Pintos Project: Introduction

CSL-Pintos

Contents

1. Introduction to Pintos

What is Pintos?

Pintos

- Pintos is an educational operating system

- This OS is designed to provide an experience to develop operating system without being excessively complex

- Pintos is developed with several limitations in terms of file system, thread scheduler, virtualization etc.

- Our purpose is to improve Pintos with advanced ideas

Pintos project

Project 1: threads

- **Alarm clock** / **Priority scheduling** / Advanced scheduler

Project 2: User program

- Argument passing / User memory / System calls

Project 3: Virtualization

- Memory management / anonymous page / stack growth / memory mapped files

Overview of Pintos source tree

src/utils
- It contains a number of help functions and utilities related to the Pintos kernel

src/threads
- It configures the behavior of kernel threads

src/devices
- It contains hardware device drivers and related code for the Pintos operating system

src/lib
- It contains various libraries and utility codes providing common functions and features for both the kernel and application programs

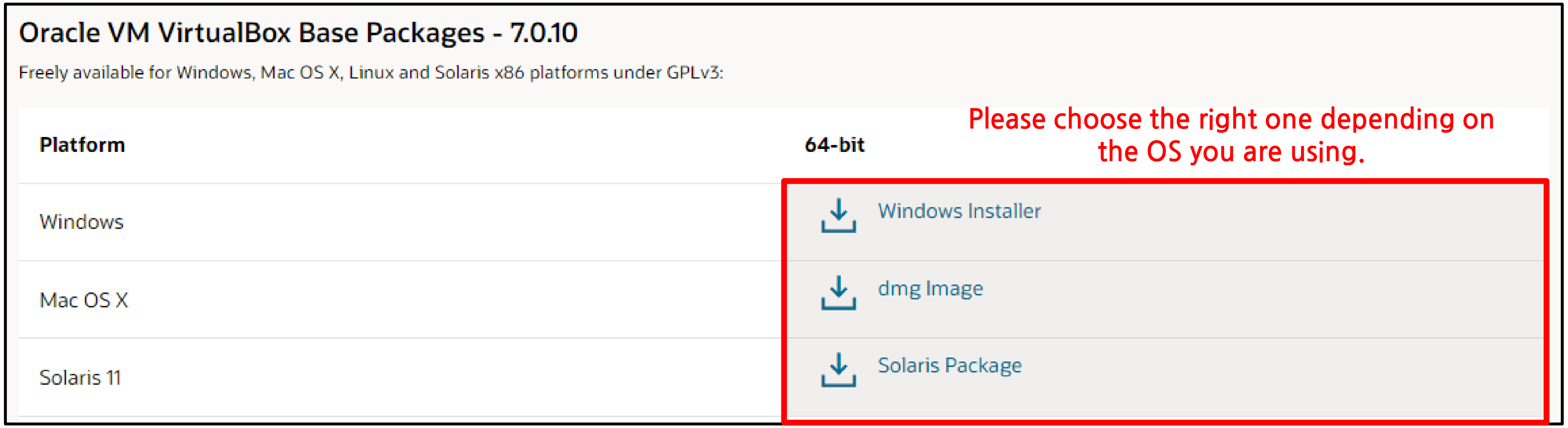
src/tests
- It contains test code and test suites used to verify various components and functionalities

2. Install Oracle VM VirtualBox

Download VirtualBox installer

Download Link

https://www.oracle.com/kr/virtualization/technologies/vm/downloads/virtualbox-downloads.html



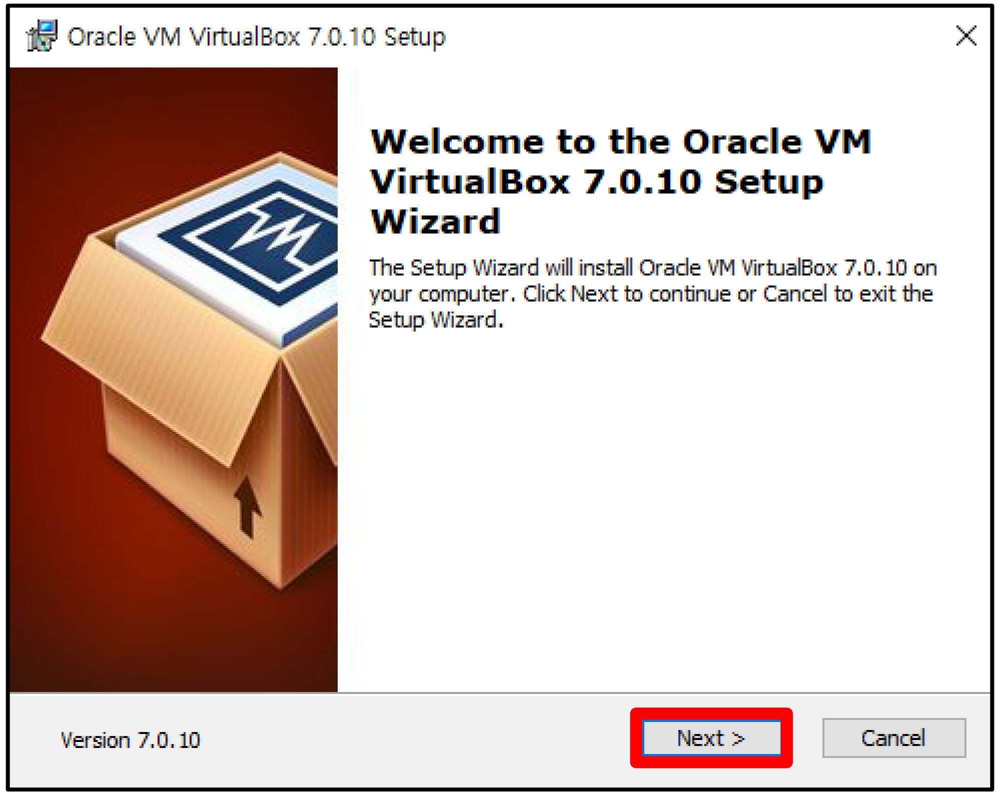
Oracle VM VirtualBox Base Packages - 7.0.10

Freely available for Windows, Mac OS X, Linux and Solaris x86 platforms under GPLv3:

Platform	64-bit
Windows	Windows Installer
Mac OS X	dmg Image
Solaris 11	Solaris Package

Please choose the right one depending on the OS you are using.

Installation Procedure ①

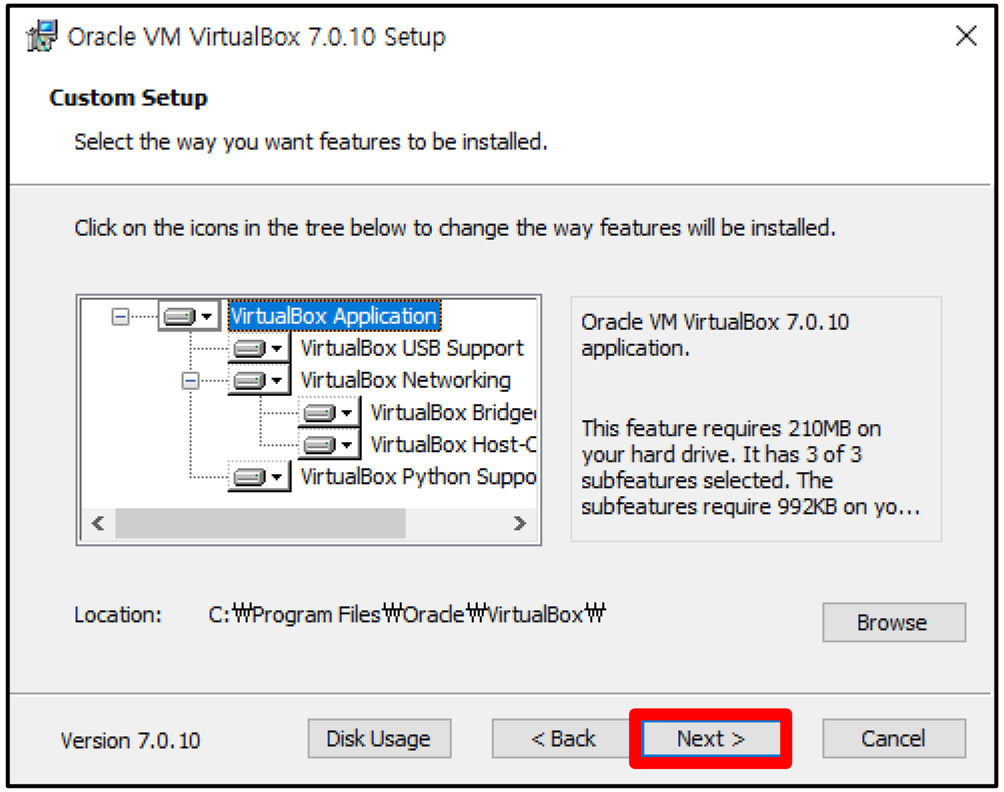


Oracle VM VirtualBox 7.0.10 Setup

Welcome to the Oracle VM VirtualBox 7.0.10 Setup Wizard

The Setup Wizard will install Oracle VM VirtualBox 7.0.10 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Version 7.0.10



Oracle VM VirtualBox 7.0.10 Setup

Custom Setup

Select the way you want features to be installed.

Click on the icons in the tree below to change the way features will be installed.

- VirtualBox Application
  - VirtualBox USB Support
  - VirtualBox Networking
    - VirtualBox Bridge
    - VirtualBox Host-C
  - VirtualBox Python Suppo

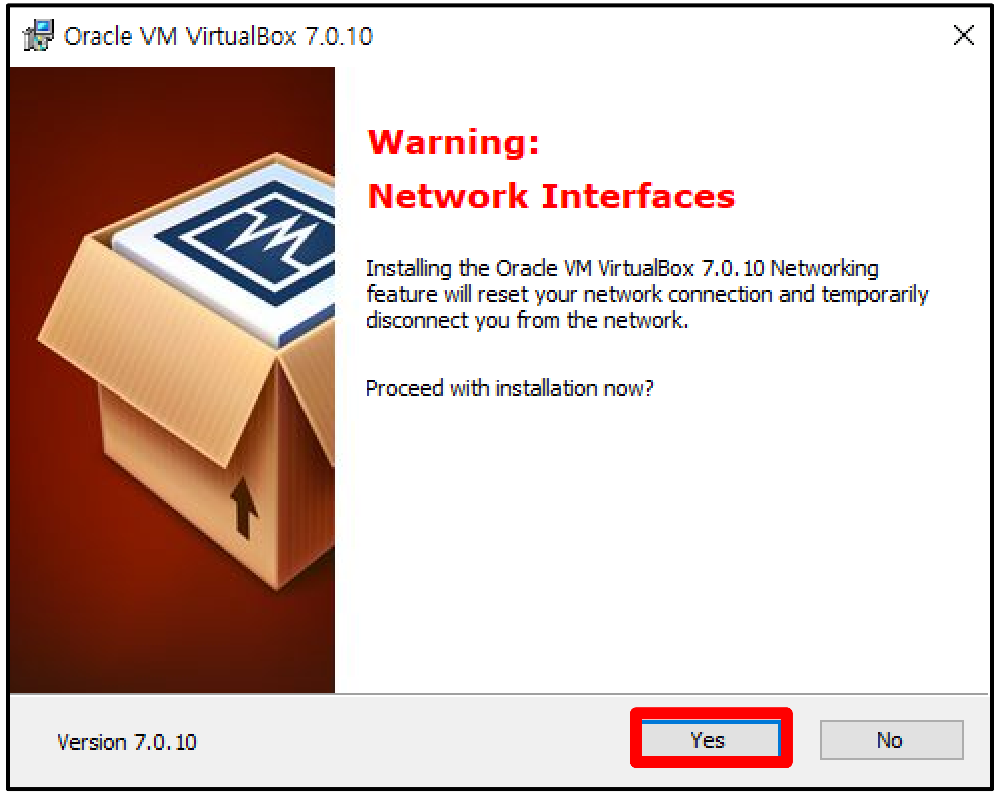
Oracle VM VirtualBox 7.0.10 application.

This feature requires 210MB on your hard drive. It has 3 of 3 subfeatures selected. The subfeatures require 992KB on yo...

Location: C:₩Program Files₩Oracle₩VirtualBox₩

Version 7.0.10

Installation Procedure ②

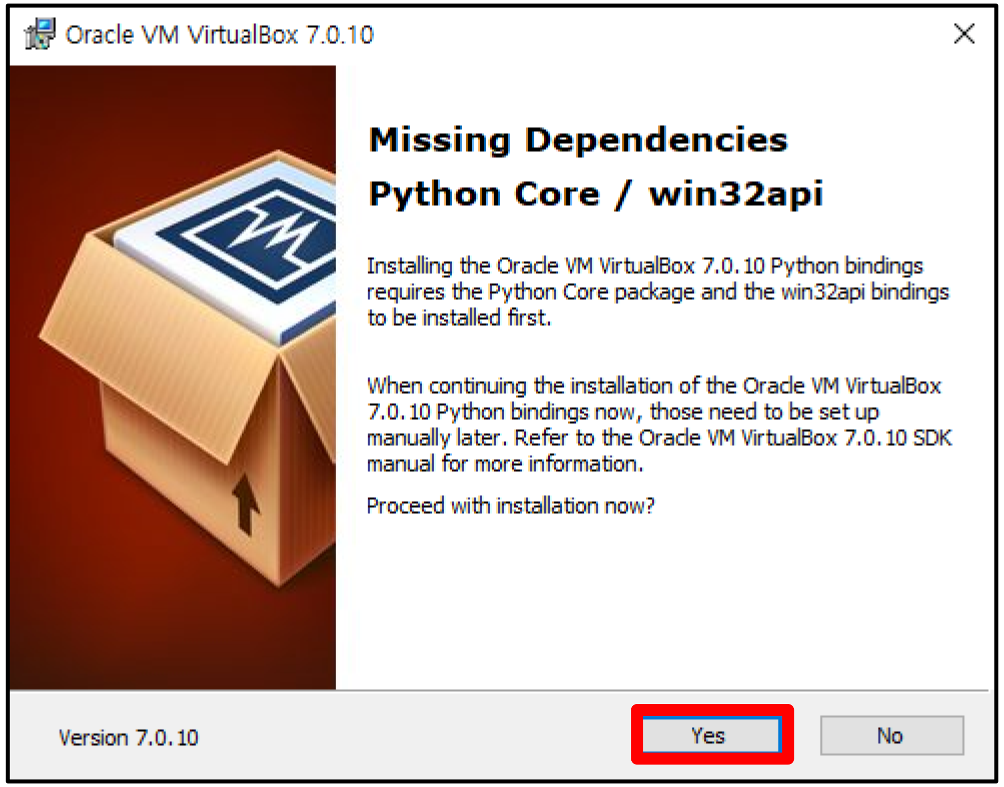

Oracle VM VirtualBox 7.0.10

Warning:
Network Interfaces

Installing the Oracle VM VirtualBox 7.0.10 Networking feature will reset your network connection and temporarily disconnect you from the network.

