



ใบงานที่ 2
เรื่อง Basic commands

เสนอ
อาจารย์ปิยพล ยืนยงสถาวร

จัดทำโดย
นายอริศ สุนทรโธม 65543206086-2

ใบงานนี้เป็นส่วนหนึ่งของรายวิชา ระบบปฏิบัติการ
หลักสูตรวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา
ประจำภาคที่ 2 ปีการศึกษา 2566

ใบงานที่ 2

Basic commands

ขั้นตอนการทดลอง

1. ให้สมัครสมาชิกเข้าใช้งานระบบใน <https://www.webminal.org/>
2. Login เข้าระบบ และเปิดเมนู Terminal
3. ปฏิบัติการทดลองคำสั่งใน Lessons 1 – 11
4. บันทึกการทดลอง และผลการทดลอง
5. อธิบายการปฏิบัติแต่ละขั้นตอนอย่างละเอียด พร้อมบันทึกภาพประกอบการปฏิบัติ
6. สรุปผลการทดลอง
7. จัดรูปแบบเอกสารใบงานให้เรียบร้อย
8. บันทึกไฟล์ในรูปแบบ PDF ส่งใน Assignment LAB2 : Basic commands

บันทึกการทดลอง และผลการทดลอง

Lesson1 - Basic commands to navigate directories

Lesson2 - Create files, display contents and stats

Lesson3 - Copy, rename, delete files

Lesson4 - Basic process commands

Lesson5 - Manipulate or parse file contents

Lesson6 - Changing file attributes

Lesson7 - Locate file and its type

Lesson8 - System and user details

Lesson9 - Linux Process Basic commands 1

Lesson10 - Linux Process Basic commands 2

Lesson11 - Linux Process states

ขั้นตอนการทดลอง

Lesson1 - Basic commands to navigate directories

```
-sh-4.2$ pwd
/home/tooy0303
```

```
-sh-4.2$ mkdir -v dir1
mkdir: created directory 'dir1'
-sh-4.2$ mkdir -v dir2
mkdir: created directory 'dir2'
-sh-4.2$
-sh-4.2$ mkdir -v dir2/dir3
mkdir: created directory 'dir2/dir3'
-sh-4.2$
-sh-4.2$ mkdir -v dir2/dir3/dir4
mkdir: created directory 'dir2/dir3/dir4'
```

```
-sh-4.2$ ls
dir1 dir2
-sh-4.2$ ls -R
.:
dir1 dir2

./dir1:

./dir2:
dir3

./dir2/dir3:
dir4

./dir2/dir3/dir4:
```

```
-sh-4.2$ : cd ..
-sh-4.2$ cd -
/home/tooy0303/dir2/dir3
-sh-4.2$ cd
```

คำสั่ง	คำอธิบาย
pwd	จะแสดงไดเรกทอรีการทำงานปัจจุบันของคุณ ใช่แล้ว บ้านของคุณคือไดเรกทอรี ตอนนี้ให้ลองสร้างไดเรกทอรีใหม่พิมพ์ข้อความต่อไปนี้บนพรอมต์
mkdir dir1	คือการสร้างโฟลเดอร์ชื่อ dir 1
ls	ระบุ dir1 dir2 เป็นเนื้อหาไดเรกทอรีใหม่ นั่นคือสิ่งที่เราต้องการ
ls -r	ใช้เพื่อแสดงรายการไฟล์และไดเรกทอรีในลำดับย้อนกลับ (reverse order) หรือจากสุดท้ายขึ้นมาที่ต้นทาง นั่นคือแสดงไฟล์และไดเรกทอรีตามลำดับตรงกันข้ามกับที่ปกติ (จากท้ายสู่ต้น) ซึ่งช่วยให้คุณได้ง่ายว่าไฟล์ไหนถูกสร้างหรือแก้ไขล่าสุดก่อน โดยปกติ ls จะแสดงในลำดับปกติ (จากต้นทางไปสุดท้าย) ถ้าเพิ่ม -r จะกลับลำดับนั้นให้เป็นจากท้ายสู่ต้น.
cd dir2	ที่ดีคือการเข้าถึงไดเรกทอรีตามชื่อที่ตั้ง
cd ..	ถอยออกจากไดเรกทอรี
cd	จะย้ายไปยังโฮมไดเรกทอรีของคุณ

