Institute of Electronics
National Yang Ming Chiao Tung University
Hsinchu, Taiwan

AI Training Course Series

Introduction to Linux

Lecture 0-1

Architecture
Development &
Algorithm
Refinement for
Intelligent Computing

Student: Meng-Hsun Hsieh, Jye-En Wu
Advisor: Juinn-Dar Huang, Ph.D.

June 15, 2024

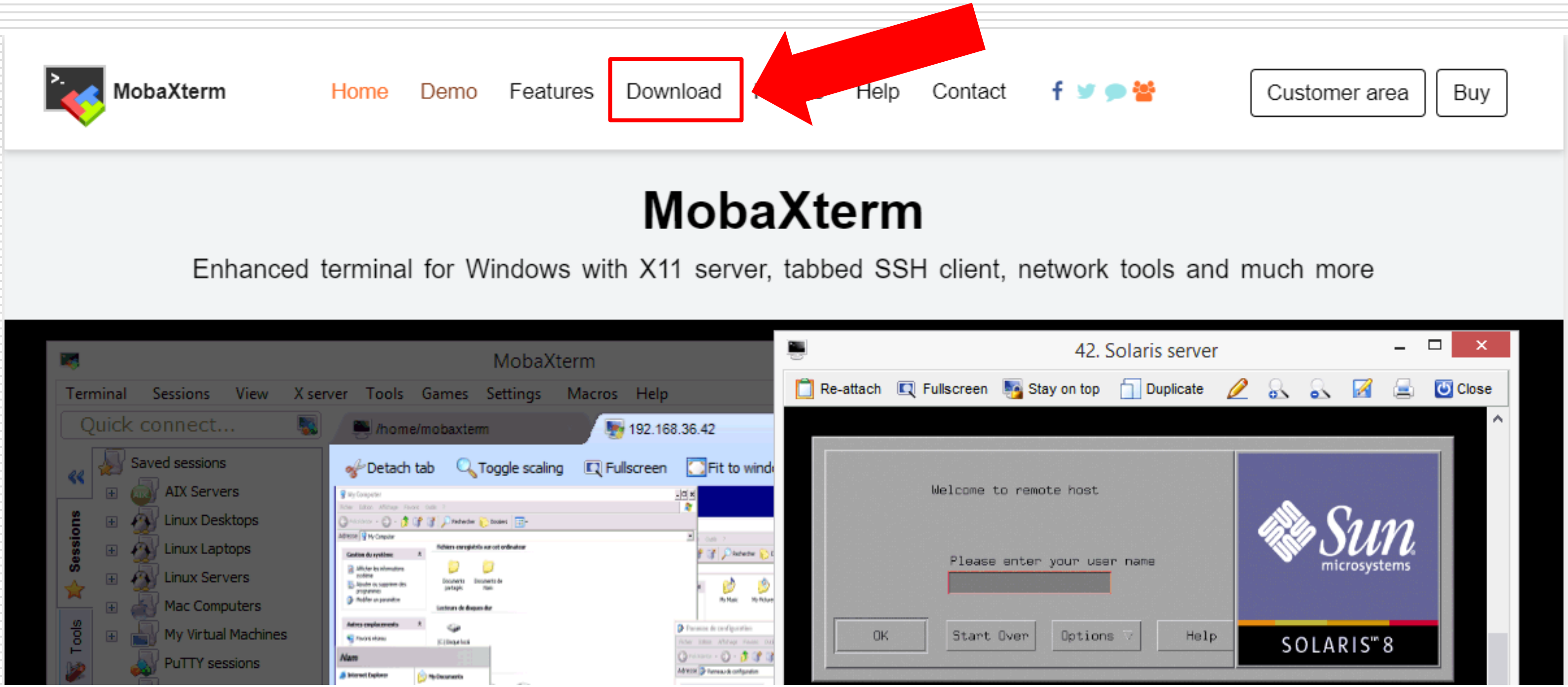
Outline

- Install MobaXterm
- Install Anaconda
- Install PyTorch
- Introduction to Linux
- Basic Linux Commands
- Advanced Linux Commands
- References

Install MobaXterm

Install MobaXterm (1/3)

- MobaXterm:
  - Enhanced terminal for Windows with X11 server, tabbed SSH client, network tools and much more
- Download from MobaXterm official website
  - https://mobaxterm.mobatek.net/



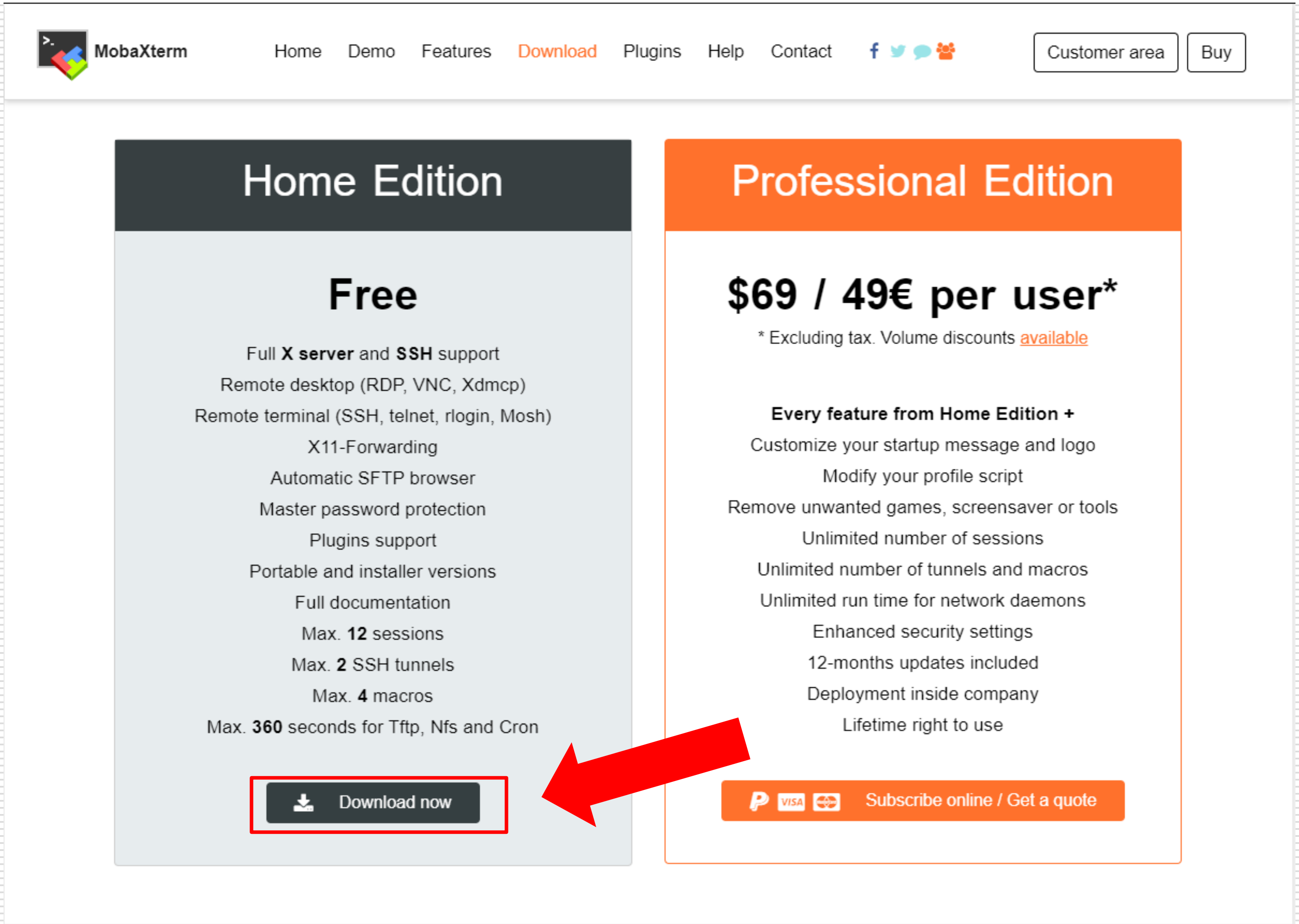
MobaXterm

Enhanced terminal for Windows with X11 server, tabbed SSH client, network tools and much more

```
Welcome to remote host

Please enter your user name
```

Install MobaXterm (2/3)

- Download Home Edition



Home Edition

Free

Full **X server** and **SSH** support
Remote desktop (RDP, VNC, Xdmcp)
Remote terminal (SSH, telnet, rlogin, Mosh)
X11-Forwarding
Automatic SFTP browser
Master password protection
Plugins support
Portable and installer versions
Full documentation
Max. **12** sessions
Max. **2** SSH tunnels
Max. **4** macros
Max. **360** seconds for Tftp, Nfs and Cron

Download now

Professional Edition

$69 / 49€ per user*

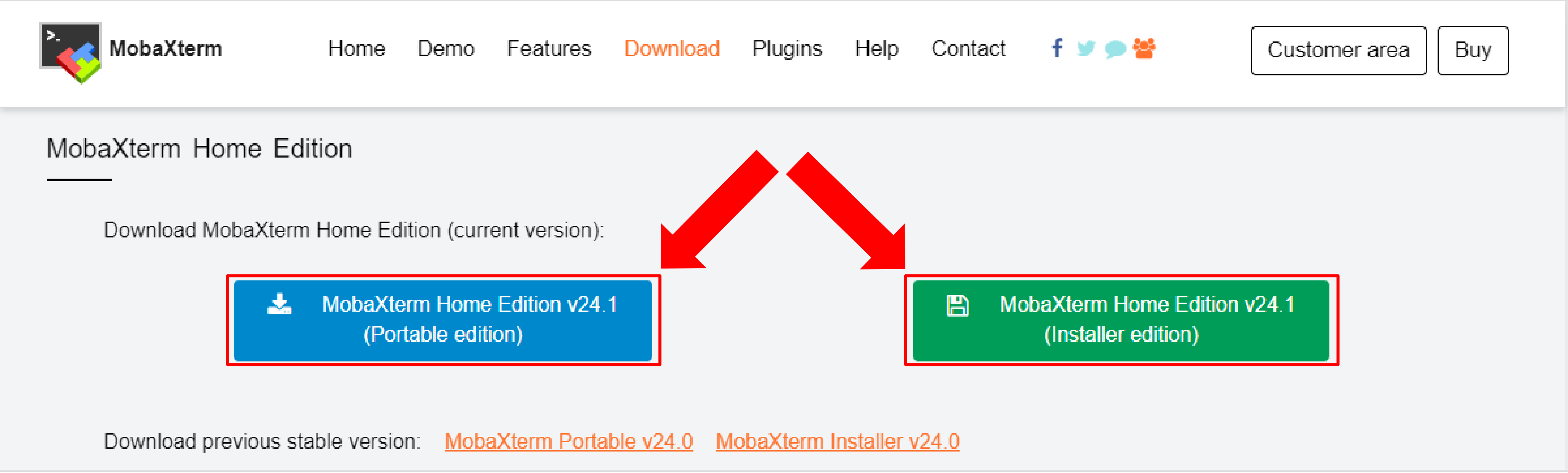
* Excluding tax. Volume discounts available

Every feature from Home Edition +
Customize your startup message and logo
Modify your profile script
Remove unwanted games, screensaver or tools
Unlimited number of sessions
Unlimited number of tunnels and macros
Unlimited run time for network daemons
Enhanced security settings
12-months updates included
Deployment inside company
Lifetime right to use

Subscribe online / Get a quote

Install MobaXterm (3/3)

- Choose Portable edition or installer edition according to your preferences



MobaXterm
Home
Demo
Features
Download
Plugins
Help
Contact
Customer area
Buy

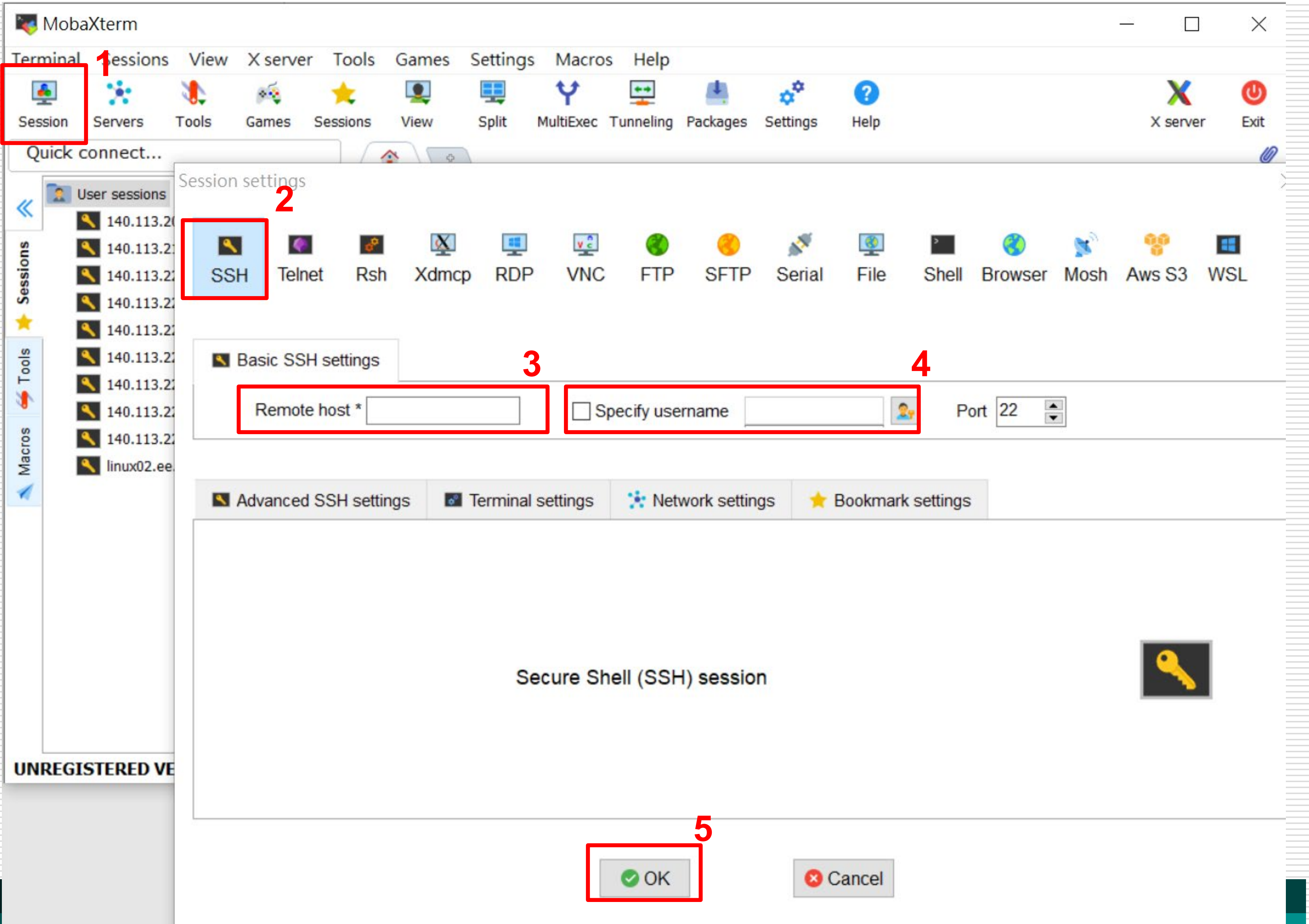
MobaXterm Home Edition

Download MobaXterm Home Edition (current version):

MobaXterm Home Edition v24.1 (Portable edition)
MobaXterm Home Edition v24.1 (Installer edition)

Download previous stable version: MobaXterm Portable v24.0 MobaXterm Installer v24.0

Connect to Server with MobaXterm (1/3)



Remote host *

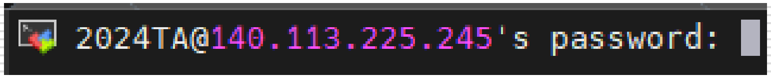
□ Specify username

Port 22

Secure Shell (SSH) session

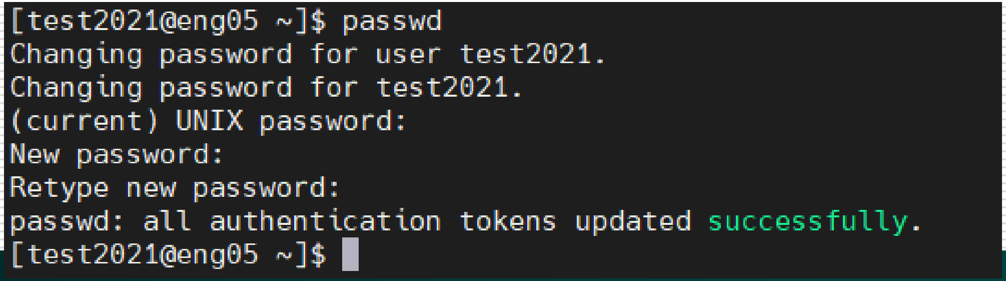
Connect to Server with MobaXterm (2/3)

- Server IP (Remote host) :
  - 140.113.225.240 (adar10, with GPUs)
  - 140.113.225.241 (eng01)
  - 140.113.225.242 (eng02, main data storage)
  - 140.113.225.243 (eng03)
  - 140.113.225.244 (eng04, with GPUs)
  - 140.113.225.245 (eng05, with GPU)
- Undergraduate students can only use eng05
- Accounts (username) are announced in Facebook group
- Default password is “adar”



```
2024TA@140.113.225.245's password:
```

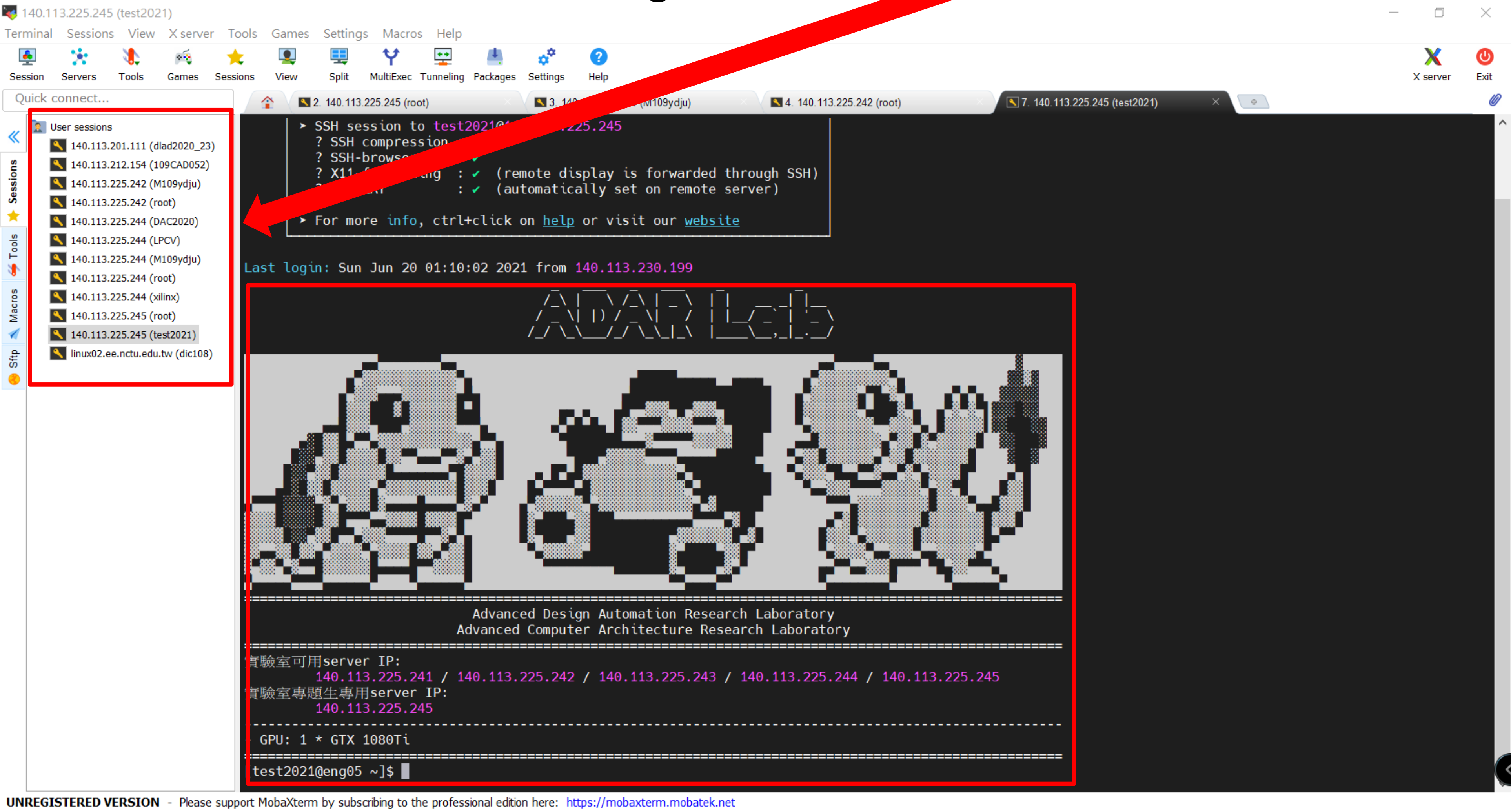
- Remember to change password after first time login
  - $ ***passwd***



```
[test2021@eng05 ~]$ passwd
Changing password for user test2021.
Changing password for test2021.
(current) UNIX password:
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[test2021@eng05 ~]$
```

Connect to Server with MobaXterm (3/3)

- MobaXterm will save your sessions in the tab
- You will see this after login


140.113.225.245 (test2021)
Terminal Sessions View X server Tools Games Settings Macros Help
Session
Servers
Tools
Games
Sessions
View
Split
MultiExec
Tunneling
Packages
Settings
Help
X server
Exit
Quick connect...
2. 140.113.225.245 (root)
3. 140 (M109ydju)
4. 140.113.225.242 (root)
7. 140.113.225.245 (test2021)
Sessions
Tools
Macros
Sftp

