contribute, and Community. The main content area is titled 'Download VirtualBox' and includes a search bar, Login, Preferences, Start Page, Index, and History links. The text explains that the page contains links to VirtualBox binaries and source code, and that downloading implies agreement to the license. It mentions the latest version is 6.1, with support ending in 2023. A section titled 'VirtualBox 7.0.8 platform packages' lists links for Windows hosts, macOS / Intel hosts (highlighted with a blue arrow), Developer preview for macOS / Arm64 (M1/M2) hosts, Linux distributions, Solaris hosts, and Solaris 11 IPS hosts. It also notes that binaries are released under GPL version 3 and provides a changelog link. A note advises upgrading guest additions after upgrading VirtualBox. The 'VirtualBox 7.0.8 Oracle VM VirtualBox Extension Pack' section lists a link for all supported platforms and mentions support for RDP, disk encryption, NVMe, and PXE boot for Intel cards, with a reference to the User Manual and the VirtualBox Personal Use and Evaluation License (PUEL).

**Download VirtualBox**

Here you will find links to VirtualBox binaries and its source code.

**VirtualBox binaries**

By downloading, you agree to the terms and conditions of the respective license.

If you're looking for the latest VirtualBox 6.1 packages, see [VirtualBox 6.1 builds](#). Version 6.1 will remain supported until December 2023.

**VirtualBox 7.0.8 platform packages**

- ⇒ [Windows hosts](#)
- ⇒ [macOS / Intel hosts](#)
- ⇒ [Developer preview for macOS / Arm64 \(M1/M2\) hosts](#)
- ⇒ [Linux distributions](#)
- ⇒ [Solaris hosts](#)
- ⇒ [Solaris 11 IPS hosts](#)

The binaries are released under the terms of the GPL version 3.

See the [changelog](#) for what has changed.

You might want to compare the checksums to verify the integrity of downloaded packages. *The SHA256 checksums should be favored as the MD5 algorithm must be treated as insecure!*

- [SHA256 checksums, MD5 checksums](#)

**Note:** After upgrading VirtualBox it is recommended to upgrade the guest additions as well.

**VirtualBox 7.0.8 Oracle VM VirtualBox Extension Pack**

- ⇒ [All supported platforms](#)

Support VirtualBox RDP, disk encryption, NVMe and PXE boot for Intel cards. See [this chapter from the User Manual](#) for an introduction to this Extension Pack. The Extension Pack binaries are released under the [VirtualBox Personal Use and Evaluation License \(PUEL\)](#). Please install the same version extension

Apple M1/M2 try UTM app and  
Ubuntu for ARM

# Let's create the first VM



Downloads - Oracle VM VirtualBox x +

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

 **VirtualBox**

search...

[Login](#) [Preferences](#)

[Start Page](#) [Index](#) [History](#)

## Download VirtualBox

Here you will find links to VirtualBox binaries and its source code.

### VirtualBox binaries

By downloading, you agree to the terms and conditions of the respective license.

If you're looking for the latest VirtualBox 6.1 packages, see [VirtualBox 6.1 builds](#). Version 6.1 will remain supported until December 2023.

### VirtualBox 7.0.8 platform packages

- ⇒ [Windows hosts](#)
- ⇒ [macOS / Intel hosts](#)
- ⇒ [Developer preview for macOS / Arm64 \(M1/M2\) hosts](#)
- ⇒ [Linux distributions](#)
- ⇒ [Solaris hosts](#)
- ⇒ [Solaris 11 IPS hosts](#)

The binaries are released under the terms of the GPL version 3.

See the [changelog](#) for what has changed.

You might want to compare the checksums to verify the integrity of downloaded packages. *The SHA256 checksums should be favored as the MD5 algorithm must be treated as insecure!*

- [SHA256 checksums, MD5 checksums](#)

**Note:** After upgrading VirtualBox it is recommended to upgrade the guest additions as well.

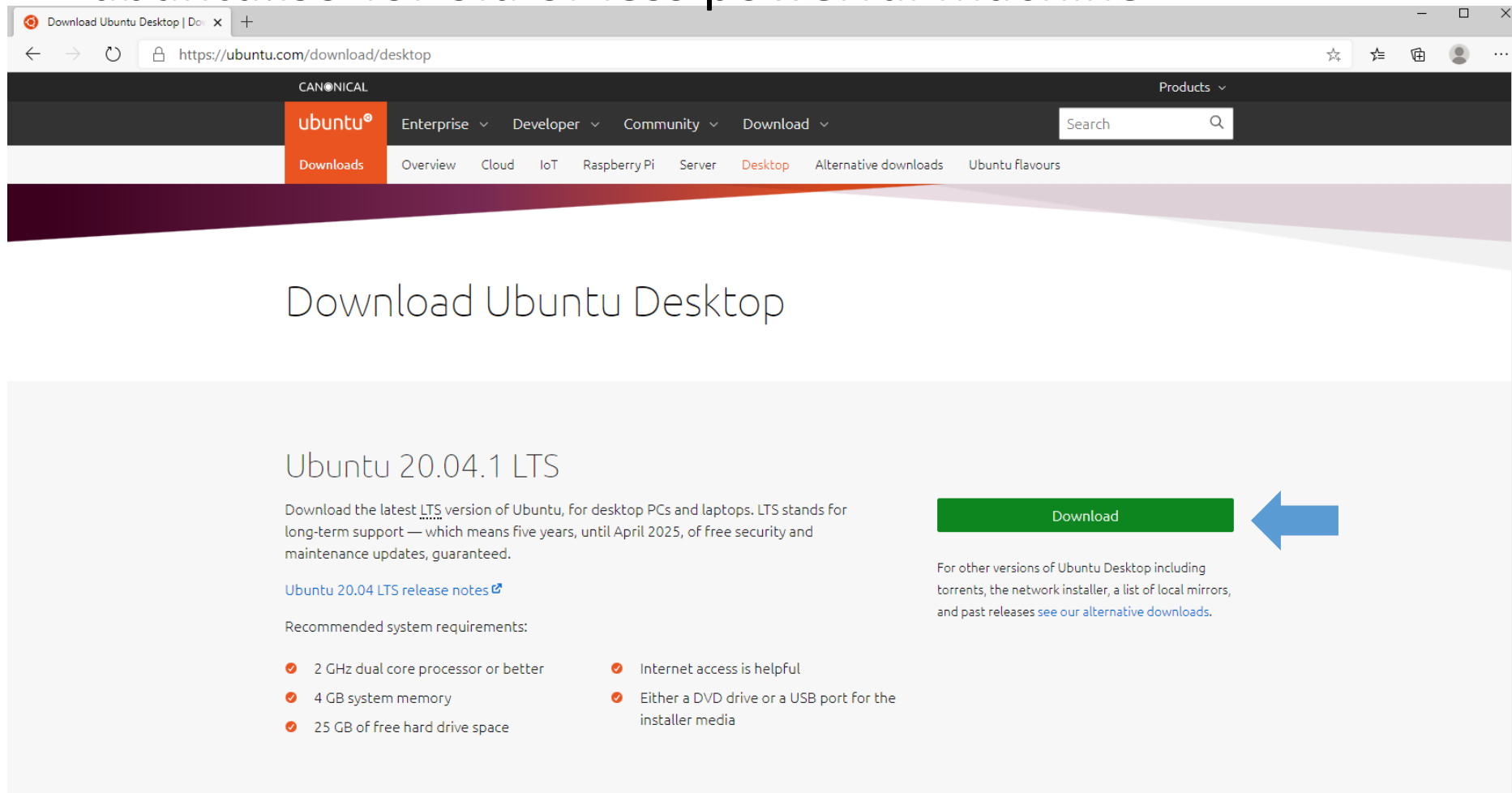
### VirtualBox 7.0.8 Oracle VM VirtualBox Extension Pack

- ⇒ [All supported platforms](#)

Support VirtualBox RDP, disk encryption, NVMe and PXE boot for Intel cards. See [this chapter from the User Manual](#) for an introduction to this Extension Pack. The Extension Pack binaries are released under the [VirtualBox Personal Use and Evaluation License \(PUEL\)](#). Please install the same version extension

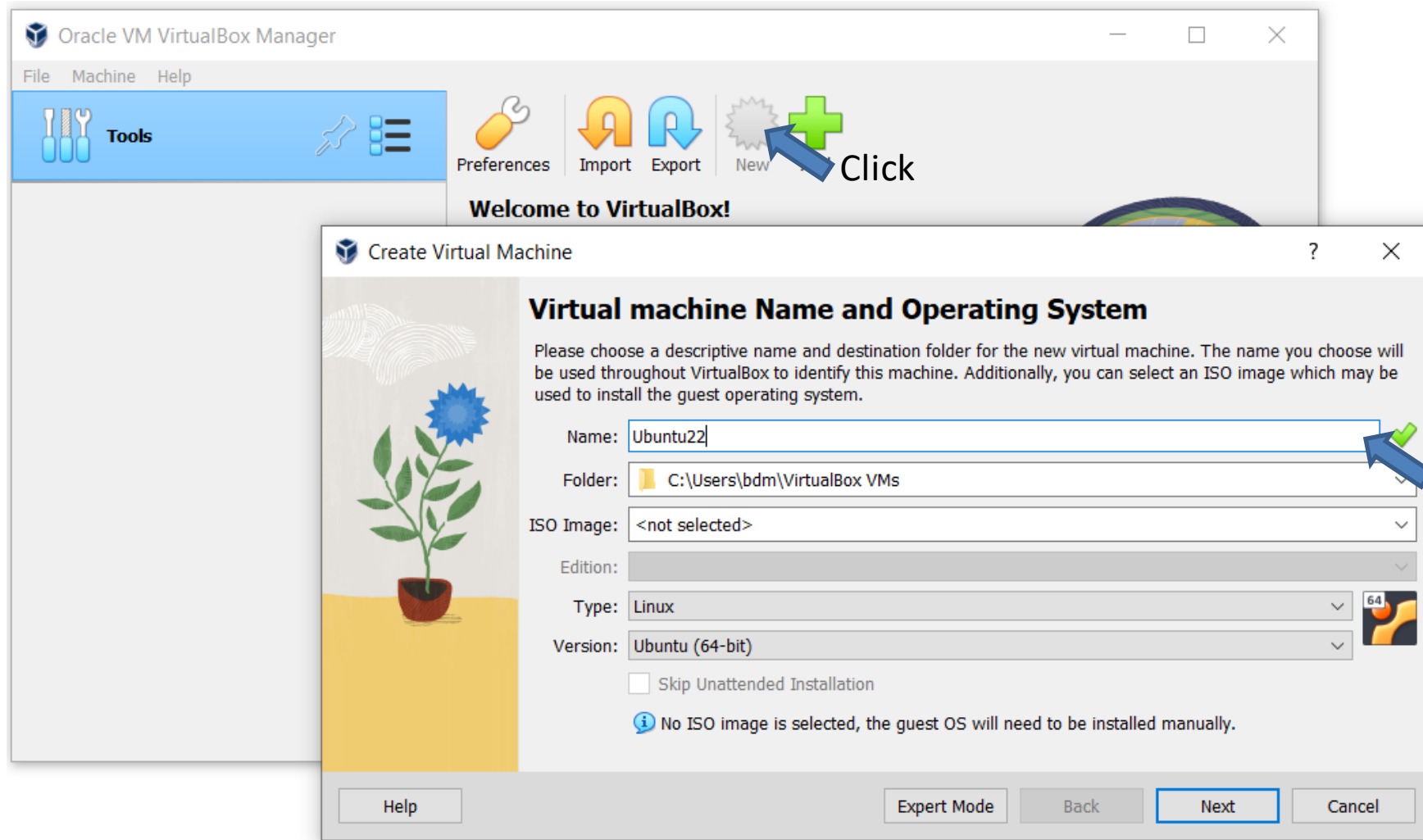
# Let's create the first VM

- Ubuntu.iso will be used for the demonstration
- Lubuntu.iso for old or less powerful machine



