

Objectives:

1. เข้าใจวิธีการติดตั้งและเริ่มใช้งาน Docker Desktop Container ขั้นพื้นฐาน พร้อมการตรวจสอบและจัดการสถานะของ container ในนั้น
2. ทดลองใช้งาน Ubuntu Linux Container ใน Docker ที่ติดตั้งขึ้นก่อนหน้านี้
3. มีทักษะเบื้องต้นในการใช้งานคำสั่งของ Linux's shell (command line UI)

Steps

1. Install Docker Desktop on your computer by following the instruction of Part 1.

Part 1 *Installation of Docker Desktop*

2. Study how to use basic Docker with Part 2.

Part 2: *Introduction to Docker*

- ☐ (ส่งตรวจ) จุดตรวจที่ 1 หลังคำสั่งลบ container ที่รัน busybox ออกไป
- ☐ (ส่งตรวจ) จุดตรวจที่ 2 หลังคำสั่ง docker container prune (แสดงให้เห็นว่าลบ images ทั้งหมดแล้ว)

Part 3: *Use Ubuntu Linux inside Docker*

- ☐ (ส่งตรวจ) จุดตรวจที่ 3 หลังออกจากการทดลองใช้ Python ใน Ubuntu ของ Docker
- ☐ (ส่งตรวจ) จุดตรวจที่ 4 หลังเรียกคำสั่ง cat passwd ใน /etc
- ☐ (ส่งตรวจ) จุดตรวจที่ 5 หลังเรียกคำสั่ง history ใน ubuntu

Part 1 *Installation of Docker Desktop*

1. Go to web of Docker to install ***Docker Desktop*** (about 511 MB!) onto your PC.

The screenshot shows the Docker Desktop installation page for Windows. The page title is "Install Docker Desktop on Windows" with an estimated reading time of 8 minutes. It includes a sidebar with navigation links like Overview, Mac, Windows, and various Docker Desktop features. The main content area has a "Download Docker Desktop for Windows" button, a "System requirements" section with tabs for WSL 2 backend and Hyper-V backend, and a list of requirements for WSL 2 backend.

Install Docker Desktop on Windows
Estimated reading time: 8 minutes

Welcome to Docker Desktop for Windows. This page contains information about Docker Desktop for Windows system requirements, download URL, instructions to install and update Docker Desktop for Windows.

Download Docker Desktop for Windows

System requirements

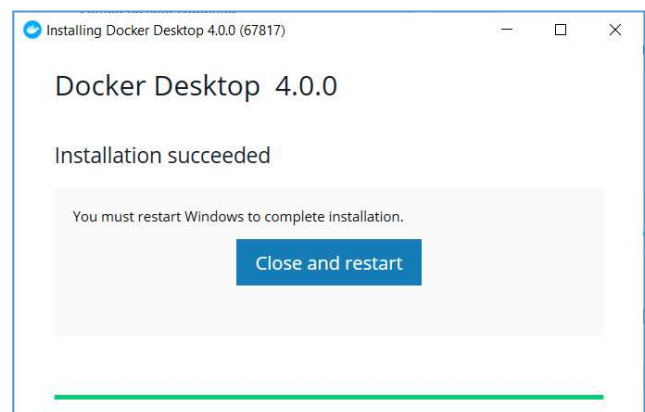
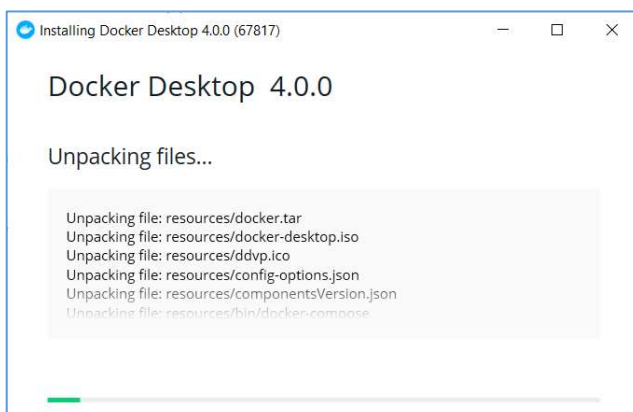
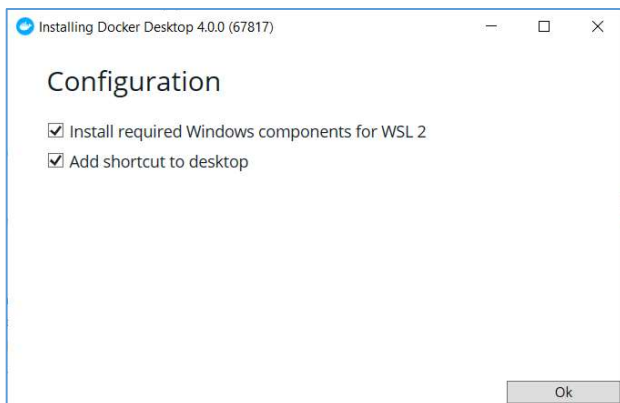
Your Windows machine must meet the following requirements to successfully install Docker Desktop.

WSL 2 backend | [Hyper-V backend and Windows containers](#)

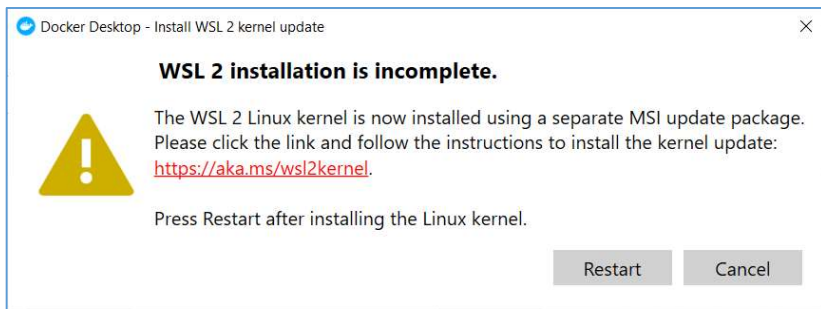
WSL 2 backend

- Windows 10 64-bit: Home or Pro 2004 (build 19041) or higher, or Enterprise or Education 1909 (build 18363) or higher.
- Enable the WSL 2 feature on Windows. For detailed instructions, refer to the [Microsoft documentation](#).
- The following hardware prerequisites are required to successfully run WSL 2 on Windows 10:
 - 64-bit processor with [Second Level Address Translation \(SLAT\)](#)
 - 4GB system RAM
 - BIOS-level hardware virtualization support must be enabled in the BIOS settings. For more information, see [Virtualization](#).
- Download and install the [Linux kernel update package](#).