User sessions
- 140.113.201.111 (dlad2020_23)
- 140.113.212.154 (109CAD052)
- 140.113.225.242 (M109ydju)
- 140.113.225.242 (root)
- 140.113.225.244 (DAC2020)
- 140.113.225.244 (LPCV)
- 140.113.225.244 (M109ydju)
- 140.113.225.244 (root)
- 140.113.225.244 (xilinx)
- 140.113.225.245 (root)
- 140.113.225.245 (test2021)
- linux02.ee.nctu.edu.tw (dic108)

```
 ➤ SSH session to test2021@      .225.245
   ? SSH compression
   ? SSH-browse
   ? X11-f       ing  : ✔  (remote display is forwarded through SSH)
                      : ✔  (automatically set on remote server)

 ➤ For more info, ctrl+click on help or visit our website

Last login: Sun Jun 20 01:10:02 2021 from 140.113.230.199
```

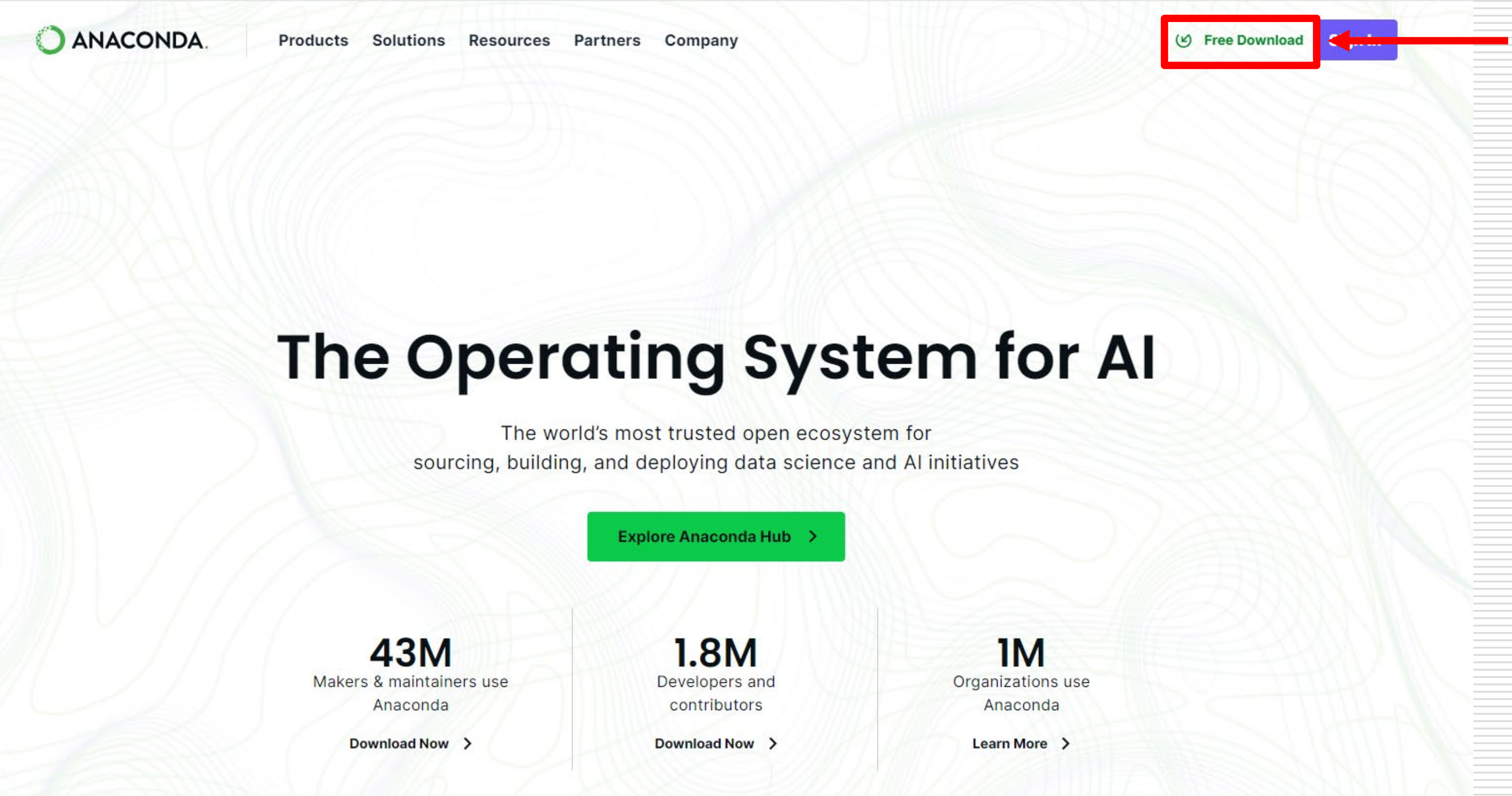
```
                         Advanced Design Automation Research Laboratory
                        Advanced Computer Architecture Research Laboratory
======================================================================================================
實驗室可用server IP:
         140.113.225.241 / 140.113.225.242 / 140.113.225.243 / 140.113.225.244 / 140.113.225.245
實驗室專題生專用server IP:
         140.113.225.245
------------------------------------------------------------------------------------------------------
  GPU: 1 * GTX 1080Ti
======================================================================================================
[test2021@eng05 ~]$
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

Install Anaconda

Get Latest Version of Anaconda (1/3)

- Get Anaconda installer archive download URL from Anaconda official website
  - https://www.anaconda.com/



ANACONDA. Products Solutions Resources Partners Company Free Download

The Operating System for AI

The world's most trusted open ecosystem for sourcing, building, and deploying data science and AI initiatives

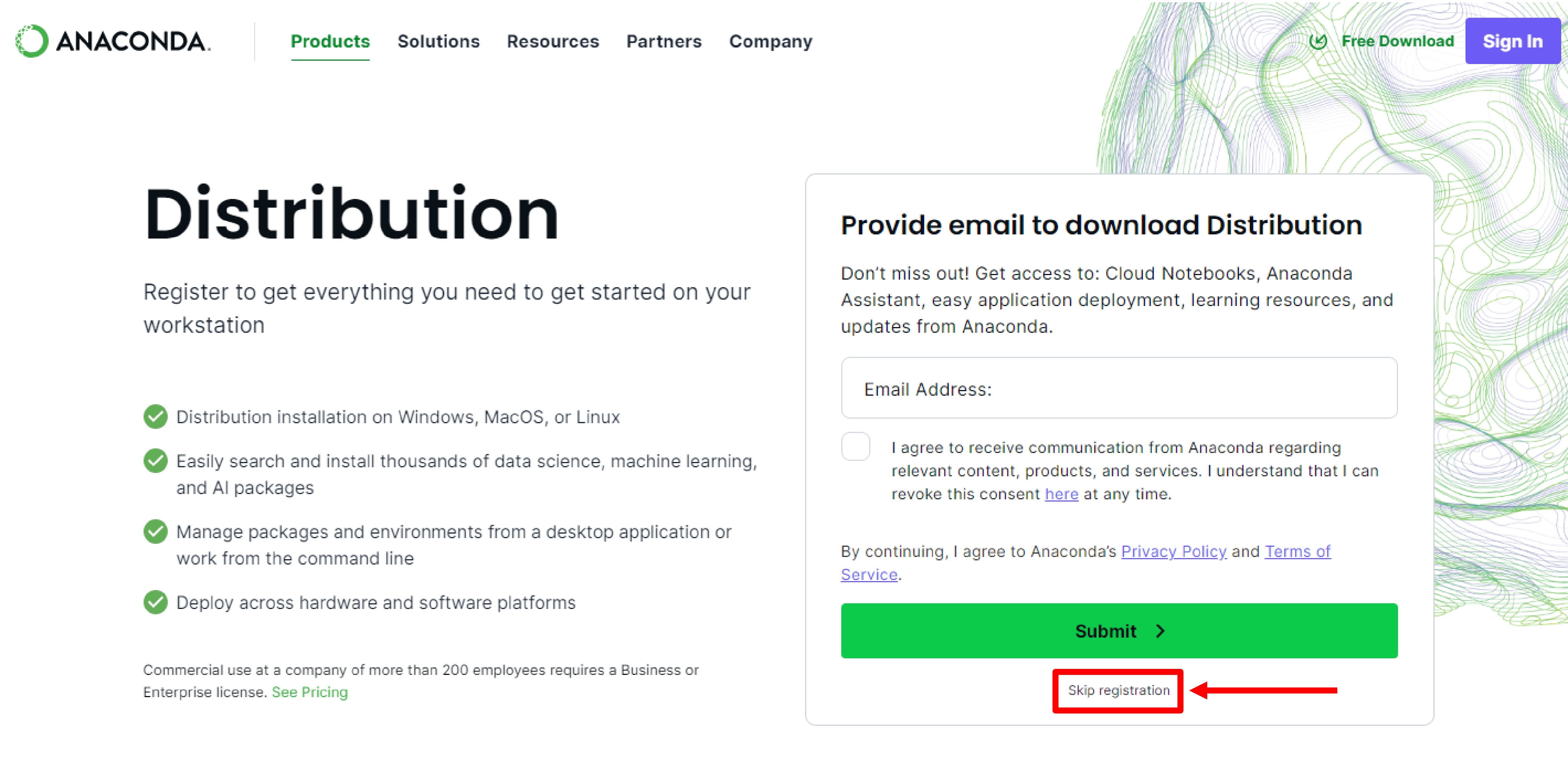
Explore Anaconda Hub >

43M
Makers & maintainers use Anaconda
Download Now >

1.8M
Developers and contributors
Download Now >

1M
Organizations use Anaconda
Learn More >

Get Latest Version of Anaconda (2/3)



Distribution

Register to get everything you need to get started on your workstation

- ✓ Distribution installation on Windows, MacOS, or Linux
- ✓ Easily search and install thousands of data science, machine learning, and AI packages
- ✓ Manage packages and environments from a desktop application or work from the command line
- ✓ Deploy across hardware and software platforms

Commercial use at a company of more than 200 employees requires a Business or Enterprise license. See Pricing

Provide email to download Distribution

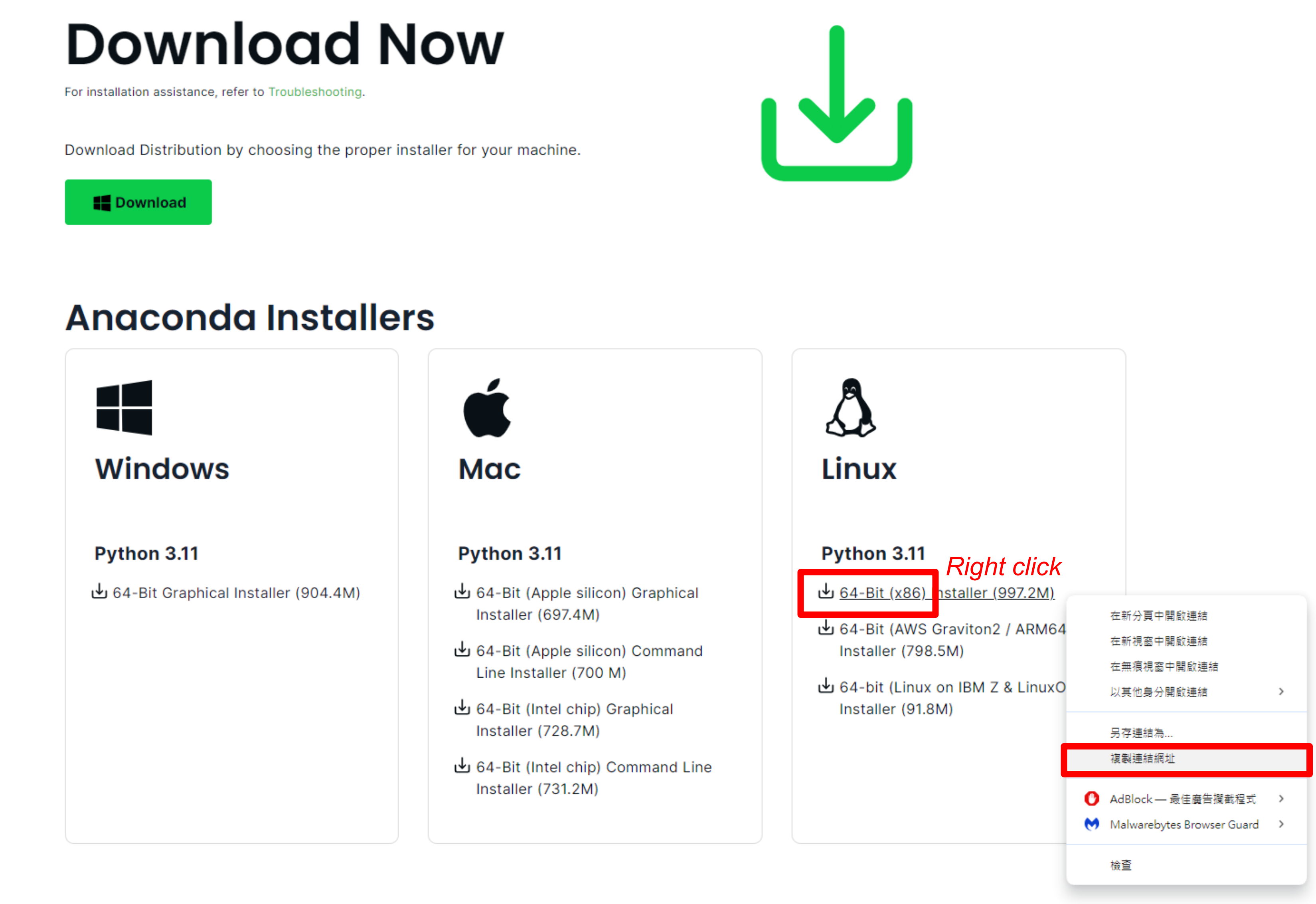
Don't miss out! Get access to: Cloud Notebooks, Anaconda Assistant, easy application deployment, learning resources, and updates from Anaconda.

Email Address:

□ I agree to receive communication from Anaconda regarding relevant content, products, and services. I understand that I can revoke this consent here at any time.

By continuing, I agree to Anaconda's Privacy Policy and Terms of Service.

Get Latest Version of Anaconda (3/3)



Download Now

For installation assistance, refer to Troubleshooting.

Download Distribution by choosing the proper installer for your machine.

Download

Anaconda Installers

Windows

Python 3.11

- 64-Bit Graphical Installer (904.4M)

Mac

Python 3.11

- 64-Bit (Apple silicon) Graphical Installer (697.4M)
- 64-Bit (Apple silicon) Command Line Installer (700 M)
- 64-Bit (Intel chip) Graphical Installer (728.7M)
- 64-Bit (Intel chip) Command Line Installer (731.2M)

Linux

Python 3.11

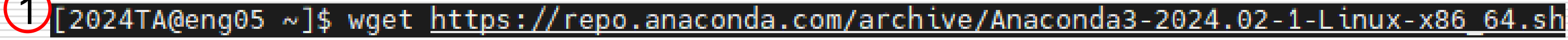
- 64-Bit (x86) Installer (997.2M)
- 64-Bit (AWS Graviton2 / ARM64 Installer (798.5M)
- 64-bit (Linux on IBM Z & LinuxO Installer (91.8M)

Right click

Install Anaconda (1/2)

- Download Installer
  - 1. $ ***wget [URL you copied]***

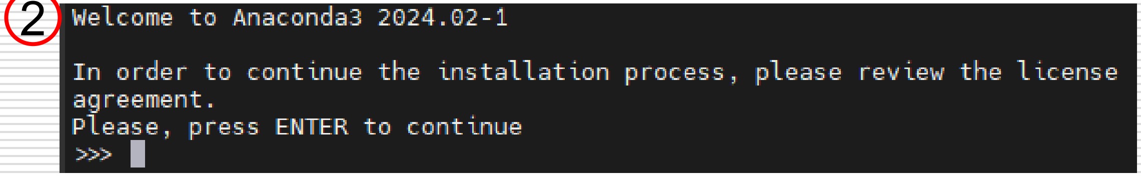
1



```
[2024TA@eng05 ~]$ wget https://repo.anaconda.com/archive/Anaconda3-2024.02-1-Linux-x86_64.sh
```

- Install
  - 2. $ ***bash Anaconda3-2024.02-1-Linux-x86_64.sh***

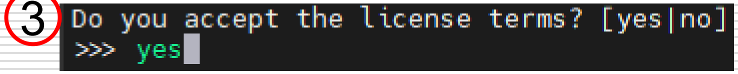
2



```
Welcome to Anaconda3 2024.02-1

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>> 
```

  - 3. Press “Enter” to continue (“q” to skip license agreement page) until this question pop. Type in “yes” and press “Enter”

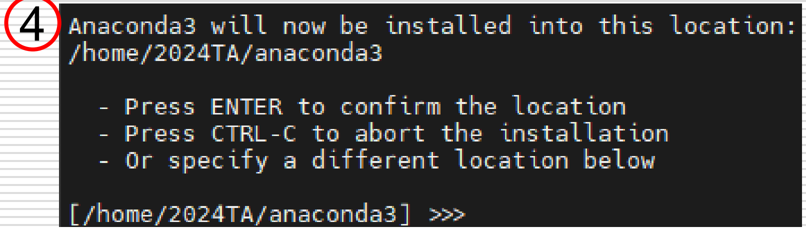
3



```
Do you accept the license terms? [yes|no]
>>> yes
```

  - 4. Press “Enter” and wait for installation

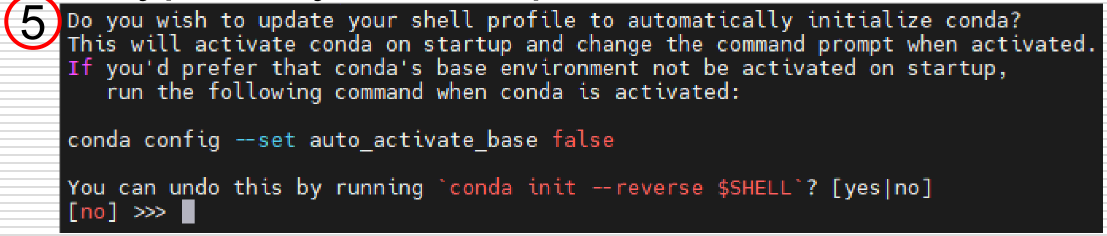
4



```
Anaconda3 will now be installed into this location:
/home/2024TA/anaconda3

  - Press ENTER to confirm the location
  - Press CTRL-C to abort the installation
  - Or specify a different location below

[/home/2024TA/anaconda3] >>>
```

  - 5.Type in “yes” and press “Enter”

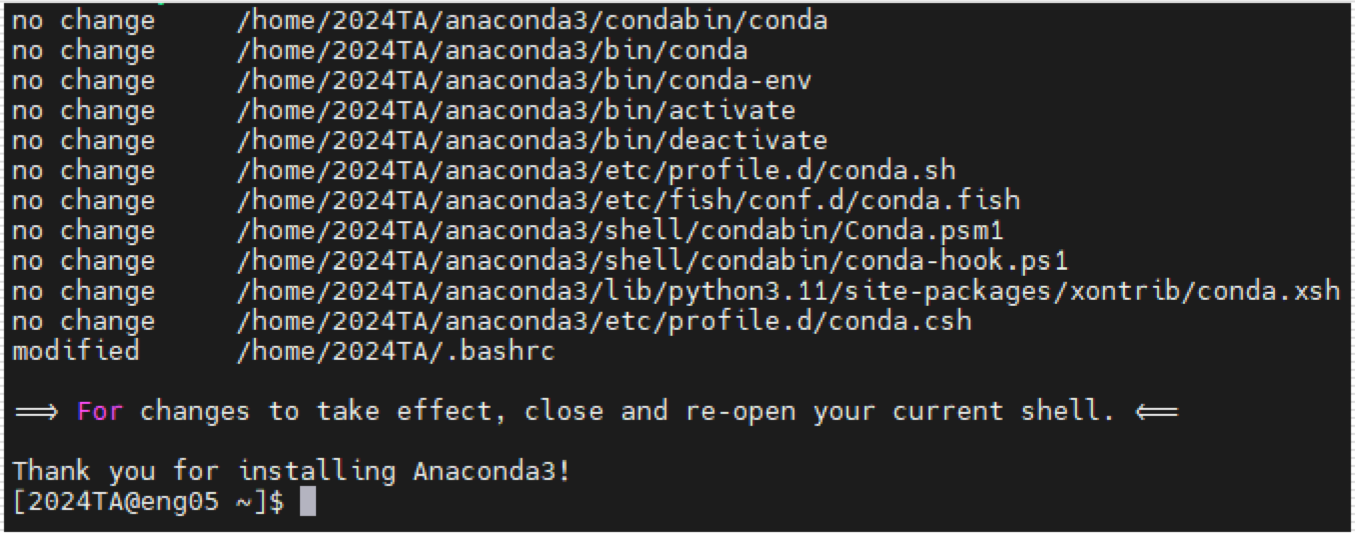
5



```
Do you wish to update your shell profile to automatically initialize conda?
This will activate conda on startup and change the command prompt when activated.
If you'd prefer that conda's base environment not be activated on startup,
   run the following command when conda is activated:

conda config --set auto_activate_base false

You can undo this by running `conda init --reverse $SHELL`? [yes|no]
[no] >>> 
```

Install Anaconda (2/2)

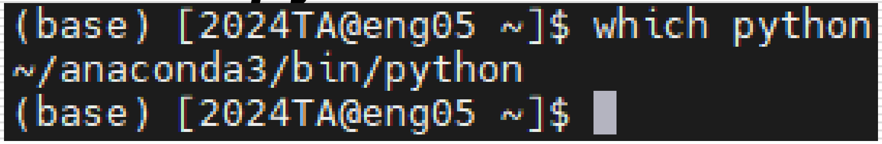
- You will see this after installing