Proceed with installation now?

Version 7.0.10


Oracle VM VirtualBox 7.0.10

Missing Dependencies
Python Core / win32api

Installing the Oracle VM VirtualBox 7.0.10 Python bindings requires the Python Core package and the win32api bindings to be installed first.

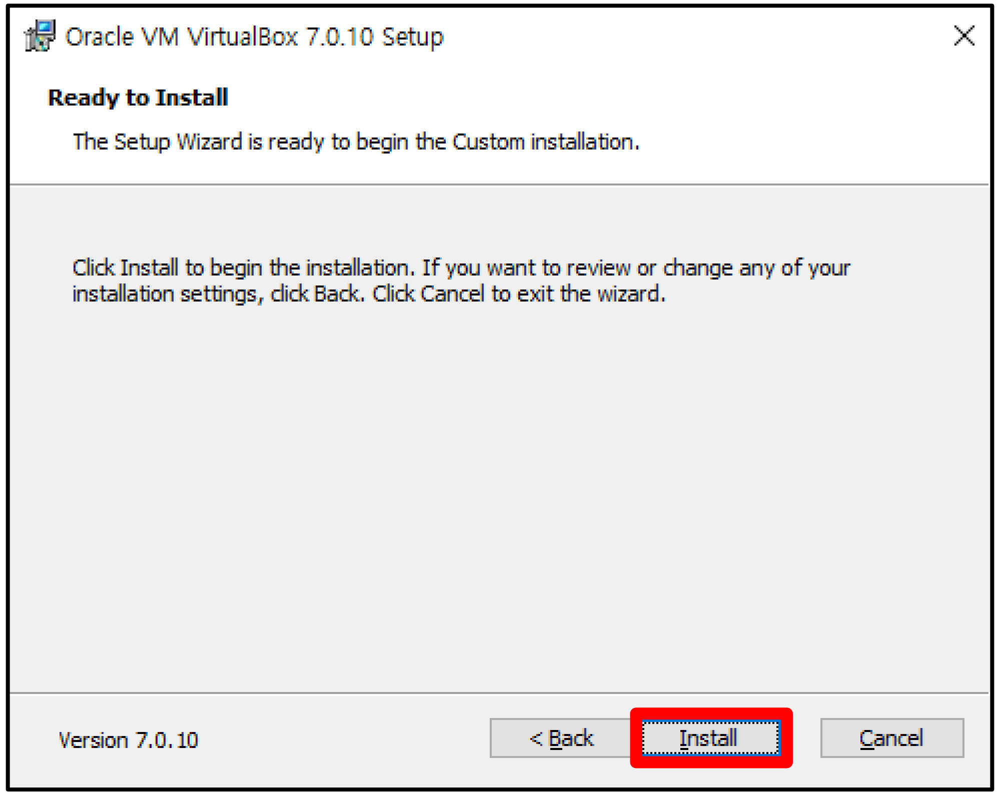
When continuing the installation of the Oracle VM VirtualBox 7.0.10 Python bindings now, those need to be set up manually later. Refer to the Oracle VM VirtualBox 7.0.10 SDK manual for more information.

Proceed with installation now?

Version 7.0.10

Computer Systems Security Laboratory

Installation Procedure ③

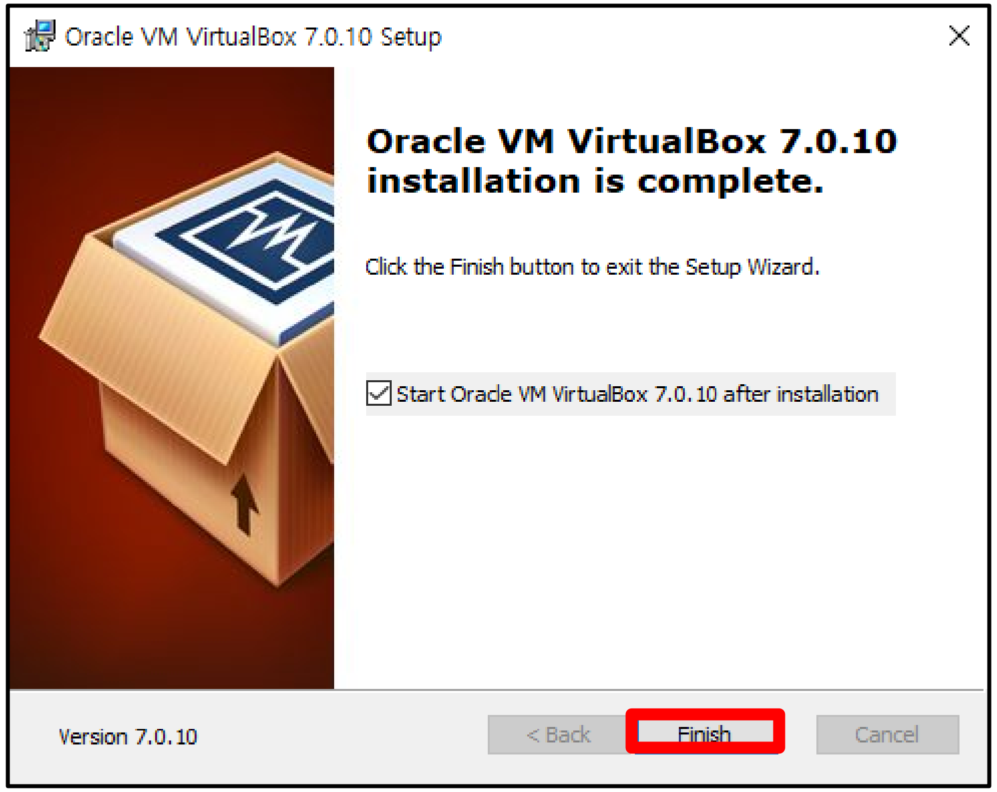

Oracle VM VirtualBox 7.0.10 Setup

Ready to Install

The Setup Wizard is ready to begin the Custom installation.

Click Install to begin the installation. If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.

Version 7.0.10


Oracle VM VirtualBox 7.0.10 Setup

Oracle VM VirtualBox 7.0.10 installation is complete.

Click the Finish button to exit the Setup Wizard.

[✓] Start Oracle VM VirtualBox 7.0.10 after installation

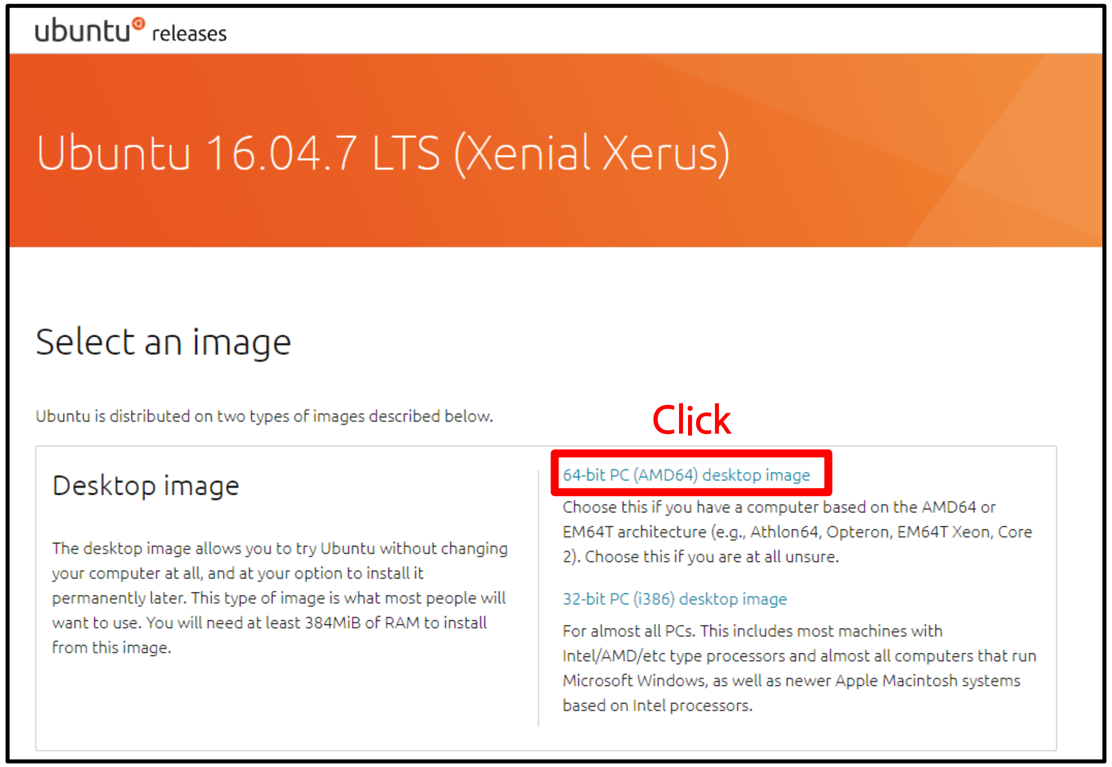
Version 7.0.10

3. Install Ubuntu16.04.7 LTS via VirtualBox

Download Ubuntu 16.04.7 LTS ISO image

Download Link

https://releases.ubuntu.com/16.04/



ubuntu releases

Ubuntu 16.04.7 LTS (Xenial Xerus)

Select an image

Ubuntu is distributed on two types of images described below.

Desktop image

The desktop image allows you to try Ubuntu without changing your computer at all, and at your option to install it permanently later. This type of image is what most people will want to use. You will need at least 384MiB of RAM to install from this image.

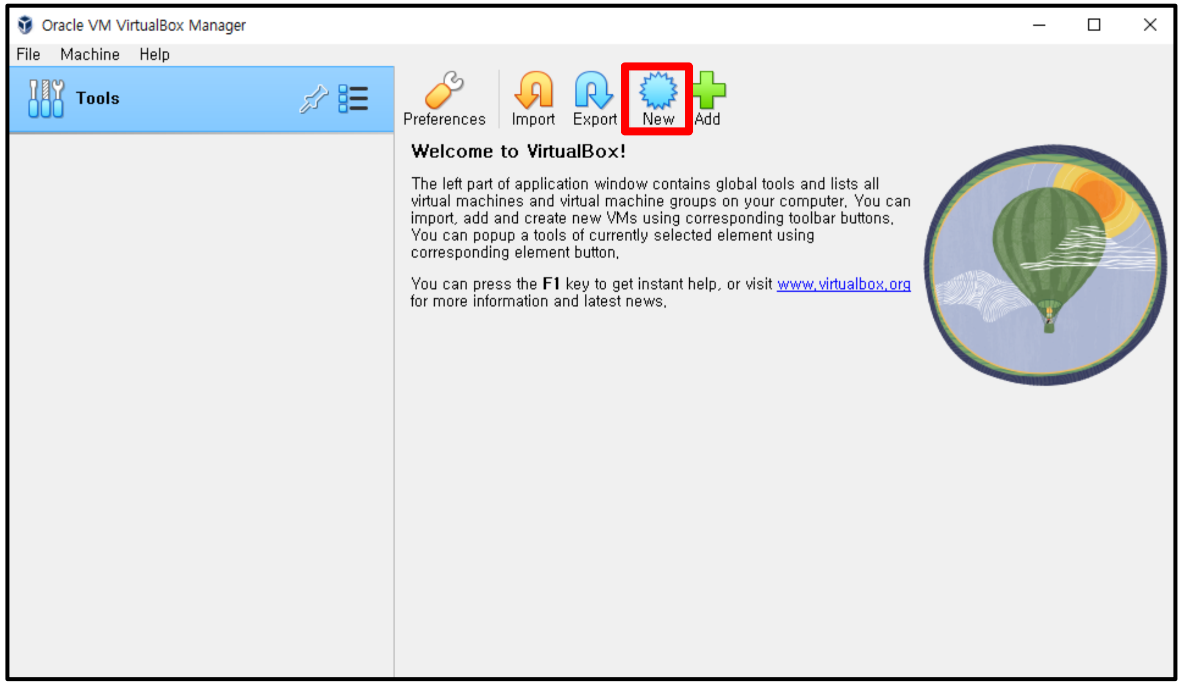
64-bit PC (AMD64) desktop image

Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.

32-bit PC (i386) desktop image

For almost all PCs. This includes most machines with Intel/AMD/etc type processors and almost all computers that run Microsoft Windows, as well as newer Apple Macintosh systems based on Intel processors.