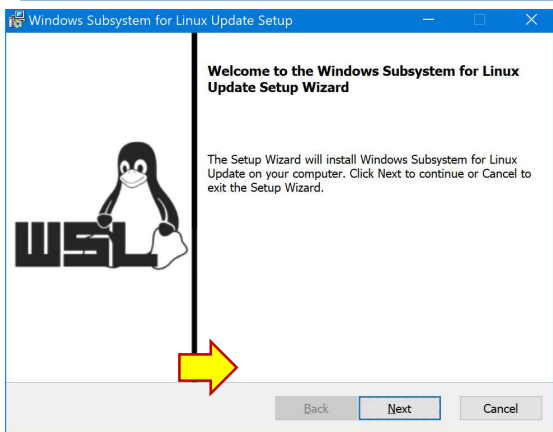
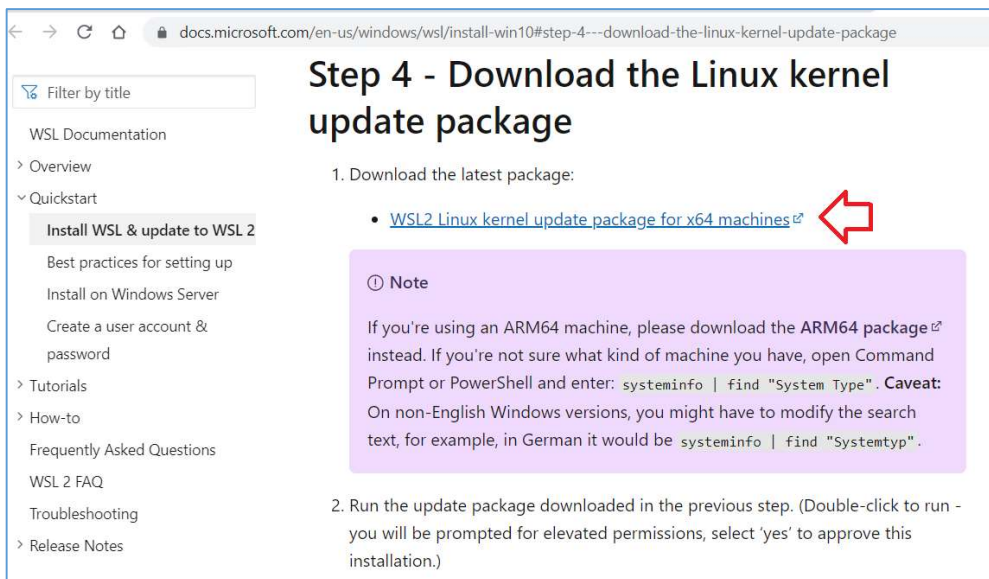
2. เมื่อสั่งให้ Install มันจะขึ้นหน้าต่างมาเพื่อยืนยันให้ติดตั้ง Windows components for WSL2 ด้วย



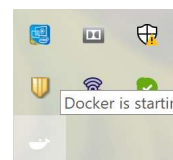
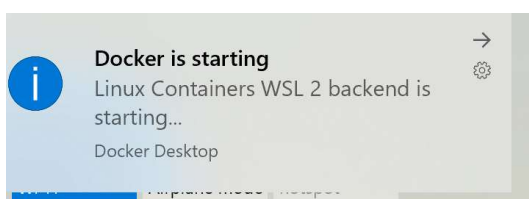
3. เมื่อ install แล้วอาจพบปัญหาว่า WSL 2 Installation is incomplete... ดังรูปต่อไปนี้



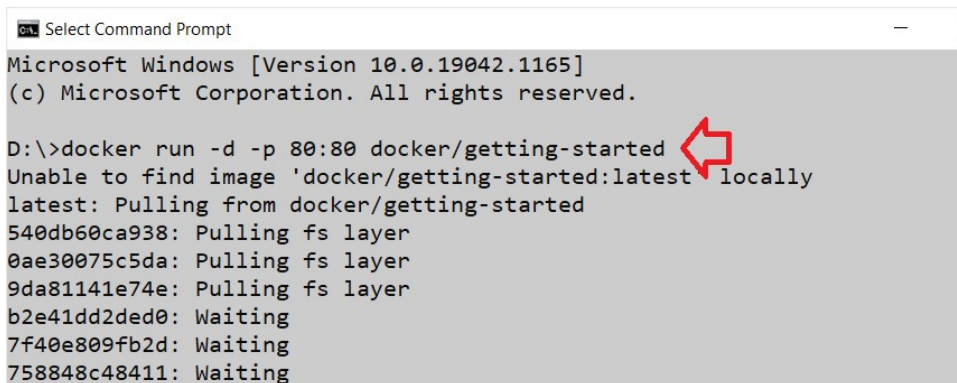
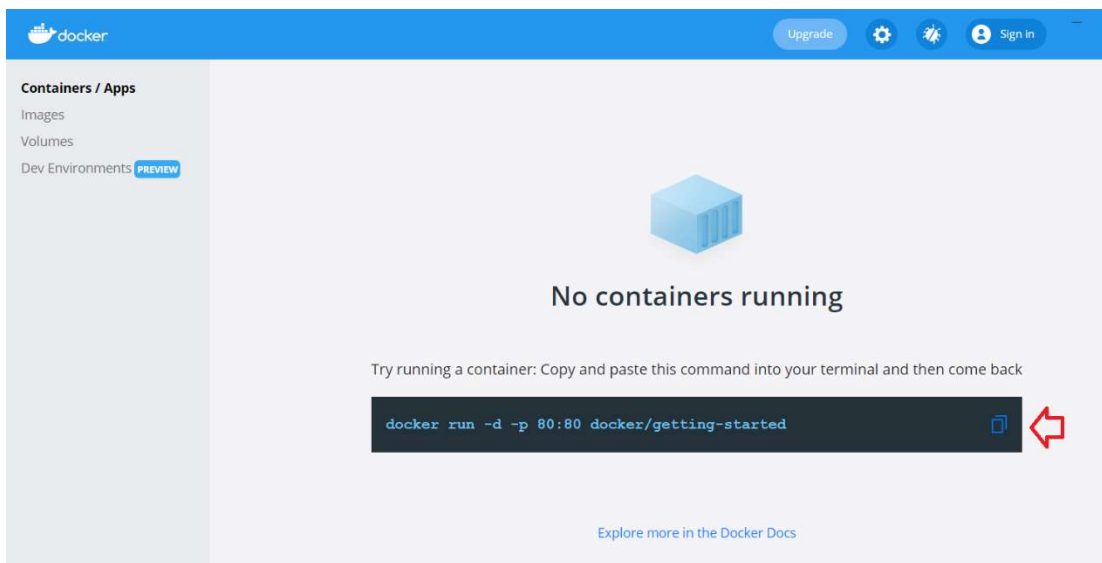
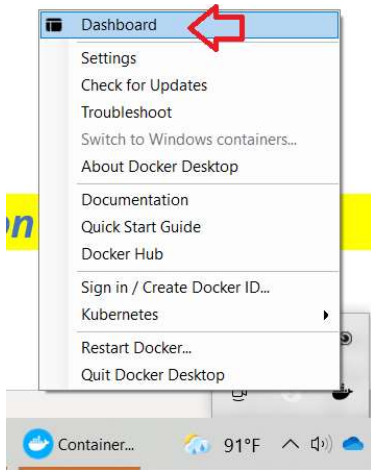
หากพบดังรูป ให้ click ไปที่ link aka.ms/wsl2kernel เพื่อติดตั้ง WSL2 Linux Kernel ให้เรียบร้อย



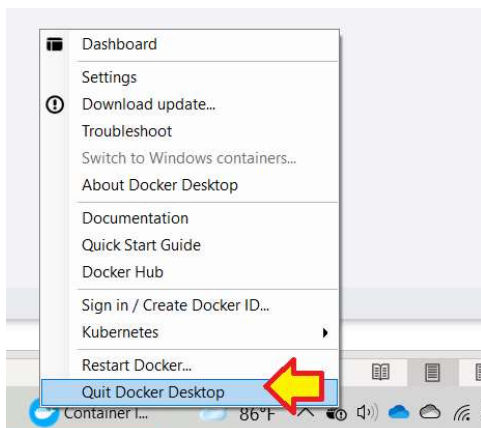
4. เมื่อติดตั้งแล้วเสร็จให้ Restart เครื่อง.... และเมื่อกลับมาจะพบ Docker icon แจ้งว่ากำลัง Starting....