```
no change     /home/2024TA/anaconda3/condabin/conda
no change     /home/2024TA/anaconda3/bin/conda
no change     /home/2024TA/anaconda3/bin/conda-env
no change     /home/2024TA/anaconda3/bin/activate
no change     /home/2024TA/anaconda3/bin/deactivate
no change     /home/2024TA/anaconda3/etc/profile.d/conda.sh
no change     /home/2024TA/anaconda3/etc/fish/conf.d/conda.fish
no change     /home/2024TA/anaconda3/shell/condabin/Conda.psm1
no change     /home/2024TA/anaconda3/shell/condabin/conda-hook.ps1
no change     /home/2024TA/anaconda3/lib/python3.11/site-packages/xontrib/conda.xsh
no change     /home/2024TA/anaconda3/etc/profile.d/conda.csh
modified      /home/2024TA/.bashrc

==> For changes to take effect, close and re-open your current shell. <==

Thank you for installing Anaconda3!
[2024TA@eng05 ~]$
```

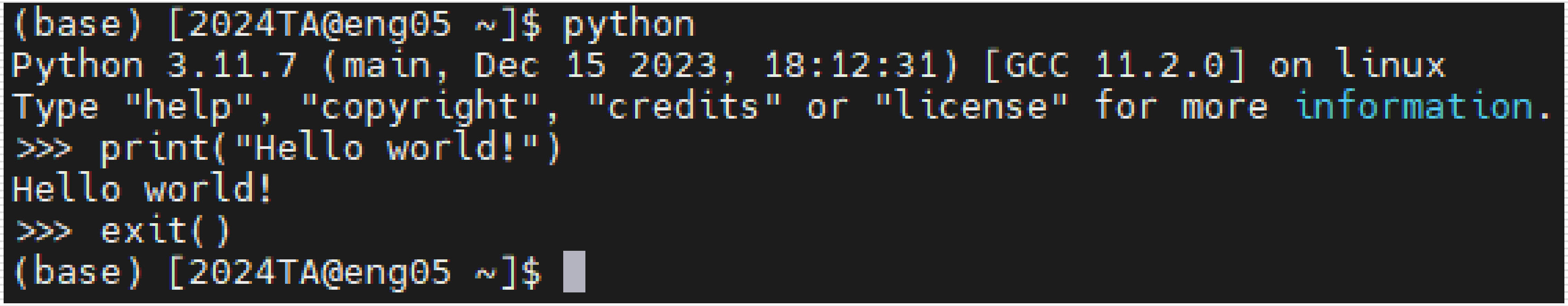
Open Python with Anaconda

- Close and re-open your shell
- Check if using anaconda python
  - $ ***which python***



```
(base) [2024TA@eng05 ~]$ which python
~/anaconda3/bin/python
(base) [2024TA@eng05 ~]$ 
```

- Start to use python
  - $ ***python***

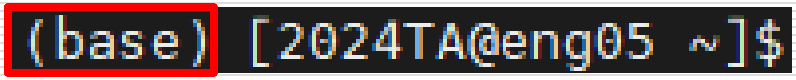


```
(base) [2024TA@eng05 ~]$ python
Python 3.11.7 (main, Dec 15 2023, 18:12:31) [GCC 11.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello world!")
Hello world!
>>> exit()
(base) [2024TA@eng05 ~]$ 
```

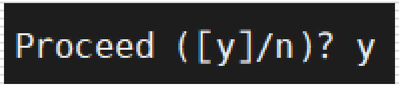
  - ***>>> exit()***

Install PyTorch

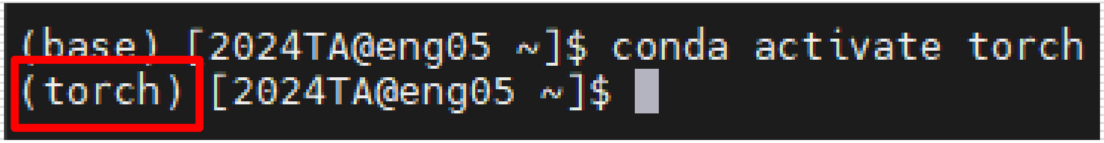
Create Environment

- Create a new conda environment
  - Stable version 

```
(base) [2024TA@eng05 ~]$
```

  - $ ***conda create --name torch python=3.8***
  - 

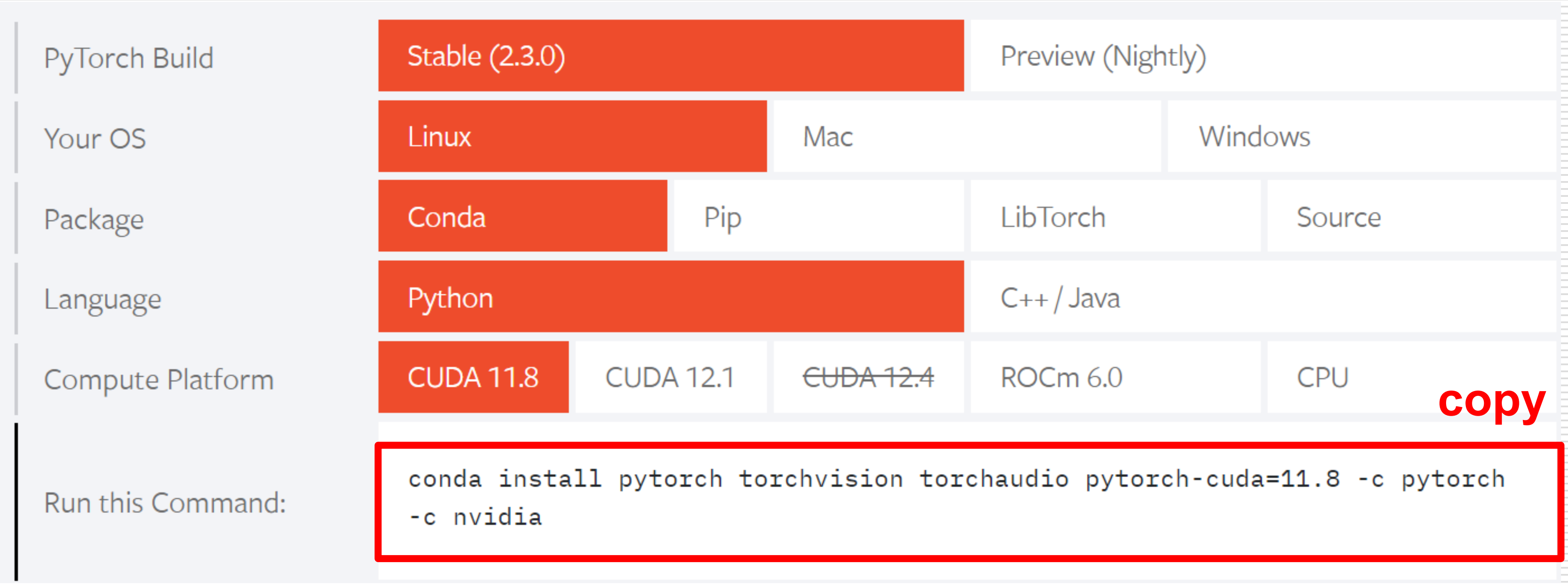
```
Proceed ([y]/n)? y
```

- Enter the environment
  - $ ***conda activate torch***
  - 

```
(base) [2024TA@eng05 ~]$ conda activate torch
(torch) [2024TA@eng05 ~]$
```

Install PyTorch (1/2)

- The following steps are for Installing the latest version (or other version) of PyTorch
- Visit PyTorch official website
  - https://pytorch.org/
- Choose the version



PyTorch Build	[✓ Stable (2.3.0)]	Preview (Nightly)		
Your OS	[✓ Linux]	Mac	Windows	
Package	[✓ Conda]	Pip	LibTorch	Source
Language	[✓ Python]	C++ / Java		
Compute Platform	[✓ CUDA 11.8]	CUDA 12.1	~~CUDA 12.4~~	ROCm 6.0	CPU

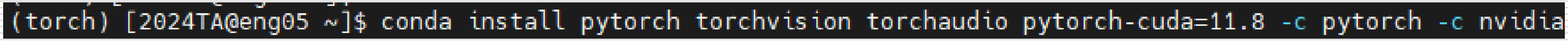
Run this Command:

```
conda install pytorch torchvision torchaudio pytorch-cuda=11.8 -c pytorch
-c nvidia
```

copy

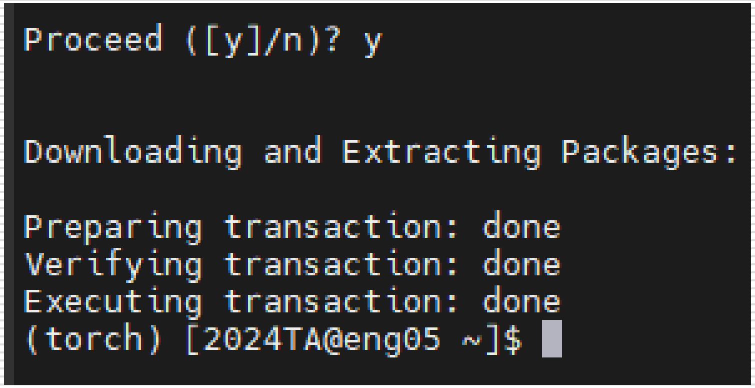
Install PyTorch (1/2)

- Run the command in conda environment on the server to install PyTorch (may take a few minutes)



```
(torch) [2024TA@eng05 ~]$ conda install pytorch torchvision torchaudio pytorch-cuda=11.8 -c pytorch -c nvidia
```

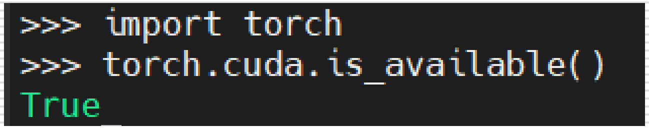
- Type in “y” and press “Enter”



```
Proceed ([y]/n)? y


Downloading and Extracting Packages:

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
(torch) [2024TA@eng05 ~]$ 
```

Check PyTorch

- Run python and import torch library
  - $ ***python***
  - >>> ***import torch***
- On eng05 , you can use cuda with gpu
  - >>> ***torch.cuda.is_available()***

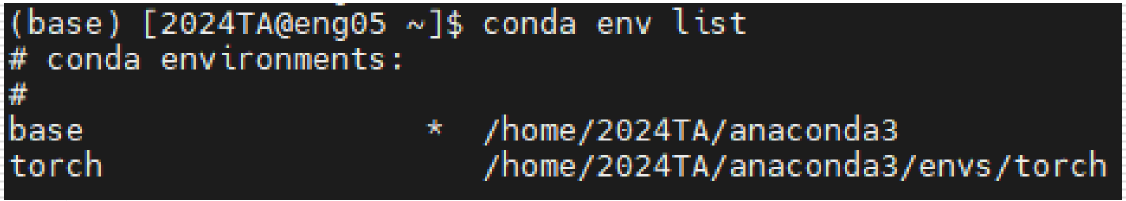


```
>>> import torch
>>> torch.cuda.is_available()
True
```

  - >>> ***exit()***

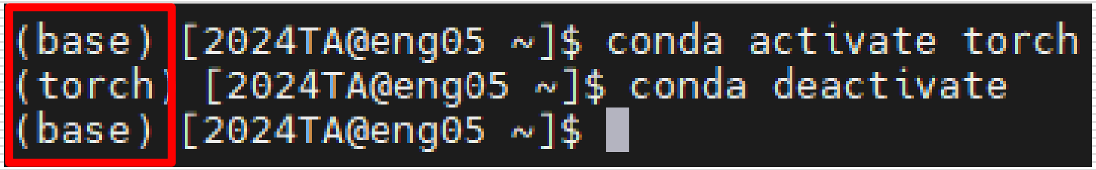
Commands in conda Environment

- Create New environment
  - conda create --name [env name] [configuration]
  - E.g. $ ***conda create --name torch python=3.6 pytorch=1.4.0***
- Show all environments
  - $ ***conda env list***



```
(base) [2024TA@eng05 ~]$ conda env list
# conda environments:
#
base                  *  /home/2024TA/anaconda3
torch                    /home/2024TA/anaconda3/envs/torch
```

- Activate conda environment
  - $ ***conda activate [env name]***
- Exit current conda environment
  - $ ***conda deactivate***



```
(base) [2024TA@eng05 ~]$ conda activate torch
(torch) [2024TA@eng05 ~]$ conda deactivate
(base) [2024TA@eng05 ~]$ 
```

- List all installed packages in environment
  - $ ***conda list***
- Remove conda environment
  - $ ***conda remove --name [env name] --all***

Introduction to Linux

Linux Distribution

- Linux was first released by Linus Torvalds
  – Linus’s MINIX
- Red Hat Enterprise Linux (RHEL)
  – CentOS and Fedora were developed from RHEL
  – Package management tool $ ***- yum***
- Debian
  – Ubuntu was developed from Debian
  – Package management tool $ ***- apt***
- All distributions use the same Linux kernels

Why Linux?

- Multi-user property
  - Allow multiple users access the system concurrently
- Friendly to programmers
  - Linux was designed for programming
- High security
- High stability
- Easy to maintenance

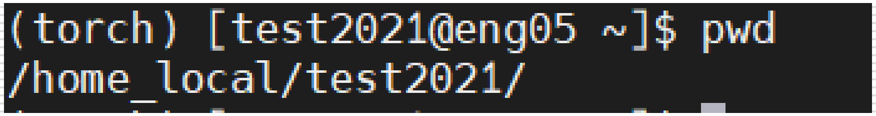
Servers in Lab

- Currently, we have six servers
  - adar10, eng01, eng02, eng03, eng04 and eng05
- eng02 is both an NIS server and an NFS server
  - NIS server: Provides account information to other servers
  - NFS server: Share home directory to other servers
- Every server has its own OS running independently
  - Can have its own account information and home directory
- Only adar10, eng04 and eng05 have GPUs
- NCHC and EDA servers are also available
  - You can ask lab members for access permissions of these servers if needed

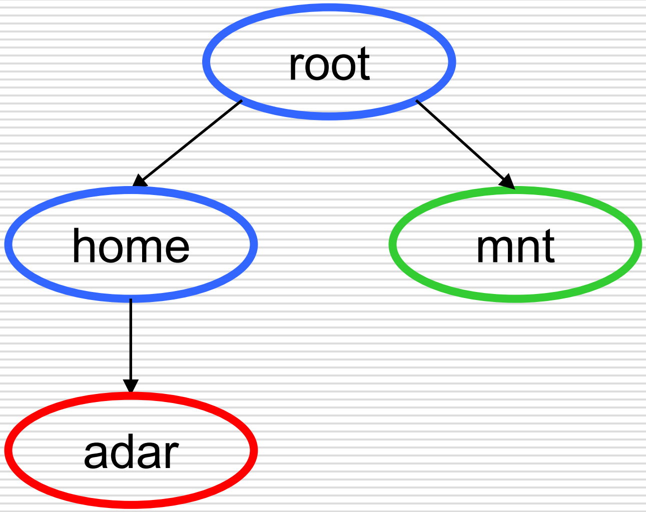
Basic Linux Commands

Paths (1/2)

- Absolute path
  - Specifying location from root directory
    - › ***/home/adar***
- Relative path
  - Specifying location related to current working directory
    - › ***../home/adar***
- Print working directory
  - $ ***pwd***
    - › ***/root/mnt***



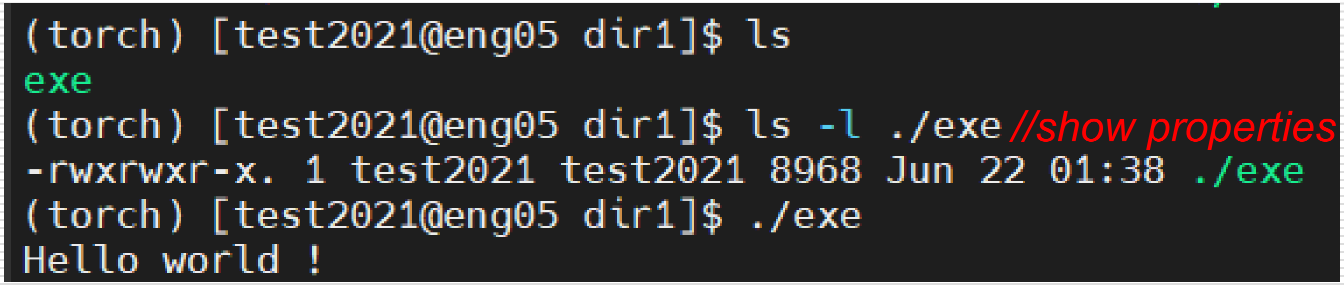
```
(torch) [test2021@eng05 ~]$ pwd
/home_local/test2021/
```



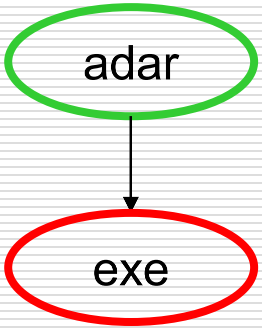
Green: Current working directory
Red: Target directory

Paths (2/2)

- “**~** ” stands for the home directory
- “**..** ” stands for the parent directory
- “**.** ” stands for the current working directory
- “***./exe*** ” equals to “exe” when specifying location
  - “**.** ” can be omitted when specifying location
- But only “***./exe*** ” can be used to run binary files



```
(torch) [test2021@eng05 dir1]$ ls
exe
(torch) [test2021@eng05 dir1]$ ls -l ./exe //show properties
-rwxrwxr-x. 1 test2021 test2021 8968 Jun 22 01:38 ./exe
(torch) [test2021@eng05 dir1]$ ./exe
Hello world !
```



Shells

- Shell is the interface between user and kernel
  - Command line interface (CLI)
    - bash, tcsh ……
  - Graphical user interface (GUI)
    - GNOME, KDE, Xfce ……

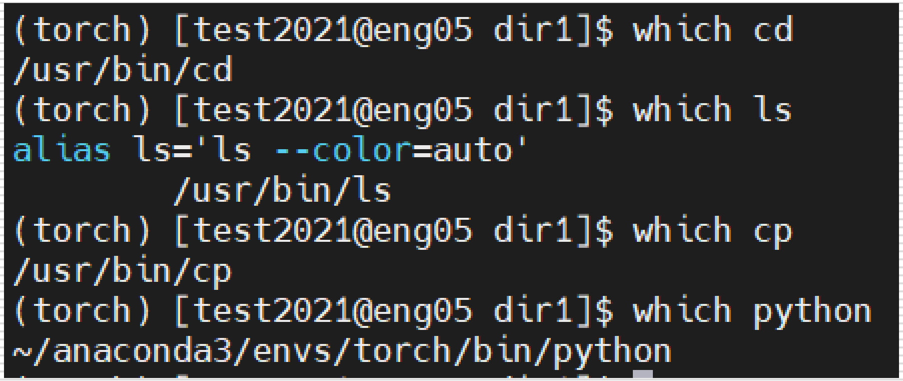
- Shell has variables for scripting and environment variables for environment setting

Command Execution (1/2)

- Commands are binary files actually
- Shell executes the command after command text is entered at the command line
  - String is broken up into words based on where spaces are in the string
  - First word of string is considered the command name
    - "***PATH*** " is an environment variable used for searching commands in sequence by the shell
    - First command which is matched will be executed
    - Doesn't contain the working directory by default for safety
  - Remaining words will be used as options and arguments
- Any Command being run is a child process of shell

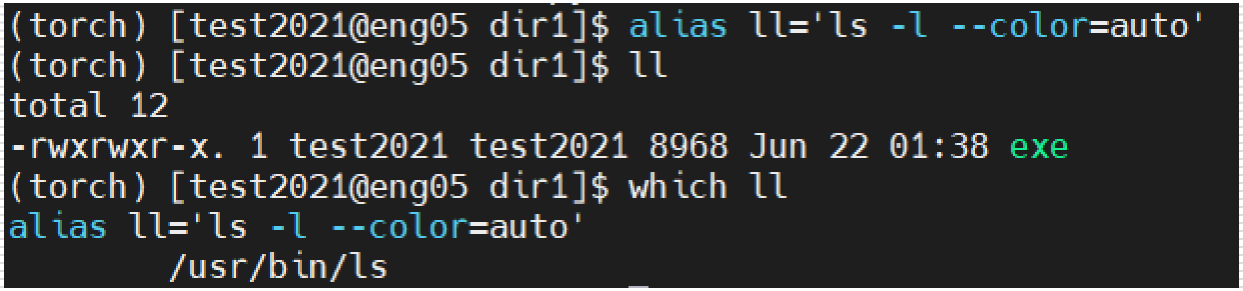
Command Execution (2/2)

- Identify the location of a given executable
  - $ ***which [command]***



```
(torch) [test2021@eng05 dir1]$ which cd
/usr/bin/cd
(torch) [test2021@eng05 dir1]$ which ls
alias ls='ls --color=auto'
        /usr/bin/ls
(torch) [test2021@eng05 dir1]$ which cp
/usr/bin/cp
(torch) [test2021@eng05 dir1]$ which python
~/anaconda3/envs/torch/bin/python
```

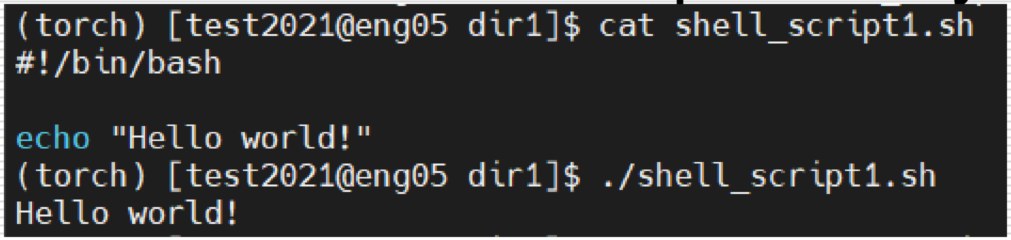
- Create simple names or abbreviations for commands
  - $ ***alias [name]=‘[command]’***