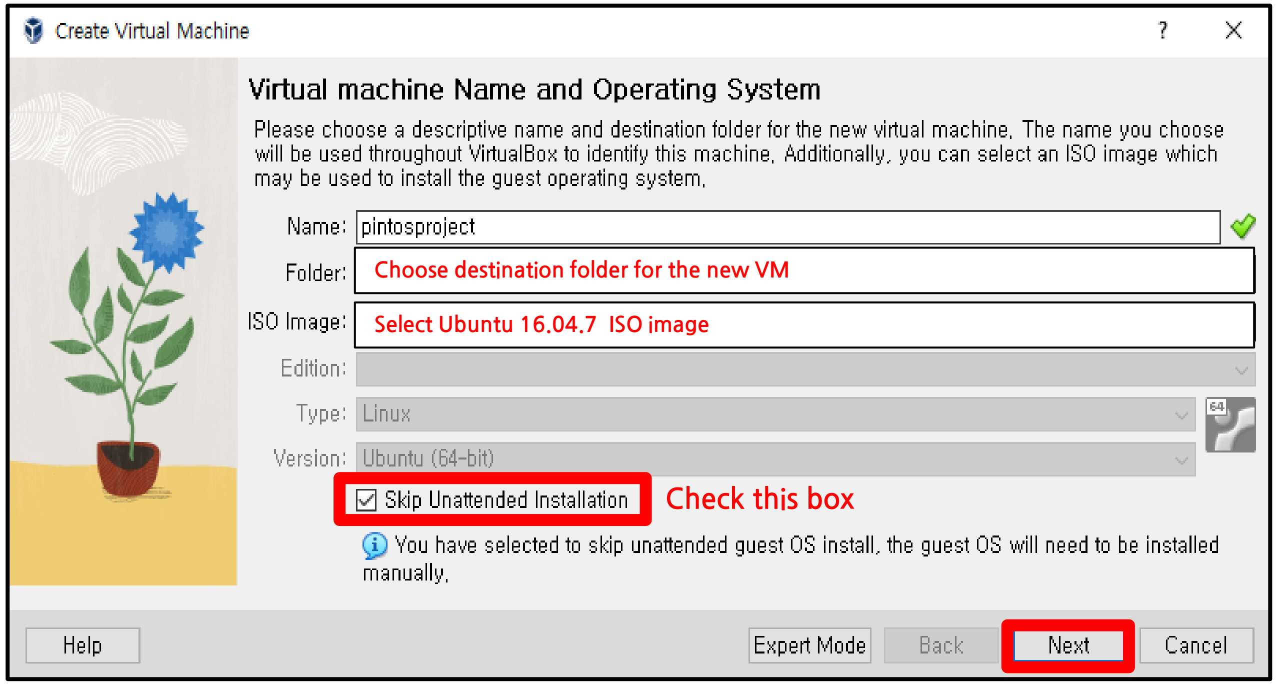
Installation Procedure ①



Welcome to VirtualBox!

The left part of application window contains global tools and lists all virtual machines and virtual machine groups on your computer. You can import, add and create new VMs using corresponding toolbar buttons. You can popup a tools of currently selected element using corresponding element button.

You can press the **F1** key to get instant help, or visit www.virtualbox.org for more information and latest news.



Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name: pintosproject

Folder: Choose destination folder for the new VM

ISO Image: Select Ubuntu 16.04.7 ISO image

Edition:

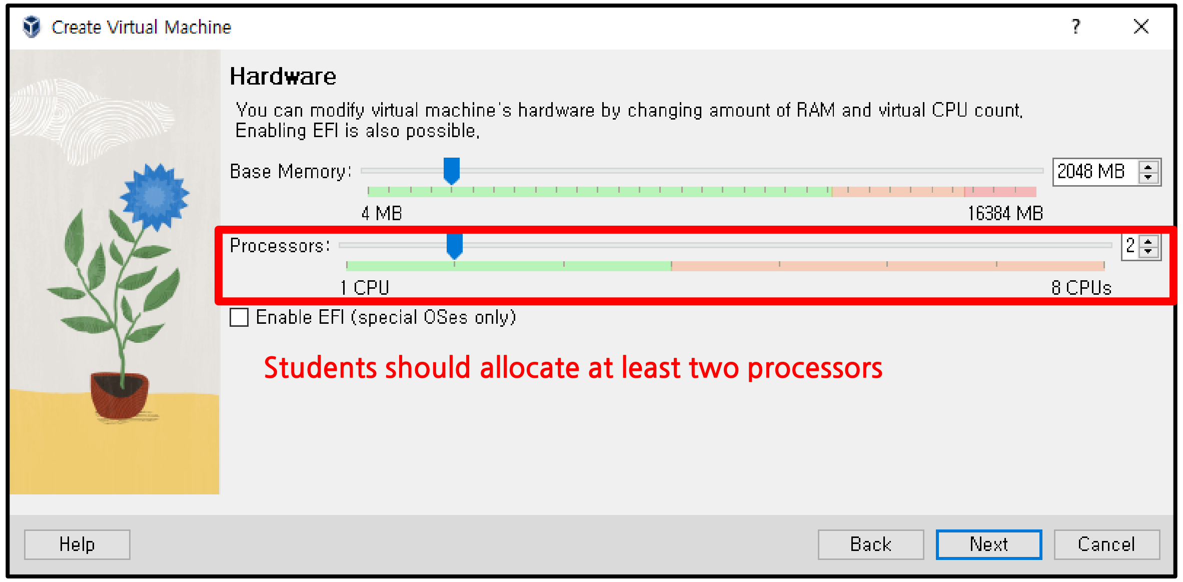
Type: Linux

Version: Ubuntu (64-bit)

☑ Skip Unattended Installation Check this box

You have selected to skip unattended guest OS install, the guest OS will need to be installed manually.

Installation Procedure ②



Create Virtual Machine

Hardware

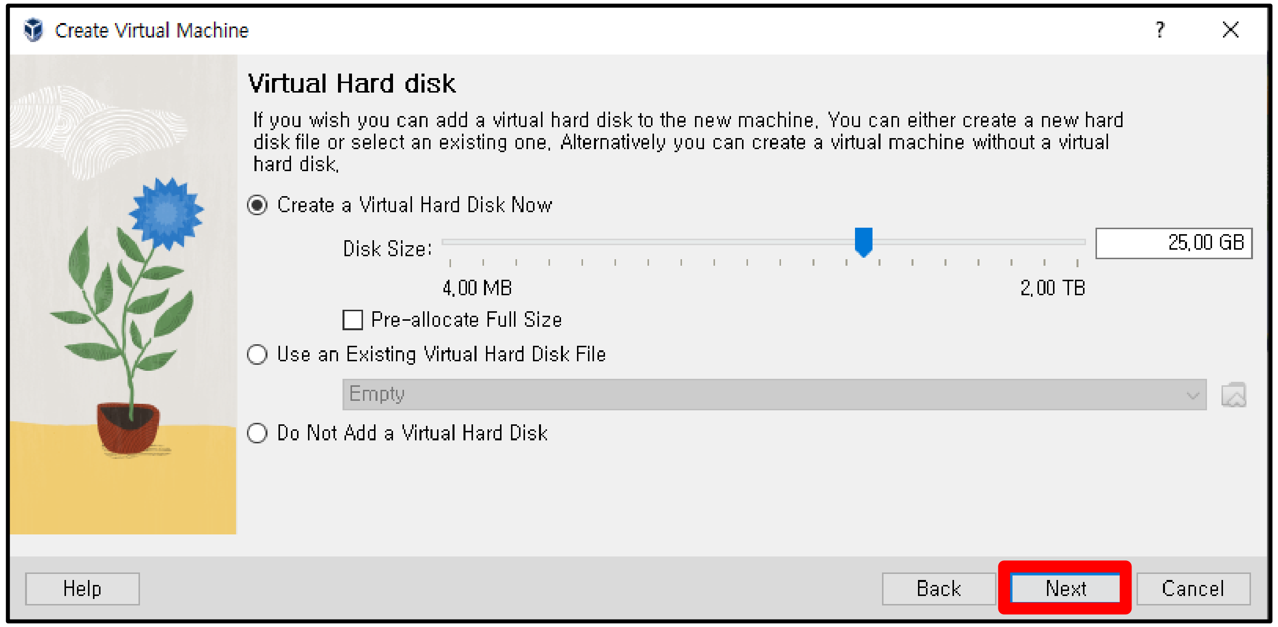
You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 2048 MB (4 MB – 16384 MB)

Processors: 2 (1 CPU – 8 CPUs)

☐ Enable EFI (special OSes only)

Students should allocate at least two processors



Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

● Create a Virtual Hard Disk Now

Disk Size: 25,00 GB (4,00 MB – 2,00 TB)

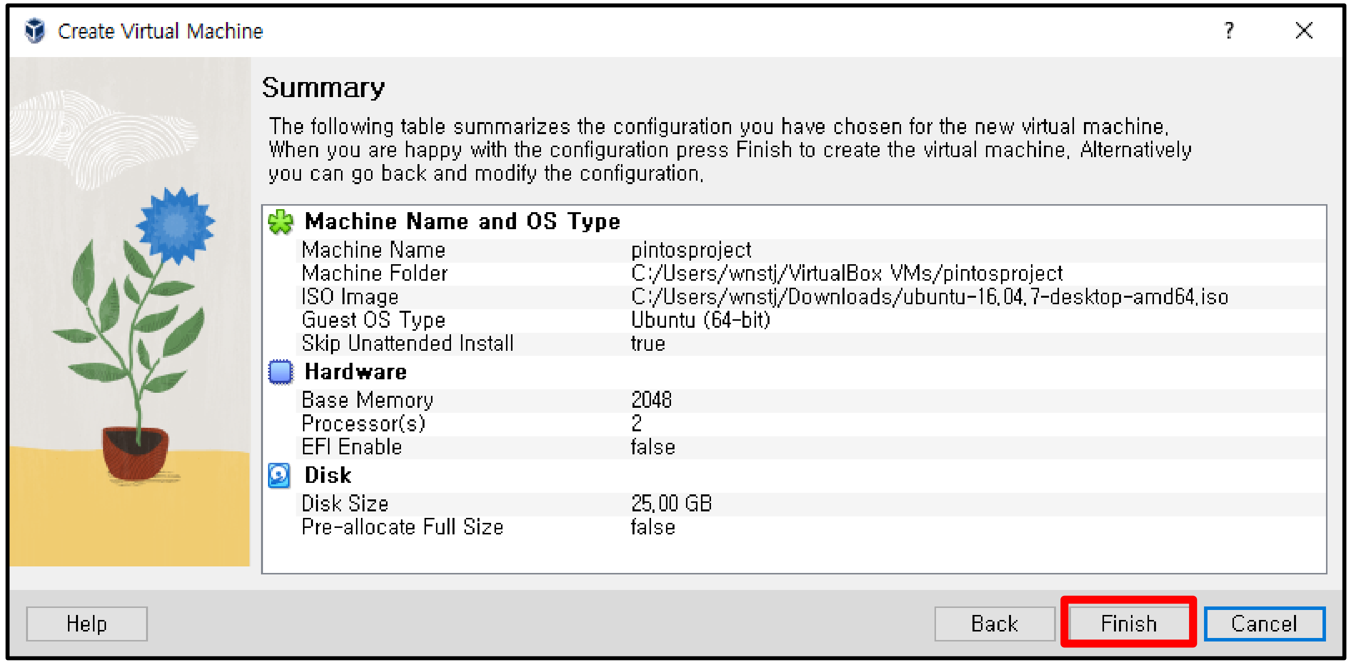
☐ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

Empty

○ Do Not Add a Virtual Hard Disk

Installation Procedure ③



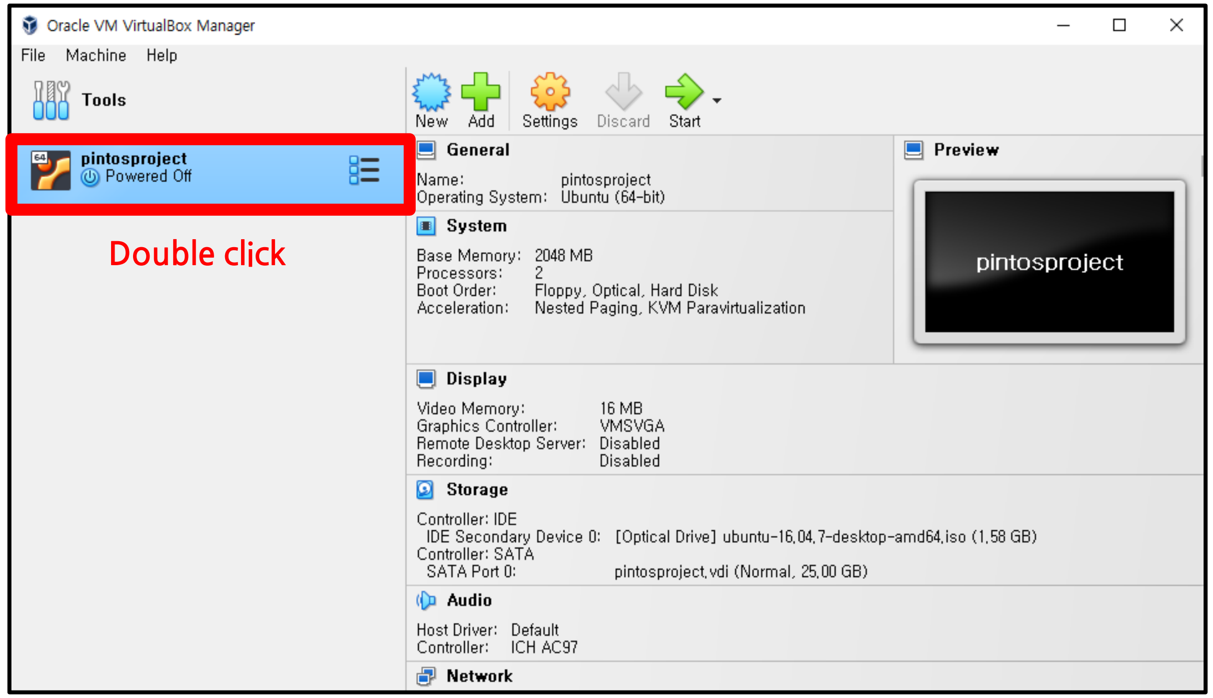
Create Virtual Machine

Summary

The following table summarizes the configuration you have chosen for the new virtual machine. When you are happy with the configuration press Finish to create the virtual machine. Alternatively you can go back and modify the configuration.