Lesson2 - Create files, display contents and stats

```
-sh-4.2$ echo "hello"
hello
-sh-4.2$ echo "hello" > hello.txt
-sh-4.2$ echo "linux" >> hello.txt
-sh-4.2$ echo "world" >> hello.txt
-sh-4.2$ cat hello.txt
hello
linux
world
-sh-4.2$ head -2 hello.txt
hello
linux
-sh-4.2$ head hello.txt
hello
linux
world
-sh-4.2$ tail -2 hello.txt
linux
world
-sh-4.2$ tail hello.txt
hello
linux
world
-sh-4.2$ stat hello.txt
  File: 'hello.txt'
  Size: 18          Blocks: 8          IO Block: 4096   regular file
Device: 810h/2064d Inode: 182255129   Links: 1
Access: (0664/-rw-rw-r--)  Uid: (290859/tooy0303)   Gid: (290918/tooy0303)
Context: guest_u:object_r:user_home_t:s0
Access: 2023-11-24 04:08:56.499469727 +0000
Modify: 2023-11-24 04:08:40.893060840 +0000
Change: 2023-11-24 04:08:40.893060840 +0000
 Birth: -
-sh-4.2$ stat dir1
  File: 'dir1'
  Size: 6           Blocks: 0          IO Block: 4096   directory
Device: 810h/2064d Inode: 271343313   Links: 2
Access: (0775/drwxrwxr-x)  Uid: (290859/tooy0303)   Gid: (290918/tooy0303)
Context: guest_u:object_r:user_home_t:s0
Access: 2023-11-24 03:55:52.256922664 +0000
Modify: 2023-11-24 03:52:25.292500216 +0000
Change: 2023-11-24 03:52:25.292500216 +0000
 Birth: -
-sh-4.2$ touch file1.txt
-sh-4.2$ touch file2.txt
```

รูปตัวอย่าง Lesson2

คำสั่ง	คำอธิบาย
touch file1.txt	สร้างไฟล์ใหม่ที่ชื่อ "file1.txt" ถ้าไฟล์นี้ยังไม่มีอยู่ คำสั่ง touch จะใช้สร้างไฟล์ใหม่ และหากไฟล์นี้มีอยู่แล้ว คำสั่ง touch จะใช้ในการอัปเดต timestamp (เวลาที่สร้างและแก้ไขล่าสุดของไฟล์) โดยไม่เปลี่ยนเนื้อหาของไฟล์.
dir	ใช้เพื่อแสดงรายการเนื้อหาไดเรกทอรี ใช่แล้ว ตามที่คุณเคยถูกต้อง dir เทียบเท่ากับ ls -C -b (ฉันรู้ว่าคุณไม่ได้เดาว่า นั่นคือ โดยค่าเริ่มต้น ไฟล์จะแสดงรายการในคอลัมน์ จัดเรียงในแนวตั้ง และอักขระพิเศษจะแสดงด้วยลำดับหลักของแบ็กสแลช
Clear	เคลียร์หน้าจอมินอล