```
(torch) [test2021@eng05 dir1]$ alias ll='ls -l --color=auto'
(torch) [test2021@eng05 dir1]$ ll
total 12
-rwxrwxr-x. 1 test2021 test2021 8968 Jun 22 01:38 exe
(torch) [test2021@eng05 dir1]$ which ll
alias ll='ls -l --color=auto'
        /usr/bin/ls
```

Shell Script (1/2)

- Shell is capable of reading the command language
- Shell script is composed of commands and command language
- Filename of shell script usually ends with “.sh”

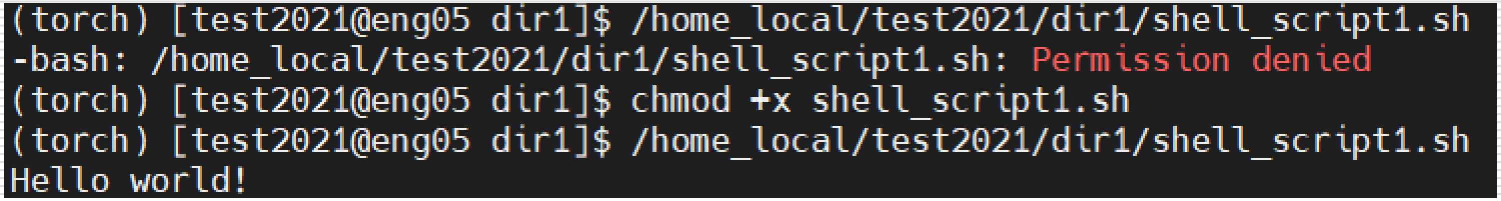


```
(torch) [test2021@eng05 dir1]$ cat shell_script1.sh
#!/bin/bash

echo "Hello world!"
(torch) [test2021@eng05 dir1]$ ./shell_script1.sh
Hello world!
```

  - $ ***echo …***
    - Print data

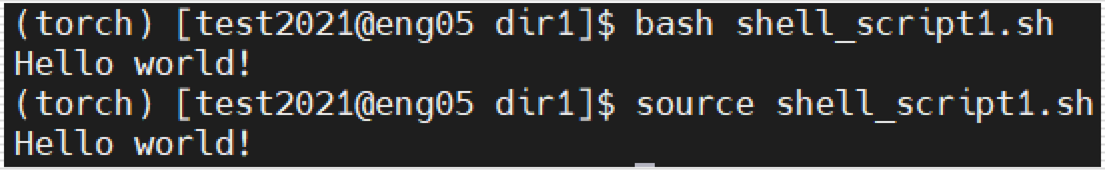
Shell Script (2/2)

- Two ways to execute shell script
  - Set an executable permission
    - $ ***chmod +x [script]***
    - ***$ [path_to_the_script]*** (Don’t omit “.” or “..” if using relative path)



```
(torch) [test2021@eng05 dir1]$ /home_local/test2021/dir1/shell_script1.sh
-bash: /home_local/test2021/dir1/shell_script1.sh: Permission denied
(torch) [test2021@eng05 dir1]$ chmod +x shell_script1.sh
(torch) [test2021@eng05 dir1]$ /home_local/test2021/dir1/shell_script1.sh
Hello world!
```

  - Specify the shell to execute the script
    - ***$ bash [script]*** (Running in child shell)

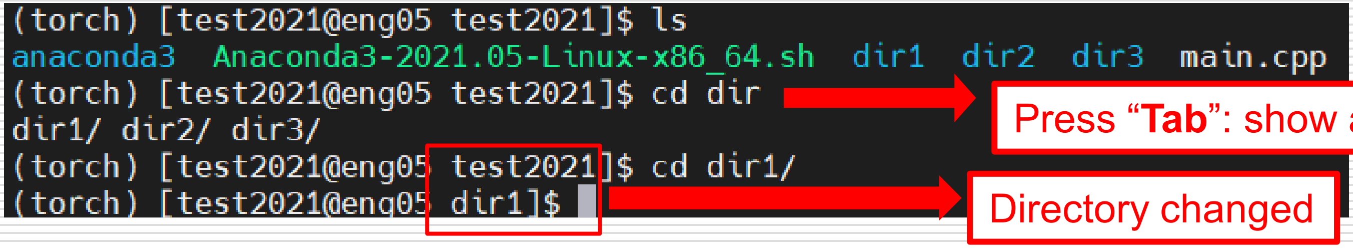


```
(torch) [test2021@eng05 dir1]$ bash shell_script1.sh
Hello world!
(torch) [test2021@eng05 dir1]$ source shell_script1.sh
Hello world!
```

    - $ ***source [script]*** (Running in current shell)

Manipulate Directories and Files (1/3)

- Change the working directory
  - $ ***cd [path]*** (Absolute path or relative path)

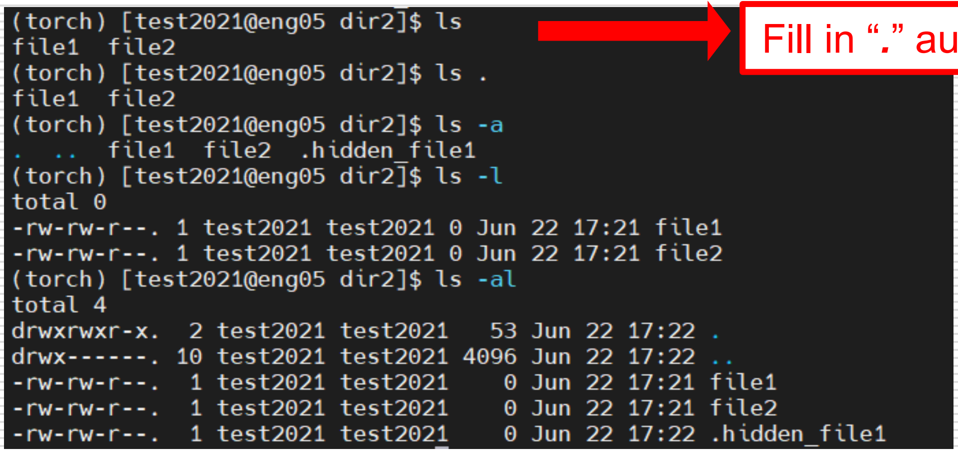


```
(torch) [test2021@eng05 test2021]$ ls
anaconda3  Anaconda3-2021.05-Linux-x86_64.sh  dir1  dir2  dir3  main.cpp
(torch) [test2021@eng05 test2021]$ cd dir
dir1/ dir2/ dir3/
(torch) [test2021@eng05 test2021]$ cd dir1/
(torch) [test2021@eng05 dir1]$ 
```

Press “**Tab**”: show all possible

Directory changed

- List directories and files
  - $ ***ls [option] [path]***
    - “***-l*** ” : Show detailed information
    - “***-a*** ” : Show hidden files and directories

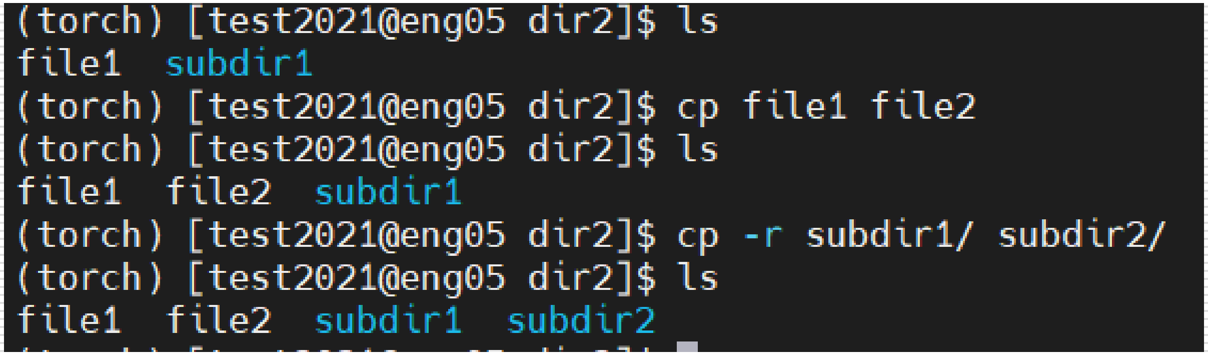


```
(torch) [test2021@eng05 dir2]$ ls
file1  file2
(torch) [test2021@eng05 dir2]$ ls .
file1  file2
(torch) [test2021@eng05 dir2]$ ls -a
.  ..  file1  file2  .hidden_file1
(torch) [test2021@eng05 dir2]$ ls -l
total 0
-rw-rw-r--. 1 test2021 test2021 0 Jun 22 17:21 file1
-rw-rw-r--. 1 test2021 test2021 0 Jun 22 17:21 file2
(torch) [test2021@eng05 dir2]$ ls -al
total 4
drwxrwxr-x.  2 test2021 test2021   53 Jun 22 17:22 .
drwx------. 10 test2021 test2021 4096 Jun 22 17:22 ..
-rw-rw-r--.  1 test2021 test2021    0 Jun 22 17:21 file1
-rw-rw-r--.  1 test2021 test2021    0 Jun 22 17:21 file2
-rw-rw-r--.  1 test2021 test2021    0 Jun 22 17:22 .hidden_file1
```

Fill in “.” automatically

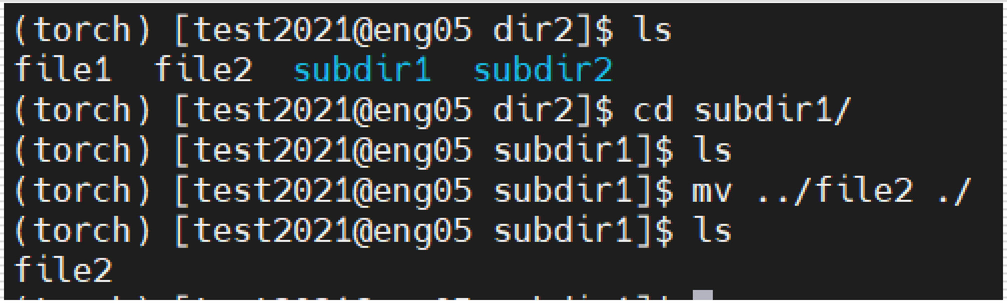
Manipulate Directories and Files (2/3)

- Copy files or directories
  - $ ***cp [option] [source] [target]***
    - ***"-r "*** : Recursive copy, using for copying directory



```
(torch) [test2021@eng05 dir2]$ ls
file1  subdir1
(torch) [test2021@eng05 dir2]$ cp file1 file2
(torch) [test2021@eng05 dir2]$ ls
file1  file2  subdir1
(torch) [test2021@eng05 dir2]$ cp -r subdir1/ subdir2/
(torch) [test2021@eng05 dir2]$ ls
file1  file2  subdir1  subdir2
```

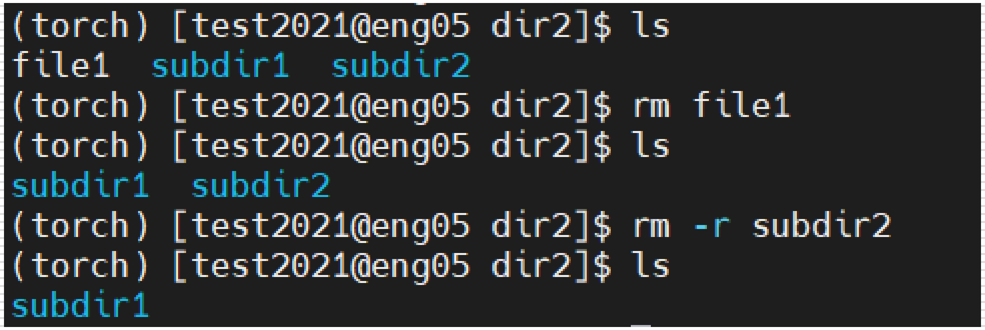
- Move files or directories
  - $ ***mv [source] [target]***



```
(torch) [test2021@eng05 dir2]$ ls
file1  file2  subdir1  subdir2
(torch) [test2021@eng05 dir2]$ cd subdir1/
(torch) [test2021@eng05 subdir1]$ ls
(torch) [test2021@eng05 subdir1]$ mv ../file2 ./
(torch) [test2021@eng05 subdir1]$ ls
file2
```

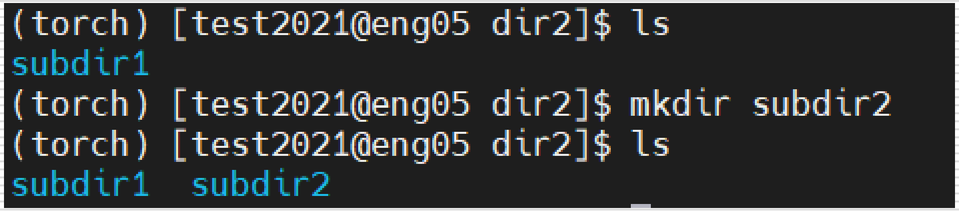
Manipulate Directories and Files (3/3)

- Remove files or directories
  - $ ***rm [option] [file or dir]***
    - “***-r*** ”: Recursive remove, using for removing directory
    - “***-f*** ”: Ignore nonexistent files and arguments



```
(torch) [test2021@eng05 dir2]$ ls
file1  subdir1  subdir2
(torch) [test2021@eng05 dir2]$ rm file1
(torch) [test2021@eng05 dir2]$ ls
subdir1  subdir2
(torch) [test2021@eng05 dir2]$ rm -r subdir2
(torch) [test2021@eng05 dir2]$ ls
subdir1
```

- Create directory
  - $ ***mkdir [directory]***



```
(torch) [test2021@eng05 dir2]$ ls
subdir1
(torch) [test2021@eng05 dir2]$ mkdir subdir2
(torch) [test2021@eng05 dir2]$ ls
subdir1  subdir2
```

File Permission (1/4)

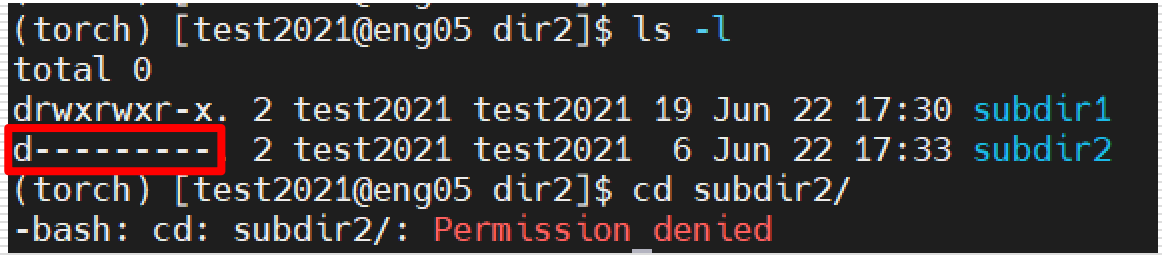
- Permission for owner, group, and others
  - Order: owner > group > others
    - If you are owner, then the permission for owner is applied
    - If you aren’t owner but in the group, then the permission for group is applied
    - If you aren’t owner and not in the group, then the permission for others is applied
  - The permission is expressed as like “-rwxr-xr--”
    - The first digit “-” indicates it is a file or a directory
    - The red digits “rwx” are the permission for owner
    - The blue digits “r-x” are the permission for group
    - The green digits “r--” are the permission for others

File Permission (2/4)

- “***r*** ” : Readable
  - File with “***r*** ” is readable
  - Directory with “***r*** ” means we can see the content inside it
- “***w*** ” : Writable
  - File with “***w*** ” is writable
  - Directory with “***w*** ” means we can alter the content inside it
- “***x*** ” : Executable
  - File with “***x*** ” is executable
    - e.g., binary files or shell scripts
  - Directory with “***x*** ” means can use it as a working directory
- “***-*** ” : No permission

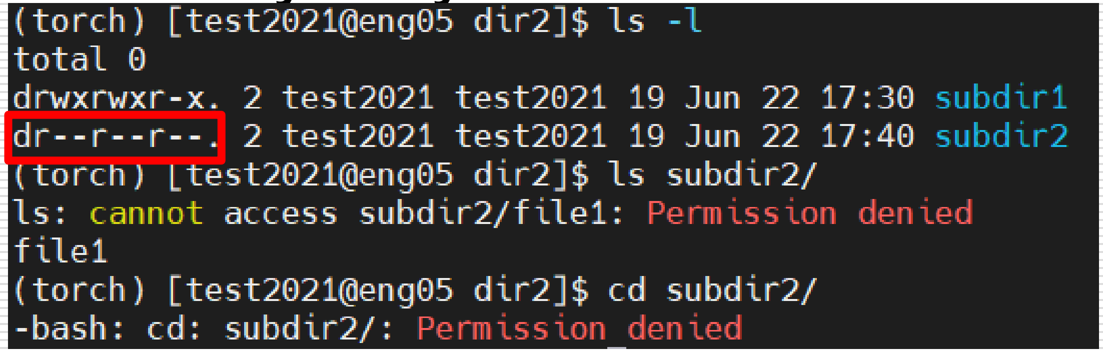
File Permission (3/4)

- Directory without permission



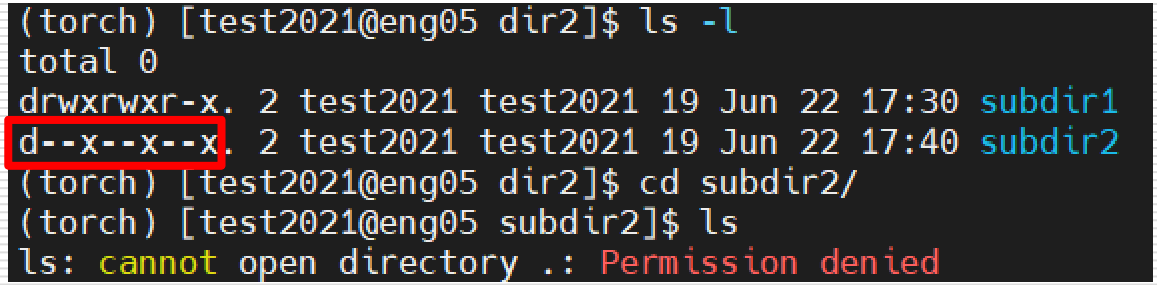
```
(torch) [test2021@eng05 dir2]$ ls -l
total 0
drwxrwxr-x. 2 test2021 test2021 19 Jun 22 17:30 subdir1
d---------  2 test2021 test2021  6 Jun 22 17:33 subdir2
(torch) [test2021@eng05 dir2]$ cd subdir2/
-bash: cd: subdir2/: Permission denied
```

- Directory only readable



```
(torch) [test2021@eng05 dir2]$ ls -l
total 0
drwxrwxr-x. 2 test2021 test2021 19 Jun 22 17:30 subdir1
dr--r--r--. 2 test2021 test2021 19 Jun 22 17:40 subdir2
(torch) [test2021@eng05 dir2]$ ls subdir2/
ls: cannot access subdir2/file1: Permission denied
file1
(torch) [test2021@eng05 dir2]$ cd subdir2/
-bash: cd: subdir2/: Permission denied
```

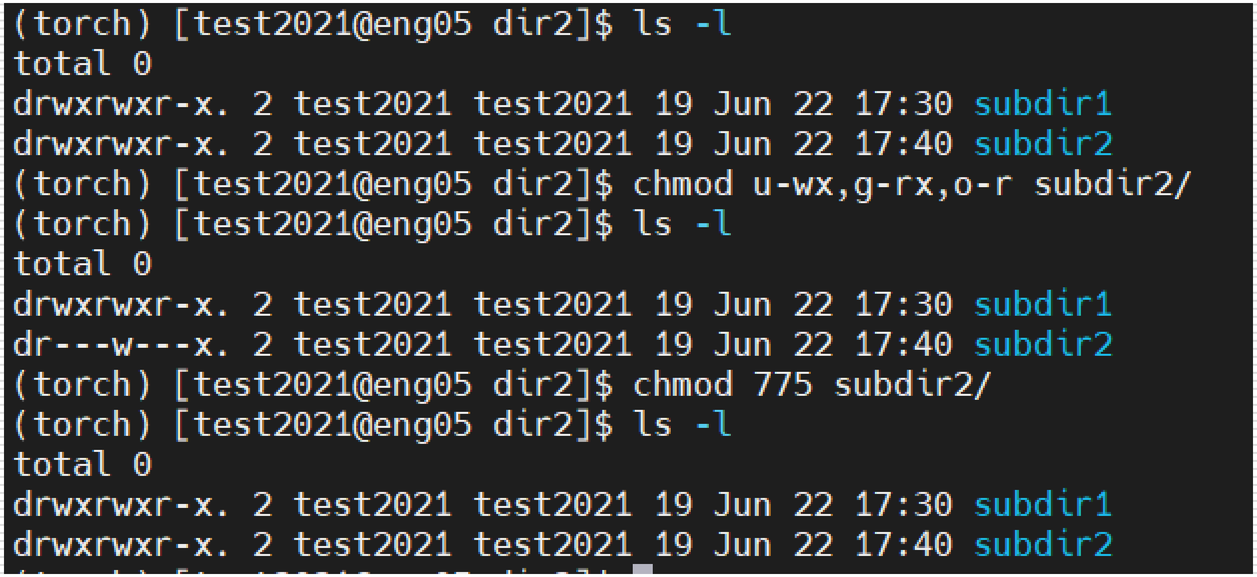
- Directory only executable



```
(torch) [test2021@eng05 dir2]$ ls -l
total 0
drwxrwxr-x. 2 test2021 test2021 19 Jun 22 17:30 subdir1
d--x--x--x. 2 test2021 test2021 19 Jun 22 17:40 subdir2
(torch) [test2021@eng05 dir2]$ cd subdir2/
(torch) [test2021@eng05 subdir2]$ ls
ls: cannot open directory .: Permission denied
```

File Permission (4/4)

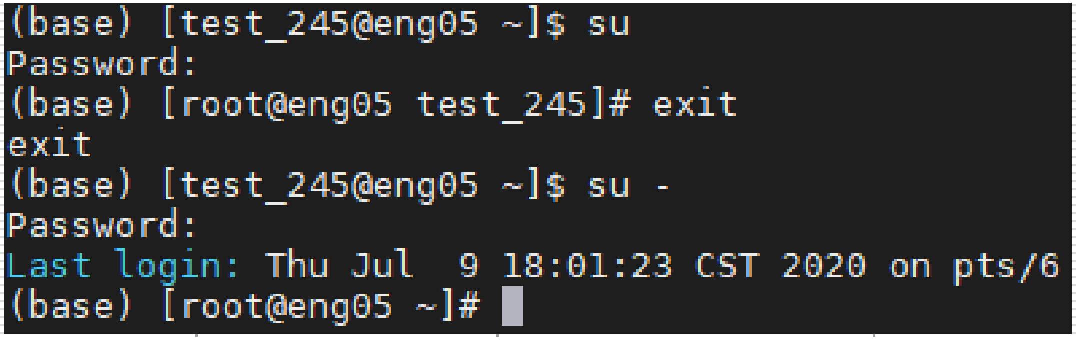
- Change permission
  - $ ***chmod [identity][operation][permission] [file or dir]***
  - Identity: u (user), g (group), o (others), a (all, ugo)
  - Operation: + (add), - (remove), = (set)
  - Permission: r (4), w (2), x (1), - (0)