5. เมื่อ Docker start เรียบร้อย ให้ลอง click ขวาที่ icon นั้น จะพบเมนูคำสั่งดังต่อไปนี้ ให้ลองเลือก Dashboard



6. หากไม่ต้องการใช้ Docker Desktop ให้ Quit ออกได้ แต่ต้องเรียก **Docker Desktop** ใหม่จึงจะใช้งานได้

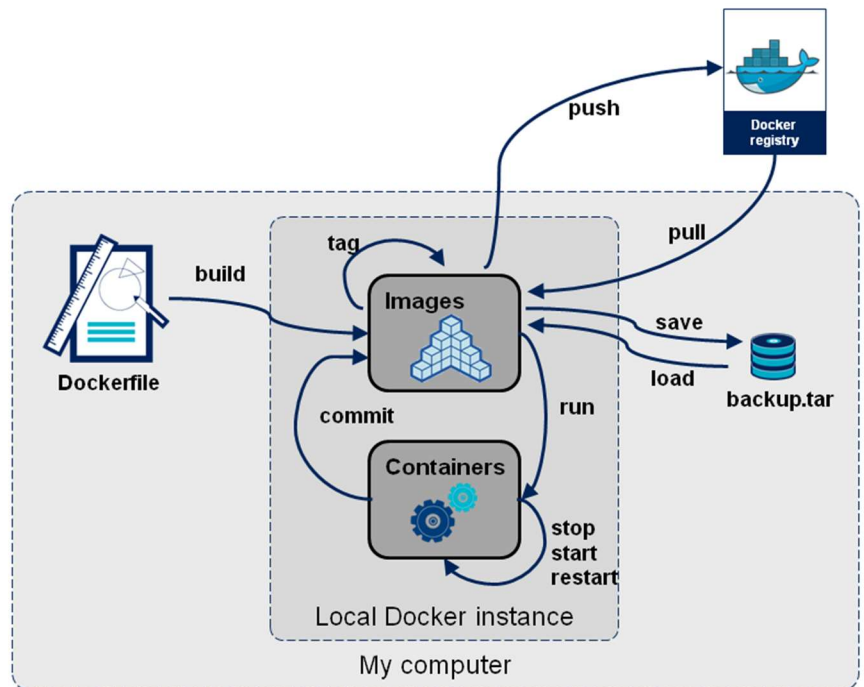


Part 2: *Introduction to Docker*



Terminology

- **Images** - The blueprints of our application which form the basis of containers. In this demo, we will be using the `docker pull` command to download the **busybox** image.
- **Containers** - Created from Docker images and run the actual application. We create a container using `docker run` which we did using the busybox image that we downloaded. A list of running containers can be seen using the `docker ps` command.
- **Docker Daemon** - The background service running on the host that manages building, running and distributing Docker containers. The daemon is the process that runs in the operating system to which clients talk to.
- **Docker Client** - The command line tool that allows the user to interact with the daemon.
- **Docker Hub** - A [registry](#) of Docker images. You can think of the registry as a directory of all available Docker images. If required, one can host their own Docker registries and can use them for pulling images.



Let's Start Our Journey

หลังจากติดตั้ง Docker เสร็จ เข้า cmd เพื่อทดสอบ...

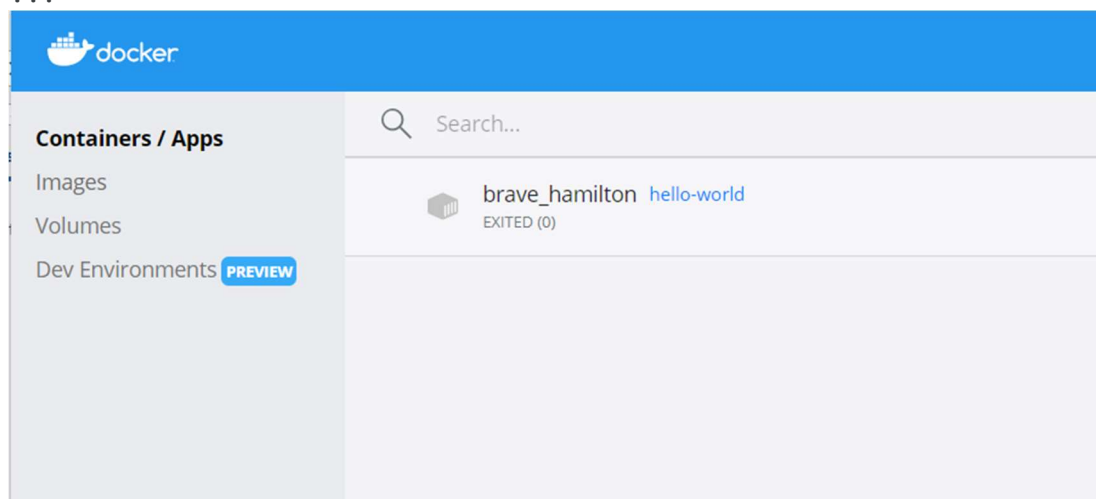
```
D:\>docker run hello-world
```

```
Unable to find image 'hello-world:latest' locally
Latest: Pulling .....
```

```
Hello from Docker.
```

```
This message shows that your installation appears to be working correctly.
```

```
...
```



D:\>**docker ps** # ดูว่ามี container ไດอยู่บ้าง

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

D:\>**docker ps -a**

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
2e131c1ccabd	hello-world	"/hello"	About a minute ago	Exited (0)	About a minute ago	brave_hamilton

D:\>**docker pull busybox** # ดึง (download) image ชื่อ busybox เข้ามา

```
Using default tag: latest
latest: Pulling from library/busybox
Digest: sha256:b37dd066f59a4961024cf4bed74cae5e68ac26b48807292bd12198afa3ecb778
Status: Image is up to date for busybox:latest
docker.io/library/busybox:latest
```

D:\>**docker run -it busybox** # เรียกใช้ image busybox พร้อม terminal + interactive mode

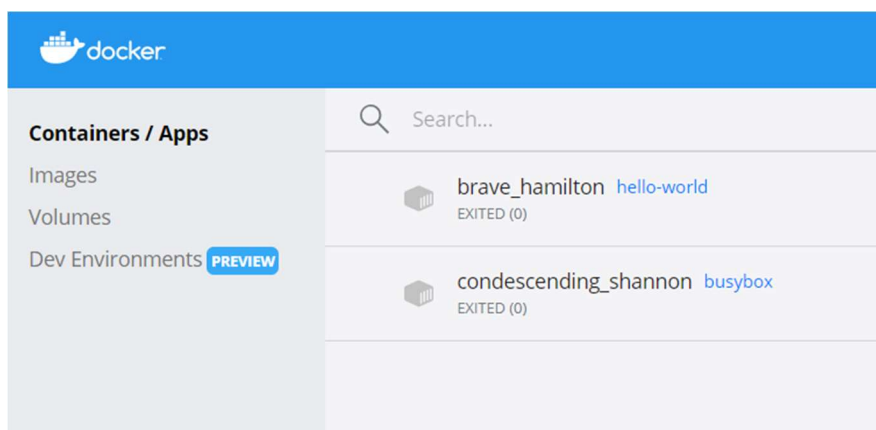
```
/ # uname
Linux
/ # uname -a
Linux 3692118e3a9e 5.10.16.3-microsoft-standard-WSL2 #1 SMP Fri Apr 2 22:23:49 UTC
2021 x86_64 GNU/Linux
/ # ls
bin dev etc home proc root sys tmp usr var
/ # pwd
/
/ # whoami
root
/ # exit
```