Machine Name and OS Type	
Machine Name	pintosproject
Machine Folder	C:/Users/wnstj/VirtualBox VMs/pintosproject
ISO Image	C:/Users/wnstj/Downloads/ubuntu-16.04.7-desktop-amd64.iso
Guest OS Type	Ubuntu (64-bit)
Skip Unattended Install	true
Hardware	
Base Memory	2048
Processor(s)	2
EFI Enable	false
Disk	
Disk Size	25.00 GB
Pre-allocate Full Size	false

Installation Procedure ④


Oracle VM VirtualBox Manager

File Machine Help

Tools

pintosproject
Powered Off

General

Name: pintosproject
Operating System: Ubuntu (64-bit)

System

Base Memory: 2048 MB
Processors: 2
Boot Order: Floppy, Optical, Hard Disk
Acceleration: Nested Paging, KVM Paravirtualization

Preview

pintosproject

Display

Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

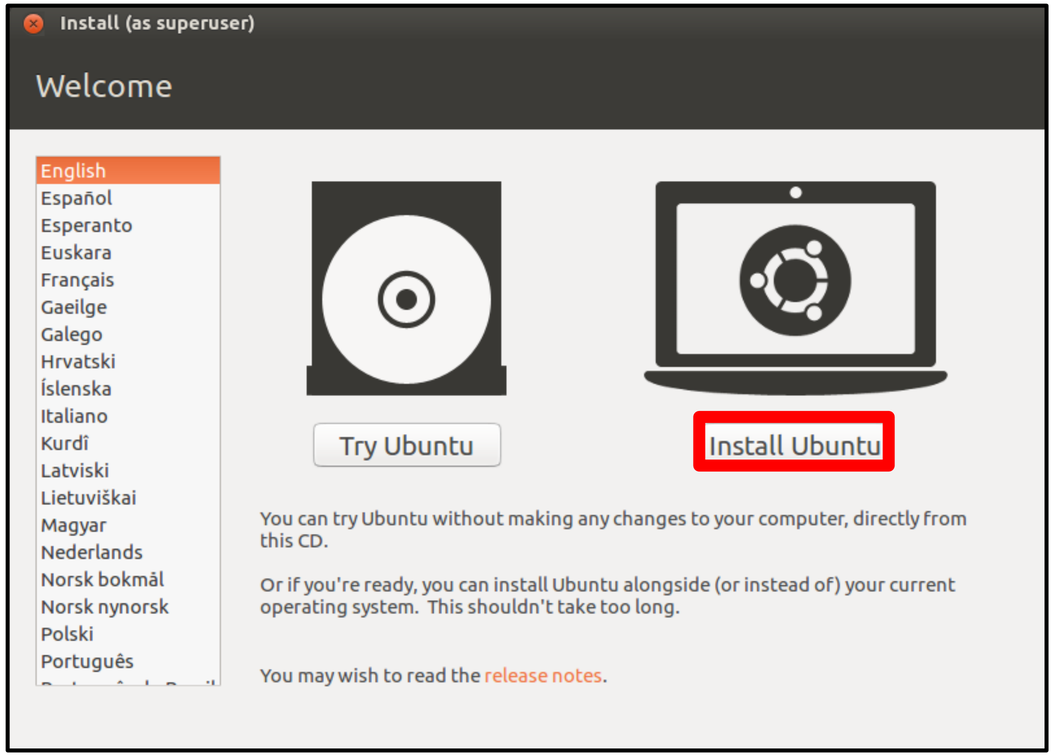
Storage

Controller: IDE
IDE Secondary Device 0: [Optical Drive] ubuntu-16.04.7-desktop-amd64.iso (1.58 GB)
Controller: SATA
SATA Port 0: pintosproject.vdi (Normal, 25.00 GB)

Audio

Host Driver: Default
Controller: ICH AC97

Network


Install (as superuser)

Welcome

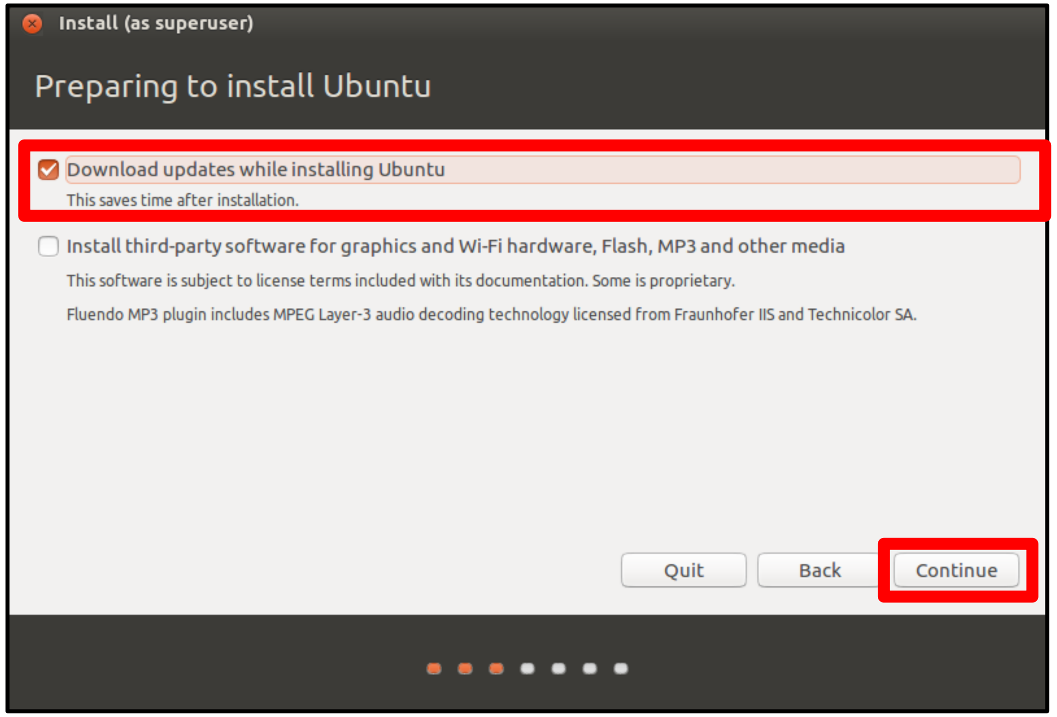
English
Español
Esperanto
Euskara
Français
Gaeilge
Galego
Hrvatski
Íslenska
Italiano
Kurdî
Latviski
Lietuviškai
Magyar
Nederlands
Norsk bokmål
Norsk nynorsk
Polski
Português

You can try Ubuntu without making any changes to your computer, directly from this CD.

Or if you're ready, you can install Ubuntu alongside (or instead of) your current operating system. This shouldn't take too long.

You may wish to read the release notes.

Installation Procedure ⑤



Install (as superuser)

Preparing to install Ubuntu

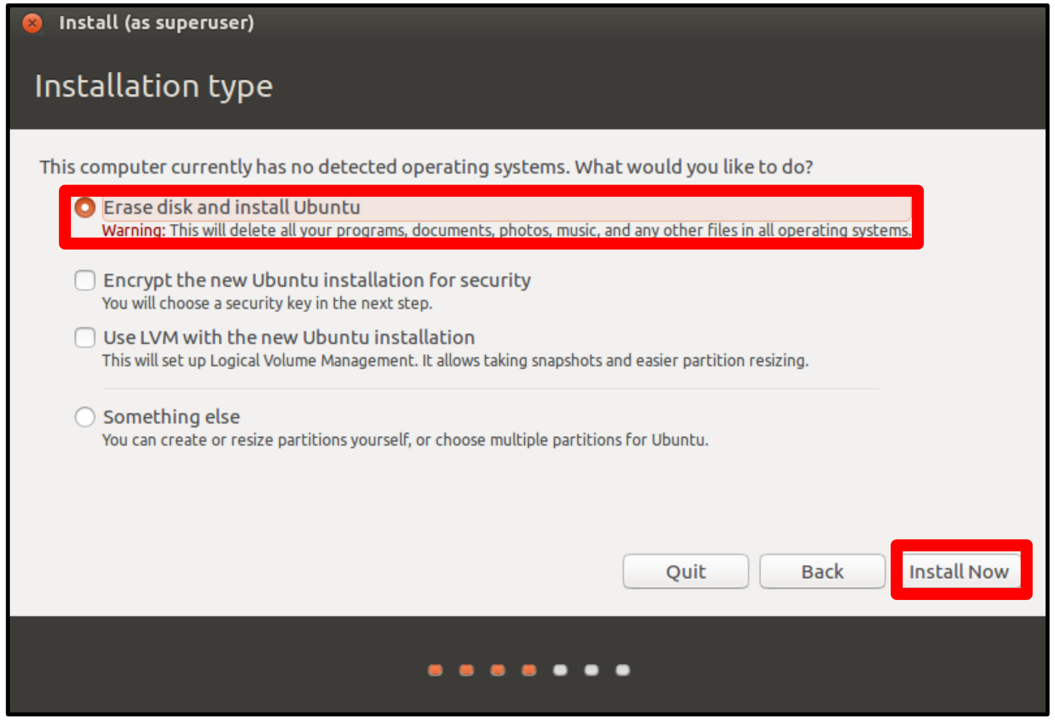
[✓] Download updates while installing Ubuntu

This saves time after installation.

[ ] Install third-party software for graphics and Wi-Fi hardware, Flash, MP3 and other media

This software is subject to license terms included with its documentation. Some is proprietary.

Fluendo MP3 plugin includes MPEG Layer-3 audio decoding technology licensed from Fraunhofer IIS and Technicolor SA.



Install (as superuser)

Installation type

This computer currently has no detected operating systems. What would you like to do?

[✓ Erase disk and install Ubuntu]

Warning: This will delete all your programs, documents, photos, music, and any other files in all operating systems.

[ ] Encrypt the new Ubuntu installation for security

You will choose a security key in the next step.

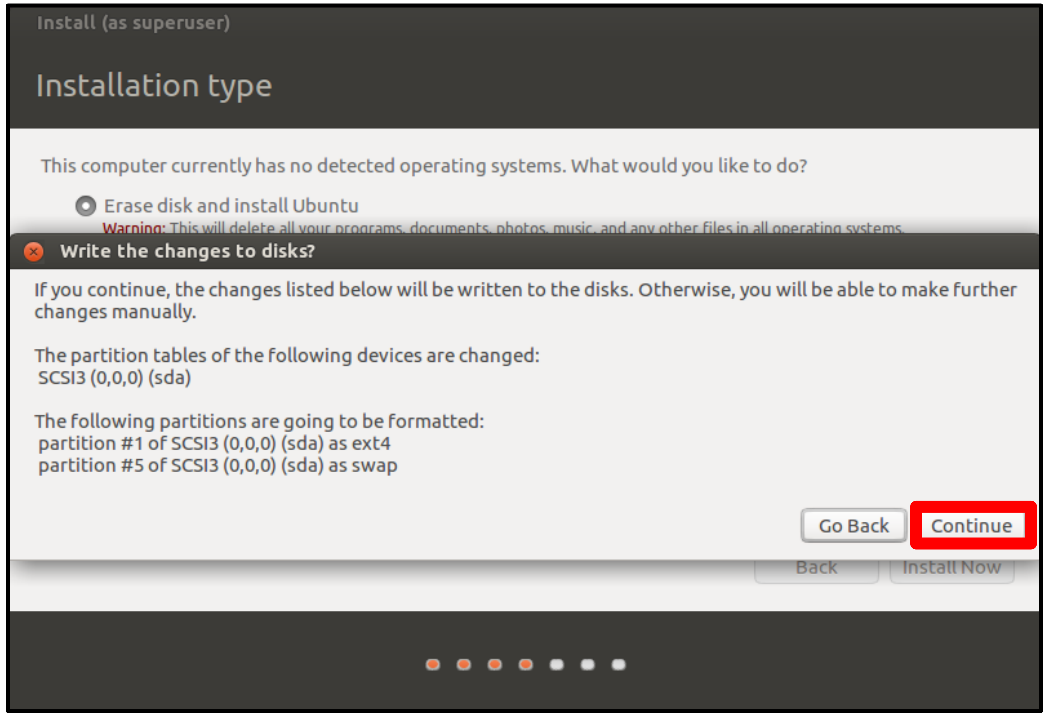
[ ] Use LVM with the new Ubuntu installation

This will set up Logical Volume Management. It allows taking snapshots and easier partition resizing.

Something else

You can create or resize partitions yourself, or choose multiple partitions for Ubuntu.

Installation Procedure ⑥


Install (as superuser)

Installation type

This computer currently has no detected operating systems. What would you like to do?

