

# **System Administration 101 (Linux)**

University of Stavanger

# Goal

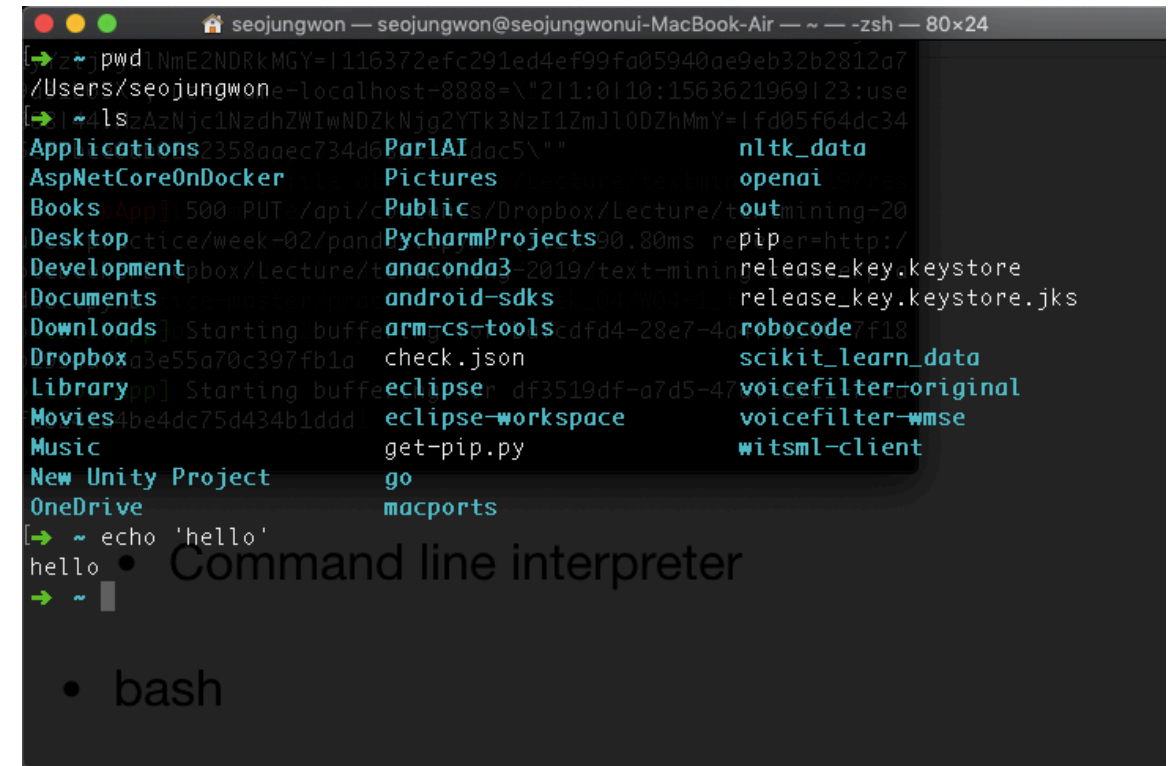
- Reviewing basic terms in the Linux environment
- Reviewing the most common Linux commands
- Looking at some git commands
- Tasks

# Terms

- shell
- terminal
- bash
- server
- host
- port
- directory
- folder
- file
- path
- resource
- memory
- cpu
- disk
- core
- permission
- root
- sudo

# Shell, Terminal, Bash

- **Terminal**
  - Text input/output environment
- **Shell**
  - Command line interpreter
- **Bash**
  - One types of shell
  - The most common shell in modern linux systems