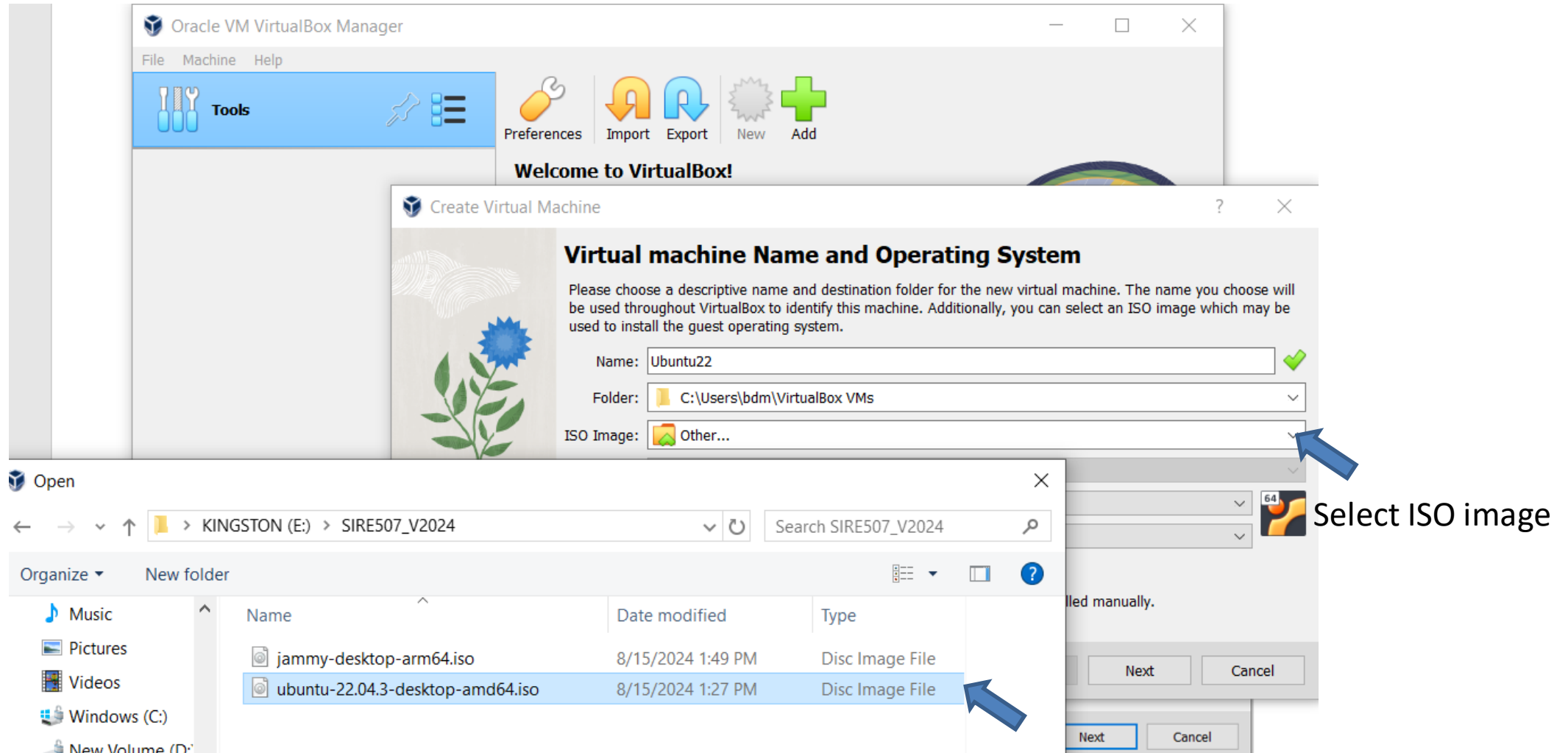


# Let's create the first VM

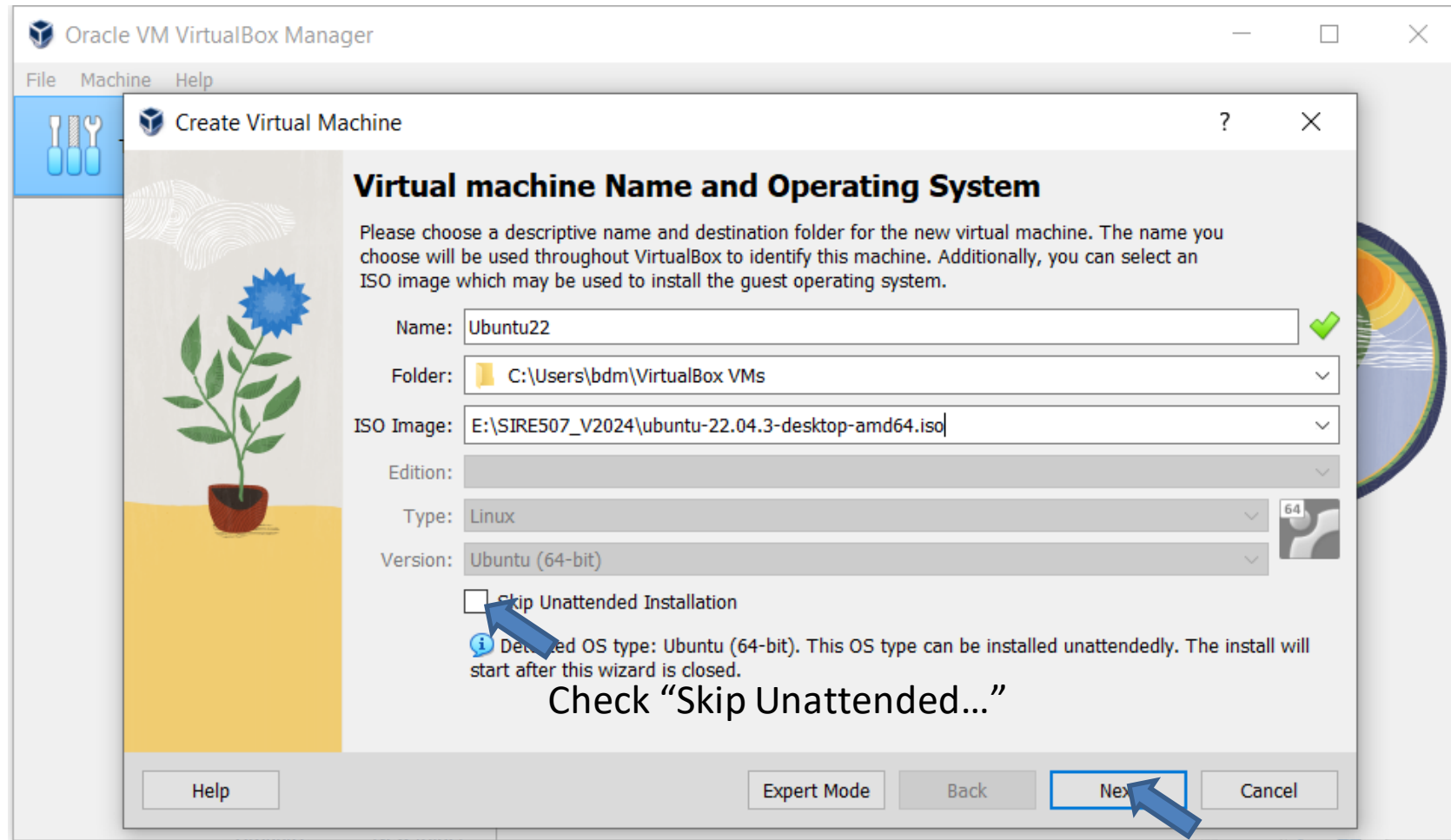


Name your VM

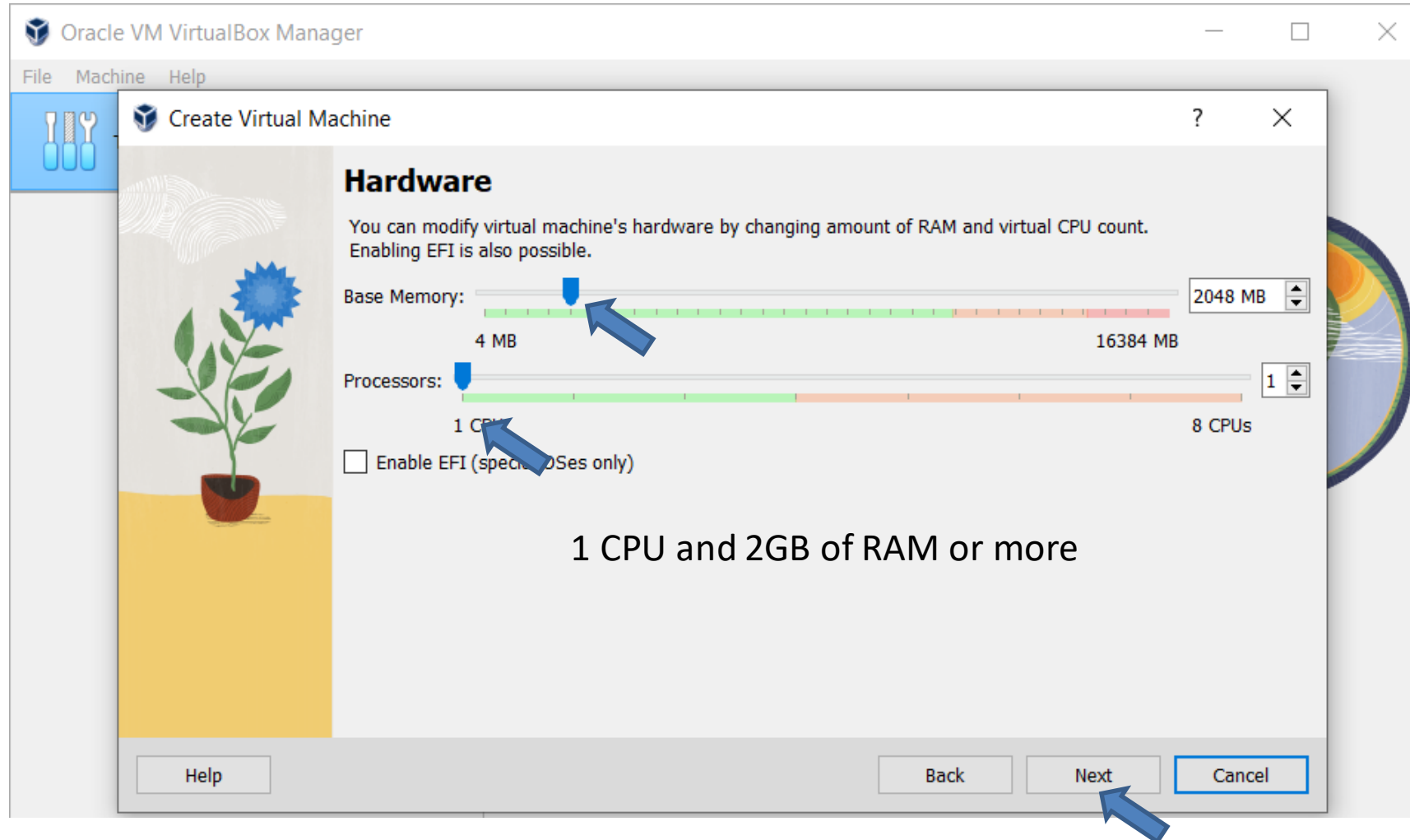
# Let's create the first VM



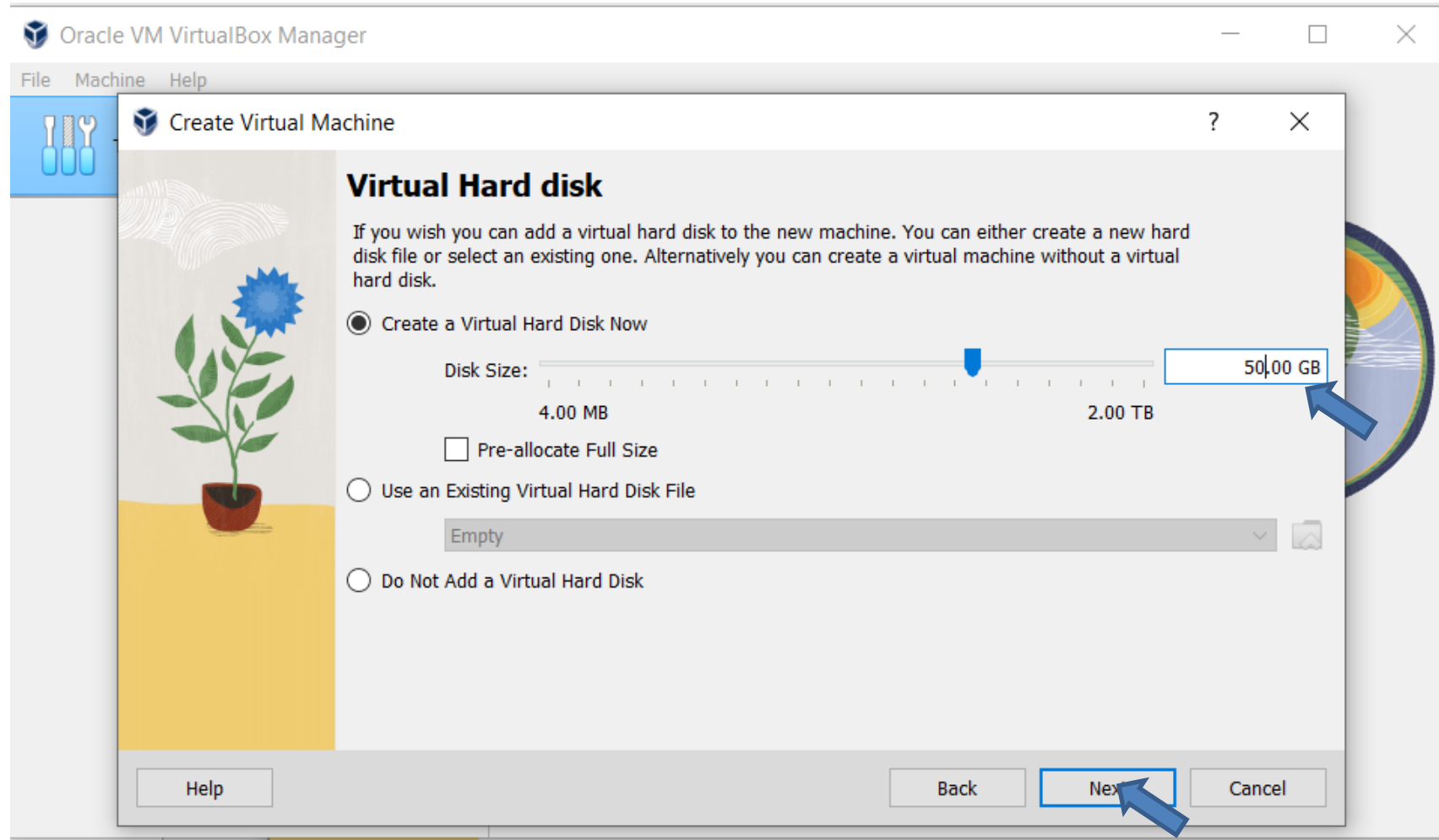
# Let's create the first VM



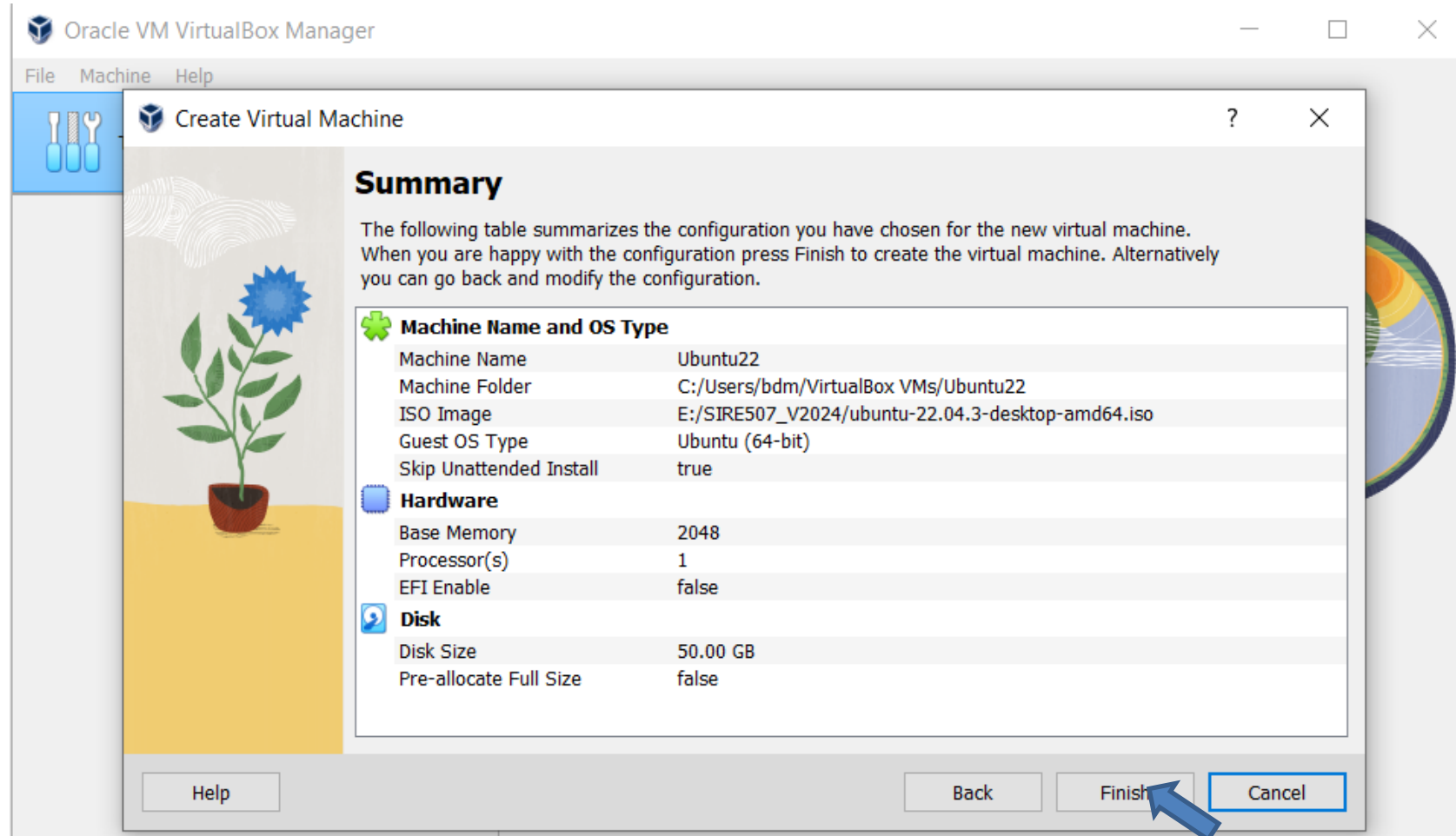
# Let's create the first VM



# Let's create the first VM

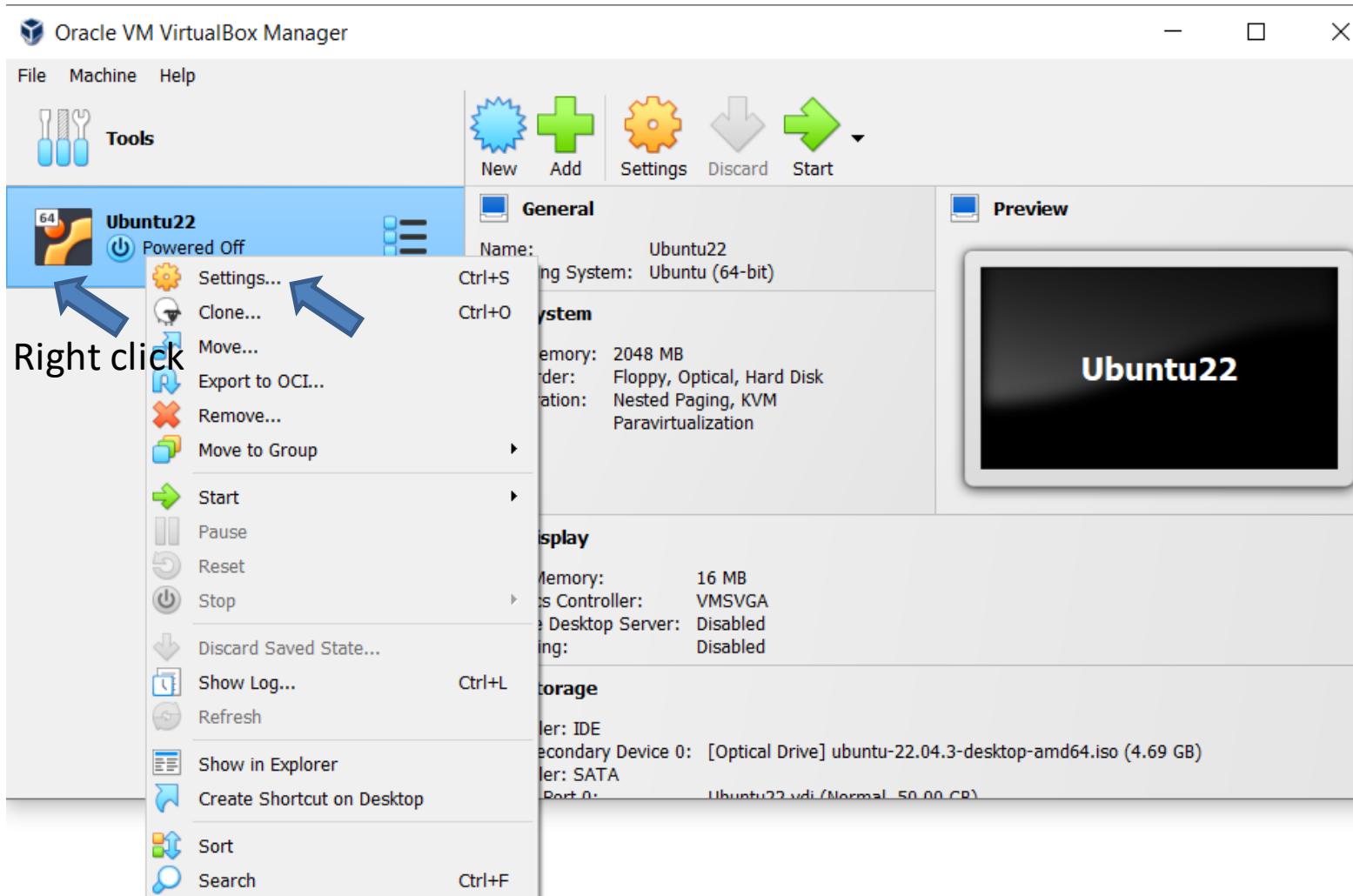


# Let's create the first VM

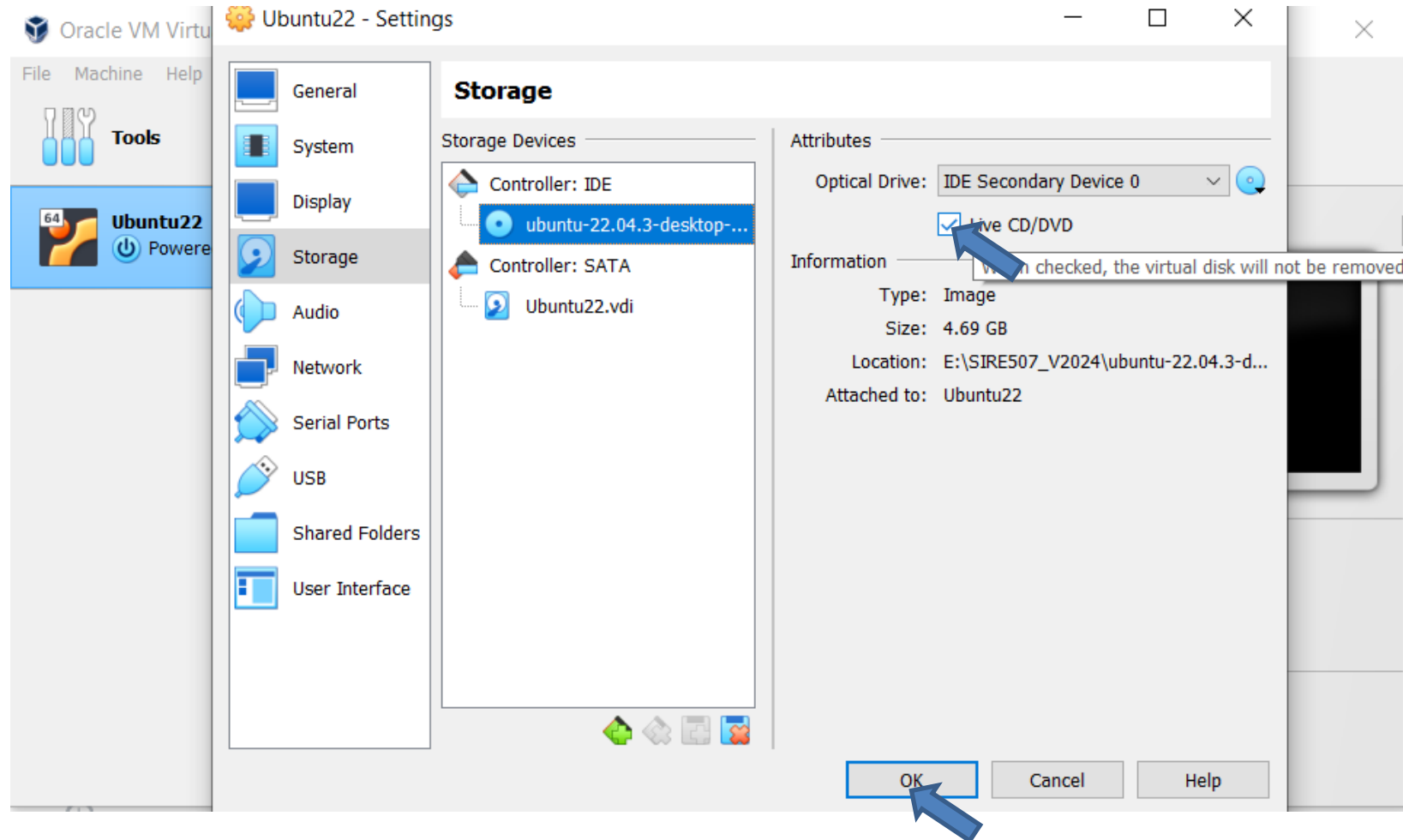