```
(torch) [test2021@eng05 dir2]$ ls -l
total 0
drwxrwxr-x. 2 test2021 test2021 19 Jun 22 17:30 subdir1
drwxrwxr-x. 2 test2021 test2021 19 Jun 22 17:40 subdir2
(torch) [test2021@eng05 dir2]$ chmod u-wx,g-rx,o-r subdir2/
(torch) [test2021@eng05 dir2]$ ls -l
total 0
drwxrwxr-x. 2 test2021 test2021 19 Jun 22 17:30 subdir1
dr---w---x. 2 test2021 test2021 19 Jun 22 17:40 subdir2
(torch) [test2021@eng05 dir2]$ chmod 775 subdir2/
(torch) [test2021@eng05 dir2]$ ls -l
total 0
drwxrwxr-x. 2 test2021 test2021 19 Jun 22 17:30 subdir1
drwxrwxr-x. 2 test2021 test2021 19 Jun 22 17:40 subdir2
(torch) [test2021@eng05 dir2]$
```

Root Privilege

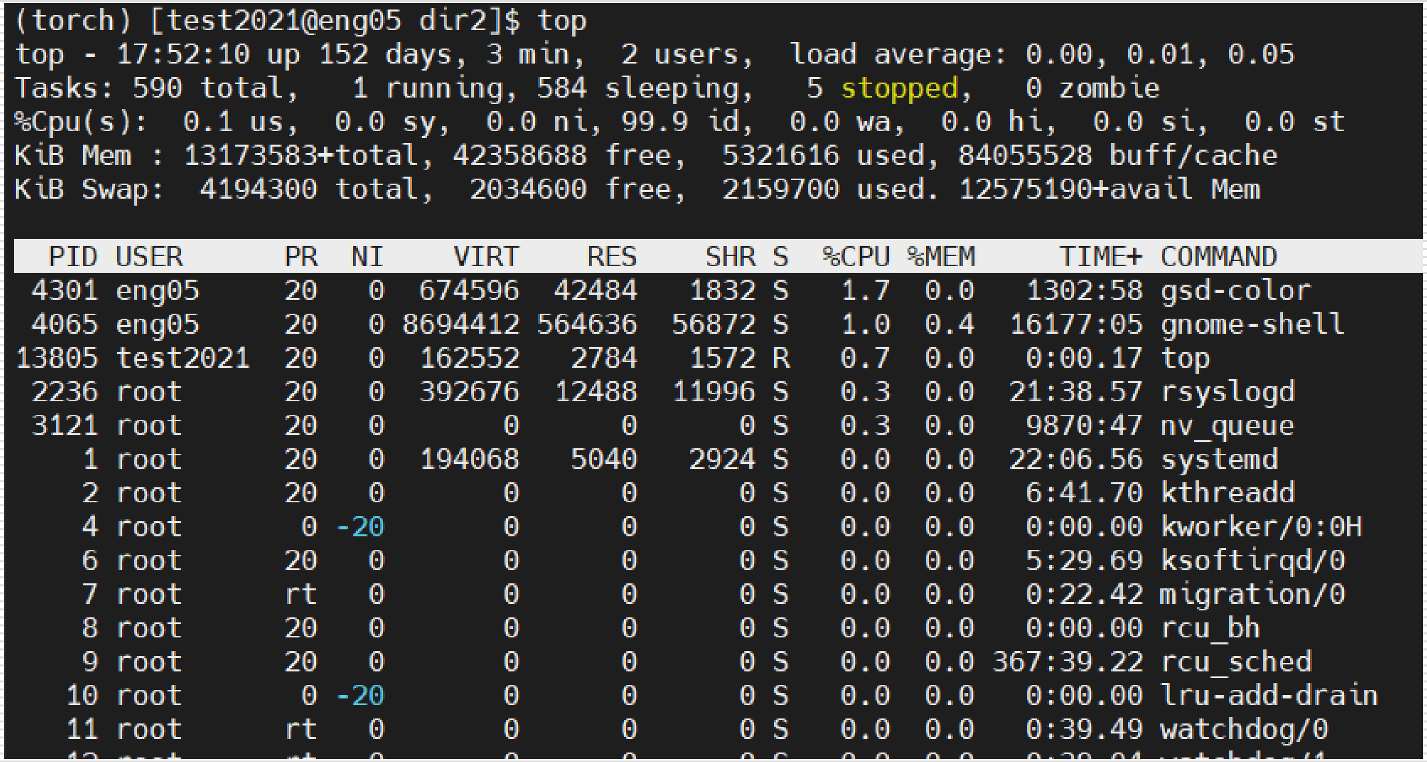
- In addition to the file permission, users cannot execute certain commands
- System administrators or users with root privilege are not regulated by these restrictions
  - User with the root privilege can use “sudo” to do anything
  - $ ***su*** : Allows ordinary user to get root privilege
  - $ ***su -*** : Enter a new shell same as using root to login



```
(base) [test_245@eng05 ~]$ su
Password:
(base) [root@eng05 test_245]# exit
exit
(base) [test_245@eng05 ~]$ su -
Password:
Last login: Thu Jul  9 18:01:23 CST 2020 on pts/6
(base) [root@eng05 ~]# 
```

System Monitoring Tool (1/3)

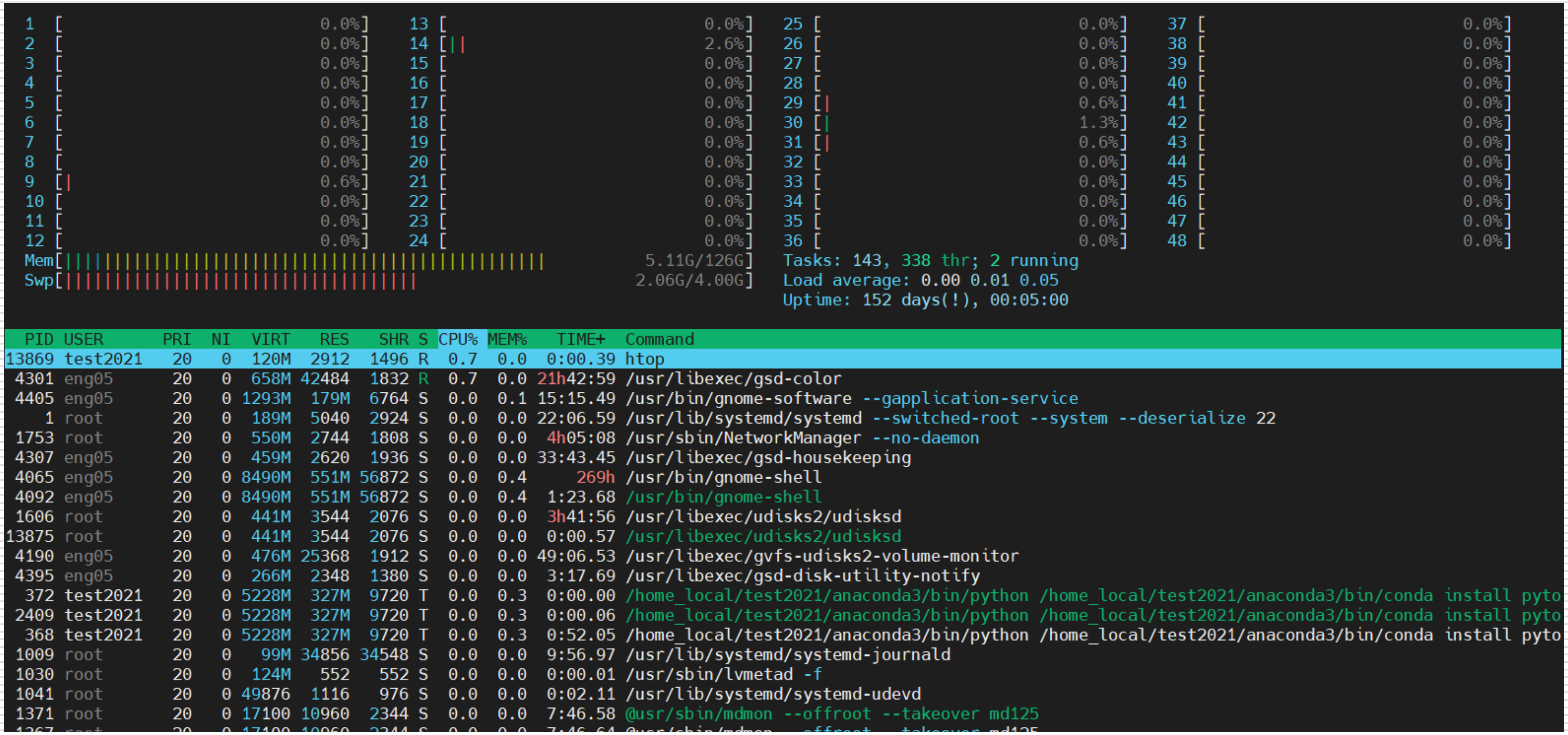
- Built-in system monitoring tool
  - $ ***top***
  - Shows a summary of system and list processes or threads currently being managed by the kernel



```
(torch) [test2021@eng05 dir2]$ top
top - 17:52:10 up 152 days, 3 min,  2 users,  load average: 0.00, 0.01, 0.05
Tasks: 590 total,   1 running, 584 sleeping,   5 stopped,   0 zombie
%Cpu(s):  0.1 us,  0.0 sy,  0.0 ni, 99.9 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem : 13173583+total, 42358688 free,  5321616 used, 84055528 buff/cache
KiB Swap:  4194300 total,  2034600 free,  2159700 used. 12575190+avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 4301 eng05     20   0  674596  42484   1832 S   1.7  0.0   1302:58 gsd-color
 4065 eng05     20   0 8694412 564636  56872 S   1.0  0.4  16177:05 gnome-shell
13805 test2021  20   0  162552   2784   1572 R   0.7  0.0   0:00.17 top
 2236 root      20   0  392676  12488  11996 S   0.3  0.0  21:38.57 rsyslogd
 3121 root      20   0       0      0      0 S   0.3  0.0   9870:47 nv_queue
    1 root      20   0  194068   5040   2924 S   0.0  0.0  22:06.56 systemd
    2 root      20   0       0      0      0 S   0.0  0.0   6:41.70 kthreadd
    4 root       0 -20       0      0      0 S   0.0  0.0   0:00.00 kworker/0:0H
    6 root      20   0       0      0      0 S   0.0  0.0   5:29.69 ksoftirqd/0
    7 root      rt   0       0      0      0 S   0.0  0.0   0:22.42 migration/0
    8 root      20   0       0      0      0 S   0.0  0.0   0:00.00 rcu_bh
    9 root      20   0       0      0      0 S   0.0  0.0 367:39.22 rcu_sched
   10 root       0 -20       0      0      0 S   0.0  0.0   0:00.00 lru-add-drain
   11 root      rt   0       0      0      0 S   0.0  0.0   0:39.49 watchdog/0
```

System Monitoring Tool (2/3)

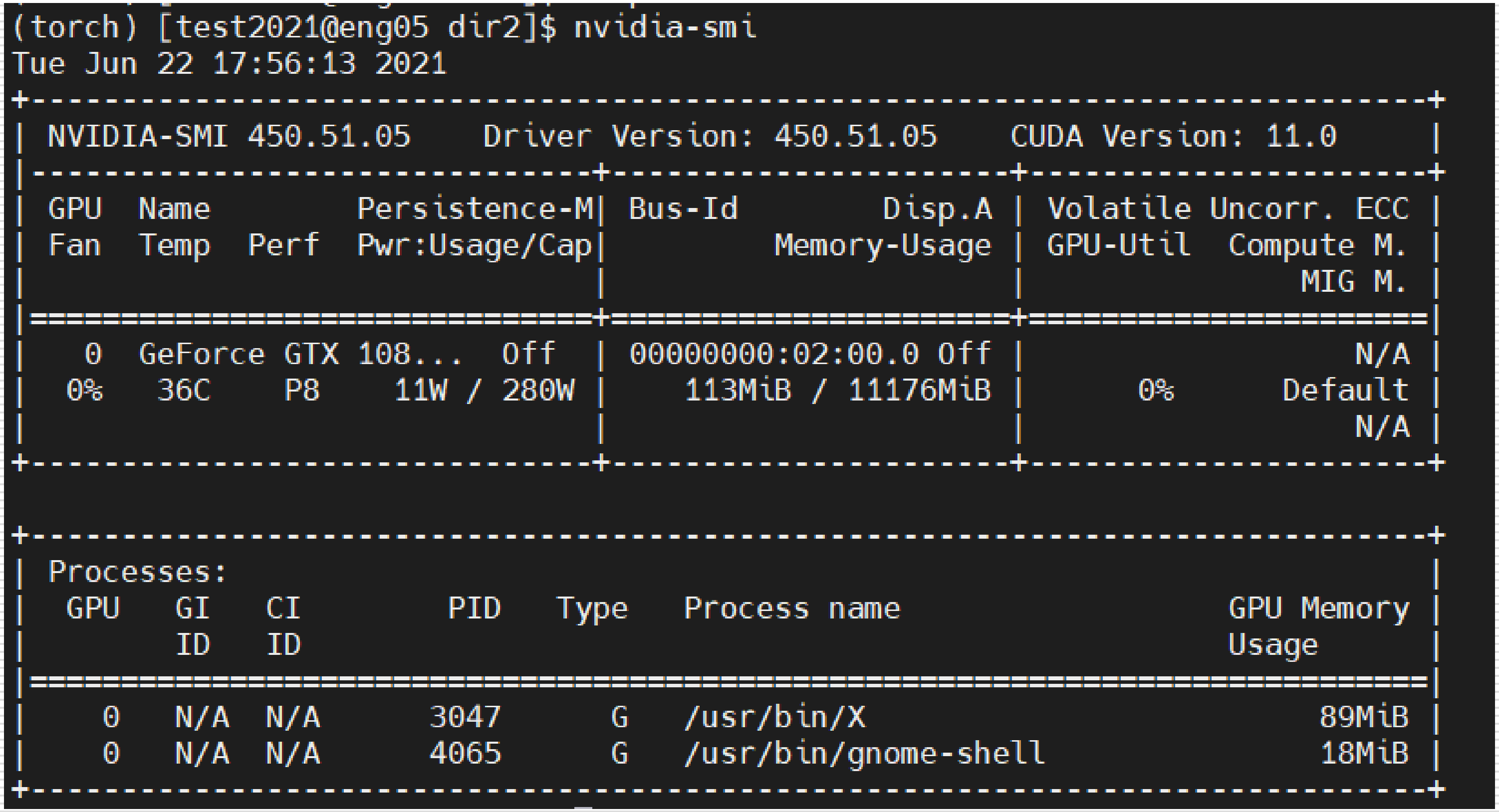
- Advance system monitoring tool
  - $ ***htop***
  - Have other instruction with it
    - E.g., Filter, Sort, Kill



```
  1  [                         0.0%]   13 [                         0.0%]   25 [                         0.0%]   37 [                         0.0%]
  2  [                         0.0%]   14 [||                       2.6%]   26 [                         0.0%]   38 [                         0.0%]
  3  [                         0.0%]   15 [                         0.0%]   27 [                         0.0%]   39 [                         0.0%]
  4  [                         0.0%]   16 [                         0.0%]   28 [                         0.0%]   40 [                         0.0%]
  5  [                         0.0%]   17 [                         0.0%]   29 [|                        0.6%]   41 [                         0.0%]
  6  [                         0.0%]   18 [                         0.0%]   30 [|                        1.3%]   42 [                         0.0%]
  7  [                         0.0%]   19 [                         0.0%]   31 [|                        0.6%]   43 [                         0.0%]
  8  [                         0.0%]   20 [                         0.0%]   32 [                         0.0%]   44 [                         0.0%]
  9  [|                        0.6%]   21 [                         0.0%]   33 [                         0.0%]   45 [                         0.0%]
  10 [                         0.0%]   22 [                         0.0%]   34 [                         0.0%]   46 [                         0.0%]
  11 [                         0.0%]   23 [                         0.0%]   35 [                         0.0%]   47 [                         0.0%]
  12 [                         0.0%]   24 [                         0.0%]   36 [                         0.0%]   48 [                         0.0%]
  Mem[|||||||||||||||||||||||||||||||||||||||||||||||||          5.11G/126G]   Tasks: 143, 338 thr; 2 running
  Swp[||||||||||||||||||||||||||||||||||||                      2.06G/4.00G]   Load average: 0.00 0.01 0.05
                                                                             Uptime: 152 days(!), 00:05:00

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
13869 test2021   20   0  120M  2912  1496 R  0.7  0.0  0:00.39 htop
 4301 eng05      20   0  658M 42484  1832 R  0.7  0.0 21h42:59 /usr/libexec/gsd-color
 4405 eng05      20   0 1293M  179M  6764 S  0.0  0.1 15:15.49 /usr/bin/gnome-software --gapplication-service
    1 root       20   0  189M  5040  2924 S  0.0  0.0 22:06.59 /usr/lib/systemd/systemd --switched-root --system --deserialize 22
 1753 root       20   0  550M  2744  1808 S  0.0  0.0  4h05:08 /usr/sbin/NetworkManager --no-daemon
 4307 eng05      20   0  459M  2620  1936 S  0.0  0.0 33:43.45 /usr/libexec/gsd-housekeeping
 4065 eng05      20   0 8490M  551M 56872 S  0.0  0.4     269h /usr/bin/gnome-shell
 4092 eng05      20   0 8490M  551M 56872 S  0.0  0.4  1:23.68 /usr/bin/gnome-shell
 1606 root       20   0  441M  3544  2076 S  0.0  0.0  3h41:56 /usr/libexec/udisks2/udisksd
13875 root       20   0  441M  3544  2076 S  0.0  0.0  0:00.57 /usr/libexec/udisks2/udisksd
 4190 eng05      20   0  476M 25368  1912 S  0.0  0.0 49:06.53 /usr/libexec/gvfs-udisks2-volume-monitor
 4395 eng05      20   0  266M  2348  1380 S  0.0  0.0  3:17.69 /usr/libexec/gsd-disk-utility-notify
  372 test2021   20   0 5228M  327M  9720 T  0.0  0.3  0:00.00 /home_local/test2021/anaconda3/bin/python /home_local/test2021/anaconda3/bin/conda install pyto
 2409 test2021   20   0 5228M  327M  9720 T  0.0  0.3  0:00.06 /home_local/test2021/anaconda3/bin/python /home_local/test2021/anaconda3/bin/conda install pyto
  368 test2021   20   0 5228M  327M  9720 T  0.0  0.3  0:52.05 /home_local/test2021/anaconda3/bin/python /home_local/test2021/anaconda3/bin/conda install pyto
 1009 root       20   0   99M 34856 34548 S  0.0  0.0  9:56.97 /usr/lib/systemd/systemd-journald
 1030 root       20   0  124M   552   552 S  0.0  0.0  0:00.01 /usr/sbin/lvmetad -f
 1041 root       20   0 49876  1116   976 S  0.0  0.0  0:02.11 /usr/lib/systemd/systemd-udevd
 1371 root       20   0 17100 10960  2344 S  0.0  0.0  7:46.58 @usr/sbin/mdmon --offroot --takeover md125
 1367 root       20   0 17100 10960  2344 S  0.0  0.0  7:46.64 @usr/sbin/mdmon --offroot --takeover md125
```

System Monitoring Tool (3/3)

- GPU monitoring tool
  - $ ***nvidia-smi***
  - Show GPU usage //Only available with GPU(eng04/eng05)

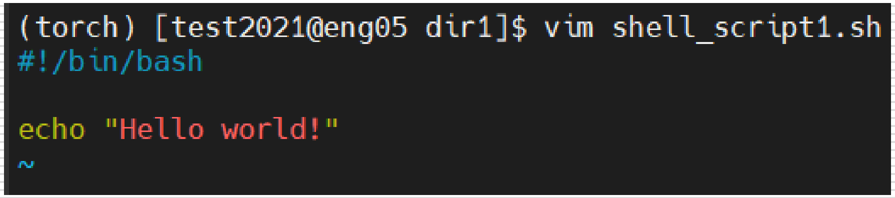


```
(torch) [test2021@eng05 dir2]$ nvidia-smi
Tue Jun 22 17:56:13 2021
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 450.51.05    Driver Version: 450.51.05    CUDA Version: 11.0     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  GeForce GTX 108...  Off  | 00000000:02:00.0 Off |                  N/A |
|  0%   36C    P8    11W / 280W |    113MiB / 11176MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A      3047      G   /usr/bin/X                         89MiB |
|    0   N/A  N/A      4065      G   /usr/bin/gnome-shell               18MiB |
+-----------------------------------------------------------------------------+
```

File Editor

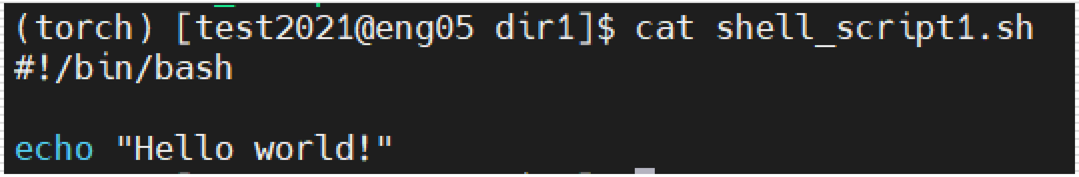
- Command line interface file editor
  - $ ***vim [file]*** : Edit file in CLI //create new if file doesn’t exist
    - “**i** " : Go to insert mode (Press “**Esc**” to quit insert mode)
    - “**V** ” : Visual line mode
    - “**:w** ” : Save file
    - “**:q** ” : Exit vim
    - “**:q!** ” : Exit vim without saving file

“:wq” : Save file and exit