echo "hello" > hello.txt	ใช้ในการสร้างไฟล์ชื่อ "hello.txt" และเขียนข้อความ "hello" เข้าไปในไฟล์นั้น โดยถ้าไฟล์ "hello.txt" มีอยู่แล้ว, คำสั่งนี้จะเขียนทับข้อมูลที่มีอยู่ในไฟล์ด้วยข้อความ "hello".
cat hello.txt	ใช้ในการแสดงเนื้อหาของไฟล์ "hello.txt" บนหน้าจอ.
head -2 hello.txt	ใช้ในการแสดงบางส่วนของ "หัว" (head) ของไฟล์ "hello.txt" บนหน้าจอ, โดยมีตัวเลข -2 ระบุว่าเราต้องการแสดงเพียง 2 บรรทัดแรกของไฟล์.
tail hello.txt	ใช้ในการแสดงบางส่วนของ "ท้าย" (tail) ของไฟล์ "hello.txt" บนหน้าจอ. คำสั่งนี้จะแสดงทั้งหมดของไฟล์ "hello.txt" หากไฟล์นั้นมีขนาดไม่เกินจำนวนบรรทัดที่ระบุไว้ตามค่าตัวเลือก (default คือ 10 บรรทัด).
stat hello.txt	ใช้ในการแสดงข้อมูลสถิติ (statistical information) เกี่ยวกับไฟล์หรือไดเรกทอรีที่ระบุ.
stat dir1	จะแสดงข้อมูลที่เกี่ยวข้องกับไดเรกทอรีนั้น ๆ รวมถึงข้อมูลเกี่ยวกับการสิทธิ์, inode, วันที่และเวลา, และอื่น ๆ.

Lesson3 - Copy, rename, delete files

```

-sh-4.2$ ps
  PID TTY          TIME CMD
 28803 pts/138    00:00:00 sh
 30500 pts/138    00:00:00 ps
-sh-4.2$ sleep 60 &
[1] 30526
-sh-4.2$ ps
  PID TTY          TIME CMD
 28803 pts/138    00:00:00 sh
 30526 pts/138    00:00:00 sleep
 30530 pts/138    00:00:00 ps
-sh-4.2$ kill 12345
sh: kill: (12345) - No such process
-sh-4.2$ ps
  PID TTY          TIME CMD
 28803 pts/138    00:00:00 sh
 30526 pts/138    00:00:00 sleep
-sh-4.2$ du
0      ./dir1
0      ./dir2/dir3/dir4
0      ./dir2/dir3
0      ./dir2
28      .
-sh-4.2$ du -xh ~
0      /home/tooy0303/dir1
0      /home/tooy0303/dir2/dir3/dir4
0      /home/tooy0303/dir2/dir3
0      /home/tooy0303/dir2
28K    /home/tooy0303/
-sh-4.2$ du --max-depth 3 ~
0      /home/tooy0303/dir1
0      /home/tooy0303/dir2/dir3/dir4
0      /home/tooy0303/dir2/dir3
0      /home/tooy0303/dir2
28      /home/tooy0303/
-sh-4.2$ cp -v hello.txt dir2
'hello.txt' -> 'dir2/hello.txt'
-sh-4.2$ cp -v hello.txt dir2/file2.txt
'hello.txt' -> 'dir2/file2.txt'
-sh-4.2$ cp -vr dir2/*.txt dir2/dir3
'dir2/file2.txt' -> 'dir2/dir3/file2.txt'
'dir2/hello.txt' -> 'dir2/dir3/hello.txt'
-sh-4.2$ cp -vr dir2/dir3 .
'dir2/dir3' -> './dir3'
'dir2/dir3/dir4' -> './dir3/dir4'
'dir2/dir3/file2.txt' -> './dir3/file2.txt'
'dir2/dir3/hello.txt' -> './dir3/hello.txt'
-sh-4.2$ md5sum hello.txt
b8d5079c5d6a9dbb3294b31d318d74c0  hello.txt
-sh-4.2$ md5sum dir2/hello.txt
b8d5079c5d6a9dbb3294b31d318d74c0  dir2/hello.txt