Review and click

# Let's create the first VM

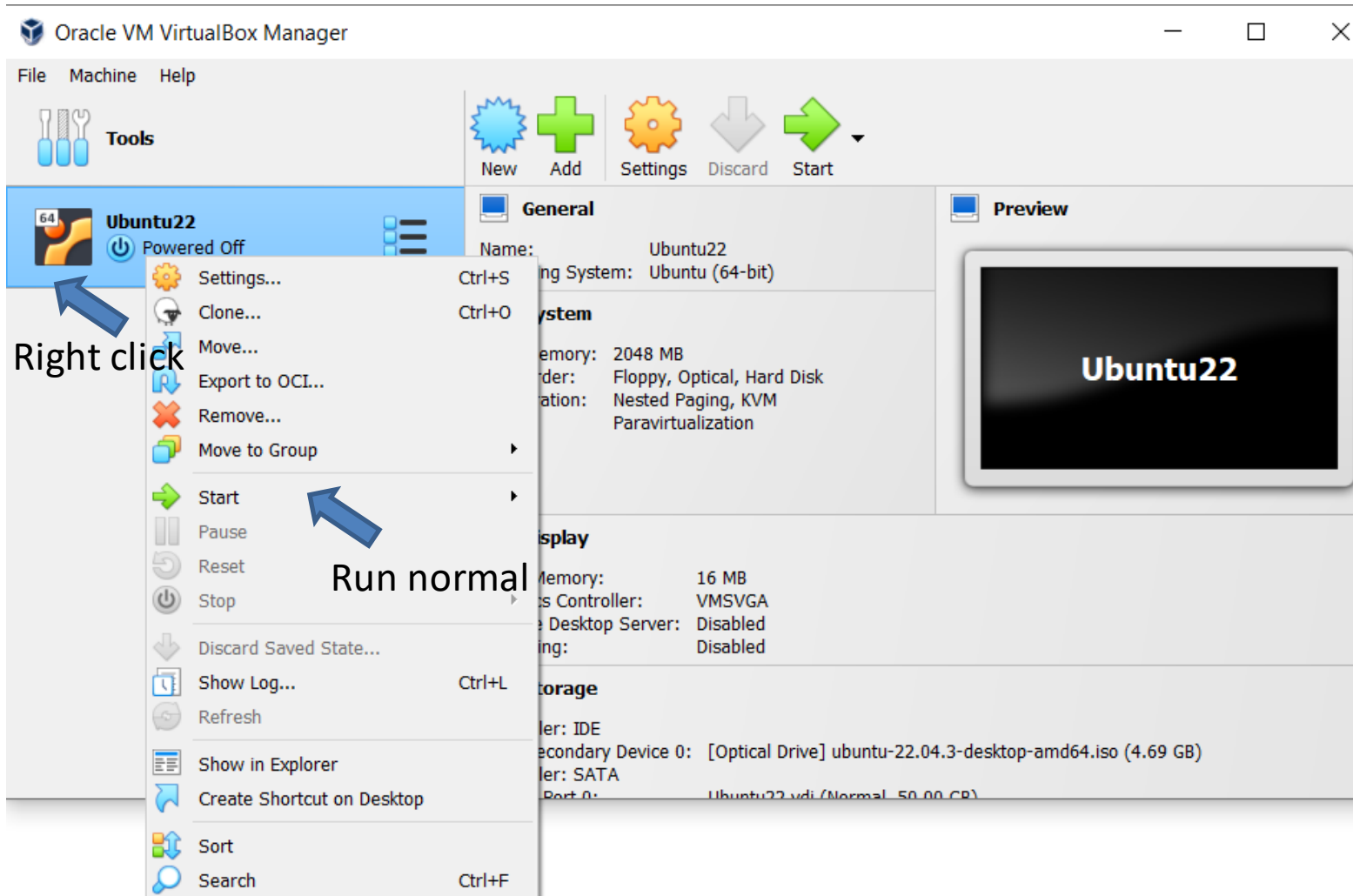


# Let's create the first VM

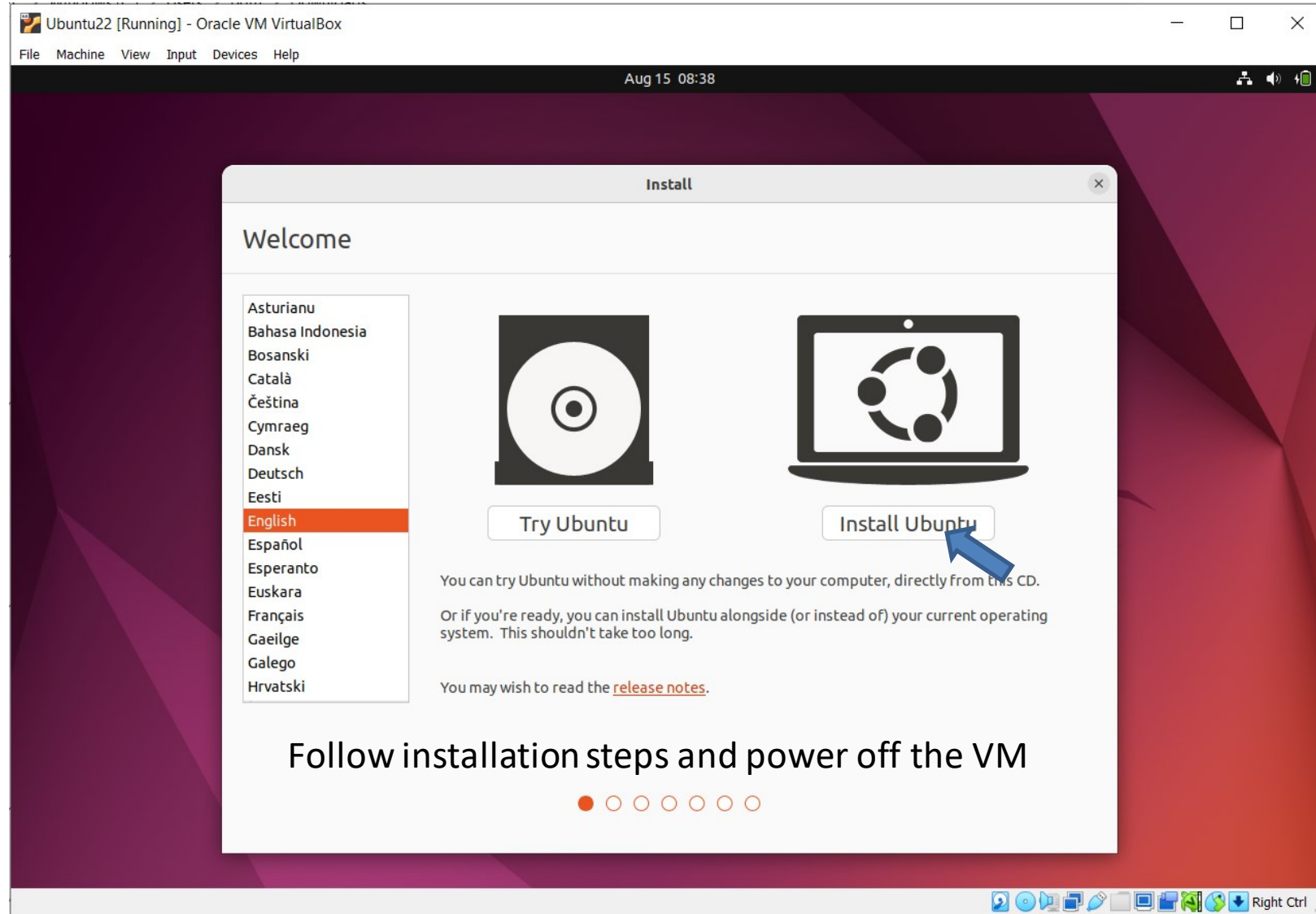




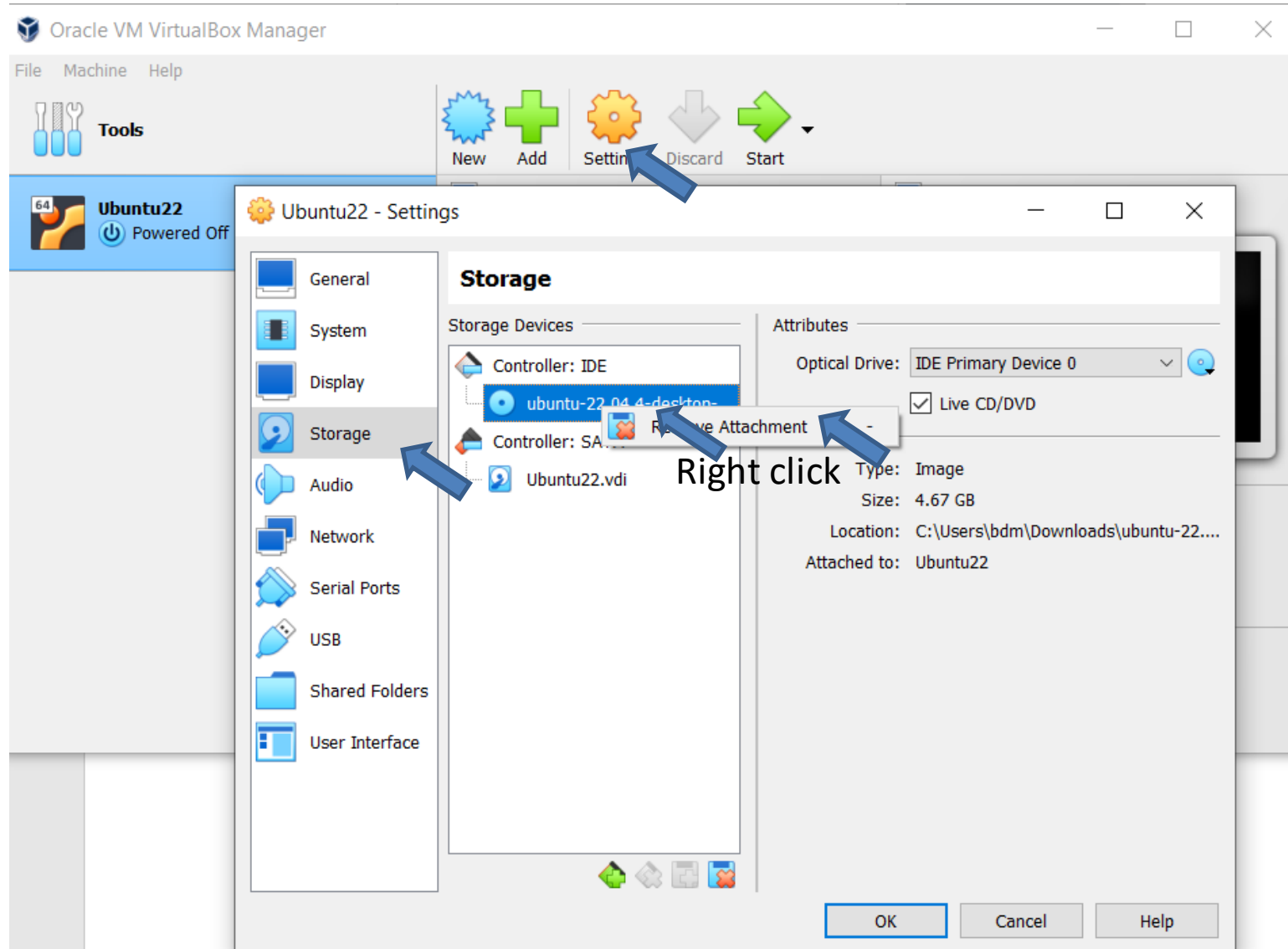
# Let's create the first VM



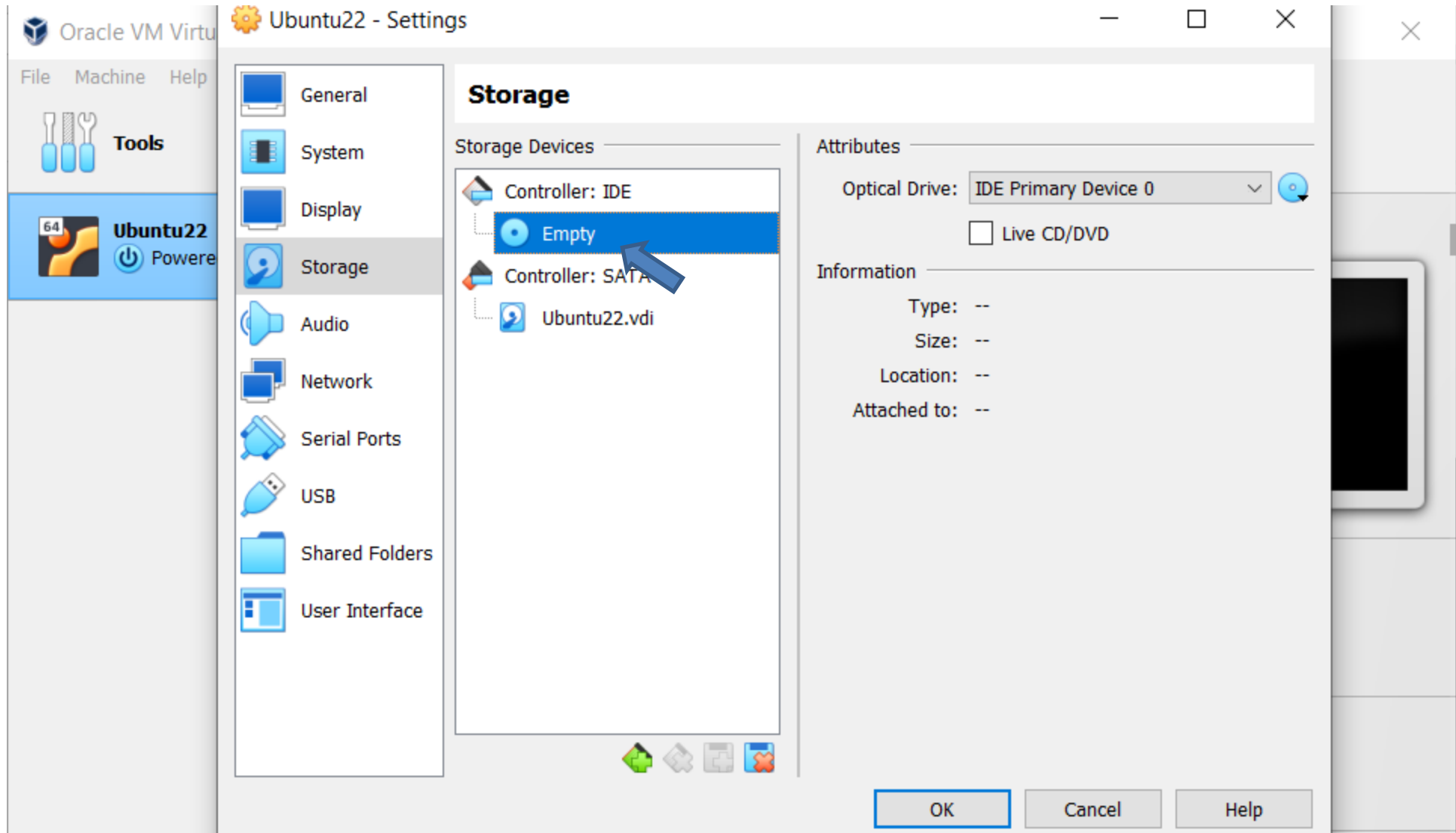
# Let's create the first VM



# Let's create the first VM



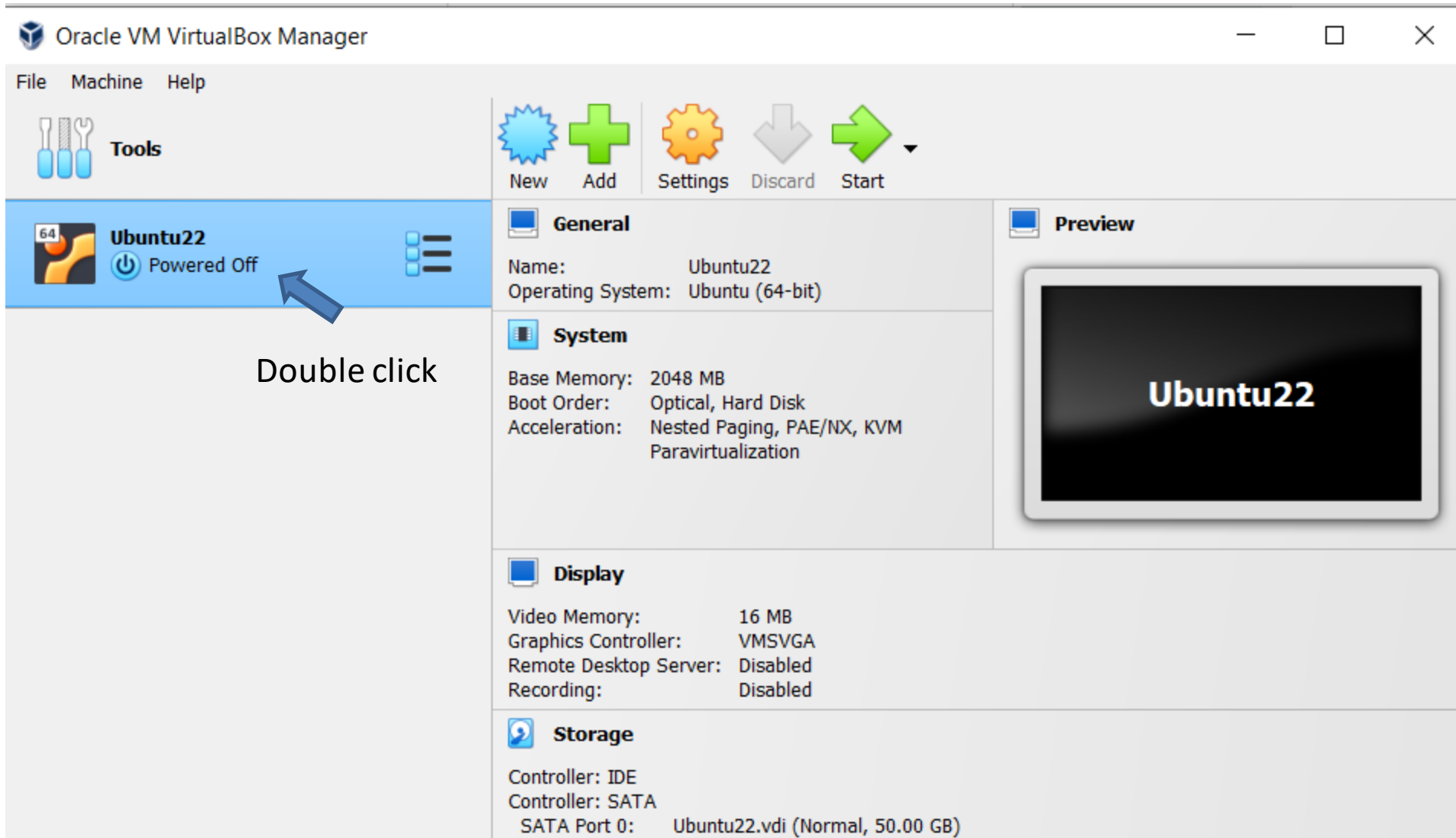
# Let's create the first VM



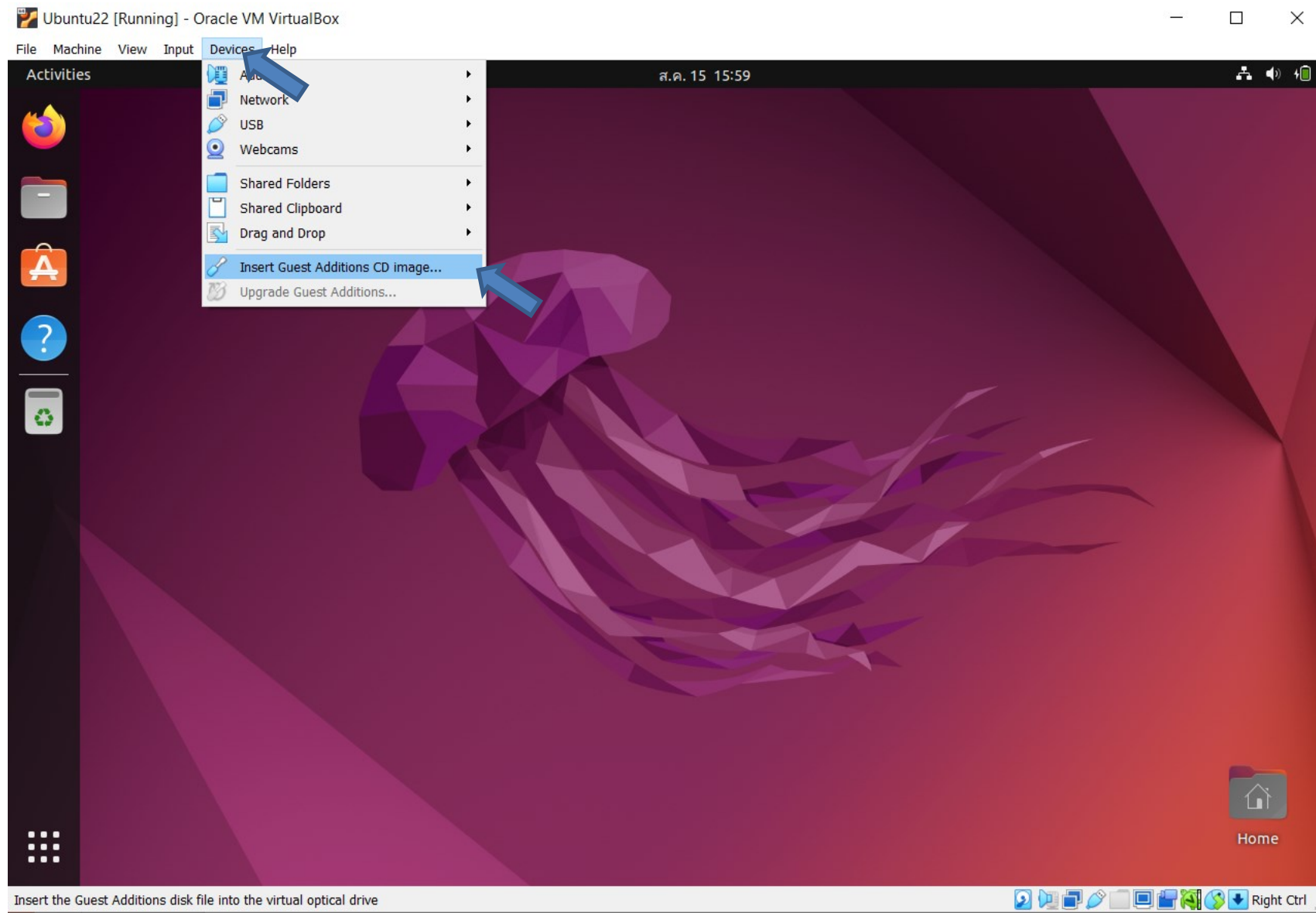
# Let's create the first VM

1. Run guest additions