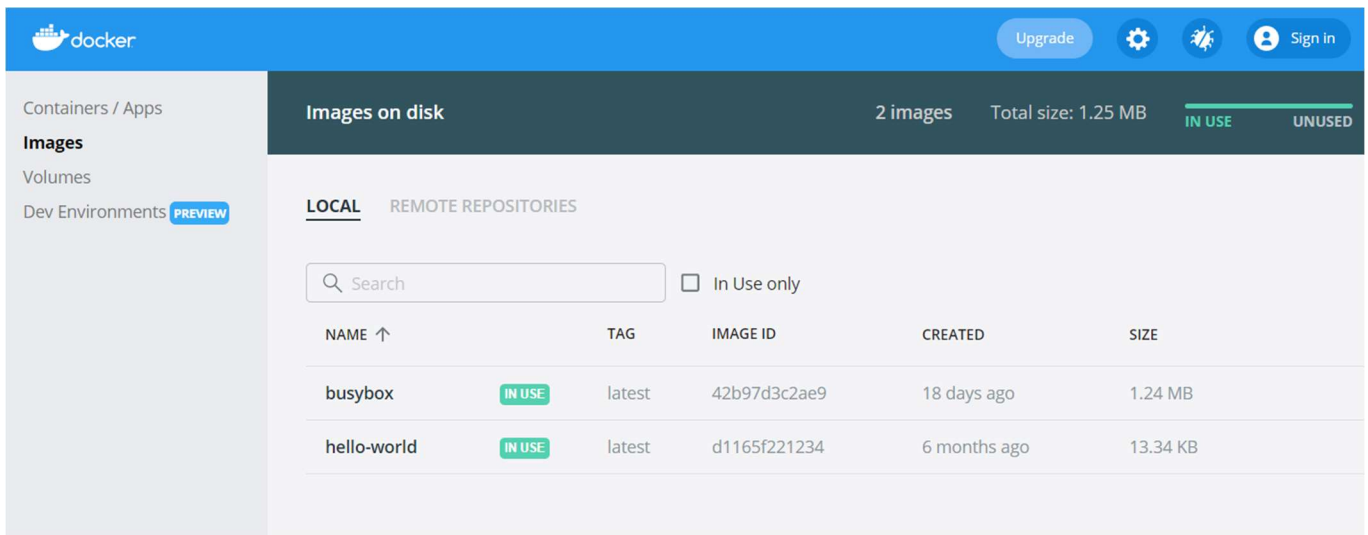
D:\>**docker ps -a** # ดู process or containers ที่มีชื่อนั้น

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
3692118e3a9e	busybox	"sh"	About a minute ago	Exited (0)
2e131c1ccabd	hello-world	condescending_shannon	5 minutes ago	Exited (0)

D:\>**docker images** # ดู images ที่มี

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
busybox	latest	42b97d3c2ae9	2 weeks ago	1.24MB
hello-world	latest	d1165f221234	6 months ago	13.3kB





D:\>**docker rm** <ชื่อ container ที่ต้องการ>**condes.....** # ลบ containers ที่ต้องการออก
condes.....

D:\>**docker ps -a**

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
2e131c1ccabd	hello-world	"/hello"	9 minutes ago	Exited (0) 9 minutes ago		brave_hamilton

☐ (ส่งตรวจ) จุดตรวจที่ 1 หลังคำสั่งลบ container ที่รัน busybox ออกไป

D:\>**docker run -it --name UNBOX busybox** # run image พร้อมกำหนดชื่อให้ container
/ # exit

ให้ นศ. ลอง run แล้วกำหนดชื่อเป็น DD-Firstname ของตัวเอง

D:\>**docker ps -a**

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
3e1d1127cd51	busybox	"sh"	6 seconds ago	Exited (0) 2 seconds ago		UNBOX
2e131c1ccabd	hello-world	"/hello"	12 minutes ago	Exited (0) 12 minutes ago		brave_hamilton

D:\>**docker image rm busybox** # ลบ image

Error response from daemon: conflict: unable to remove repository reference "busybox" (must force) - container 3e1d1127cd51 is using its referenced image 42b97d3c2ae9

D:\>**docker image rm --force busybox**

Untagged: busybox:latest

Untagged: busybox@sha256:b37dd066f59a4961024cf4bed74cae5e68ac26b48807292bd12198afa3ecb778

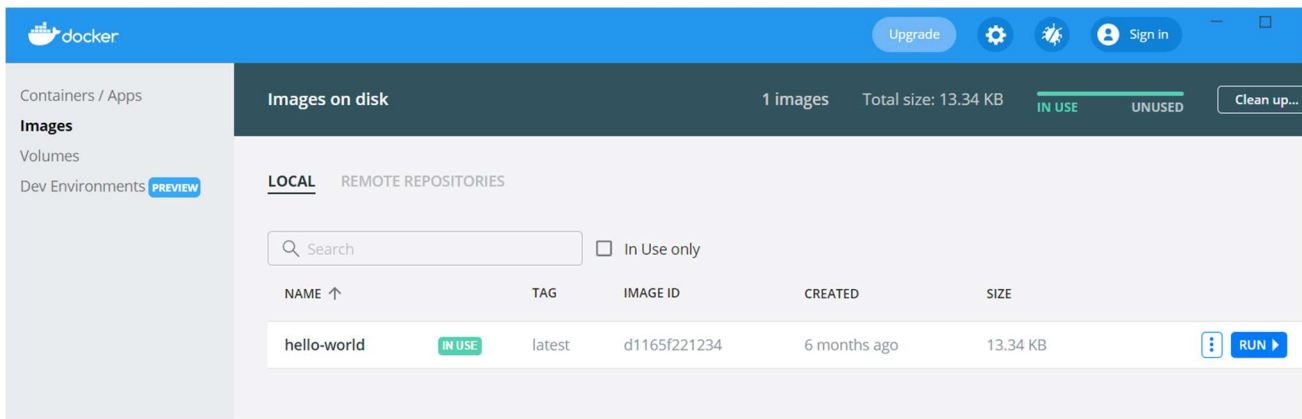
Deleted: sha256:42b97d3c2ae95232263a04324aaf656dc80e7792dee6629a9eff276cdfb806c0

D:\>**docker images**

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
hello-world	latest	d1165f221234	6 months ago	13.3kB

D:\>**docker ps -a**

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
3e1d1127cd51	42b97d3c2ae9	"sh"	3 minutes ago	Exited (0) 3 minutes ago		UNBOX
2e131c1ccabd	hello-world	"/hello"	15 minutes ago	Exited (0) 15 minutes ago		brave_hamilton



D:\>**docker container prune** # Removes all stopped containers.

WARNING! This will remove all stopped containers.

Are you sure you want to continue? [y/N] y

Deleted Containers:

3e1d1127cd51be47124142cb7a6396b0bb28a34db631a6f6e36a6b32ad1f202c
2e131c1ccabd4f9aef641c84d0a8102b98f1d37692699ff4291abd8096400229

Total reclaimed space: 5B

D:\>**docker ps -a**

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

Q: When you need some helps...