Erase disk and install Ubuntu

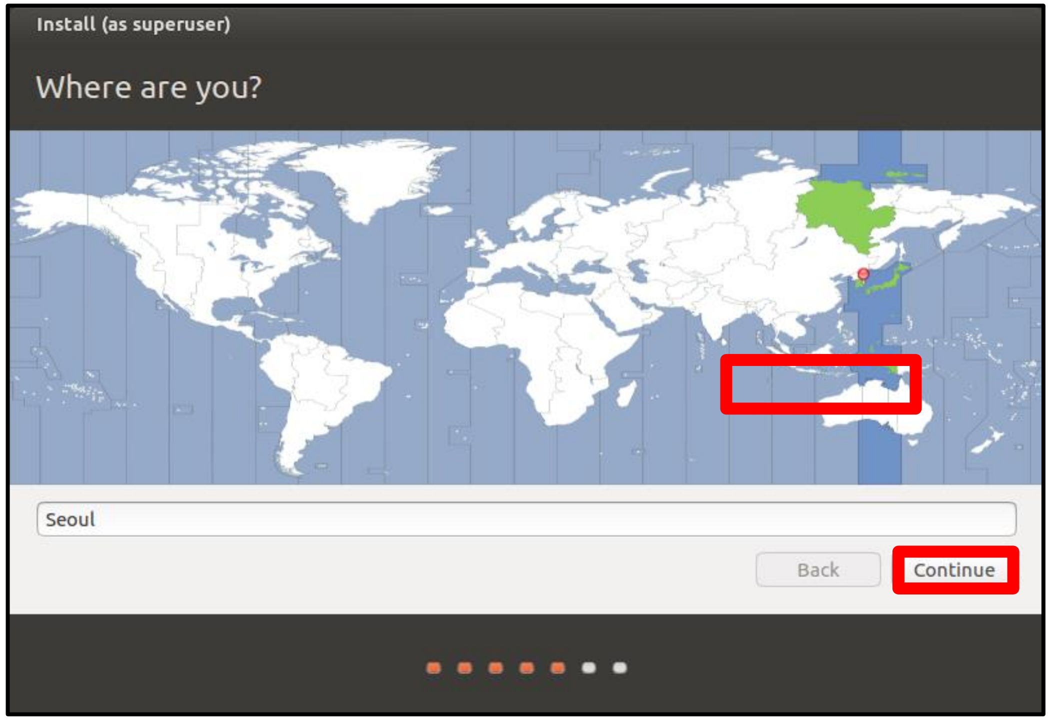
Warning: This will delete all your programs, documents, photos, music, and any other files in all operating systems.

Write the changes to disks?

If you continue, the changes listed below will be written to the disks. Otherwise, you will be able to make further changes manually.

The partition tables of the following devices are changed:
SCSI3 (0,0,0) (sda)

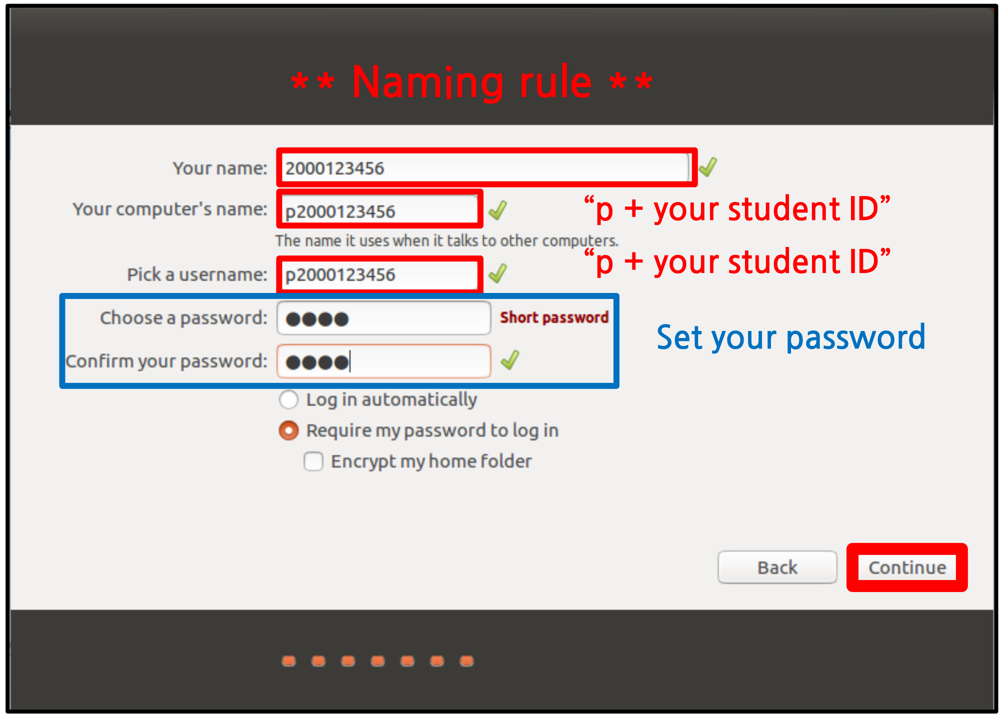
The following partitions are going to be formatted:
partition #1 of SCSI3 (0,0,0) (sda) as ext4
partition #5 of SCSI3 (0,0,0) (sda) as swap


Install (as superuser)

Where are you?

Seoul

Installation Procedure ⑦


** Naming rule **

Your name: 2000123456

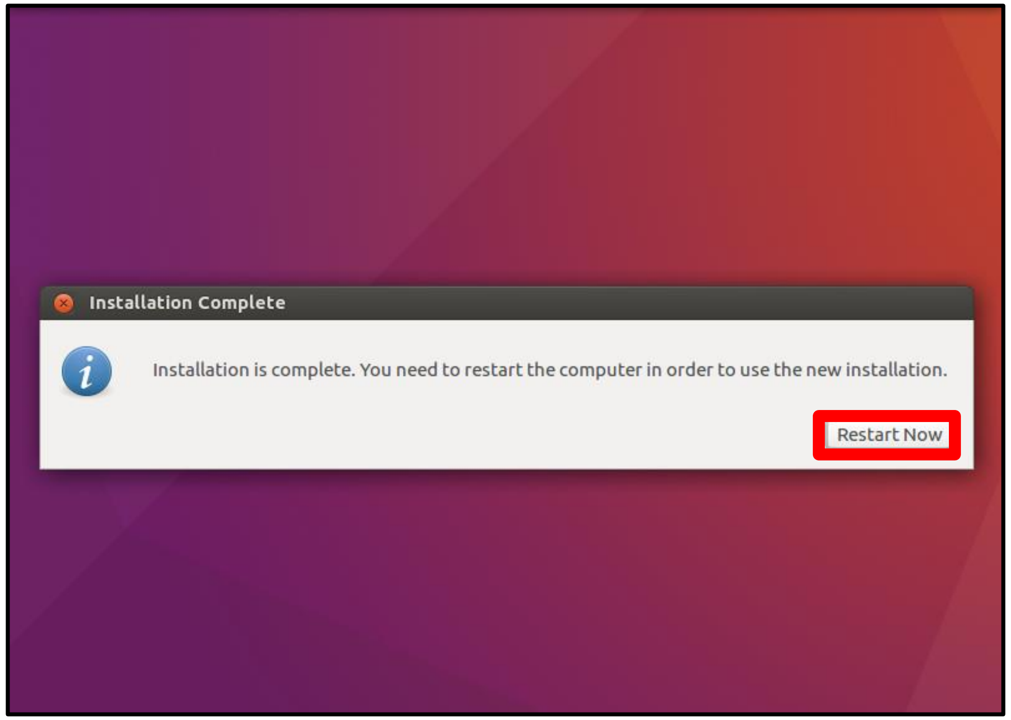
Your computer's name: p2000123456

The name it uses when it talks to other computers.

Pick a username: p2000123456

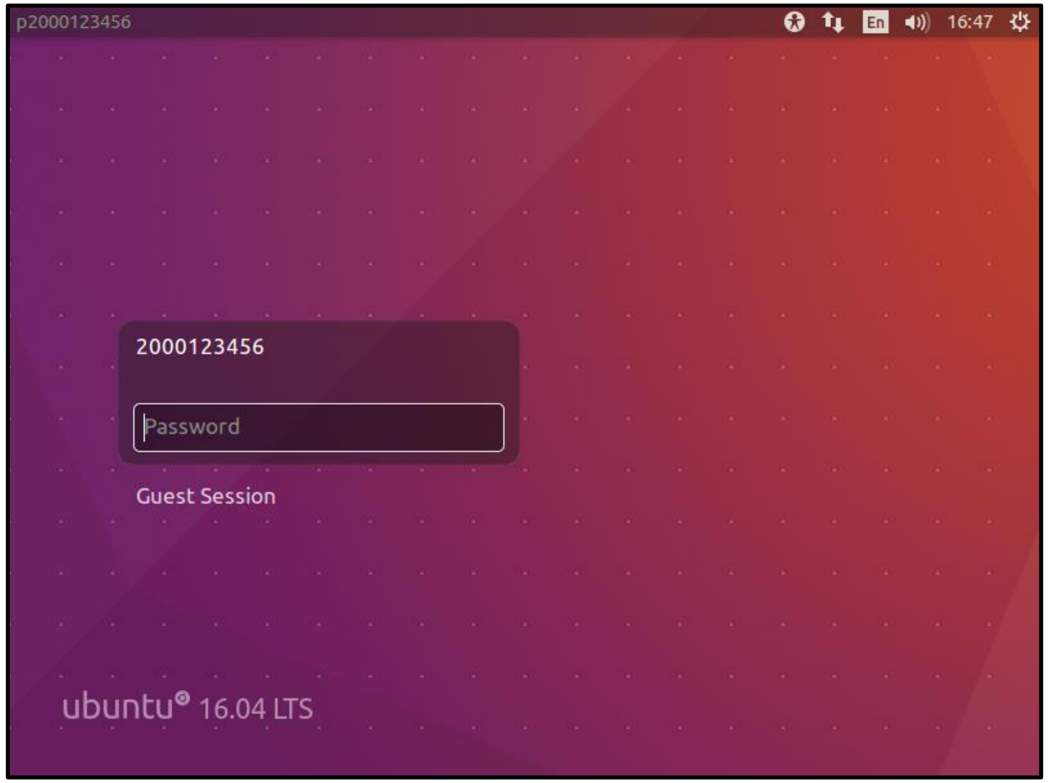
Choose a password: ●●●● Short password

Confirm your password: ●●●●


Installation Complete

Installation is complete. You need to restart the computer in order to use the new installation.

Installation Procedure ⑧



Two additional options for convenient use

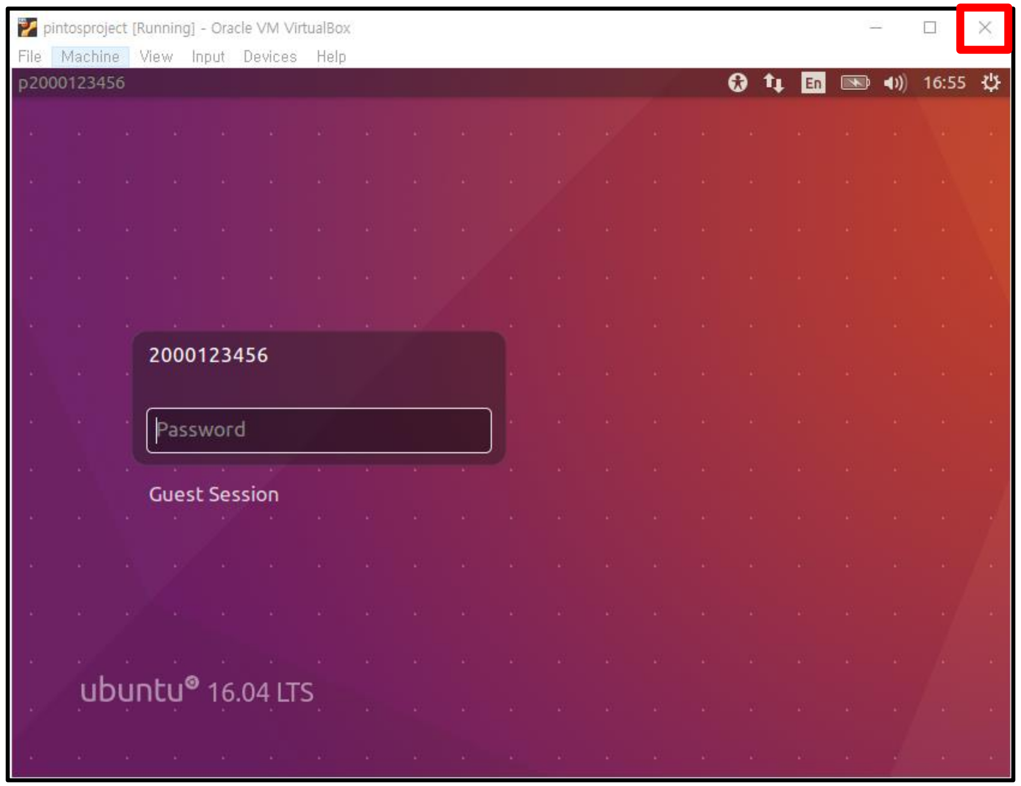
(a) **Shared clipboard / Drag & drop**

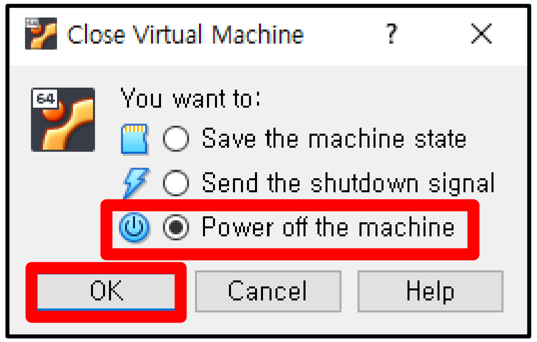
- This option allows guests to have *read or write access* to the clipboard, even when they are not focused on a window in the virtual machine.