The screenshot shows a terminal window titled 'sejungwon — sejungwon@sejungwonui-MacBook-Air — ~ — zsh — 80x24'. The user has executed 'pwd' and 'ls' commands. The 'ls' output is color-coded and shows a standard macOS file structure including Applications, Books, Desktop, Downloads, Dropbox, Library, Movies, Music, New Unity Project, and OneDrive. Below the directory listing, the user has executed 'echo 'hello'', which outputs 'hello'.

```
sejungwon — sejungwon@sejungwonui-MacBook-Air — ~ — zsh — 80x24
[➜] ~ pwd
/Users/sejungwon
[➜] ~ ls
Applications  Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
Books         Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
Desktop       Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
Downloads     Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
Dropbox       Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
Library       Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
Movies        Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
Music         Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
New Unity Project Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
OneDrive      Desktop  Downloads  Library  Movies  Music  New Unity Project  OneDrive
[➜] ~ echo 'hello'
hello
[➜] ~
```

e.g.,  
***“Open your terminal,  
and generate the file using a bash script”***

# Server, Host, Port

- ‘Server’ and ‘Host’ are often used interchangeably.
- **Host** is a computer or another device that connects to the network while a **server** is a software or a hardware device that provides services to other programs or devices in the network.
  - e.g., *“A web host (or web hosting company) maintains multiple web servers and provides web hosting services for clients.”*
- Port
  - A port is an **addressable network location** implemented in an operating system to help differentiate traffic destined for different services or applications.

\* source: <https://techterms.com/definition/host>

\*\* source: <https://pediaa.com/difference-between-host-and-server/>

\*\*\* source: <http://www.linuxandubuntu.com/home/what-are-ports-how-to-find-open-ports-in-linux>

# Path, Directory, Folder, File

- **Path:** the general form of the name of a file or directory, specifies a unique location in a file system.
  - **Absolute path:** full path from root directory  
e.g., : C:\Users\Desktop, /home/User1
  - **Relative path:** way to specify the directory from another  
e.g., : ../, ../../, ../Users
- **Directory = Folder**
  - A directory is a **file system cataloging structure** which contains references to other computer files, and possibly other directories.
- **File:** *“In Linux, everything is a File”*
  - A file is an object on a computer that stores data, information, settings, or commands used with a computer program.

\* source: <https://www.tecmint.com/explanation-of-everything-is-a-file-and-types-of-files-in-linux/>

\*\* source: <https://www.computerhope.com/jargon/f/file.htm>

# Resource, memory, disk, core

- **Resource** is any physical or virtual component of limited availability within a computer system.
- **Memory** mainly means the size of RAM.
- **Disk** means disk space (HDD, SSD).
- **Core** means each processor in CPU.

# Permission, Root, Sudo

- Each file and process is owned by different users.
- We do not have '**permission**' to manipulate others'.
- However, the **root** account has permission for everything.
- When we run a command as a root account in Linux, we start the command with **sudo**.



# Linux commands

- File and directory commands
- Archives related commands
- Network related commands
- Other useful commands

# File and directory commands

- **cd**: move to the certain directory
- **pwd**: show the current path
- **ls**: display the files in the specified directory
- **touch**: create a new file
- **rm**: remove the file
- **cp**: copy the file to others
- **mv**: move (rename) the file.
- **cat**: display the contents of file
- **less**: browse through a text file
- **head**: display the first n-lines of a file
- **tail**: display the last n-lines of a file
- **find**: find files in the specified directory

# Archive related commands

- **tar**
  - **(compress)** tar cf archive.tar directory
  - **(extract)** tar xf archive.tar
  - **(compress)** tar czf archive.tar.gz directory
  - **(extract)** tar xzf archive.tar.gz
- **(compress)** tar cjf archive.tar.bz2 directory
- **(extract)** tar xjf archive.tar.bz2
- **zip**
  - **(compress)** zip archive.zip directory
  - **(extract)** unzip archive.zip