D:\>**docker ps --help**

Usage: docker ps [OPTIONS]

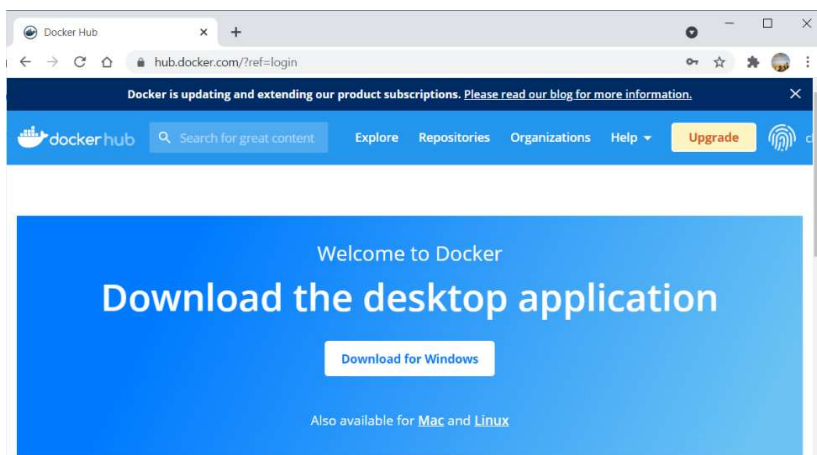
List containers

Options:

-a, --all	Show all containers (default shows just running)
-f, --filter filter	Filter output based on conditions provided
--format string	Pretty-print containers using a Go template
-n, --last int	Show n last created containers (includes all states) (default -1)
-l, --latest	Show the latest created container (includes all states)
--no-trunc	Don't truncate output
-q, --quiet	Only display container IDs
-s, --size	Display total file sizes

Q: แล้ว busybox หรือ ubuntu images มาจากไหน?

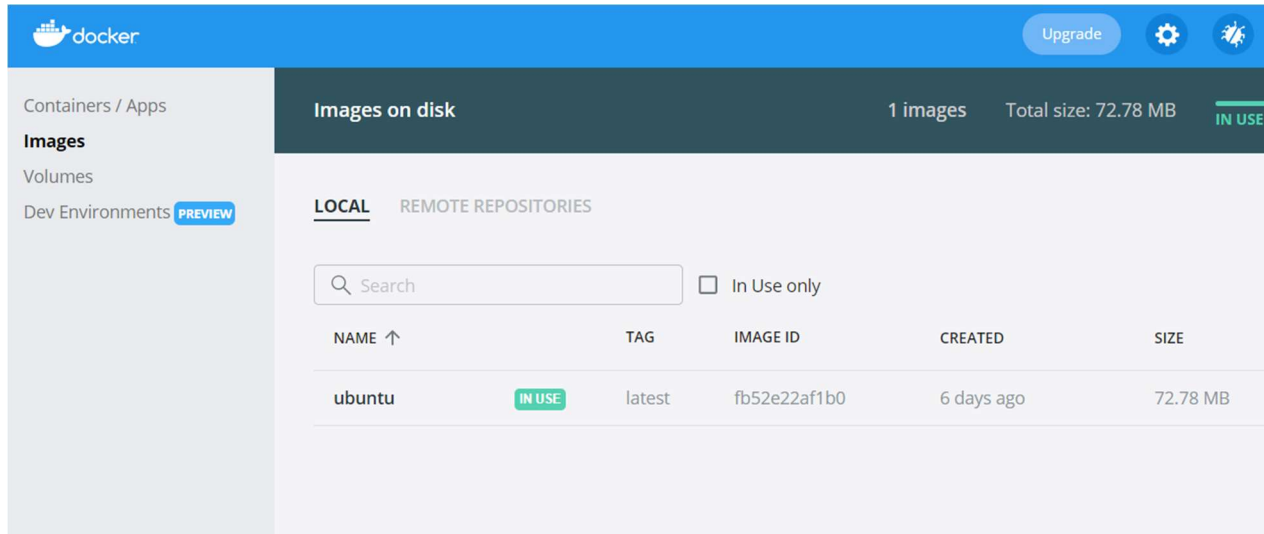
[Docker hub](https://hub.docker.com/) repository



□ (ส่งตรวจ) จุดตรวจที่ 2 หลังคำสั่ง **docker container prune** (แสดงให้เห็นว่าลบ images ทั้งหมดแล้ว)

Part 3: Use Ubuntu Linux inside Docker

Start by calling **cmd** or **command** of Windows 10



```
Microsoft Windows [Version 10.0.19042.1165]
(c) Microsoft Corporation. All rights reserved.
```

```
D:\> docker run -h gandalf -it ubuntu
```

เรียกใช้ Ubuntu บน
Docker โดยตั้งชื่อ
hostname เป็น gandalf

ให้ นศ. ตั้งชื่อเครื่อง (hostname) เป็น ID-Firstname (โดย ID คือ ID สองหลักท้าย)

```
root@99-gandalf:/# apt-get update
```

เพื่อ update software ต่างๆ ใน ubuntu ที่ใช้งานอยู่

```
Get:1 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal InRelease [265 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security/restricted amd64
Packages [525 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal-backports InRelease [101 kB]
.....
Get:17 http://archive.ubuntu.com/ubuntu focal-backports/main amd64 Packages
[2668 B]
Get:18 http://archive.ubuntu.com/ubuntu focal-backports/universe amd64
Packages [6324 B]
Fetched 17.8 MB in 1min 2s (286 kB/s)
Reading package lists... Done
```

```
root@gandalf:/# apt install python3.9
```

ติดตั้ง python 3.9

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ca-certificates libpython3.9-minimal libpython3.9-stdlib openssl python3.9-
minimal tzdata
Suggested packages:
  python3.9-venv python3.9-doc binutils binfmt-support
The following NEW packages will be installed:
  ca-certificates libpython3.9-minimal libpython3.9-stdlib openssl python3.9
python3.9-minimal tzdata
0 upgraded, 7 newly installed, 0 to remove and 0 not upgraded.
Need to get 6039 kB of archives.
After this operation, 25.6 MB of additional disk space will be used.
```

Do you want to continue? [Y/n] **y**

```
Get:1 http://archive.ubuntu.com/ubuntu focal-updates/universe amd64
libpython3.9-minimal amd64 3.9.5-3~20.04.1 [756 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal-updates/universe amd64 python3.9-
minimal amd64 3.9.5-3~20.04.1 [2022 kB]
Get:3 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 openssl amd64
1.1.1f-1ubuntu2.8 [620 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 ca-certificates all
20210119~20.04.1 [146 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 tzdata all 2021a-
0ubuntu0.20.04 [295 kB]
```