```
(torch) [test2021@eng05 dir1]$ vim shell_script1.sh
#!/bin/bash

echo "Hello world!"
~
```

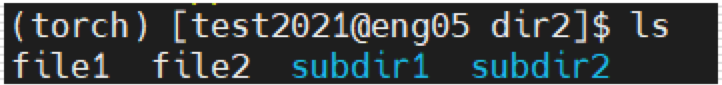
- Graphical user interface
  - $ ***gedit [file]*** : Edit file in GUI
- View the content of a file on the terminal
  - $ ***cat [file]***



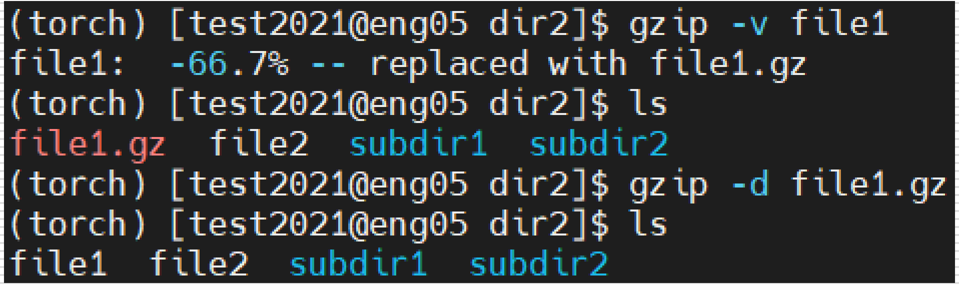
```
(torch) [test2021@eng05 dir1]$ cat shell_script1.sh
#!/bin/bash

echo "Hello world!"
```

Compression and Decompression (1/2)

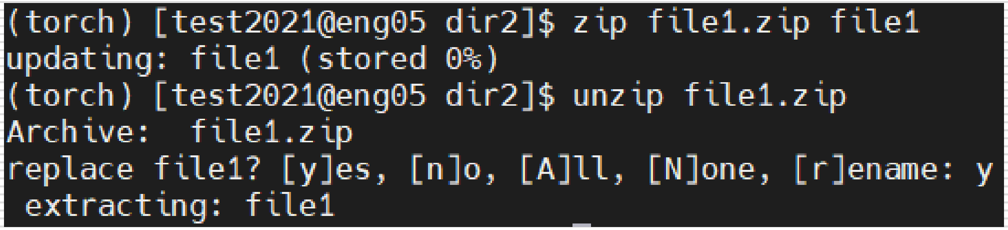
- File compression (Replace ***file*** with ***file.gz***)
  - $ ***gzip -v [file]*** : Show compression ratio
  - $ ***gzip -c [file]*** : Show the content of the compressed file
- File decompression (Replace ***file.gz*** with ***file***)
  - $ ***gzip -d [file]***



```
(torch) [test2021@eng05 dir2]$ ls
file1  file2  subdir1  subdir2
```



```
(torch) [test2021@eng05 dir2]$ gzip -v file1
file1:   -66.7% -- replaced with file1.gz
(torch) [test2021@eng05 dir2]$ ls
file1.gz  file2  subdir1  subdir2
(torch) [test2021@eng05 dir2]$ gzip -d file1.gz
(torch) [test2021@eng05 dir2]$ ls
file1  file2  subdir1  subdir2
```



```
(torch) [test2021@eng05 dir2]$ zip file1.zip file1
updating: file1 (stored 0%)
(torch) [test2021@eng05 dir2]$ unzip file1.zip
Archive:  file1.zip
replace file1? [y]es, [n]o, [A]ll, [N]one, [r]ename: y
 extracting: file1
```

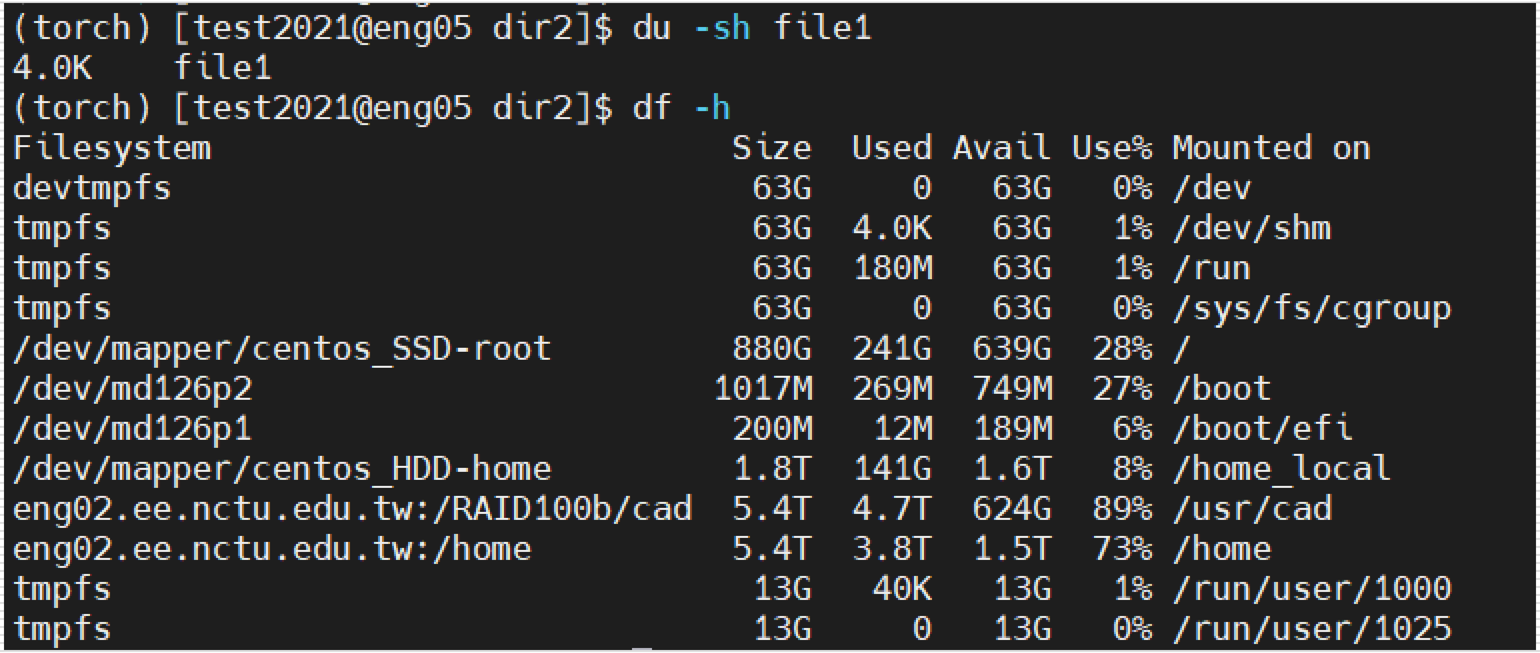
- Zip file
  - $ zip [compressed file] [file] (Compress file to file.zip)
  - $ unzip [compressed file] (File decompression)

Compression and Decompression (2/2)

- Tarfile
  - Multiple files **stored** in a single file
- Tarball
  - Multiple files **compressed** in a single file (Using ***gzip***)
- Build or extract tar files
  - Build a tarfile (.tar)
    - $ ***tar -cvf [tarfile] [file 1] [file 2] …***
  - Extract a tarfile
    - $ ***tar -xvf [tarfile] (-C [dir])***
  - Build a tarball (.tar.gz)
    - $ ***tar -zcvf [tarball] [file 1] [file 2] …***
  - Extract a tarball
    - $ ***tar -zxvf [tarball] (-C [dir])***

Check disk space

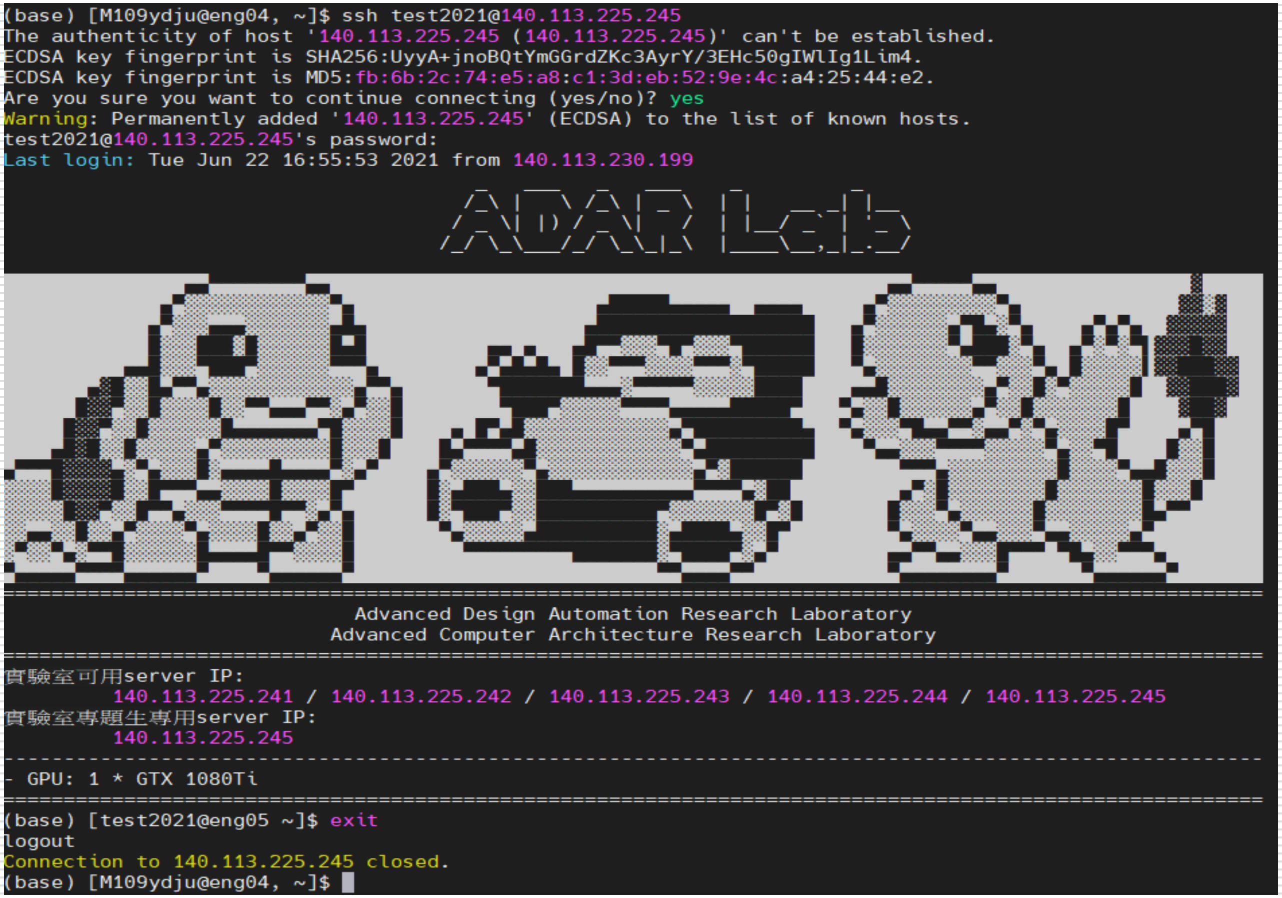
- Show the space usage of a file or directory
  – $ ***du -sh [file or dir]***
- Show the information related to file systems about total space and available space
  – $ ***df -h***



```
(torch) [test2021@eng05 dir2]$ du -sh file1
4.0K    file1
(torch) [test2021@eng05 dir2]$ df -h
Filesystem                            Size  Used Avail Use% Mounted on
devtmpfs                               63G     0   63G   0% /dev
tmpfs                                  63G  4.0K   63G   1% /dev/shm
tmpfs                                  63G  180M   63G   1% /run
tmpfs                                  63G     0   63G   0% /sys/fs/cgroup
/dev/mapper/centos_SSD-root           880G  241G  639G  28% /
/dev/md126p2                         1017M  269M  749M  27% /boot
/dev/md126p1                          200M   12M  189M   6% /boot/efi
/dev/mapper/centos_HDD-home           1.8T  141G  1.6T   8% /home_local
eng02.ee.nctu.edu.tw:/RAID100b/cad  5.4T  4.7T  624G  89% /usr/cad
eng02.ee.nctu.edu.tw:/home            5.4T  3.8T  1.5T  73% /home
tmpfs                                  13G   40K   13G   1% /run/user/1000
tmpfs                                  13G     0   13G   0% /run/user/1025
```

Connect to Other Servers (1/2)

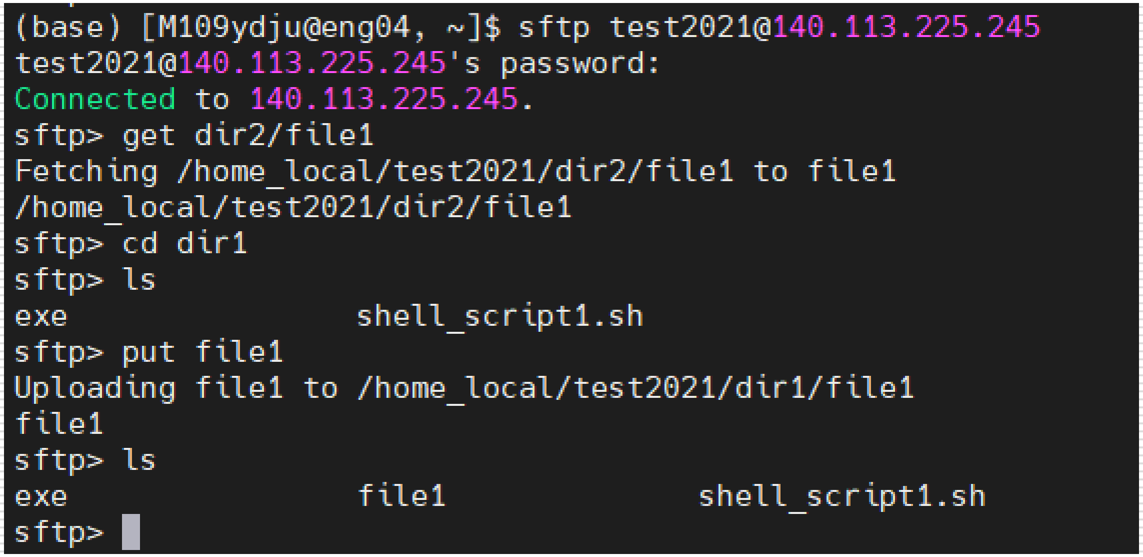
- Connect to a Linux server (SSH)
  - $ ***ssh [account]@[ip] (-X)***
    - "***-X*** " : Enabled when GUI command is needed



```
(base) [M109ydju@eng04, ~]$ ssh test2021@140.113.225.245
The authenticity of host '140.113.225.245 (140.113.225.245)' can't be established.
ECDSA key fingerprint is SHA256:UyyA+jnoBQtYmGGrdZKc3AyrY/3EHc50gIWlIg1Lim4.
ECDSA key fingerprint is MD5:fb:6b:2c:74:e5:a8:c1:3d:eb:52:9e:4c:a4:25:44:e2.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '140.113.225.245' (ECDSA) to the list of known hosts.
test2021@140.113.225.245's password:
Last login: Tue Jun 22 16:55:53 2021 from 140.113.230.199

=========================================================================================================
                              Advanced Design Automation Research Laboratory
                            Advanced Computer Architecture Research Laboratory
=========================================================================================================
實驗室可用server IP:
         140.113.225.241 / 140.113.225.242 / 140.113.225.243 / 140.113.225.244 / 140.113.225.245
實驗室專題生專用server IP:
         140.113.225.245
---------------------------------------------------------------------------------------------------------
- GPU: 1 * GTX 1080Ti
=========================================================================================================
(base) [test2021@eng05 ~]$ exit
logout
Connection to 140.113.225.245 closed.
(base) [M109ydju@eng04, ~]$
```

Connect to Other Servers (2/2)

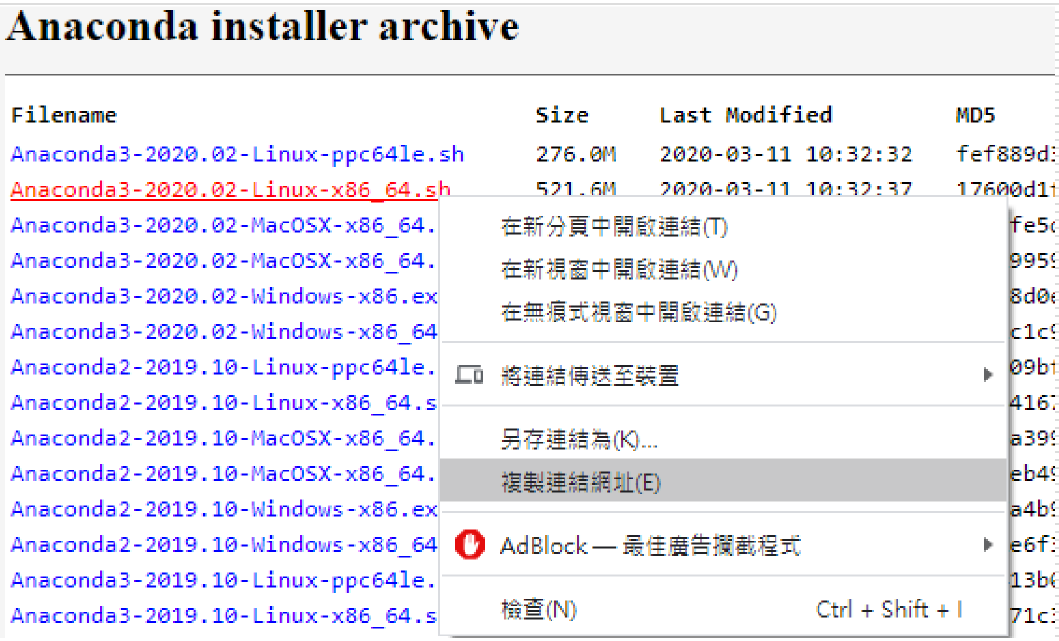
- Transfer file to a Linux server (SFTP)
  - $ ***sftp [account]@[ip]***
    - $ ***put (-r)*** : Upload file (directory)
    - $ ***get (-r)*** : Download file (directory)



```
(base) [M109ydju@eng04, ~]$ sftp test2021@140.113.225.245
test2021@140.113.225.245's password:
Connected to 140.113.225.245.
sftp> get dir2/file1
Fetching /home_local/test2021/dir2/file1 to file1
/home_local/test2021/dir2/file1
sftp> cd dir1
sftp> ls
exe                    shell_script1.sh
sftp> put file1
Uploading file1 to /home_local/test2021/dir1/file1
file1
sftp> ls
exe                    file1                  shell_script1.sh
sftp> 
```

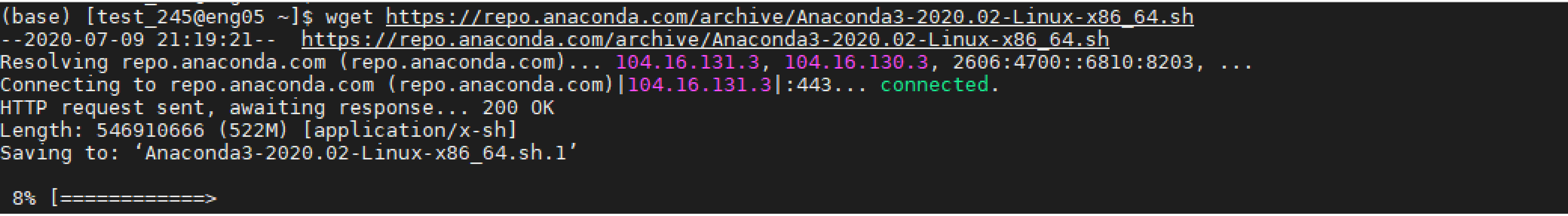
Download Files from a URL

- Download files from a URL via terminal
  - $ ***wget [URL]***



Anaconda installer archive

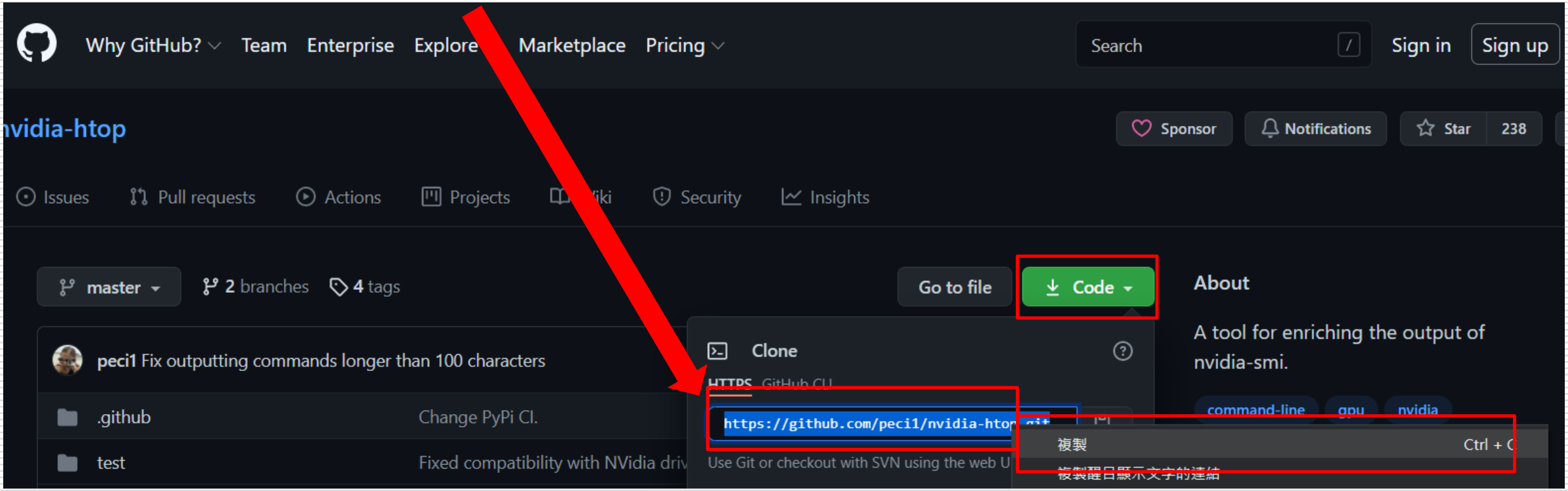
```
Filename                                 Size     Last Modified         MD5
Anaconda3-2020.02-Linux-ppc64le.sh       276.0M   2020-03-11 10:32:32   fef889d
Anaconda3-2020.02-Linux-x86_64.sh        521.6M   2020-03-11 10:32:37   17600d1
Anaconda3-2020.02-MacOSX-x86_64.                                        fe5
Anaconda3-2020.02-MacOSX-x86_64.                                        995
Anaconda3-2020.02-Windows-x86.ex                                        8d0
Anaconda3-2020.02-Windows-x86_64                                        c1c
Anaconda2-2019.10-Linux-ppc64le.                                        09b
Anaconda2-2019.10-Linux-x86_64.s                                        416
Anaconda2-2019.10-MacOSX-x86_64.                                        a39
Anaconda2-2019.10-MacOSX-x86_64.                                        eb4
Anaconda2-2019.10-Windows-x86.ex                                        a4b
Anaconda2-2019.10-Windows-x86_64                                        e6f
Anaconda3-2019.10-Linux-ppc64le.                                        13b
Anaconda3-2019.10-Linux-x86_64.s                                        71c
```