```

```

b8d5079c5d6a9dbb3294b31d318d74c0 hello.txt
-sh-4.2$ md5sum dir2/hello.txt
b8d5079c5d6a9dbb3294b31d318d74c0 dir2/hello.txt
-sh-4.2$ 
-sh-4.2$ md5sum hello.txt
b8d5079c5d6a9dbb3294b31d318d74c0 hello.txt
-sh-4.2$ mv hello.txt dir2/dir3/dir4/hi.txt
-sh-4.2$ mkdir dir5
-sh-4.2$ mv dir2/*.txt dir5
-sh-4.2$ mv dir5 dir50
-sh-4.2$ ln dir2/dir3/dir4/hi.txt hello
-sh-4.2$ stat hello
  File: 'hello'
  Size: 18          Blocks: 8          IO Block: 4096   regular file
Device: 810h/2064d Inode: 182255129   Links: 2
Access: (0664/-rw-rw-r--)  Uid: (290859/tooy0303)   Gid: (290918/tooy0303)
Context: guest_u:object_r:user_home_t:s0
Access: 2023-11-24 04:08:56.499469727 +0000
Modify: 2023-11-24 04:08:40.893060840 +0000
Change: 2023-11-24 04:20:21.446415268 +0000
Birth: -
-sh-4.2$ stat dir2/dir3/dir4/hi.txt
  File: 'dir2/dir3/dir4/hi.txt'
  Size: 18          Blocks: 8          IO Block: 4096   regular file
Device: 810h/2064d Inode: 182255129   Links: 2
Access: (0664/-rw-rw-r--)  Uid: (290859/tooy0303)   Gid: (290918/tooy0303)
Context: guest_u:object_r:user_home_t:s0
Access: 2023-11-24 04:08:56.499469727 +0000
Modify: 2023-11-24 04:08:40.893060840 +0000
Change: 2023-11-24 04:20:21.446415268 +0000
Birth: -
-sh-4.2$ ln -s dir2/dir3/dir4/hi.txt softlink
-sh-4.2$ stat softlink
  File: 'softlink' -> 'dir2/dir3/dir4/hi.txt'
  Size: 21          Blocks: 0          IO Block: 4096   symbolic link
Device: 810h/2064d Inode: 182255134   Links: 1
Access: (0777/lrwxrwxrwx)  Uid: (290859/tooy0303)   Gid: (290918/tooy0303)
Context: guest_u:object_r:user_home_t:s0
Access: 2023-11-24 04:20:45.979058021 +0000
Modify: 2023-11-24 04:20:45.979058021 +0000
Change: 2023-11-24 04:20:45.979058021 +0000
Birth: -
-sh-4.2$ rm -i file2.txt
rm: remove regular empty file 'file2.txt'? y
-sh-4.2$ rm -ri dir50/*
rm: remove regular file 'dir50/file2.txt'?
rm: remove regular file 'dir50/hello.txt'? y
-sh-4.2$ rm -rf junk/*
-sh-4.2$ rmdir dir50
rmdir: failed to remove 'dir50': Directory not empty