```
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the
Term::ReadLine module) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-
gnu/perl/5.30.0 /usr/local/share/perl/5.30.0 /usr/lib/x86_64-linux-gnu/perl5/5.30
/usr/share/perl5 /usr/lib/x86_64-linux-gnu/perl/5.30 /usr/share/perl/5.30
/usr/local/lib/site_perl /usr/lib/x86_64-linux-gnu/perl-base) at
/usr/share/perl5/Debconf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
Configuring tzdata
-----
```

Please select the geographic area in which you live. Subsequent configuration questions will narrow this down by presenting a list of cities, representing the time zones in which they are located.

1. Africa 3. Antarctica 5. Arctic 7. Atlantic 9. Indian 11. SystemV 13. Etc
2. America 4. Australia 6. Asia 8. Europe 10. Pacific 12. US
Geographic area: **6**

Please select the city or region corresponding to your time zone.

1. Aden	14. Beirut	27. Gaza	40. Karachi	53. Muscat	66. Riyadh	79. Tokyo
2. Almaty	15. Bishkek	28. Harbin	41. Kashgar	54. Nicosia	67. Sakhalin	80. Tomsk
3. Amman	16. Brunei	29. Hebron	42. Kathmandu	55. Novokuznetsk	68. Samarkand	81. Ujung_Pandang
4. Anadyr	17. Chita	30. Ho_Chi_Minh	43. Khandyga	56. Novosibirsk	69. Seoul	82. Ulaanbaatar
5. Aqtou	18. Chobalsan	31. Hong_Kong	44. Kolkata	57. Omsk	70. Shanghai	83. Urumqi
6. Aqtobe	19. Chongqing	32. Hovd	45. Krasnoyarsk	58. Oral	71. Singapore	84. Ust-Nera
7. Ashgabat	20. Colombo	33. Irkutsk	46. Kuala_Lumpur	59. Phnom_Penh	72. Srednekolymysk	85. Vientiane
8. Atyrau	21. Damascus	34. Istanbul	47. Kuching	60. Pontianak	73. Taipei	86. Vladivostok
9. Baghdad	22. Dhaka	35. Jakarta	48. Kuwait	61. Pyongyang	74. Tashkent	87. Yakutsk
10. Bahrain	23. Dili	36. Jayapura	49. Macau	62. Qatar	75. Tbilisi	88. Yangon
11. Baku	24. Dubai	37. Jerusalem	50. Magadan	63. Qostanay	76. Tehran	89. Yekaterinburg
12. Bangkok	25. Dushanbe	38. Kabul	51. Makassar	64. Qyzylorda	77. Tel_Aviv	90. Yerevan
13. Barnaul	26. Famagusta	39. Kamchatka	52. Manila	65. Rangoon	78. Thimphu	

Time zone: **12**

```
Current default time zone: 'Asia/Bangkok'
Local time is now:      Wed Sep  8 13:46:04 +07 2021.
Universal Time is now:  Wed Sep  8 06:46:04 UTC 2021.
Run 'dpkg-reconfigure tzdata' if you wish to change it.
```

```
Setting up python3.9-minimal (3.9.5-3~20.04.1) ...
Setting up openssl (1.1.1f-1ubuntu2.8) ...
Setting up libpython3.9-stdlib:amd64 (3.9.5-3~20.04.1) ...
Setting up ca-certificates (20210119~20.04.1) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend
cannot be used. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 76.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the
Term::ReadLine module) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-
gnu/perl/5.30.0 /usr/local/share/perl/5.30.0 /usr/lib/x86_64-linux-gnu/perl5/5.30
/usr/share/perl5 /usr/lib/x86_64-linux-gnu/perl/5.30 /usr/share/perl/5.30
/usr/local/lib/site_perl /usr/lib/x86_64-linux-gnu/perl-base) at
/usr/share/perl5/Debconf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
Updating certificates in /etc/ssl/certs...
129 added, 0 removed; done.
Setting up python3.9 (3.9.5-3~20.04.1) ...
```

```
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for ca-certificates (20210119~20.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
```

```
root@gandalf:/usr# python3.9
```

เรียก python3.9 ที่ลงไว้

```
Python 3.9.5 (default, May 19 2021, 11:32:47)
[GCC 9.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>> import math
>>> a = math.pi / 2
>>> print(math.sin(a))
1.0
```

ทดลองใช้ python3.9

```
>>> exit
Use exit() or Ctrl-D (i.e. EOF) to exit
>>> exit()
root@gandalf:/usr#
```

ออกจาก python3.9

□ (ส่งตรวจ) จุดตรวจที่ 3 หลังออกจากการทดลองใช้ Python ใน Ubuntu ของ Docker

```
root@gandalf:/# ls
```

```
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv
sys tmp usr var
```

```
root@gandalf:/# ls -la
```

ดูข้อมูล (ไฟล์ subdirectory ฯลฯ) ทั้งหมด
รวมทั้งที่ hidden (ขึ้นต้นด้วย .) ด้วย

```
total 72
drwxr-xr-x  1 root root 4096 Aug 28 06:02 .
drwxr-xr-x  1 root root 4096 Aug 28 06:02 ..
-rwxr-xr-x  1 root root    0 Aug 28 06:02 .dockerenv
drwxr-xr-x  2 root root 4096 Aug  7 13:03 bin
drwxr-xr-x  2 root root 4096 Apr 24 2018 boot
drwxr-xr-x  5 root root  360 Aug 28 06:02 dev
drwxr-xr-x  1 root root 4096 Aug 28 06:02 etc
drwxr-xr-x  2 root root 4096 Apr 24 2018 home
drwxr-xr-x  8 root root 4096 May 23 2017 lib
drwxr-xr-x  2 root root 4096 Aug  7 13:03 lib64
drwxr-xr-x  2 root root 4096 Aug  7 13:02 media
drwxr-xr-x  2 root root 4096 Aug  7 13:02 mnt
drwxr-xr-x  2 root root 4096 Aug  7 13:02 opt
dr-xr-xr-x 142 root root    0 Aug 28 06:02 proc
drwx-----  2 root root 4096 Aug  7 13:03 root
drwxr-xr-x  1 root root 4096 Aug 15 07:28 run
drwxr-xr-x  1 root root 4096 Aug 15 07:28 sbin
drwxr-xr-x  2 root root 4096 Aug  7 13:02 srv
dr-xr-xr-x 13 root root    0 Aug 28 05:43 sys
drwxrwxrwt  2 root root 4096 Aug  7 13:03 tmp
drwxr-xr-x  1 root root 4096 Aug  7 13:02 usr
drwxr-xr-x  1 root root 4096 Aug  7 13:03 var
```

```
root@gandalf:/# cd
```

ย้าย working directory ไปที่ \$HOME
นั่นคือ ~

```
root@gandalf:~# ls -a
.  ..  .bashrc  .profile
```

```
root@gandalf:~# uname -a
```

```
Linux 690f4ae411bb 5.10.16.3-microsoft-standard-WSL2 #1 SMP Fri Apr 2  
22:23:49 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
```

```
# see and study at https://linux.die.net/man/1/uname
```

```
root@gandalf:~# cd /
```

```
root@gandalf:/# ls -a
```

```
.    .dockerenv  boot  etc    lib      media  opt    root  sbin  sys  usr  
..   bin          dev   home  lib64  mnt     proc  run   srv  tmp  var
```

```
root@gandalf:/# ls bin
```

```
bash          bzless  dir      gzexe    more      rmdir      touch      zdiff  
bunzip2       bzmores dmesg    gzip     mount     run-parts  true       zegrep  
bzip2         cat     dnsdomainname hostname mountpoint sed         umount     zfgrep  
bzcat         chgrp   domainname kill      mv         sh         uname      zforce  
bzdiff        chmod   echo     ln        nisdomainname sh.distrib  uncompress zgrep  
bzegrep       chown   egrep    login    pidof     sleep      vdir       zless  
bzexe         cp      false    ls        ps         stty       wdctl      zmore  
bzfgrep       dash    fgrep    lsblk    pwd        su         which      znew  
bzgrep        date    findmnt  mkdir    rbash     sync       ypdomainname  
bzip2         dd      grep     mknod    readlink  tar        zcat  
bzip2recover  df      gunzip   mktemp   rm         tempfile   zcmp
```