# Network related commands

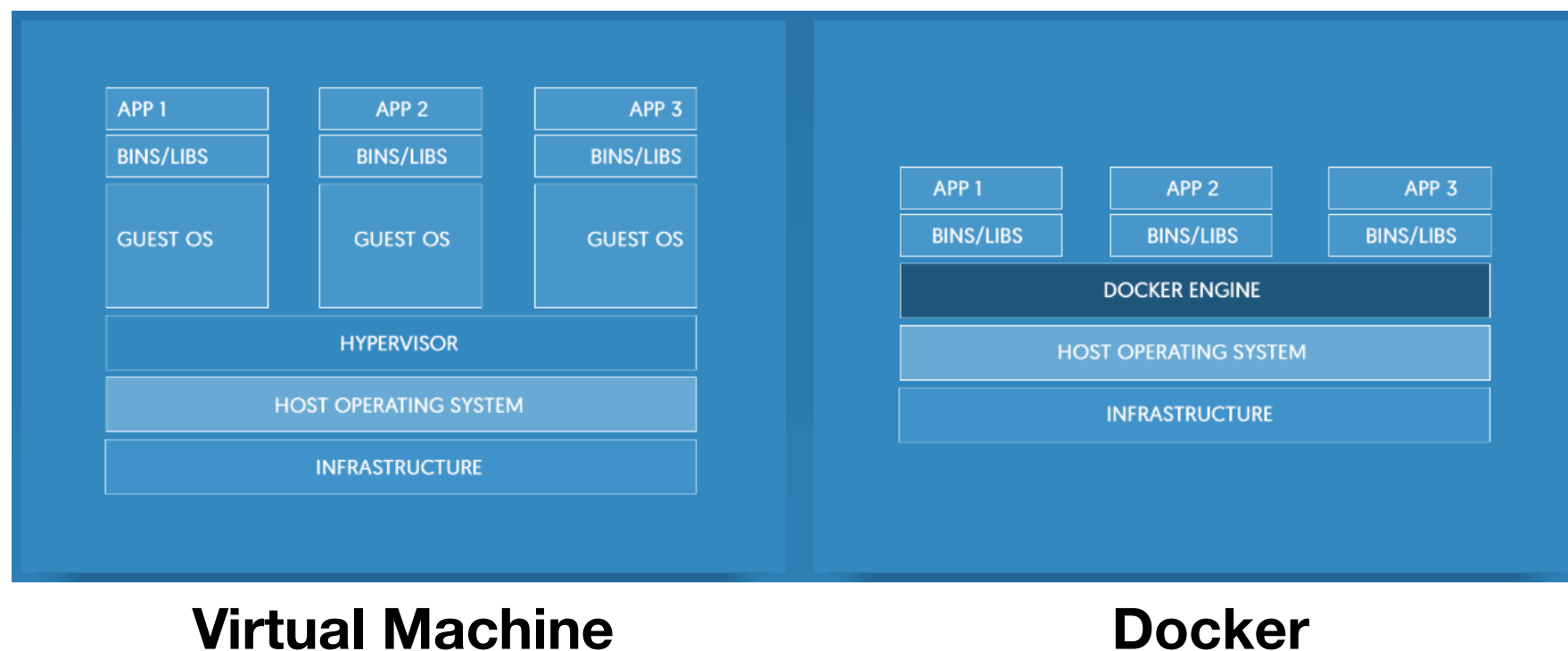
- **ssh:** connect to host using port
  - `ssh -p port user@host`
- **scp:** secure copy to/from host from/to local
  - (to server) `scp file host:/path`
  - (from server) `scp host:/path/file /localpath`
- **wget / curl:** download the file
  - `wget host/file.txt`
  - `curl host/file.txt --output some.file`
- **ping:** send echo request to host
  - `ping www.google.com`

# Other useful commands

- **top (htop)**: display and manage the top processes
- **df -h (-i)**: show free and used space (inodes) on mounted filesystems
- **kill pid**: kill process with process ID of pid
- **chmod**: change the permission of a file
- **chown**: change the ownership of a file
- **man command**: show documentation of the given command

# Virtualization

- How can we set up linux environment in our local machine? (when we use Windows or Mac)
  - Option #1: VirtualBox or VMware
  - Option #2: Docker



# Tasks

# Git demo

- Creating a new GitHub repo
- Initialising a git repo
- Making changes
- Pushing changes
- Pulling changes
- Branching