(b) **VBoxVGA**

- This option allows guest VM to use full screen

Enabling additional options ①

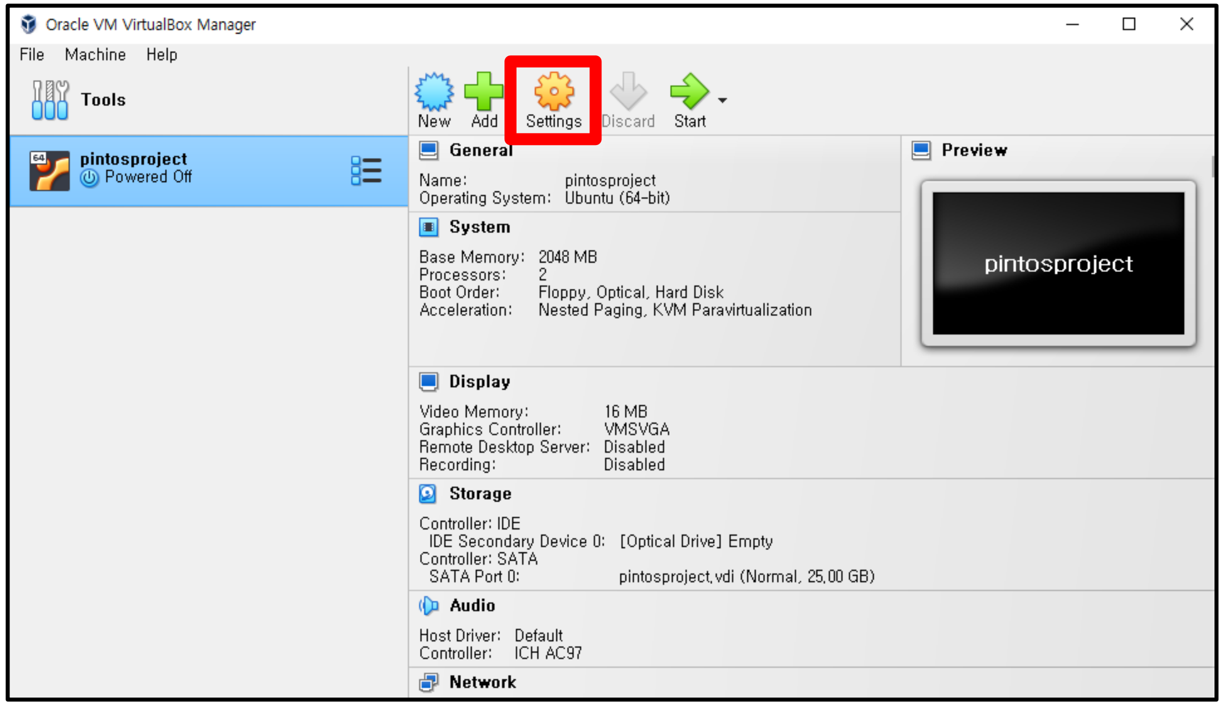



Close Virtual Machine

You want to:

- Save the machine state
- Send the shutdown signal
- [✓ Power off the machine]

Enabling additional options ②



Oracle VM VirtualBox Manager

File Machine Help

Tools

pintosproject
Powered Off

New Add Settings Discard Start

General
Name: pintosproject
Operating System: Ubuntu (64-bit)

System
Base Memory: 2048 MB
Processors: 2
Boot Order: Floppy, Optical, Hard Disk
Acceleration: Nested Paging, KVM Paravirtualization

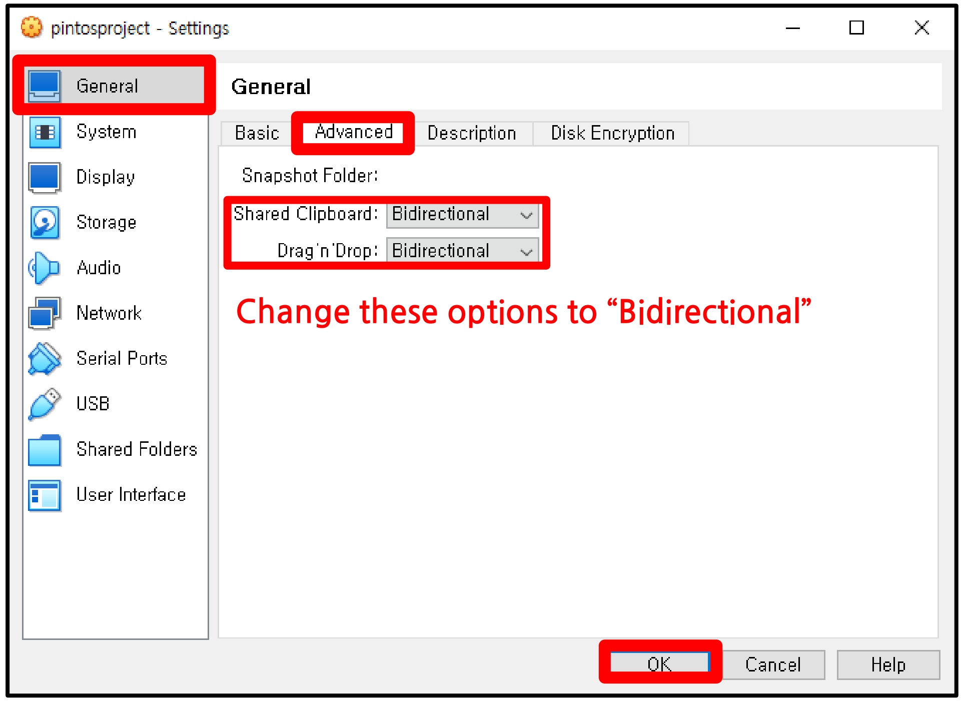
Preview
pintosproject

Display
Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage
Controller: IDE
IDE Secondary Device 0: [Optical Drive] Empty
Controller: SATA
SATA Port 0: pintosproject.vdi (Normal, 25.00 GB)

Audio
Host Driver: Default
Controller: ICH AC97

Network



pintosproject - Settings

General, System, Display, Storage, Audio, Network, Serial Ports, USB, Shared Folders, User Interface

General

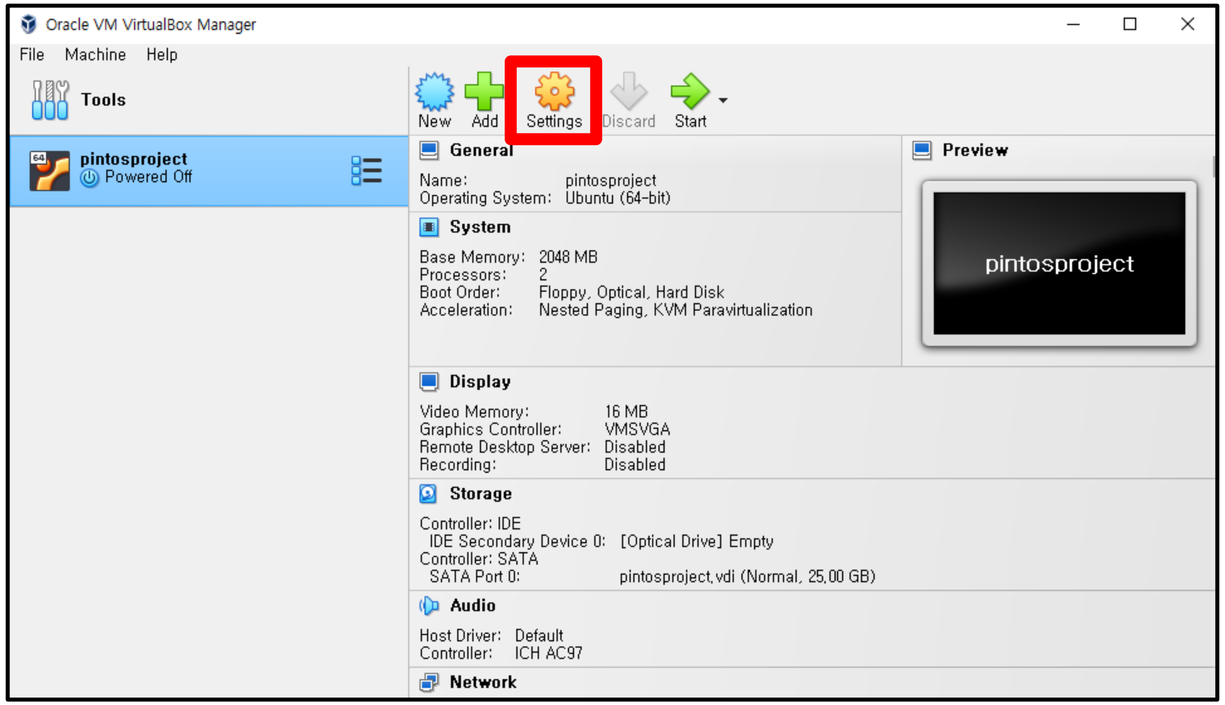
Basic | Advanced | Description | Disk Encryption

Snapshot Folder:
Shared Clipboard: Bidirectional
Drag'n'Drop: Bidirectional

Change these options to "Bidirectional"

OK Cancel Help

Enabling additional options ③



pintosproject
Powered Off

General
Name: pintosproject
Operating System: Ubuntu (64-bit)

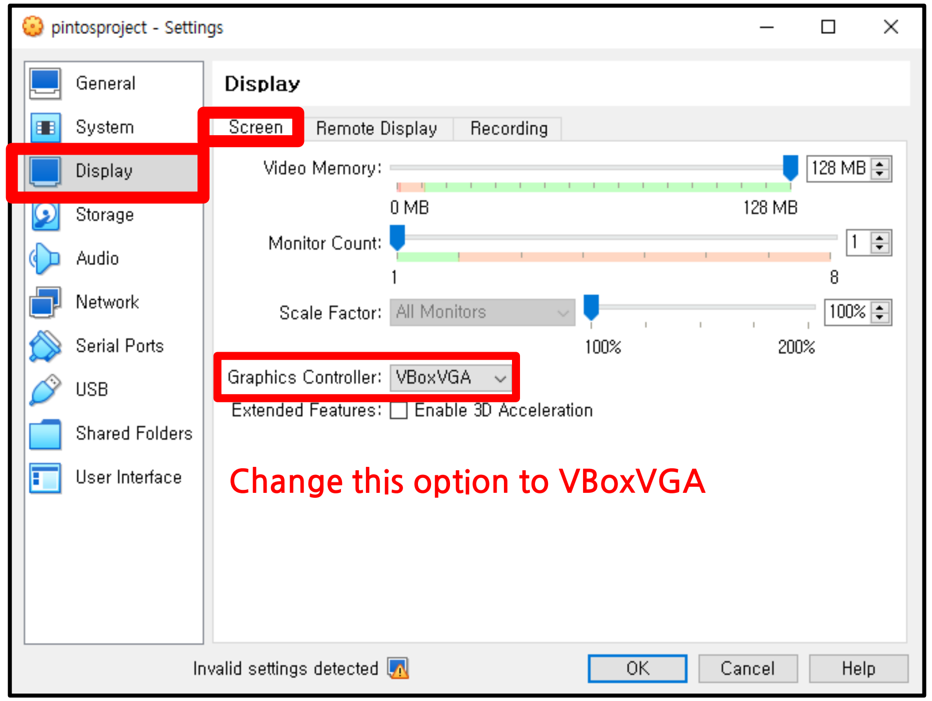
System
Base Memory: 2048 MB
Processors: 2
Boot Order: Floppy, Optical, Hard Disk
Acceleration: Nested Paging, KVM Paravirtualization

Display
Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage
Controller: IDE
IDE Secondary Device 0: [Optical Drive] Empty
Controller: SATA
SATA Port 0: pintosproject.vdi (Normal, 25.00 GB)