```
root@gandalf:/# cd etc
```

```
root@gandalf:/etc# cat passwd
```

```
root:x:0:0:root:/root:/bin/bash  
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin  
bin:x:2:2:bin:/bin:/usr/sbin/nologin  
sys:x:3:3:sys:/dev:/usr/sbin/nologin  
sync:x:4:65534:sync:/bin:/bin/sync  
games:x:5:60:games:/usr/games:/usr/sbin/nologin  
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin  
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin  
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin  
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin  
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin  
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin  
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin  
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin  
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin  
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin  
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin  
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin  
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
```

เข้าไป /etc เพื่อดูโปรแกรมและข้อมูล
ระบบที่สำคัญๆ

□ (สังตรวจ) จุตตรวจที่ 4 หลังเรืยกคําลั้ **cat passwd** ใน **/etc** พร้อมอธิบายเขตข้อมูล (field) ต่าง
ๆ ของข้อมูลในไฟล์ดังกล่าวได้ ว่าแต่ละ field คืออะไร

```
root@gandalf:/etc# whoami
```

```
root
```

```
root@gandalf:/etc# ps
```

PID	TTY	TIME	CMD
1	pts/0	00:00:00	bash
17	pts/0	00:00:00	ps

แสดง processes ต่างๆ ของ
current user

```
root@gandalf:/etc# ps -aux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.1	18504	3416	pts/0	Ss	06:02	0:00	/bin/bash
root	18	0.0	0.1	34396	2844	pts/0	R+	06:04	0:00	ps -aux

```
root@gandalf:/etc# cd
```

```
root@gandalf:~# mkdir programs
```

```
root@gandalf:~# cd prog*
```

สร้าง subdirectory ด้วยชื่อของตัวเอง แล้ว
ย้าย (cd) เข้าไปข้างในนั้น

```
root@gandalf:~/programs# ls -al /bin > list-of-bin
```

```
root@gandalf:~/programs# cat list-*
```

```
total 4948
drwxr-xr-x 2 root root    4096 Aug  7 13:03 .
drwxr-xr-x 1 root root    4096 Aug 28 06:02 ..
-rwxr-xr-x 1 root root 1113504 Jun  6 22:28 bash
.....
-rwxr-xr-x 1 root root    35064 Jan 18  2018 cat
-rwxr-xr-x 1 root root    59608 Jan 18  2018 chmod
-rwxr-xr-x 1 root root    67768 Jan 18  2018 chown
-rwxr-xr-x 1 root root   141528 Jan 18  2018 cp
-rwxr-xr-x 1 root root    84776 Jan 18  2018 df
.....
-rwxr-xr-x 1 root root   219528 Jul 12  2017 grep
-rwxr-xr-x 2 root root     2301 Apr 28  2017 gunzip
-rwxr-xr-x 1 root root     5927 Apr 28  2017 gzexe
-rwxr-xr-x 1 root root   101560 Apr 28  2017 gzip
-rwxr-xr-x 1 root root    18504 Jan 31  2018 hostname
-rwxr-xr-x 1 root root    26704 May 14  2018 kill
-rwxr-xr-x 1 root root    67808 Jan 18  2018 ln
-rwxr-xr-x 1 root root    52664 Mar 22 19:05 login
-rwxr-xr-x 1 root root   133792 Jan 18  2018 ls
.....
-rwxr-xr-x 1 root root   133432 May 14  2018 ps
-rwxr-xr-x 1 root root    35000 Jan 18  2018 pwd
lrwxrwxrwx 1 root root         4 Jun  6 22:28 rbash -> bash
-rwxr-xr-x 1 root root    63704 Jan 18  2018 rm
-rwxr-xr-x 1 root root    43192 Jan 18  2018 rmdir
.....
-rwsr-xr-x 1 root root    44664 Mar 22 19:05 su
-rwxr-xr-x 1 root root   423312 Jan 21  2019 tar
```

redirect (เปลี่ยนทิศทาง) output ให้ไปลง
ใน ไฟล์แทน. More at:
<https://www.tutorialspoint.com/unix/unix-io-redirections.htm>

```
root@gandalf:~/programs# ls -l
total 8
-rw-r--r-- 1 root root 4510 Aug 28 08:45 list-of-bin
root@gandalf:~/programs# history
 1  ls
 2  ls -la
 3  cd
 4  ls -a
 5  cd /
 6  ls -a
 7  ls bin
 8  cd etc
 9  cat passwd
10  whoami
11  ps
12  ps -aux
13  cd
14  mkdir programs
15  cd prog*
16  ls -al /bin > list-of-bin
17  cat list-*
18  ls -l
19  history
```

ดูรายการคำสั่งทั้งหมดที่เรียกไป ในการ
login session ปัจจุบัน

□ (ส่งตรวจ) จดตรวจที่ 5 หลังเรียกคำสั่ง **history** ใน ubuntu

```
root@gandalf:~/programs# exit
```

Q1 หลังจากออกมาจาก Ubuntu นี้แล้ว Docker ที่ใช้งานมี Container ชื่ออะไรอยู่บ้าง แจ้ง Image Names กับ (Container) Names มาให้หมด

Q2 หากเรากลับไปเรียก `docker run ubuntu` อีกครั้ง เราจะเห็น subdirectory ชื่อตัวเองที่สร้างไว้ก่อนหน้านี้หรือไม่ เพราะเหตุ?

Q3 หากเรากลับไปเรียก `docker run ubuntu` อีกครั้ง เราจะใช้ python ได้หรือไม่ เพราะอะไร?

More basic-intermediate Linux's shell commands

<https://www.guru99.com/linux-commands-cheat-sheet.html>

Widely-Used Linux/Unix Shells Commands

ในการใช้งานหลายคำสั่ง จะต้องระบุ option เพิ่มเพื่อให้ทำงานได้ หรือกำหนดรูปแบบการทำงาน เช่น คำสั่ง cp อย่างง่ายที่สุดคือ cp sourcePath DestPath เป็นต้น หากไม่ทราบวิธีใช้งานคำสั่งไหนให้พิมพ์คำสั่งนั้นแล้วตามด้วย --help เช่น cp --help แล้วอ่านรายละเอียดการใช้งานคำสั่งนั้น ๆ

No.	Command	Description
1	adduser	ใช้เพิ่มหรือเปลี่ยนแปลง user บน Linux
2	arch	ใช้แสดงรุ่น hardware ของเครื่อง server
3	awk	ใช้ค้นหาข้อมูล text ในรูปแบบที่ซับซ้อน
4	basename	ใช้แสดงเฉพาะส่วนของชื่อ filename
5	bc	คำนวณตัวเลข ตามสูตรทางคณิตศาสตร์
6	cal	แสดงปฏิทิน วันเดือนปี
7	cat	แสดงผลข้อมูลภายใน file ในรูปแบบ text
8	chgrp	เปลี่ยนเจ้าของ group ของ file
9	chmod	เปลี่ยนสิทธิ์ในการเข้าถึง file
10	chown	เปลี่ยนเจ้าของ file หรือ directory
11	cksum	นับจำนวน bytes ของ file
12	clear	ล้างหน้าจอ screen
13	cmp	วิเคราะห์เปรียบเทียบ files ในระดับ bytes
14	comm	คำสั่งวิเคราะห์เปรียบเทียบ file ที่ละบรรทัด
15	cp	คำสั่งทำสำเนาหรือ copy ข้อมูล
16	cron	ควบคุมการเริ่มทำงานของ job schedule
17	crontab	ใช้ตั้งเวลาให้คำสั่งเริ่มทำงานตามที่ต้องการ
18	csplit	แตก file ตามจำนวนบรรทัด
19	cut	ตัดข้อมูล file เป็น field column
20	date	แสดงเวลาวันเดือนปี
21	dc	เครื่องคิดเลขแบบตั้งโต๊ะ
22	dd	backup ข้อมูลใน harddisk
23	df	แสดงข้อมูลพื้นที่ disk ทั้งหมด
24	diff	วิเคราะห์เปรียบเทียบ file ที่ละบรรทัด
25	dir	แสดงข้อมูล directory
26	dircolors	ที่ใช้ในการปรับสีของผลลัพธ์ ls
27	dirname	แสดงชื่อ directory ของ file
28	du	ดูข้อมูลรายละเอียดขนาด file
29	echo	ในการแสดงผลบนหน้าจอ screen
30	ed	editor file ชนิดหนึ่ง
31	egrep	ค้นหาบรรทัดใน file ที่ตรงเงื่อนไข
32	env	สร้าง environment ในการ run program
33	expand	เปลี่ยนข้อมูล file จาก tab เป็น space
34	expr	ที่ใช้ประมวลผลตรรกะคณิตศาสตร์
35	factor	แยกตัวประกอบทางคณิตศาสตร์
36	fdisk	บริหารจัดการ disk partition
37	find	ใช้ในการค้นหา file หรือ directory
38	fmt	จัดเรียงข้อมูลภายใน file ในรูปแบบ format