2. Set a shared folder

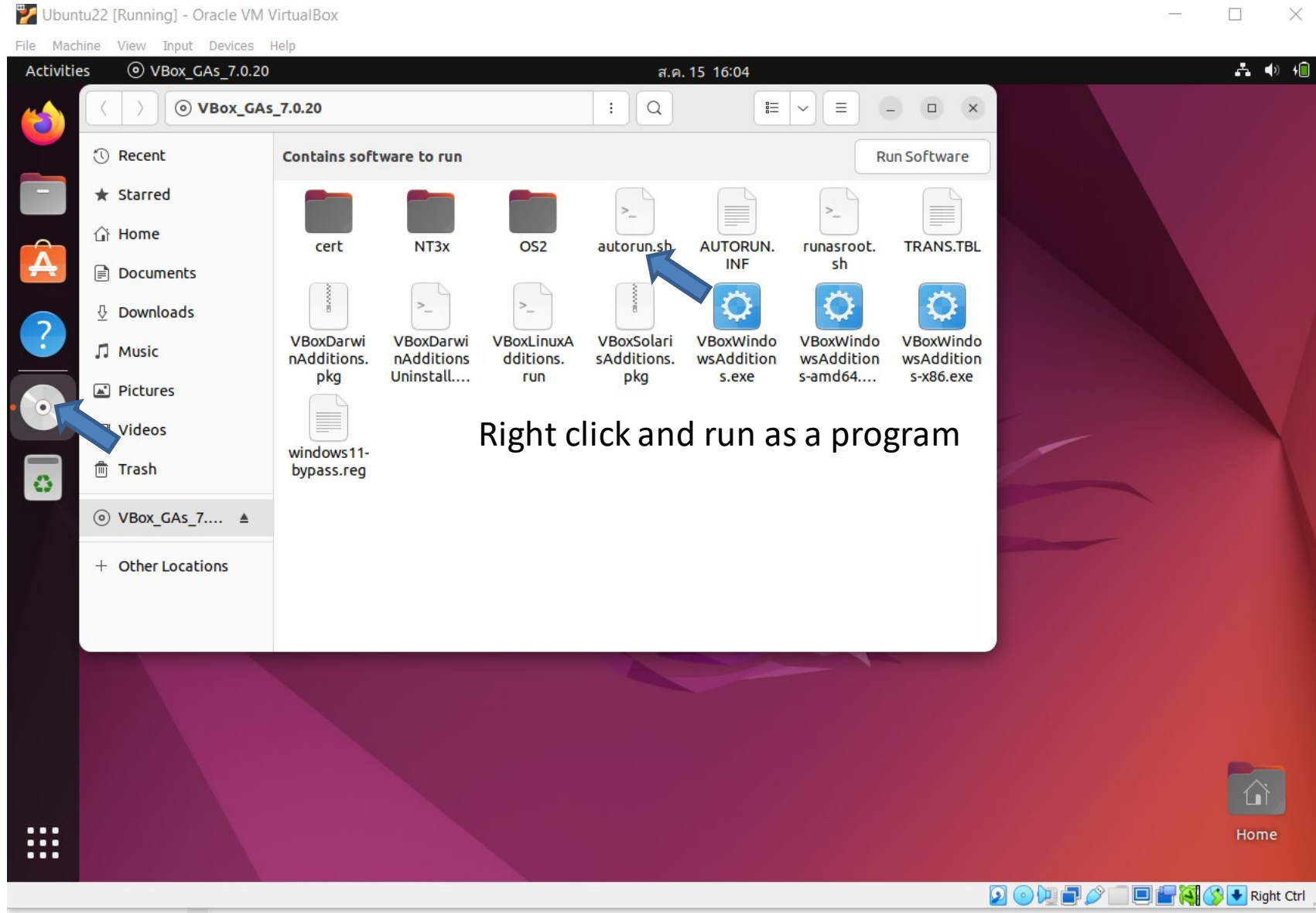
# Guest additions



# Guest additions

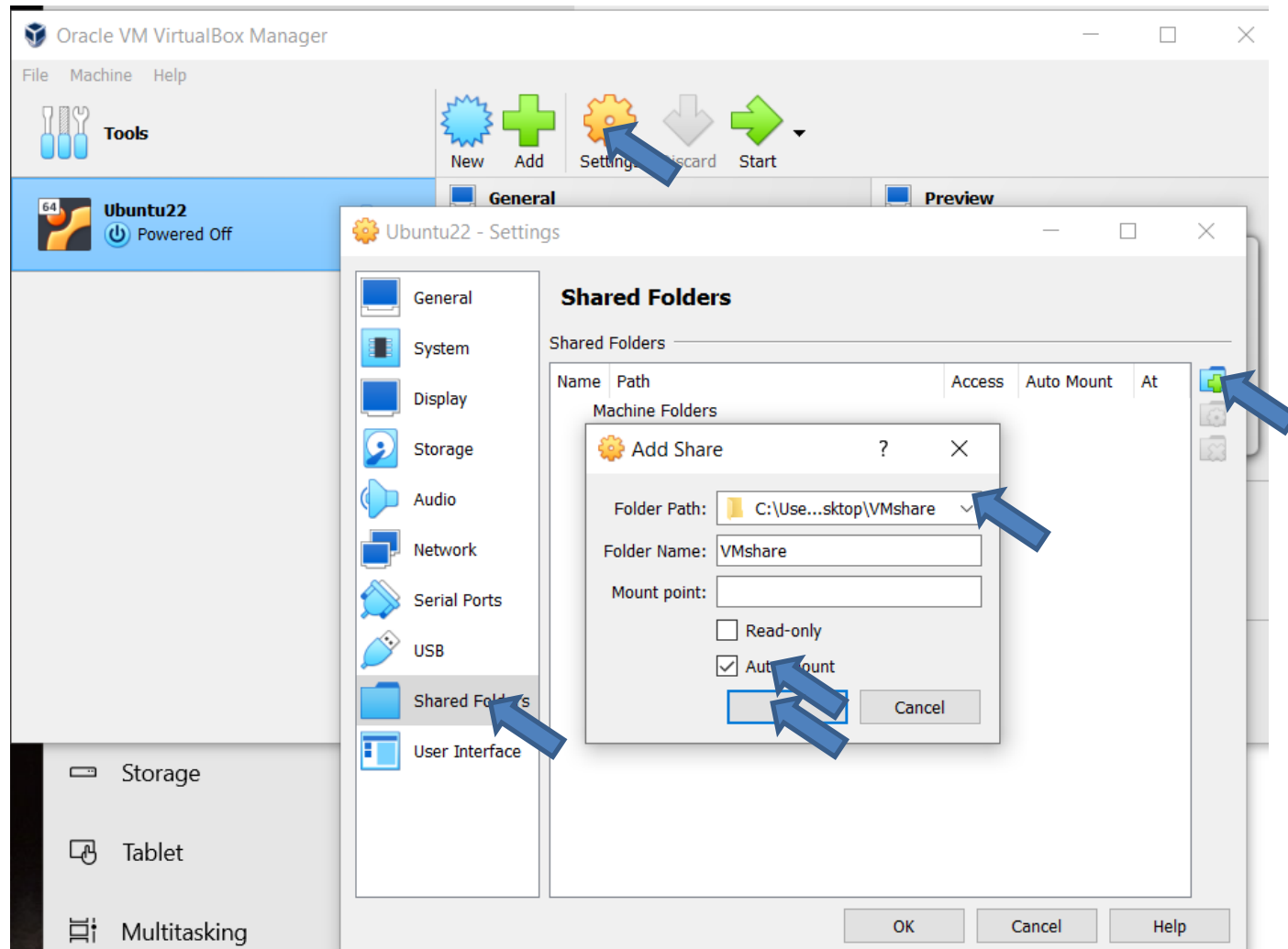


# Guest additions





# Set a shared folder



# Set a shared folder

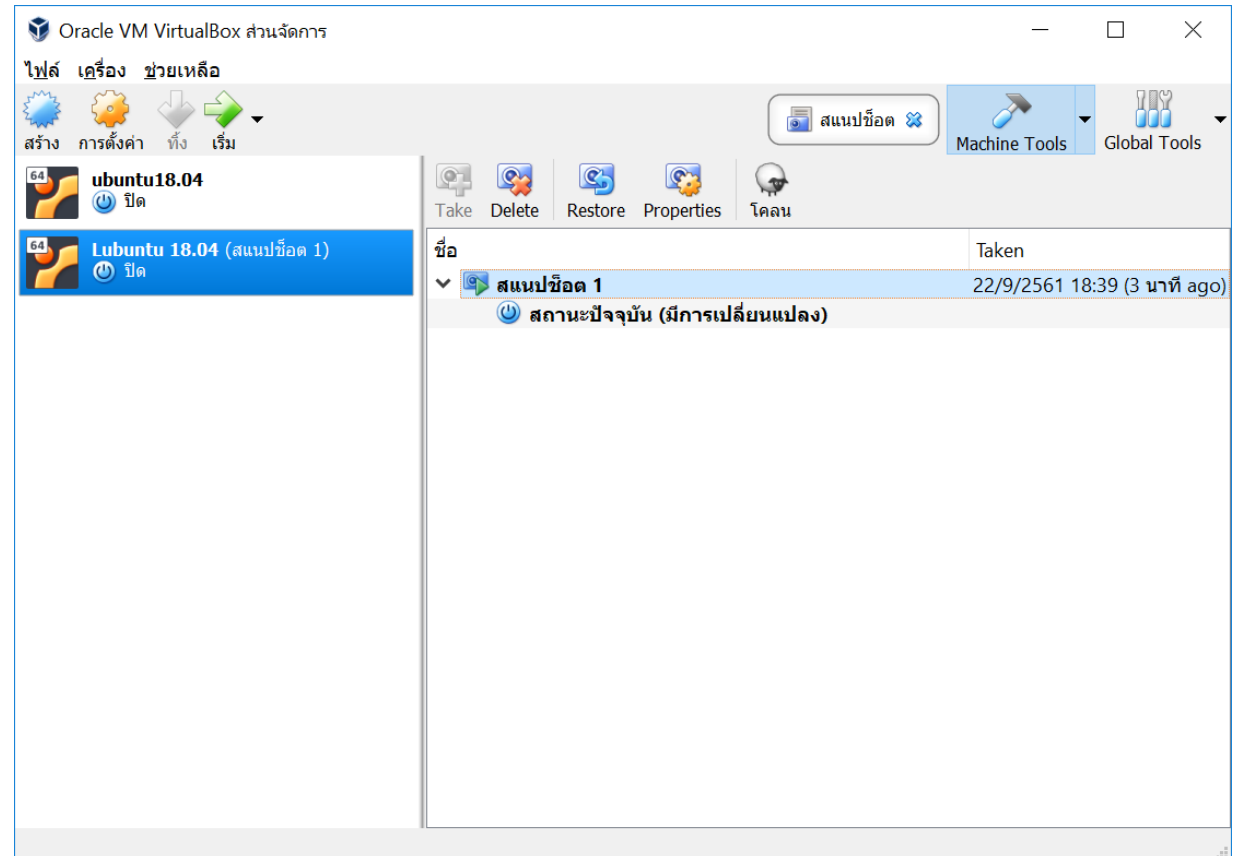
- Switch on the VM
- Open the Terminal

```
sudo usermod -aG vboxsf YourName
```

- Shutdown and restart

# Backups by snapshot

- Backup = A copy of files from a computer's hard disk, usually made on some external medium such as CD-ROM or flash drive