```
(base) [test_245@eng05 ~]$ wget https://repo.anaconda.com/archive/Anaconda3-2020.02-Linux-x86_64.sh
--2020-07-09 21:19:21--  https://repo.anaconda.com/archive/Anaconda3-2020.02-Linux-x86_64.sh
Resolving repo.anaconda.com (repo.anaconda.com)... 104.16.131.3, 104.16.130.3, 2606:4700::6810:8203, ...
Connecting to repo.anaconda.com (repo.anaconda.com)|104.16.131.3|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 546910666 (522M) [application/x-sh]
Saving to: ‘Anaconda3-2020.02-Linux-x86_64.sh.1’

 8% [============>
```

Download Files from github

- $ *git clone [link]*

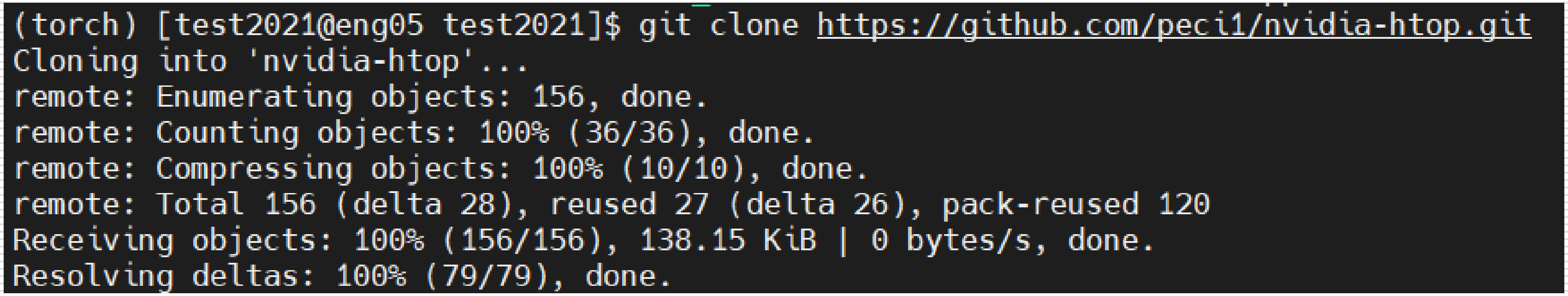


peci1 Fix outputting commands longer than 100 characters

https://github.com/peci1/nvidia-htop.git

Use Git or checkout with SVN using the web U

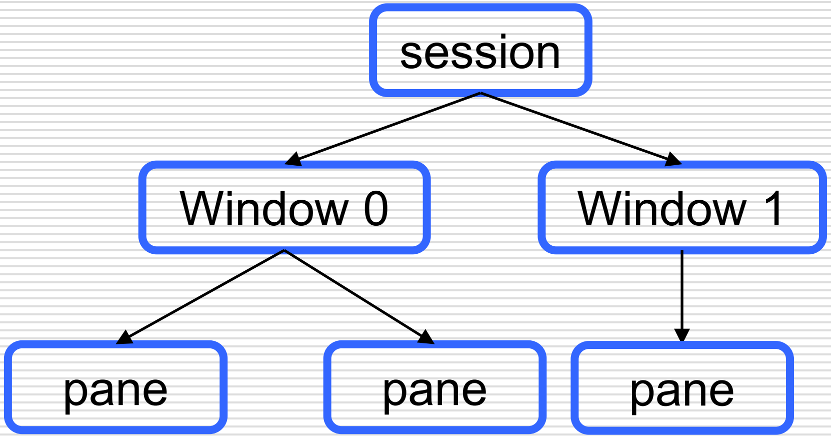
A tool for enriching the output of nvidia-smi.



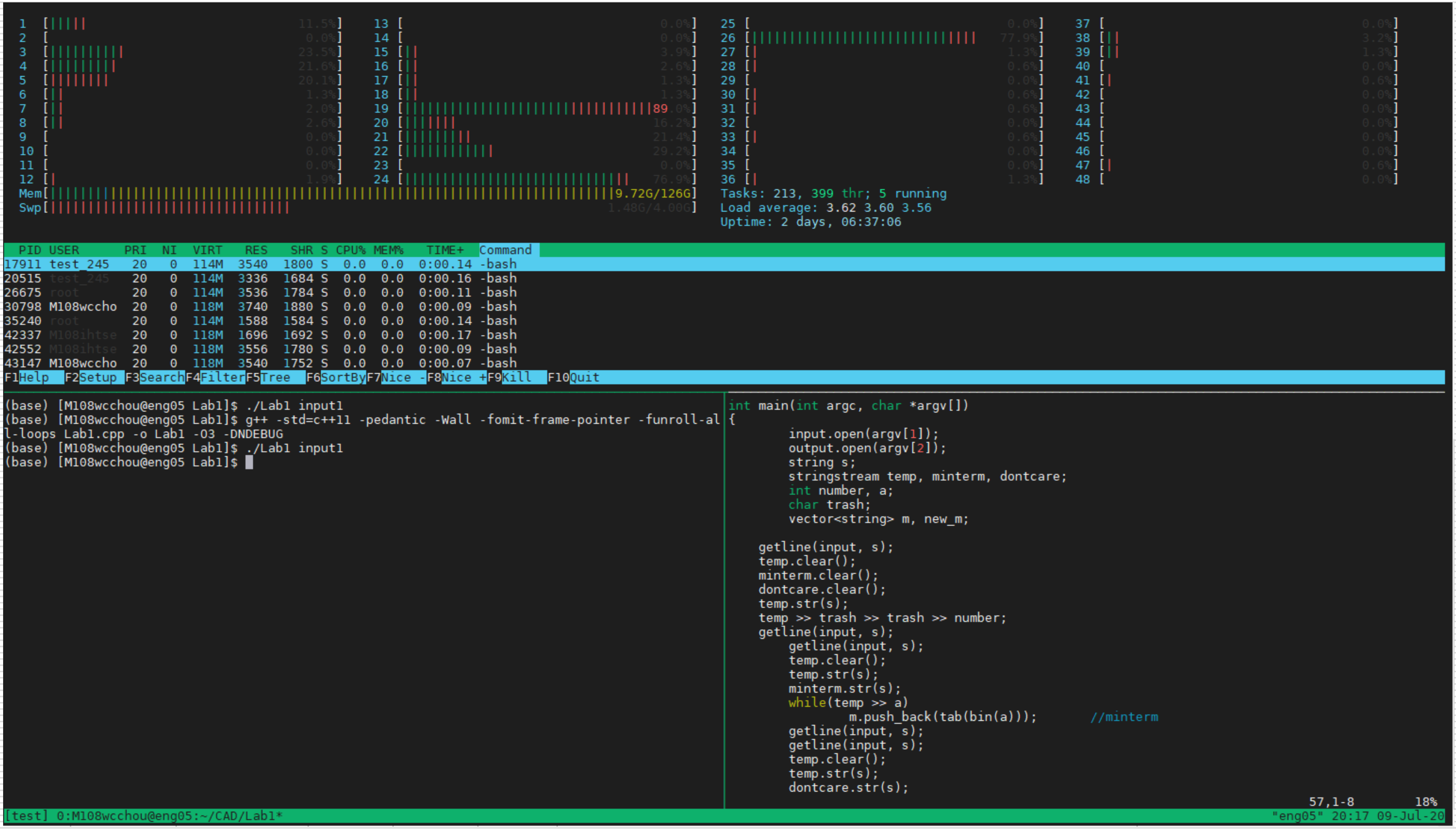
```
(torch) [test2021@eng05 test2021]$ git clone https://github.com/peci1/nvidia-htop.git
Cloning into 'nvidia-htop'...
remote: Enumerating objects: 156, done.
remote: Counting objects: 100% (36/36), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 156 (delta 28), reused 27 (delta 26), pack-reused 120
Receiving objects: 100% (156/156), 138.15 KiB | 0 bytes/s, done.
Resolving deltas: 100% (79/79), done.
```

Terminal Multiplexer (1/4)

- Tool that can make your works remain running after disconnecting from the server
  - $ ***tmux***
  - Allow window splitting
  - Project management
- Example
  - A session with 2 windows
  - Window 0 has 2 panes
  - Window 1 has 1 pane



Terminal Multiplexer (2/4)



```
  1  [|||||                      11.5%]    13 [                           0.0%]    25 [                           0.0%]    37 [                           0.0%]
  2  [                            0.0%]    14 [                           0.0%]    26 [|||||||||||||||||||||||||||| 77.9%]    38 [||                         3.2%]
  3  [||||||||||                 23.5%]    15 [||                         3.9%]    27 [|                          1.3%]    39 [||                         1.3%]
  4  [|||||||||                  21.6%]    16 [||                         2.6%]    28 [|                          0.6%]    40 [                           0.0%]
  5  [||||||||                   20.1%]    17 [||                         1.3%]    29 [                           0.0%]    41 [|                          0.6%]
  6  [||                          1.3%]    18 [||                         1.3%]    30 [|                          0.6%]    42 [                           0.0%]
  7  [||                          2.0%]    19 [||||||||||||||||||||||||||||||||||89.0%]    31 [|                          0.6%]    43 [                           0.0%]
  8  [||                          2.6%]    20 [|||||||                   16.2%]    32 [                           0.0%]    44 [                           0.0%]
  9  [                            0.0%]    21 [|||||||||                 21.4%]    33 [|                          0.6%]    45 [                           0.0%]
  10 [                            0.0%]    22 [||||||||||||              29.2%]    34 [                           0.0%]    46 [                           0.0%]
  11 [                            0.0%]    23 [                           0.0%]    35 [                           0.0%]    47 [|                          0.6%]
  12 [|                           1.9%]    24 [||||||||||||||||||||||||||||||  76.9%]    36 [|                          1.3%]    48 [                           0.0%]
  Mem[||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||9.72G/126G]    Tasks: 213, 399 thr; 5 running
  Swp[|||||||||||||||||||||||||||||||                              1.48G/4.00G]    Load average: 3.62 3.60 3.56
                                                                                   Uptime: 2 days, 06:37:06

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
17911 test_245   20   0  114M  3540  1800 S  0.0  0.0  0:00.14 -bash
20515 test_245   20   0  114M  3336  1684 S  0.0  0.0  0:00.16 -bash
26675 root       20   0  114M  3536  1784 S  0.0  0.0  0:00.11 -bash
30798 M108wccho  20   0  118M  3740  1880 S  0.0  0.0  0:00.09 -bash
35240 root       20   0  114M  1588  1584 S  0.0  0.0  0:00.14 -bash
42337 M108ihtse  20   0  118M  1696  1692 S  0.0  0.0  0:00.17 -bash
42552 M108ihtse  20   0  118M  3556  1780 S  0.0  0.0  0:00.09 -bash
43147 M108wccho  20   0  118M  3540  1752 S  0.0  0.0  0:00.07 -bash
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

```
(base) [M108wcchou@eng05 Lab1]$ ./Lab1 input1
(base) [M108wcchou@eng05 Lab1]$ g++ -std=c++11 -pedantic -Wall -fomit-frame-pointer -funroll-al
l-loops Lab1.cpp -o Lab1 -O3 -DNDEBUG
(base) [M108wcchou@eng05 Lab1]$ ./Lab1 input1
(base) [M108wcchou@eng05 Lab1]$ █
```

```
int main(int argc, char *argv[])
{
        input.open(argv[1]);
        output.open(argv[2]);
        string s;
        stringstream temp, minterm, dontcare;
        int number, a;
        char trash;
        vector<string> m, new_m;

    getline(input, s);
    temp.clear();
    minterm.clear();
    dontcare.clear();
    temp.str(s);
    temp >> trash >> trash >> number;
    getline(input, s);
        getline(input, s);
        temp.clear();
        temp.str(s);
        minterm.str(s);
        while(temp >> a)
                m.push_back(tab(bin(a)));       //minterm
        getline(input, s);
        getline(input, s);
        temp.clear();
        temp.str(s);
        dontcare.str(s);
                                                                        57,1-8        18%
```

```
[test] 0:M108wcchou@eng05:~/CAD/Lab1*                                        "eng05" 20:17 09-Jul-20
```

Terminal Multiplexer (3/4)

- Session manipulation

command	action
tmux ls	list existed sessions
tmux new (-s) ([session_name])	create new session
tmux kill-session –t [session_name]	delete a specific session
tmux attach –t [session_name]	attach to a existed session
tmux detach / ctrl-b + d	detach from a session
tmux rename-session –t [old_name] [new_name]	rename a specific session

- Window manipulation

command	action
ctrl-b + c	create a new window
ctrl-b + n	move to the next window
ctrl-b + p	move to the previous window
ctrl-b + [number]	move to the designated window
ctrl-b + &	delete the current window

Terminal Multiplexer (4/4)

- Pane manipulation

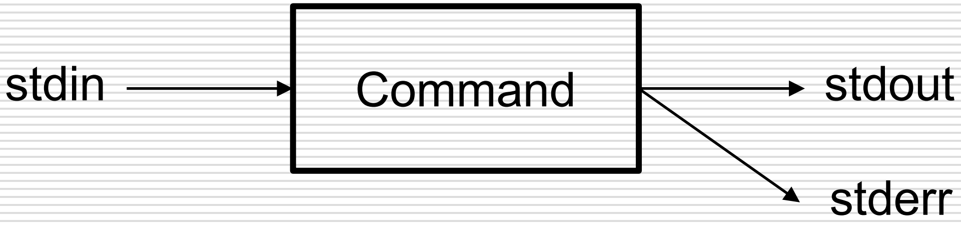
command	action
ctrl-b + %	split the current pane horizontally
ctrl-b + “	split the current pane vertically
ctrl-b-↑↓←→	resize the current pane
ctrl-b + x	delete the current pane
ctrl-b + ↑↓←→	switch the pane

- Practice:
  – Use tmux to create a session with 1 window which contains 3 panes

Advanced Linux Commands

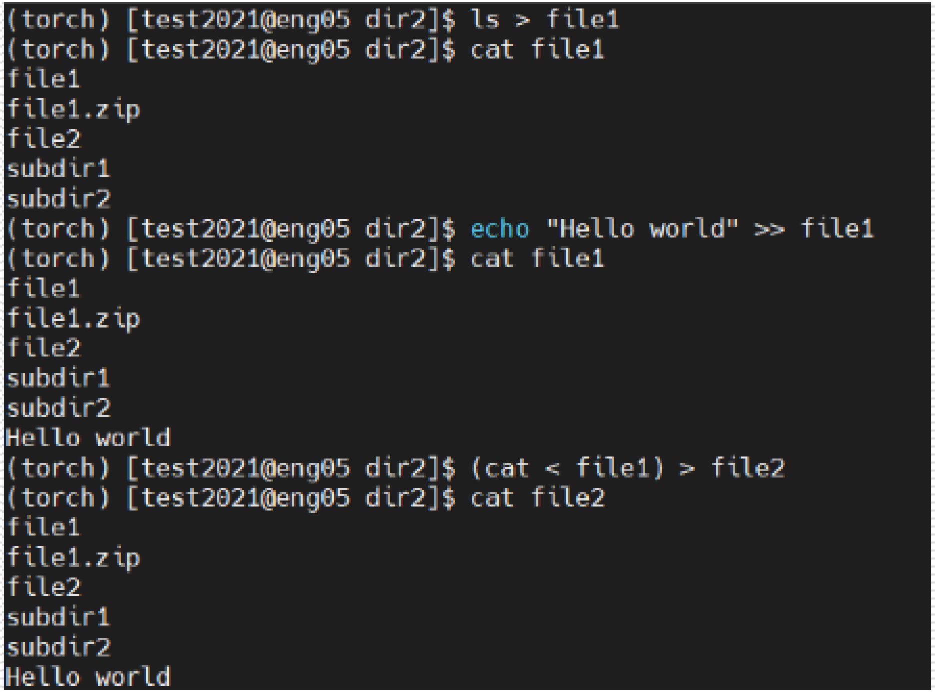
I/O Redirection (1/3)

- stdin
  - Standard input stream
  - Read input through the terminal (e.g., Keyboard input)
  - File descriptor 0
- stdout
  - Standard output stream
  - Produces output to the terminal
  - File descriptor 1
- stderr
  - standard error stream
  - Produces “error” output to the console
  - File descriptor 2



I/O Redirection (2/3)

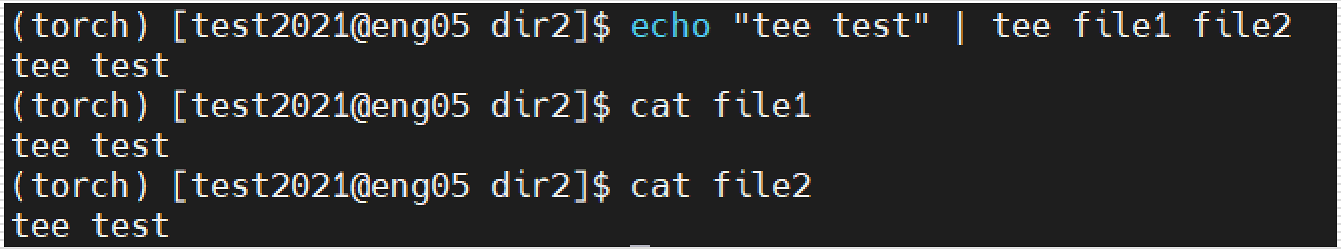
- Redirecting input (stdin)
  - “**<** ” : redirecting input from a file
- Redirecting output (stdout or stderr)
  - ***[file descriptor]>*** : Rewrite file
  - ***[file descriptor]>>*** : Do not rewrite file (append)



```
(torch) [test2021@eng05 dir2]$ ls > file1
(torch) [test2021@eng05 dir2]$ cat file1
file1
file1.zip
file2
subdir1
subdir2
(torch) [test2021@eng05 dir2]$ echo "Hello world" >> file1
(torch) [test2021@eng05 dir2]$ cat file1
file1
file1.zip
file2
subdir1
subdir2
Hello world
(torch) [test2021@eng05 dir2]$ (cat < file1) > file2
(torch) [test2021@eng05 dir2]$ cat file2
file1
file1.zip
file2
subdir1
subdir2
Hello world
```

I/O Redirection (3/3)

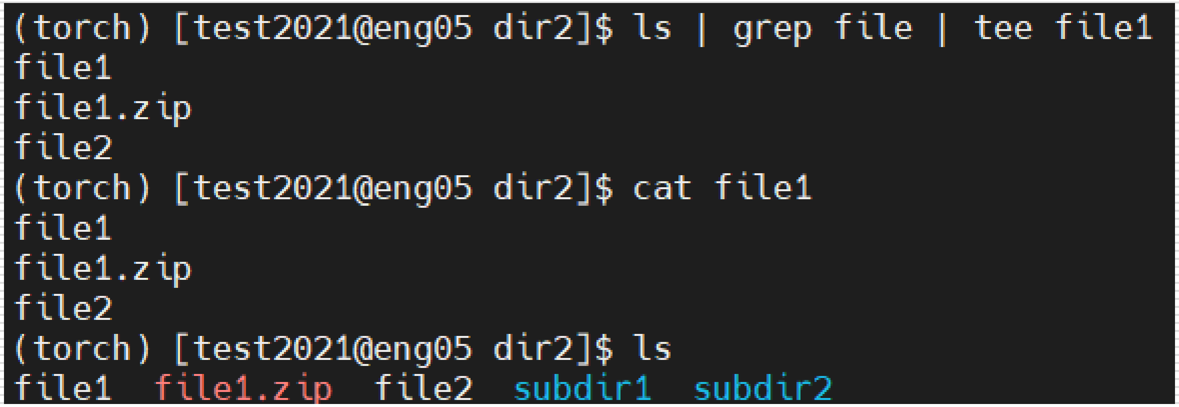
- Pipe the command
  - Use the stdout of previous command as the stdin of next command
    - $ ***[command 1] | [command 2] | …***
      - The command before “**|** ” must be able to read stdin
- Read the stdin and write it to both of the stdout and one or more files
  - $ ***[command] | tee [file]***



```
(torch) [test2021@eng05 dir2]$ echo "tee test" | tee file1 file2
tee test
(torch) [test2021@eng05 dir2]$ cat file1
tee test
(torch) [test2021@eng05 dir2]$ cat file2
tee test
```