Audio
Host Driver: Default
Controller: ICH AC97

Network



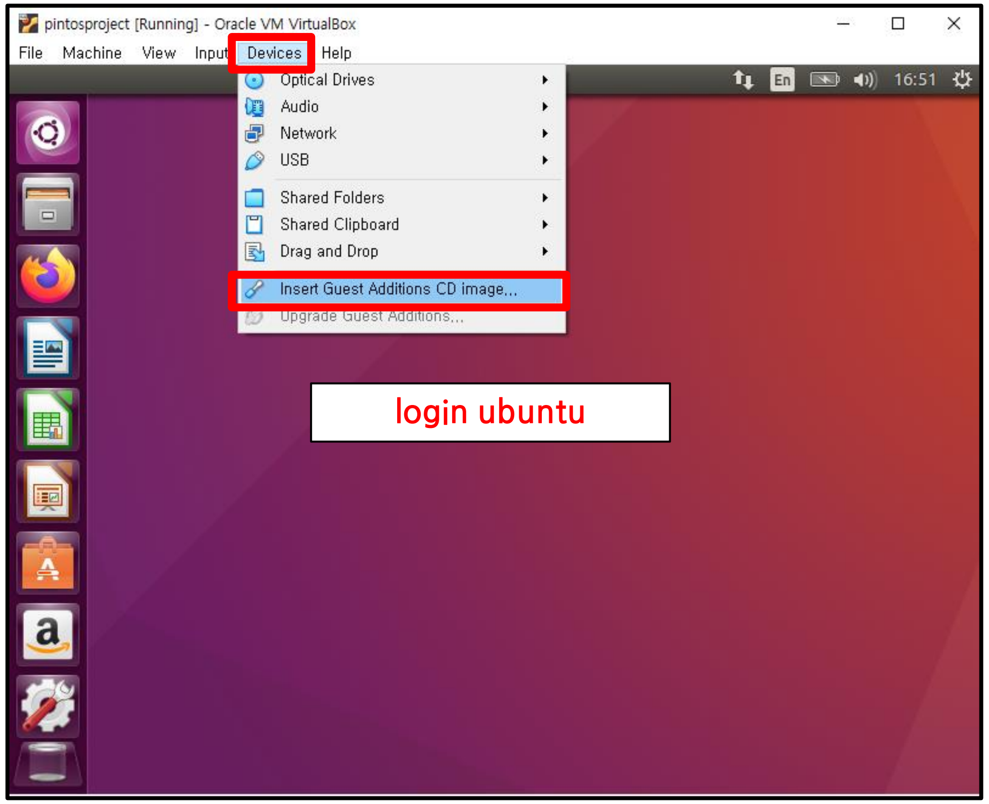
Display

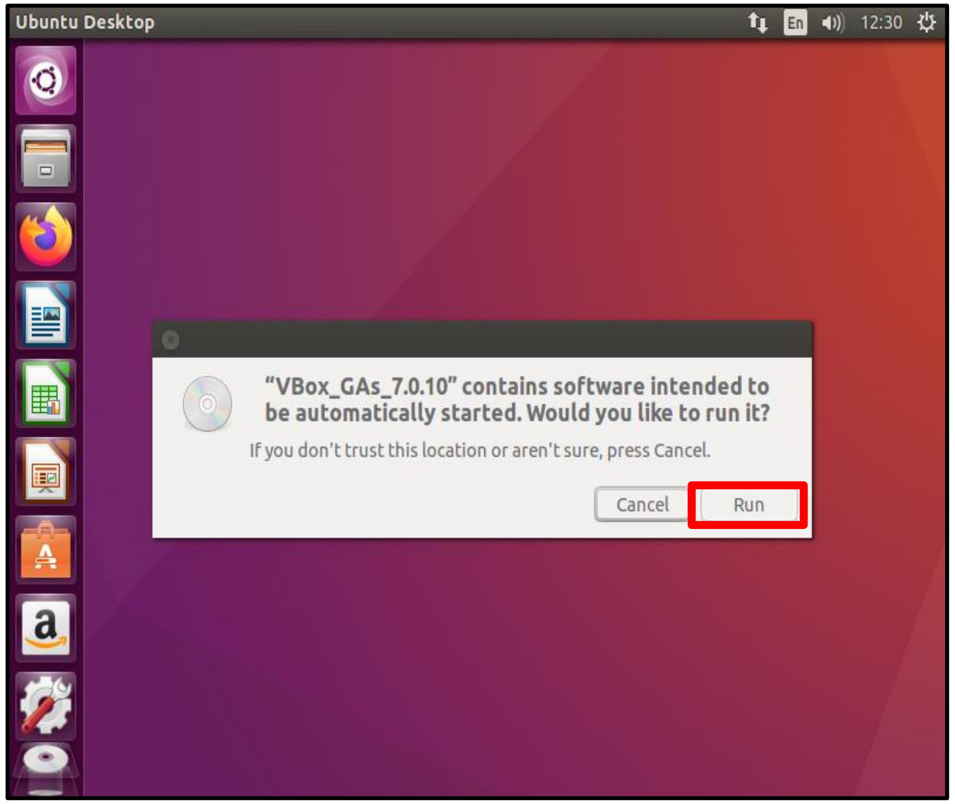
Video Memory: 128 MB
Monitor Count: 1
Scale Factor: All Monitors 100%
Graphics Controller: VBoxVGA
Extended Features: ☐ Enable 3D Acceleration

Change this option to VBoxVGA

Invalid settings detected

Enabling additional options ④



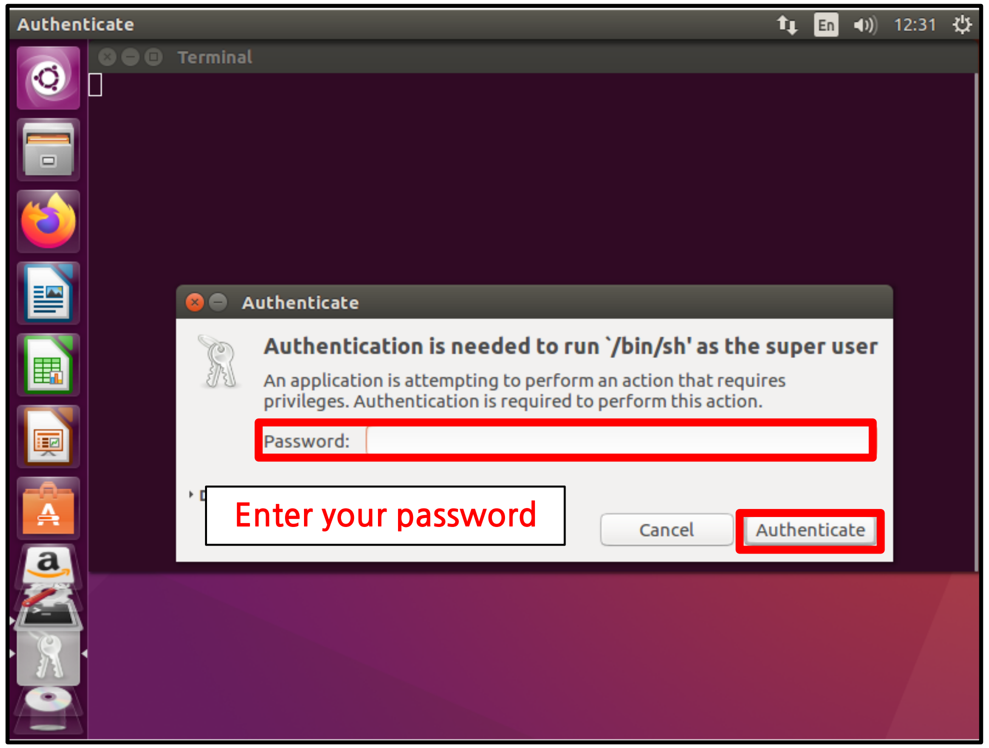


"VBox_GAs_7.0.10" contains software intended to be automatically started. Would you like to run it?

If you don't trust this location or aren't sure, press Cancel.

Computer Systems Security Laboratory

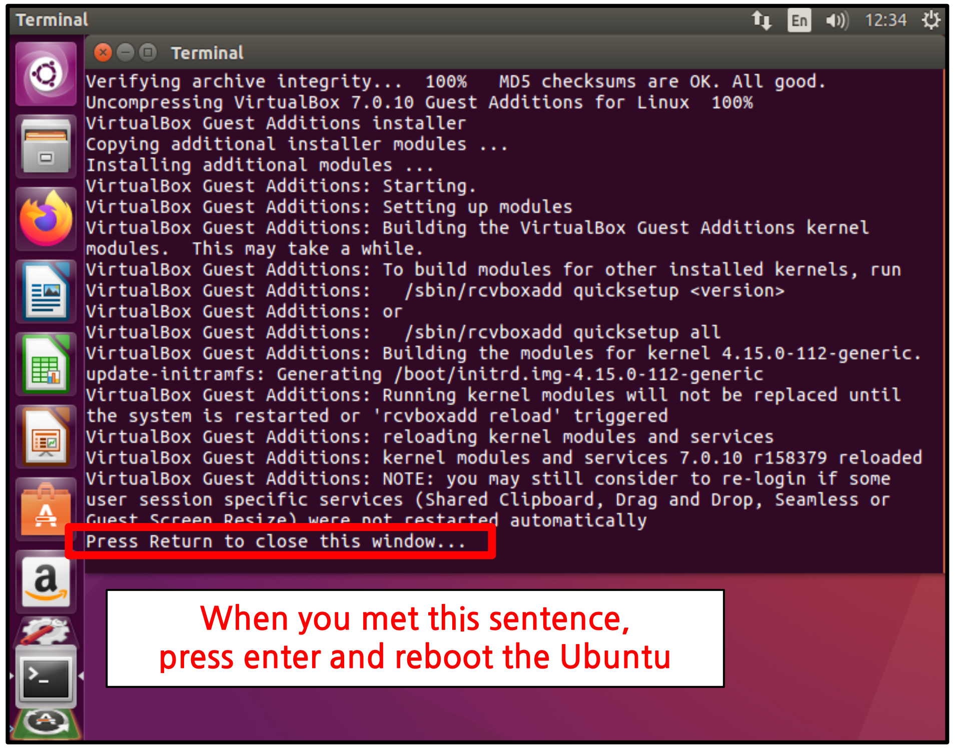
Enabling additional options ⑤



Authentication is needed to run `/bin/sh' as the super user

An application is attempting to perform an action that requires privileges. Authentication is required to perform this action.

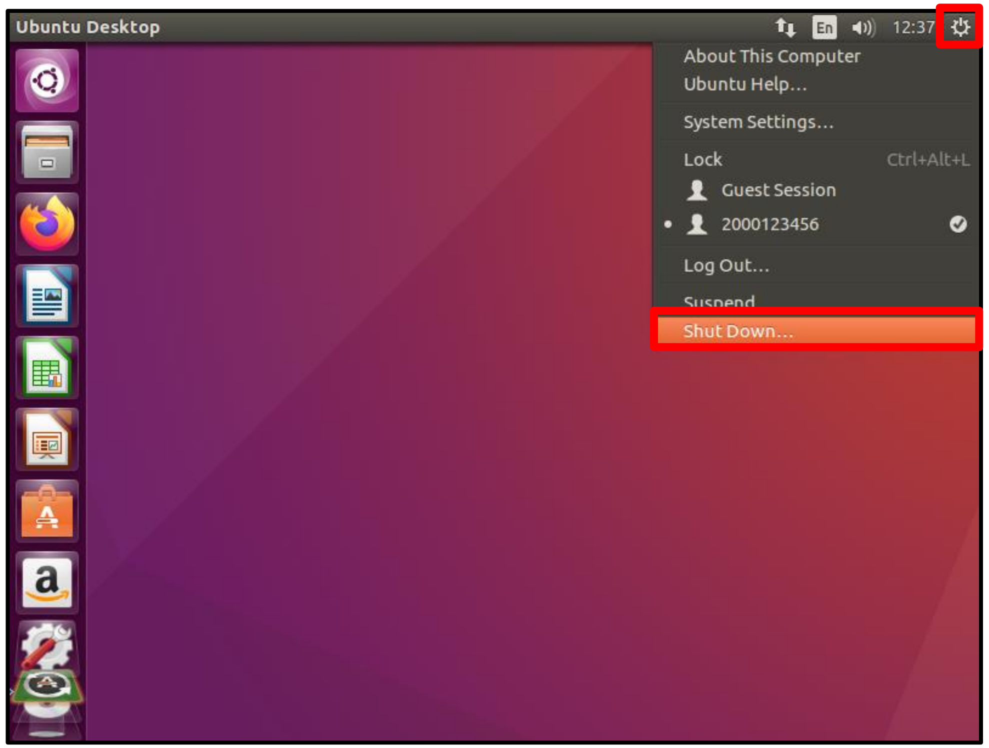
Enter your password

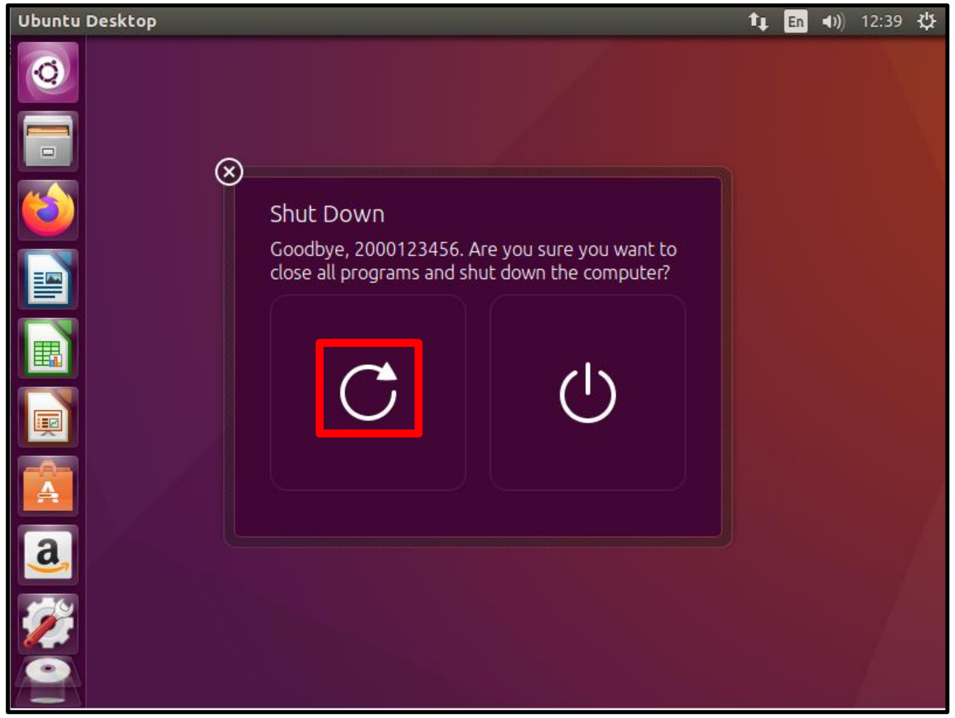


```
Verifying archive integrity...  100%   MD5 checksums are OK. All good.
Uncompressing VirtualBox 7.0.10 Guest Additions for Linux  100%
VirtualBox Guest Additions installer
Copying additional installer modules ...
Installing additional modules ...
VirtualBox Guest Additions: Starting.
VirtualBox Guest Additions: Setting up modules
VirtualBox Guest Additions: Building the VirtualBox Guest Additions kernel
modules.  This may take a while.
VirtualBox Guest Additions: To build modules for other installed kernels, run
VirtualBox Guest Additions:   /sbin/rcvboxadd quicksetup <version>
VirtualBox Guest Additions: or
VirtualBox Guest Additions:   /sbin/rcvboxadd quicksetup all
VirtualBox Guest Additions: Building the modules for kernel 4.15.0-112-generic.
update-initramfs: Generating /boot/initrd.img-4.15.0-112-generic
VirtualBox Guest Additions: Running kernel modules will not be replaced until
the system is restarted or 'rcvboxadd reload' triggered
VirtualBox Guest Additions: reloading kernel modules and services
VirtualBox Guest Additions: kernel modules and services 7.0.10 r158379 reloaded
VirtualBox Guest Additions: NOTE: you may still consider to re-login if some
user session specific services (Shared Clipboard, Drag and Drop, Seamless or
Guest Screen Resize) were not restarted automatically
Press Return to close this window...
```