No.	Command	Description
39	fold	จัดเรียงความยาวตัวอักษรแต่ละบรรทัด
40	free	แสดงข้อมูลการใช้งาน memory
41	fsck	ตรวจสอบและซ่อมแซม file system
42	gawk	ใช้ค้นหาข้อมูล text ในรูปแบบเดียวกับ awk
43	grep	ค้นหาบรรทัดใน file ที่ตรงเงื่อนไข
44	groups	แสดงข้อมูล group ของ system user
45	gunzip	ยกเลิกการบีบอัดข้อมูล file
46	gzip	บีบอัดข้อมูล file หรือ การ zip file
47	head	แสดงข้อมูลบางส่วนภายใน file
48	hostname	แสดงข้อมูลชื่อของเครื่อง server
49	id	แสดงข้อมูล user, group ในระบบ
50	ifconfig	แสดงข้อมูลและเปลี่ยนค่า interface server
51	info	ข้อมูลโปรแกรมบนระบบทั้งหมดที่ใช้งาน
52	iptables	จัดการกรอง ip port ที่เข้ามาใช้งาน
53	join	เชื่อมข้อมูล 2 file ด้วย field ที่เหมือนกัน
54	kill	ส่ง Signal หรือยกเลิกการทำงานของ process
55	less	อ่านข้อมูลและค้นหาข้อมูลใน file
56	ln	สร้าง link เชื่อมโยงกันระหว่าง file
57	locate	ใช้ในการค้นหา file หรือ directory
58	logname	แสดงชื่อ user login
59	ls	แสดงข้อมูลภายใน directory
60	man	แสดงคู่มือการใช้งาน program
61	mkdir	สร้าง directory
62	more	อ่านข้อมูลและค้นหาข้อมูลใน file
63	mount	ติดตั้งใช้งานอุปกรณ์ที่เชื่อมต่อ
64	mv	ย้ายตำแหน่ง file หรือ directory
65	netstat	แสดงสถานะ network connection ทั้งหมด
66	nice	จัดลำดับความสำคัญของ process
67	nl	แสดงเลขที่บรรทัดของข้อมูลใน file
68	nohup	ป้องกันการหยุดของ background process
69	passwd	เปลี่ยน password ของ System user
70	paste	เชื่อมข้อมูลที่แต่ละบรรทัดจากหลาย file
71	ping	ตรวจสอบสถานะ server ปลายทาง
72	pr	แสดงข้อมูลภายใน file ในรูปแบบสิ่งพิมพ์
73	printf	แสดงผลข้อมูลบนหน้าจอ screen
74	ps	แสดง process ต่าง ๆ (process status)
75	pwd	แสดง directory หรือ path ที่อยู่ปัจจุบัน
76	rcp	คัดลอก file ข้ามเครื่อง server
77	rm	ลบ file หรือ directory
78	rmdir	ลบ directory
79	rsync	sync ข้อมูล file ระหว่าง server
80	screen	สร้าง session screen ขึ้นมาใหม่อีกจอ
81	sdiff	วิเคราะห์เปรียบเทียบข้อมูล file ที่ละบรรทัด

No.	Command	Description
82	sed	เปลี่ยนแปลงข้อมูล text ที่มีรูปแบบซับซ้อน
83	shutdown	ปิดการทำงานของระบบ
84	sleep	หน่วงเวลา
85	sort	ในการจัดเรียงข้อมูล file ที่ละบรรทัด
86	split	แตก file ตามจำนวนบรรทัด
87	su	login ด้วย user id อื่น
88	sum	การตรวจสอบ checksum และ ขนาด block
89	sync	เขียนข้อมูล memory ลง disk
90	tac	แสดงข้อมูลใน file แบบกลับหลัง
91	tail	แสดงข้อมูลบางส่วนภายใน file
92	tar	จัดเก็บรวบรวม file ข้อมูล
93	tee	อ่านข้อมูลพร้อมกับเขียนข้อมูลลง file
94	time	จับเวลาการทำงาน process
95	top	จัดเรียงอันดับแสดงการทำงานของ process
96	touch	เปลี่ยนแปลง file timestamps
97	tr	ค้นหาและเปลี่ยนแปลงข้อมูล text
98	traceroute	แสดงเส้นทางการทำงาน network
99	tty	แสดงชนิดของ terminal ที่ใช้งาน
100	uname	แสดงชื่อระบบของ server
101	unexpand	เปลี่ยน space เป็น tab
102	uniq	ในการจัดเรียงข้อมูลแบบไม่ซ้ำกัน
103	units	ในการแปลงค่าหน่วยวัด
104	useradd	สร้าง user และจัดการ user บนระบบ
105	userdel	ลบ user ออกจากระบบ
106	usermod	เปลี่ยนแปลงข้อมูลของ user
107	w	แสดง user ที่ login รวมถึงที่ใช้งาน
108	watch	monitor process ที่ทำงานอยู่
109	wc	นับจำนวนคำและบรรทัดจาก file
110	whereis	ค้นหาตำแหน่ง file program
111	which	ตำแหน่ง file program
112	who	แสดงข้อมูล user ที่ login ขณะนั้น
113	whoami	แสดงชื่อ user ที่ใช้ login
114	nano	ในการสร้างหรือแก้ไข file ข้อมูล text
115	vi	ในการสร้างหรือแก้ไข file ข้อมูล text
116	telnet	โปรโตคอลเชื่อมต่อสื่อสารด้วยข้อมูลตัวอักษร
117	ssh	เชื่อมต่อ shell server แบบเข้ารหัส
118	scp	คัดลอก file ข้อมูลแบบเข้ารหัสความปลอดภัย

ในการใช้งานหลายคำสั่ง จะต้องระบุ option เพิ่มเพื่อให้ทำงานได้ หรือกำหนดรูปแบบการทำงาน เช่น คำสั่ง cp ง่ายที่สุดคือ cp sourcePath DestPath เป็นต้น หากไม่ทราบวิธีใช้งานคำสั่งไหนให้พิมพ์คำสั่งนั้นแล้วตามด้วย --help เช่น cp --help แล้วอ่านรายละเอียดการใช้งานคำสั่งนั้น ๆ