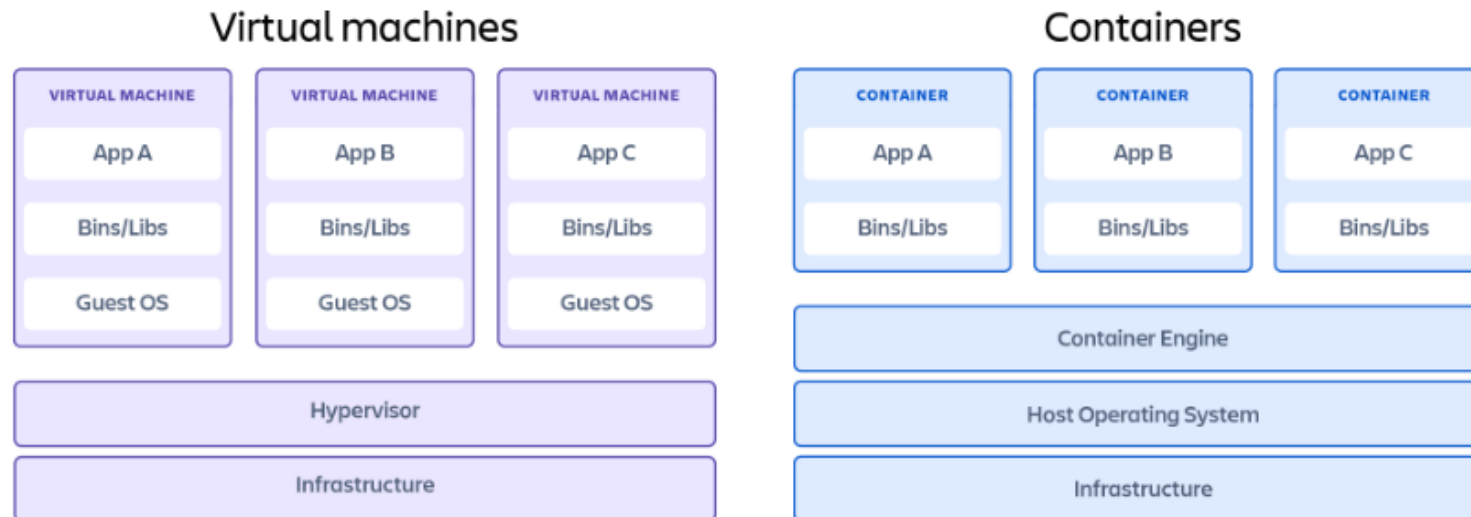
# Portable VM

- You can save file.vdi to external HDD or USB flash drive
- Connect to any host machine with Virtualbox to quickly create a copy of VM in file.vdi
- **NOTE:**
  - USB flash drive must be formatted as NTFS
  - FAT32 format (default format in some USB flash drives) cannot accommodate a single file larger than 4GB

# VM versus Container



- VM requires resources (e.g. RAM, CPU, HDD space) allocation while a container shares resources with the host.
- VM virtualizes an entire machine while a container virtualizes an “environment” for specific software.



# VM versus Container



- Container use cases:
  - Run old software in new machine/OS
    - Software needs to run on 32-bit OS with Python 2
  - Deployment with reproducibility
    - Run your software developed on your machine on HPC
  - Microservices
    1. Container for database
    2. Container for web application
    3. Container for backup

# VM versus Container

- **Docker** (<https://docs.docker.com/>)
  - Most popular container (i.e. good documentation)
  - Large repository of container images (DockerHub)
- **Singularity** (<https://docs.sylabs.io/guides/latest/user-guide/>)
  - “Portable” container (Singularity Image Format, SIF)
  - Other containers may be converted to Singularity container (especially Docker)
  - Run on HPC with job scheduler (e.g. Slurm)



# Cloud computing

- What is it?