When you met this sentence,
press enter and reboot the Ubuntu

Enabling additional options ⑥





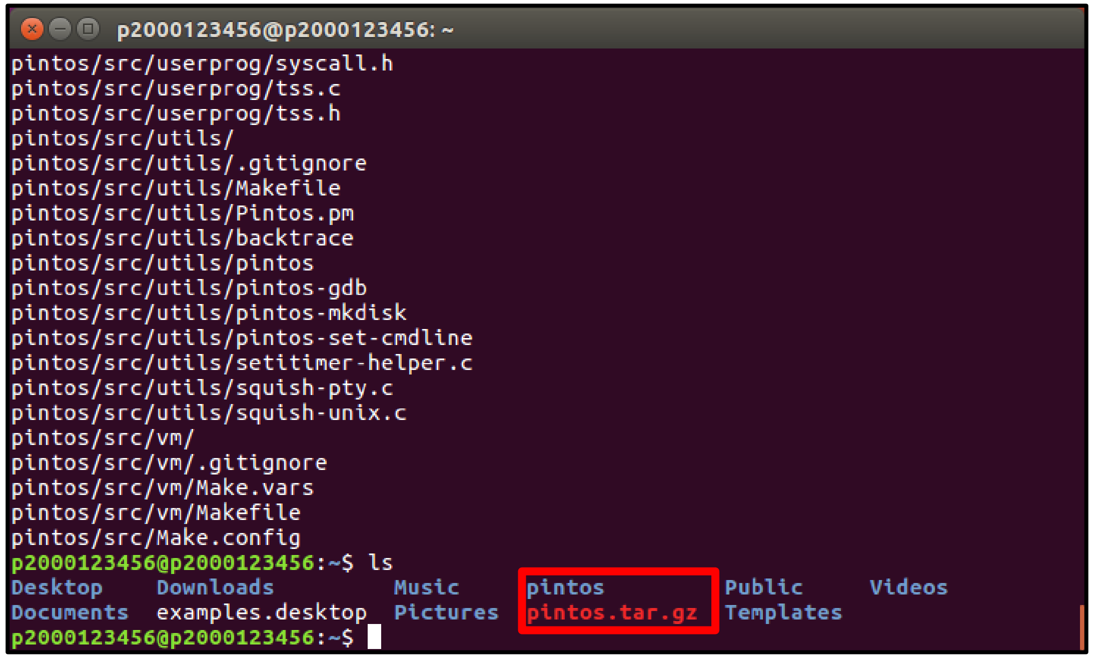
Shut Down

Goodbye, 2000123456. Are you sure you want to close all programs and shut down the computer?

4. Install Pintos

Install packages related to Pintos

```
$ sudo apt update

$ sudo apt install qemu libc6-dev g++ gcc

$ sudo ln -s /usr/bin/qemu-system-i386 /usr/bin/qemu

$ wget https://web.stanford.edu/class/cs140/projects/pintos/pintos.tar.gz

$ tar -xvf pintos.tar.gz
```



```
p2000123456@p2000123456: ~
pintos/src/userprog/syscall.h
pintos/src/userprog/tss.c
pintos/src/userprog/tss.h
pintos/src/utils/
pintos/src/utils/.gitignore
pintos/src/utils/Makefile
pintos/src/utils/Pintos.pm
pintos/src/utils/backtrace
pintos/src/utils/pintos
pintos/src/utils/pintos-gdb
pintos/src/utils/pintos-mkdisk
pintos/src/utils/pintos-set-cmdline
pintos/src/utils/setitimer-helper.c
pintos/src/utils/squish-pty.c
pintos/src/utils/squish-unix.c
pintos/src/vm/
pintos/src/vm/.gitignore
pintos/src/vm/Make.vars
pintos/src/vm/Makefile
pintos/src/Make.config
p2000123456@p2000123456:~$ ls
Desktop    Downloads         Music     pintos         Public     Videos
Documents  examples.desktop  Pictures  pintos.tar.gz  Templates
p2000123456@p2000123456:~$
```

Computer Systems Security Laboratory

Modify several configuration files

① pintos/src/threads/Make.vars
SIMULATOR = —bochs → SIMULATOR = --qemu

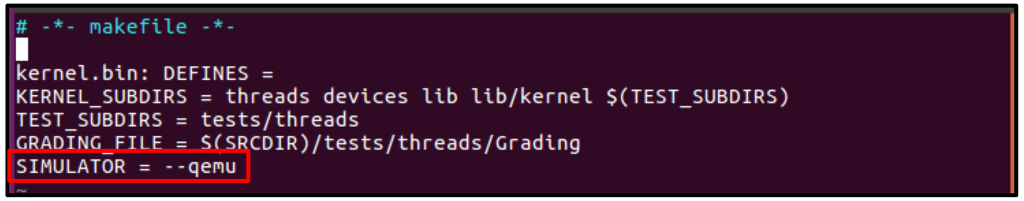
② pintos/src/utils/pintos (line 103)
($sim = "qemu" if !defined $sim;)

③ pintos/src/utils/pintos (line 259)
('your directory/pintos/src/threads/build/kernel.bin');

④ pintos/src/utils/Pintos.pm (line 362)
("your directory/pintos/src/threads/build/loader.bin")

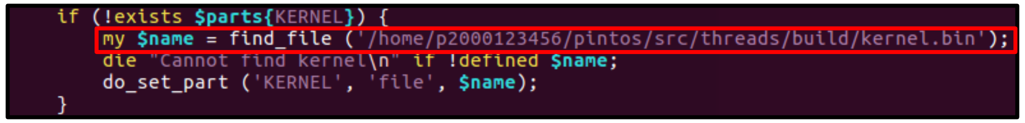
⑤ pintos/src/device/shutdown.c (line 100)
outw(0x604, 0x0|0x2000); <- Add on the 100th line

⑥ pintos/src/utils/Makefile
"LDFLAGS = -lm" → "LDLIBS = -lm"



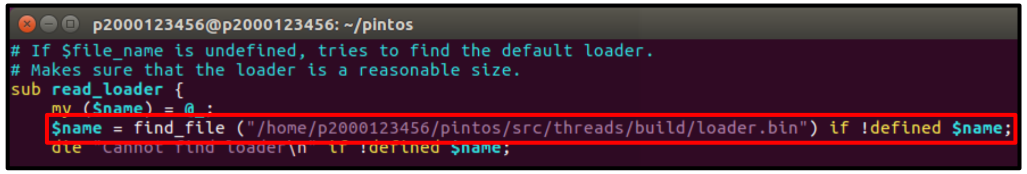
```
# -*- makefile -*-

kernel.bin: DEFINES =
KERNEL_SUBDIRS = threads devices lib lib/kernel $(TEST_SUBDIRS)
TEST_SUBDIRS = tests/threads
GRADING_FILE = $(SRCDIR)/tests/threads/Grading
SIMULATOR = --qemu
```

①



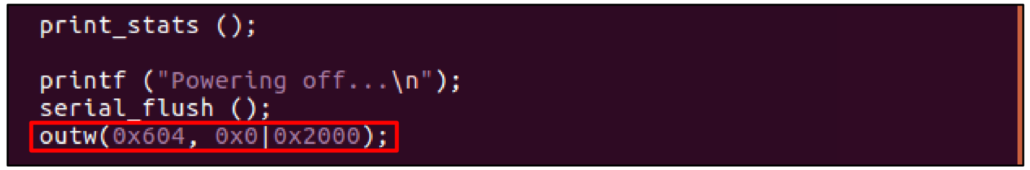
```
    if (!exists $parts{KERNEL}) {
        my $name = find_file ('/home/p2000123456/pintos/src/threads/build/kernel.bin');
        die "Cannot find kernel\n" if !defined $name;
        do_set_part ('KERNEL', 'file', $name);
    }
```

③



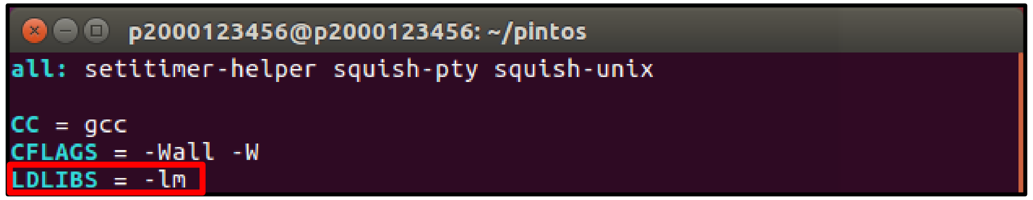
```
p2000123456@p2000123456: ~/pintos
# If $file_name is undefined, tries to find the default loader.
# Makes sure that the loader is a reasonable size.
sub read_loader {
    my ($name) = @_;
    $name = find_file ("/home/p2000123456/pintos/src/threads/build/loader.bin") if !defined $name;
    die "Cannot find loader\n" if !defined $name;
```

④



```
  print_stats ();

  printf ("Powering off...\n");
  serial_flush ();
  outw(0x604, 0x0|0x2000);
```

⑤



```
p2000123456@p2000123456: ~/pintos
all: setitimer-helper squish-pty squish-unix

CC = gcc
CFLAGS = -Wall -W
LDLIBS = -lm
```

⑥

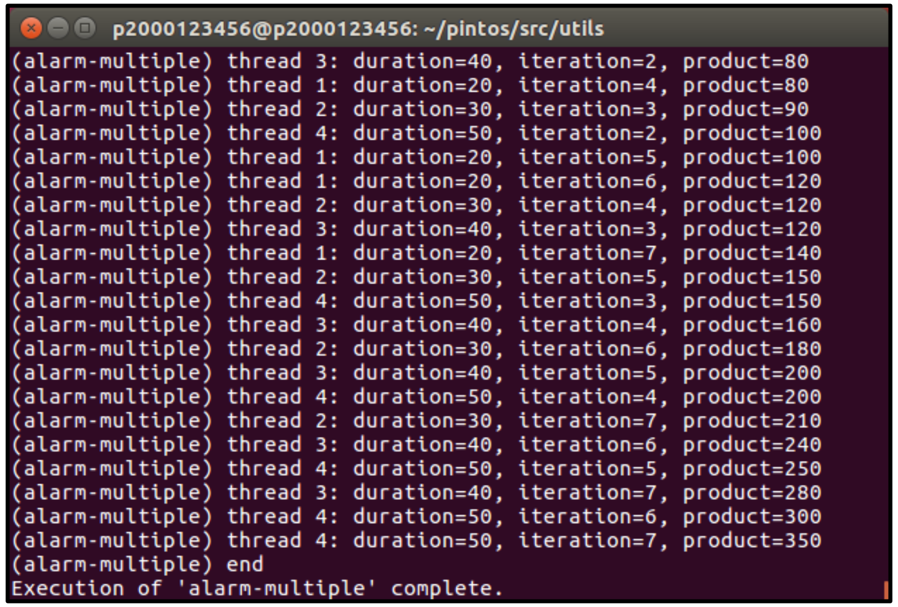
Build Pintos & set environment variable

Build threads, utils and set environment variable

```
$ cd pintos/src/threads

$ make

$ cd pintos/src/utils

$ echo "export PATH=\"\$PATH:~/pintos/src/utils\"" >>
~/.bashrc

$ source ~/.bashrc
```

Run pintos

```
$ pintos -q run alarm-multiple
```



```
p2000123456@p2000123456: ~/pintos/src/utils
(alarm-multiple) thread 3: duration=40, iteration=2, product=80
(alarm-multiple) thread 1: duration=20, iteration=4, product=80
(alarm-multiple) thread 2: duration=30, iteration=3, product=90
(alarm-multiple) thread 4: duration=50, iteration=2, product=100
(alarm-multiple) thread 1: duration=20, iteration=5, product=100
(alarm-multiple) thread 1: duration=20, iteration=6, product=120
(alarm-multiple) thread 2: duration=30, iteration=4, product=120
(alarm-multiple) thread 3: duration=40, iteration=3, product=120
(alarm-multiple) thread 1: duration=20, iteration=7, product=140
(alarm-multiple) thread 2: duration=30, iteration=5, product=150
(alarm-multiple) thread 4: duration=50, iteration=3, product=150
(alarm-multiple) thread 3: duration=40, iteration=4, product=160
(alarm-multiple) thread 2: duration=30, iteration=6, product=180
(alarm-multiple) thread 3: duration=40, iteration=5, product=200
(alarm-multiple) thread 4: duration=50, iteration=4, product=200
(alarm-multiple) thread 2: duration=30, iteration=7, product=210
(alarm-multiple) thread 3: duration=40, iteration=6, product=240
(alarm-multiple) thread 4: duration=50, iteration=5, product=250
(alarm-multiple) thread 3: duration=40, iteration=7, product=280
(alarm-multiple) thread 4: duration=50, iteration=6, product=300
(alarm-multiple) thread 4: duration=50, iteration=7, product=350
(alarm-multiple) end
Execution of 'alarm-multiple' complete.
```

Result of “$ pintos -q run alarm-multiple”

Thank you

:-)