Search within a File

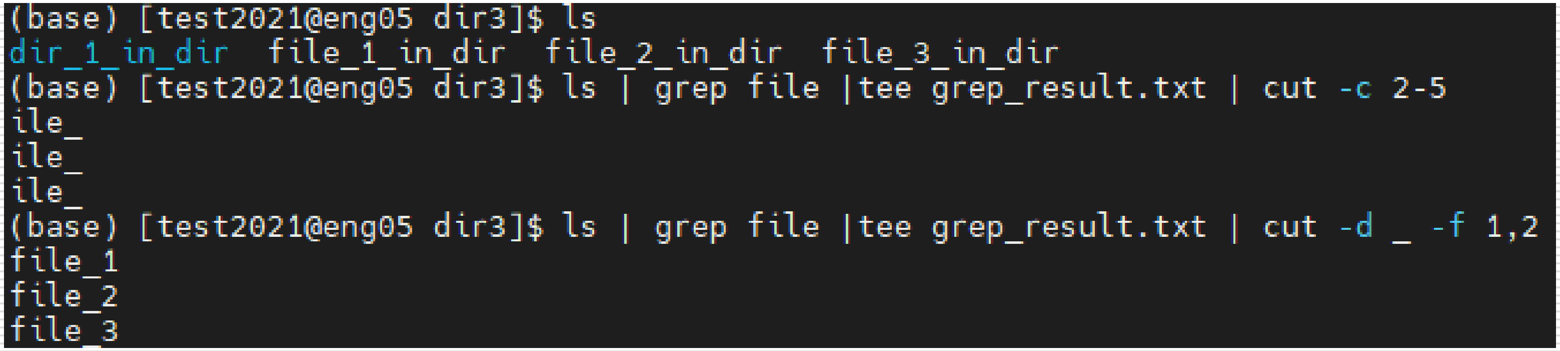
- Display the searching result of a particular pattern
  - $ ***grep [pattern] [file]***
  - $ ***[command]* | *grep [pattern]***



```
(torch) [test2021@eng05 dir2]$ ls | grep file | tee file1
file1
file1.zip
file2
(torch) [test2021@eng05 dir2]$ cat file1
file1
file1.zip
file2
(torch) [test2021@eng05 dir2]$ ls
file1  file1.zip  file2  subdir1  subdir2
```

Show Section of a File

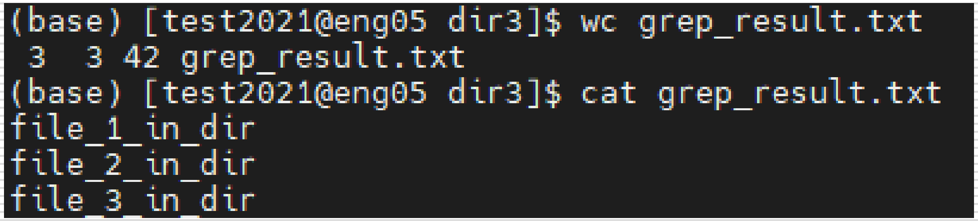
- Display the result of cutting out the section from each line of file to stdout
  - $ ***cut -c [range 1],[range 2]… [file]***
    - Cut out the section from the range of characters of each line
  - $ ***cut -d [separator] -f [field 1],[field 2]… [file]***
    - Cut the section from the field separating by the separator
  - $ ***[command] | cut -d [separator] -f [field 1],[field 2]…***



```
(base) [test2021@eng05 dir3]$ ls
dir_1_in_dir  file_1_in_dir  file_2_in_dir  file_3_in_dir
(base) [test2021@eng05 dir3]$ ls | grep file |tee grep_result.txt | cut -c 2-5
ile_
ile_
ile_
(base) [test2021@eng05 dir3]$ ls | grep file |tee grep_result.txt | cut -d _ -f 1,2
file_1
file_2
file_3
```

Count in a File

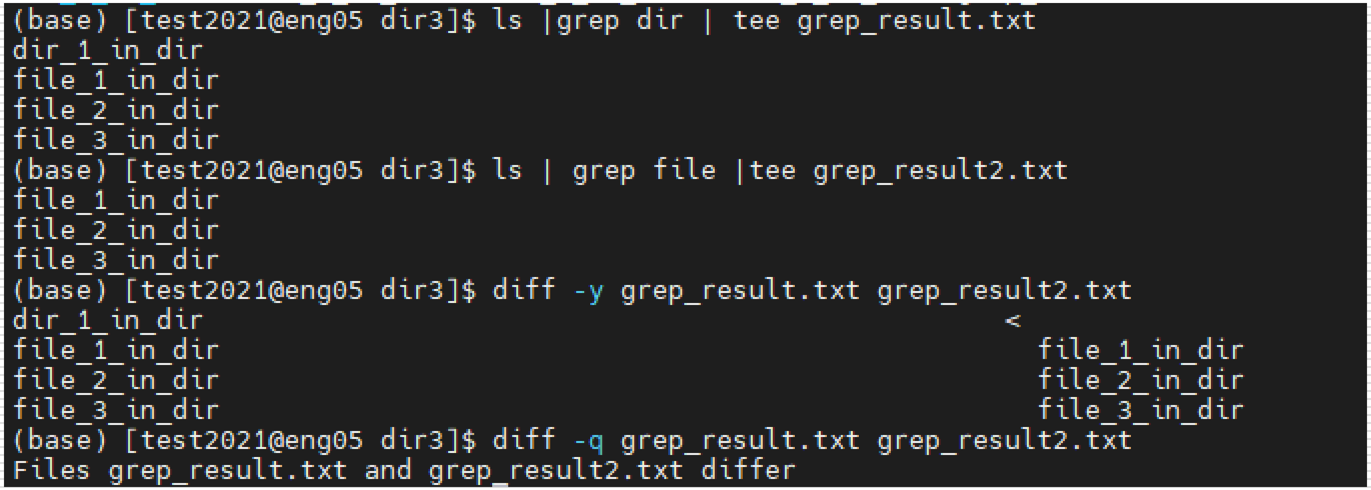
- Display the number of lines, words, and bytes in a file
  - $ ***wc [option] [file 1] [file 2] …***
  - $ ***[command] | wc [option]***
    - “***-l*** ” : Count lines
    - “***-w*** ” : Counting words
    - “***-b*** ” : Counting bytes
    - “***-m*** ” : Counting characters



```
(base) [test2021@eng05 dir3]$ wc grep_result.txt
 3  3 42 grep_result.txt
(base) [test2021@eng05 dir3]$ cat grep_result.txt
file_1_in_dir
file_2_in_dir
file_3_in_dir
```

Find Differences between Files

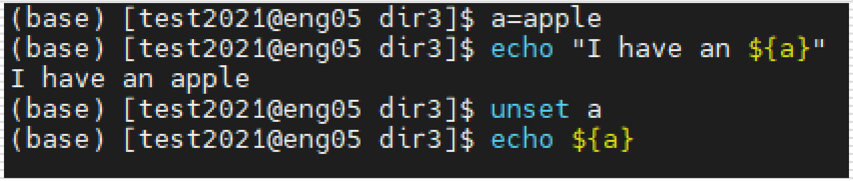
- Display the differences between files
  - $ ***diff [option] [file 1] [file 2]***
  - $ ***[command] | diff [file 1] - [option]***
    - “***-y*** ” : Show the files side-by-side and mark the difference
    - “***-q*** ” : Only show if there exists difference



```
(base) [test2021@eng05 dir3]$ ls |grep dir | tee grep_result.txt
dir_1_in_dir
file_1_in_dir
file_2_in_dir
file_3_in_dir
(base) [test2021@eng05 dir3]$ ls | grep file |tee grep_result2.txt
file_1_in_dir
file_2_in_dir
file_3_in_dir
(base) [test2021@eng05 dir3]$ diff -y grep_result.txt grep_result2.txt
dir_1_in_dir                                                  <
file_1_in_dir                                                   file_1_in_dir
file_2_in_dir                                                   file_2_in_dir
file_3_in_dir                                                   file_3_in_dir
(base) [test2021@eng05 dir3]$ diff -q grep_result.txt grep_result2.txt
Files grep_result.txt and grep_result2.txt differ
```

Shell Variables (1/3)

- We can define variable in shell
  - Name of shell variable can only contain letters (***a***-***z***, ***A***-***Z***), numbers (***0***-***9***), or underscore (**_**)
  - Only valid in current shell
  - Define shell variable
    - $ ***[variable name]=[variable value]***
      - Variable value can only be strings
      - Use “ ” to enclose the string if it contains “***space***”
  - Print shell variable
    - $ ***echo $[variable name]***
    - Use “{}” to separate variable name from other strings
  - Erase shell variable
    - $ ***unset [variable name]***



```
(base) [test2021@eng05 dir3]$ a=apple
(base) [test2021@eng05 dir3]$ echo "I have an ${a}"
I have an apple
(base) [test2021@eng05 dir3]$ unset a
(base) [test2021@eng05 dir3]$ echo ${a}
```

Shell Variables (2/3)

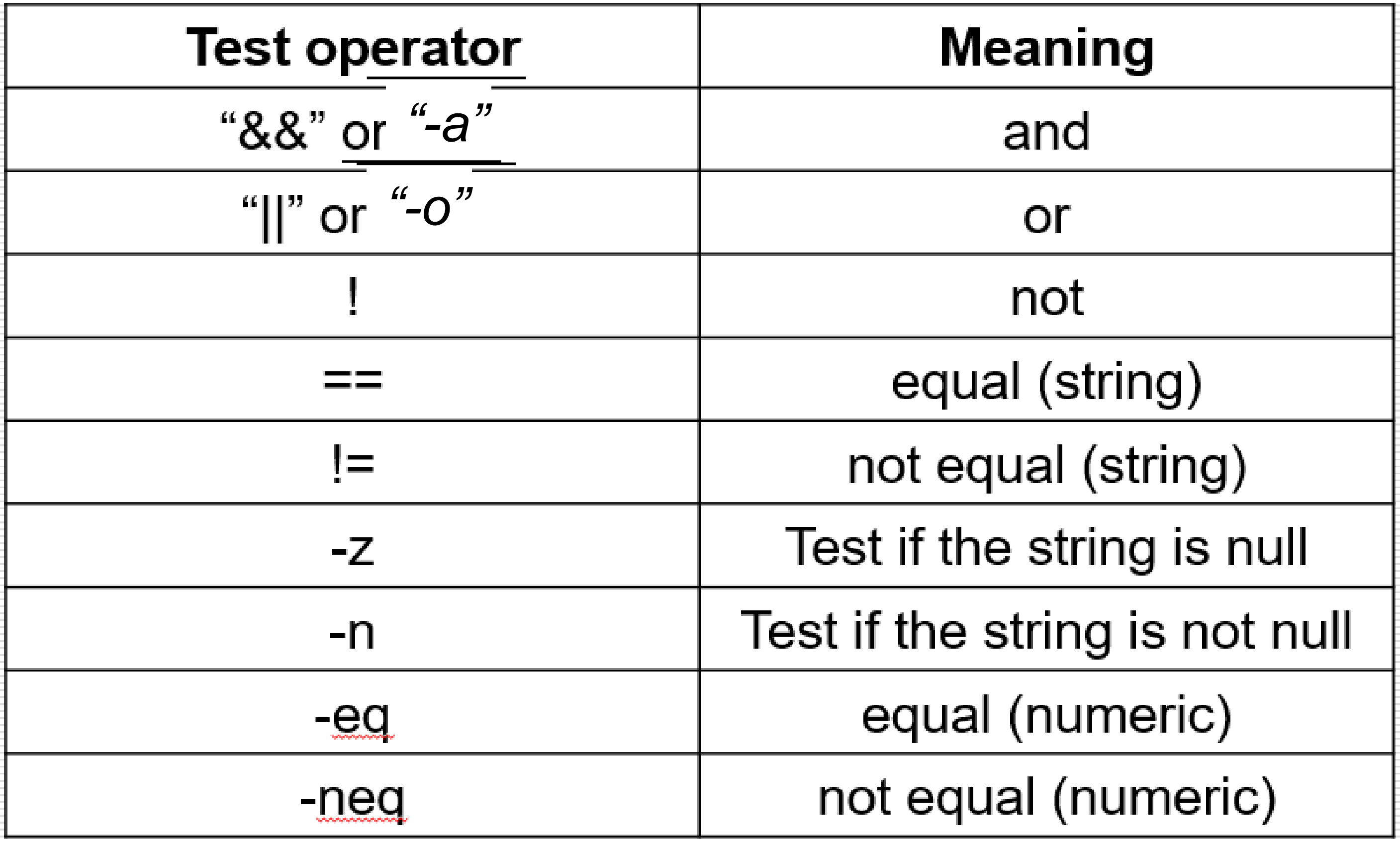
- Child process inherits all the environment variables from parent process
- Define environment variable
  - $ ***export [variable name]=[variable value]***
- Print environment variable
  - $ ***echo $[variable name]***
  - $ ***printenv [variable name]***
- Erase environment variable
  - $ ***unset [variable name]***

Shell Variables (3/3)

- Return value (***$?***)
  - Command returns a value as its status when it is finished
    - Zero : Command execute successfully
    - Non-zero : There are some failure during executing
  - Same as the value returned by “return” in C++ main function
  - Shell script can also return value with “***exit*** ” command

Conditional Expressions (1/2)

- Test condition
  - *[[condition]]*
- Test operators



Test operator	Meaning
“&&” or *“-a”*	and
“||” or *“-o”*	or
!	not
==	equal (string)
!=	not equal (string)
-z	Test if the string is null
-n	Test if the string is not null
-eq	equal (numeric)
-neq	not equal (numeric)

Conditional Expressions (2/2)

- Test operators

Test operator	Meaning
-gt	greater than
-lt	less than
-ge	greater than or equal
-le	less than or equal
-e	Test if a file or directory exists
-f	Test if a file exists
-d	Test if a directory exists

if & else Statement in Shell

- Syntax
  - *if [expression 1];*

    then

    …

    elif [expression 2];

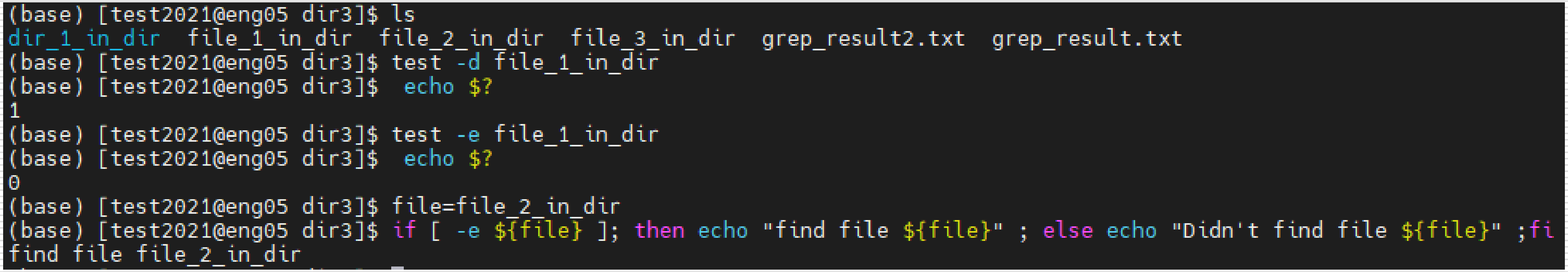
    then

    …

    else

    …

    fi



```
(base) [test2021@eng05 dir3]$ ls
dir_1_in_dir  file_1_in_dir  file_2_in_dir  file_3_in_dir  grep_result2.txt  grep_result.txt
(base) [test2021@eng05 dir3]$ test -d file_1_in_dir
(base) [test2021@eng05 dir3]$  echo $?
1
(base) [test2021@eng05 dir3]$ test -e file_1_in_dir
(base) [test2021@eng05 dir3]$  echo $?
0
(base) [test2021@eng05 dir3]$ file=file_2_in_dir
(base) [test2021@eng05 dir3]$ if [ -e ${file} ]; then echo "find file ${file}" ; else echo "Didn't find file ${file}" ;fi
find file file_2_in_dir
```

Loop (1/2)

- Syntax of while loop
  - ***while [expression]***

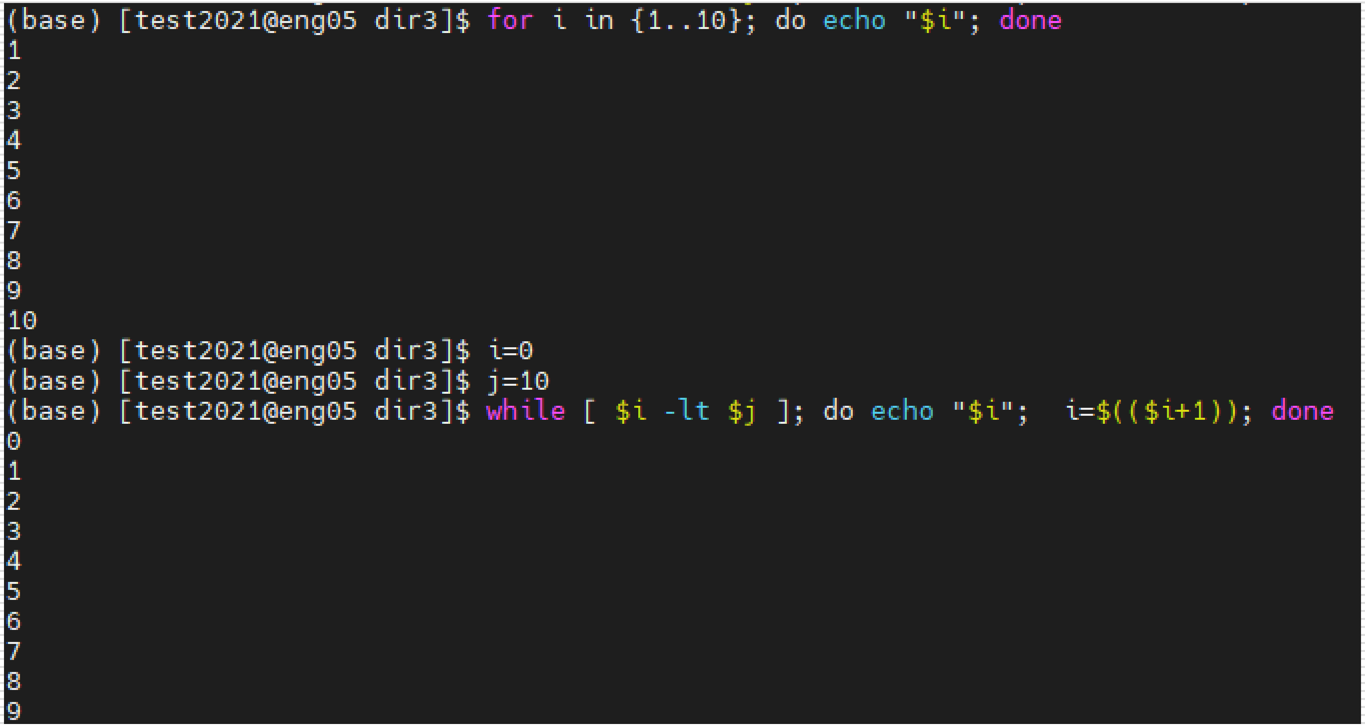
    do

    …

    done
- Syntax of for loop
  - ***for [variable] in [val 1] [val 2] …***
  - ***for [var] in $[val_var]***
  - ***for (([initializer]; [test expression]; [counting expresion]))***
- "***break*** ", and "***continue*** " are available

Loop (2/2)

- Example



```
(base) [test2021@eng05 dir3]$ for i in {1..10}; do echo "$i"; done
1
2
3
4
5
6
7
8
9
10
(base) [test2021@eng05 dir3]$ i=0
(base) [test2021@eng05 dir3]$ j=10
(base) [test2021@eng05 dir3]$ while [ $i -lt $j ]; do echo "$i";  i=$(($i+1)); done
0
1
2
3
4
5
6
7
8
9
```

Pass Arguments to Shell Scripts

- Arguments can be passed to the script when it is executed
  - ***[path/to/script] [argument 1] [argument 2] …***

    $0 ***$1*** ***$2***
  - In script
    - “***$0*** ” : Current script
    - “***$n*** ” : Specific argument passing to script, ***n*** **=** ***1***, ***2***, …
    - “***$#*** ” : Number of arguments passing to script
    - “***$**** ” : All arguments passing to script (seem as a string)
    - “***$@*** ”: All arguments passing to script
    - “***$?*** ” : Returned value of the latest command
    - “***$$*** ” : Working ID of current script

Regular Expressions (1/2)

- Not every command supports the regular expression

Special Character	Meaning
^word	a line begins with "word"
word$	a line ends with "word"
.	any character
\	escape character
*	repetition of the previous character for any times
[n1-n2]	Any character between "n1" to "n2"
[^list]	Any character in "list"
\{n,m\}	Repetition of previous character for the times beteen "n" and "m"

Regular Expressions (2/2)

特殊符號	代表意義
[:alnum:]	代表英文大小寫字元及數字，亦即 0-9, A-Z, a-z
[:alpha:]	代表任何英文大小寫字元，亦即 A-Z, a-z
[:blank:]	代表空白鍵與 [Tab] 按鍵兩者
[:cntrl:]	代表鍵盤上面的控制按鍵，亦即包括 CR, LF, Tab, Del.. 等等
[:digit:]	代表數字而已，亦即 0-9
[:graph:]	除了空白字元 (空白鍵與 [Tab] 按鍵) 外的其他所有按鍵
[:lower:]	代表小寫字元，亦即 a-z
[:print:]	代表任何可以被列印出來的字元
[:punct:]	代表標點符號 (punctuation symbol)，亦即：" ' ? ! ; : # $...
[:upper:]	代表大寫字元，亦即 A-Z
[:space:]	任何會產生空白的字元，包括空白鍵, [Tab], CR 等等
[:xdigit:]	代表 16 進位的數字類型，因此包括： 0-9, A-F, a-f 的數字與字元

References

References

- 鳥哥的 Linux 私房菜
  - http://linux.vbird.org/
- Linux 命令大全| 菜鳥教程
  - https://www.runoob.com/linux/linux-command-manual.html
- Linux命令大全(手冊) – 真正好用的Linux命令在線查詢網站
  - https://www.linuxcool.com/
- 簡明Linux Shell Script 入門教學 - TechBridge 技術共筆部落格
  - https://blog.techbridge.cc/2019/11/15/linux-shell-script-tutorial/

Thank you