# Cloud computing

- What is it?
- Cloud computing, often referred to as simply “the cloud,” is the delivery of on-demand computing resources — everything from applications to data centers — over the internet on a pay-for-use basis.
  - Elastic resources — Scale up or down quickly and easily to meet demand
  - Metered service so you only pay for what you use
  - Self service — All the IT resources you need with self-service access



IBM **Cloud**






<https://www.ibm.com/cloud/learn/what-is-cloud-computing>

Why do you need to know Cloud Computing?

# Why do you need to know Cloud Computing?

- Web application deployment
- Requiring intensive but transient computing power
- Backup and archive
- Sharing and hosting (Public cloud)
- Anything else?

# Cloud service providers

- Amazon web services The AWS logo consists of the lowercase letters "aws" in a dark blue, sans-serif font, with an orange curved arrow underneath that points from the 'a' to the 's'.
- Google cloud platform The Google Cloud Platform logo features a hexagonal icon composed of six smaller hexagons in blue, red, yellow, and grey, followed by the text "Google Cloud Platform" in a bold, black, sans-serif font.
- Microsoft Azure The Azure logo features a stylized blue triangle icon to the left of the word "Azure" in a blue, sans-serif font.
- IBM cloud The IBM Cloud logo features a blue cloud icon with three small vertical lines above it, followed by the text "IBM Cloud" in a bold, black, sans-serif font.
- Digital Ocean The DigitalOcean logo features a blue circular icon with a white square and a plus sign inside, followed by the text "DigitalOcean" in a bold, blue, sans-serif font.

# Hands on: Cloud Computing

# Overview

- Demo for creating a cloud VM
- Practice: remotely access the VM
- Practice: remotely transfer files from/to the VM
- Back-up and clean-up

# Creating a VM

- Platform will be based on “your advisor”
  - Digital ocean (Simple Unix/Linux applications, web application)
  - Google Cloud Platform/AWS (Machine learning, NextFlow)
  - IPGG server/MU cluster/Thai-SC (Intensive calculation with CPUs and RAM)

# Creating a VM

- Calculate/Estimate your resource wisely
  - CPU/RAM/HDD
  - OS
  - Location of your data center
  - Safety features

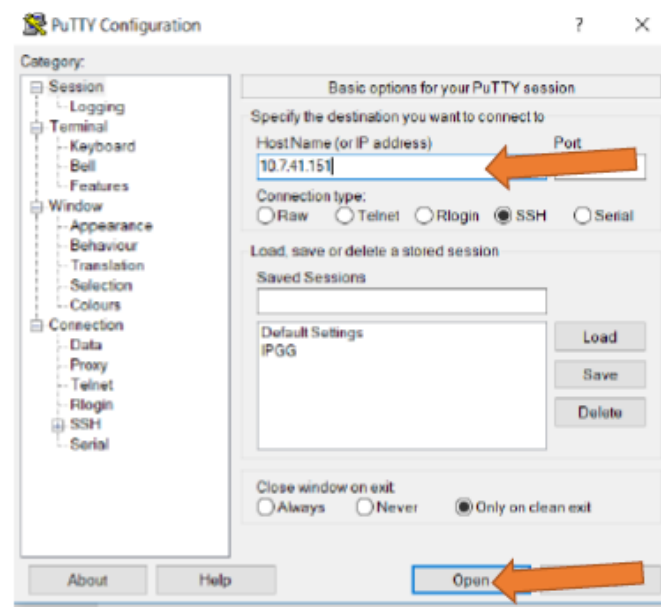


# Creating a VM on



# Remotely access the VM (or server)

- Windows
  - Require a software
  - Install “putty”
  - Login as ‘root’
    - Windows user: Open PuTTY → enter droplet’s IP Address → click “Open”



# Remotely access the VM (or server)

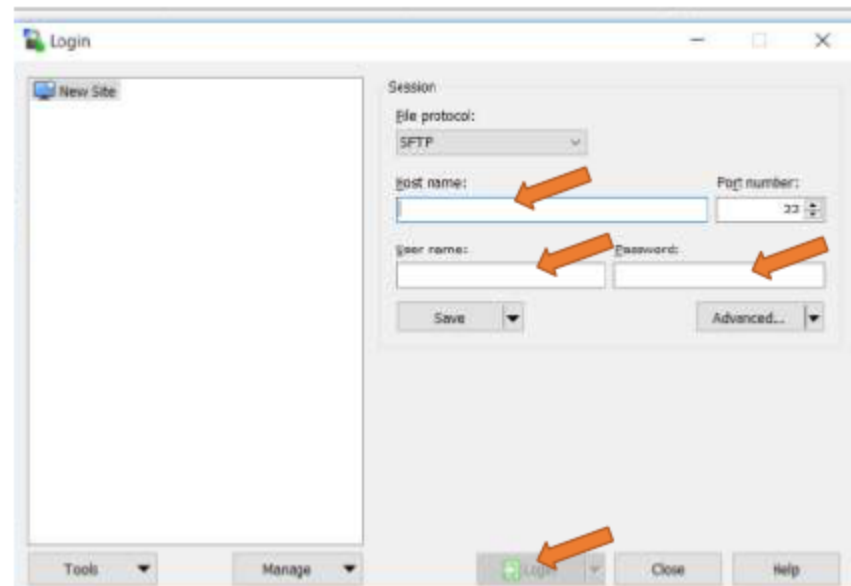
- Mac and Linux
  - Use your 'terminal'
  - `ssh root@<VM IP address>`

# Test basic Unix/Linux commands

- `ls`
- `pwd`
- `touch`
- `cp`
- `mv`
- `rm`
- `mkdir`
- `cd`
- `*nano*`

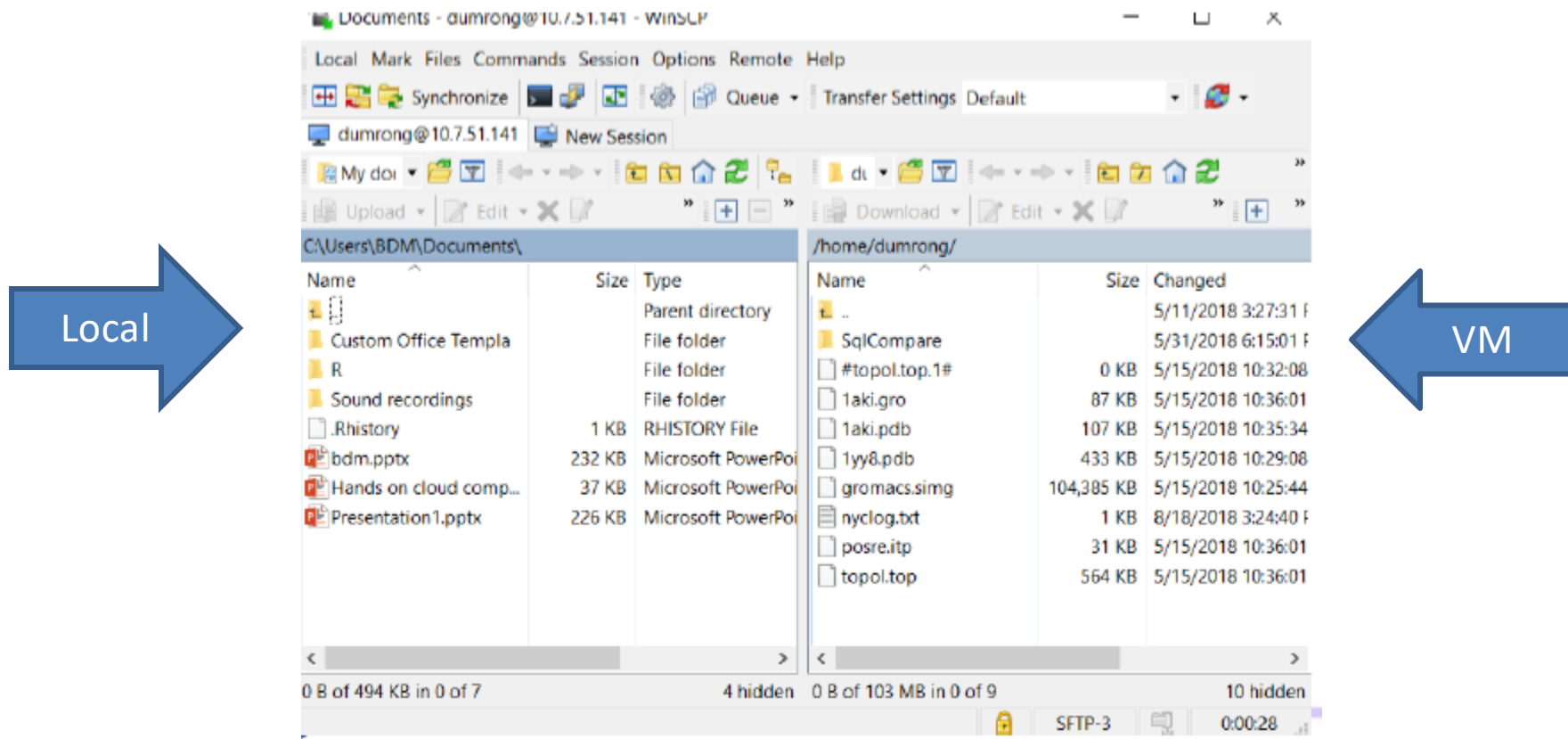
# Remotely transfer files from/to the VM

- Windows
  - Require a software
  - Install “WinSCP”
    - Windows user: Open WinSCP → enter droplet’s IP Address  
→ Enter username (‘root’) and password → click “Login”



# Remotely transfer files from/to the VM

- Windows
  - Require a software
  - Install “WinSCP”



# Remotely transfer files from/to the VM

- Mac and Linux
  - Use your 'terminal'
  - Download:

```
scp root@<VM IP address>:/path/to/file /local/path
```

- Upload:

```
scp /local/path/to/file root@<VM IP address>:/path/
```

# Practice file transfer (to VM)

1. Open notepad/text editor in your computer
2. Write "Thank you." in a new text file
3. Save the file as `thank.txt`
4. Upload this file to your VM (WinSCP or scp)
5. Access your VM (PuTTY or ssh)
6. Type `ls` see if you can find your file
7. Type `cat thank.txt` to read the content in the file



# Practice file transfer (from VM)

1. Access your droplet (PuTTY or ssh)
2. Type `echo "you are welcome" > yaw.txt`
3. Connect to your VM with WinSCP (or use scp)
4. Download `yaw.txt` to you computer
5. Use notepad or word to read the file

# Back-up and clean-up

- Back-up with 'snapshot':
  - Freeze everything in your VM at the time of the snapshot.
  - The snapshot is used to instantly restore the VM.
  - Use for:
    - Routine back-up
    - Temporary paused VM
  - Cheaper than let the VM run. (Snapshot is **NOT** free!)

# Back-up and clean-up

- Clean-up:
  - Running and power-off VMs cost the same!
  - **Everything** must be destroyed to stop the cloud provider from charging you.
  - Snapshots and volumes cost money!
  - Clean-up after:
    - Finish the project
    - Results and scripts are downloaded to local machines
- After a VM is destroyed, data in the VM are gone for good!