```

คำสั่ง	คำอธิบาย
du	ใช้ในการแสดงขนาดที่ใช้พื้นที่บนดิสก์ของไดเรกทอรีและไฟล์. คำสั่งนี้สามารถใช้งานร่วมกับตัวเลือกต่าง ๆ เพื่อแสดงข้อมูลเพิ่มเติม เช่น ขนาดทั้งหมดที่ใช้, ขนาดไฟล์แต่ละไฟล์, หรือแสดงในรูปแบบที่เข้าใจง่าย.
du -xh ~	ใช้ในการแสดงขนาดที่ใช้พื้นที่บนดิสก์ของไดเรกทอรีที่ถูกระบุ (ในที่นี้คือ ~ ซึ่งแทนไดเรกทอรีของผู้ใช้ปัจจุบัน) โดยไม่สนใจการเข้าถึงไฟล์หรือไดเรกทอรีที่ตั้งอยู่ในไดเรกทอรีอื่น. du: คำสั่งแสดงขนาดที่ใช้พื้นที่บนดิสก์ของไดเรกทอรีและไฟล์. -x: กำหนดให้คำสั่งไม่ค้นหาขนาดที่ใช้พื้นที่บนดิสก์ของไดเรกทอรีและไฟล์ที่อยู่ในไดเรกทอรีที่ต่างจากไดเรกทอรี -h: แสดงขนาดในรูปแบบที่อ่านง่าย (human-readable).
cp -v hello.txt dir2	ใช้ในการคัดลอก (copy) ไฟล์ "hello.txt" ไปยังไดเรกทอรี "dir2" และแสดงข้อความที่ระบุว่าไฟล์ถูกคัดลอก
cp -vr dir2/*.txt dir2/dir3	ใช้ในการคัดลอกไฟล์ทั้งหมดที่มีนามสกุล ".txt" จากไดเรกทอรี "dir2" ไปยังไดเรกทอรี "dir3" และแสดงข้อความที่ระบุว่าไฟล์ถูกคัดลอก
md5sum hello.txt	ใช้ในการคำนวณและแสดงค่า MD5 checksum (hash) ของไฟล์. MD5 checksum เป็นตัวเลขที่เกิดจากการประมวลผลข้อมูลในไฟล์เพื่อตรวจสอบความถูกต้องของไฟล์นั้น.
mv dir2/*.txt dir5	ใช้ในการย้าย (หรือเปลี่ยนชื่อ) ไฟล์ทั้งหมดที่มีนามสกุล ".txt" จากไดเรกทอรี "dir2" ไปยังไดเรกทอรี "dir5".
ln dir2/dir3/dir4/hi.txt hello	ใช้ในการสร้าง symbolic link (ที่ระบุโดยตัวชี้ไปยังไฟล์หรือไดเรกทอรีอื่น) ของไฟล์ "dir2/dir3/dir4/hi.txt" ในไดเรกทอรีปัจจุบัน และให้ symbolic link มีชื่อ "hello".
ln -s dir2/dir3/dir4/hi.txt softlink	ใช้สร้าง symbolic link ที่ชื่อ "softlink" และชี้ไปยังไฟล์ "dir2/dir3/dir4/hi.txt". ln: คำสั่งสร้าง symbolic link หรือ hard link. -s: เป็นตัวเลือกที่ระบุให้ ln สร้าง symbolic link. dir2/dir3/dir4/hi.txt: ไฟล์ที่เราต้องการสร้าง symbolic link ชี้ไปยัง. softlink: ชื่อ symbolic link ที่เราต้องการสร้าง.
rm -i file2.txt	ใช้ในการลบไฟล์ "file2.txt" โดยให้ระบบถามยืนยันจากผู้ใช้ก่อนที่จะทำการลบ (interactive mode). rm: คำสั่งลบไฟล์หรือไดเรกทอรี. -i: เปิดโหมด interactive ซึ่งจะระบุให้ระบบถามคำยืนยันจากผู้ใช้ก่อนที่จะทำการลบ.
rm -ri dir50/*	ใช้ในการลบทั้งหมดของไฟล์และไดเรกทอรีที่อยู่ในไดเรกทอรี "dir50" โดยให้ระบบถามคำยืนยันจากผู้ใช้ก่อนที่จะทำการลบแบบแอดซันตลอด.
rm -rf junk/*	ใช้ในการลบทั้งหมดของไฟล์และไดเรกทอรีที่อยู่ในไดเรกทอรี "junk" โดยไม่ต้องถามคำยืนยันจากผู้ใช้ (force mode).
rmdir dir50	ใช้เพื่อยกเลิกไดเรกทอรีที่ว่างเปล่าเท่านั้น.

Lesson4 - Basic process commands

```
-sh-4.2$ ps
  PID TTY          TIME CMD
 2241 pts/138    00:00:00 ps
28803 pts/138    00:00:00 sh
-sh-4.2$ sleep 60 &
[1] 2264
-sh-4.2$ kill 12345
-sh: kill: (12345) - No such process
-sh-4.2$ killall sleep
[1]+  Terminated                  sleep 60
-sh-4.2$ killall -u webminal
Cannot find user webminal
-sh-4.2$ killall -w find
find: no process found
-sh-4.2$ pidof bash
-sh-4.2$ pidof -s bash
-sh-4.2$ nice -n 19 sleep 30 &
[1] 2442
-sh-4.2$ renice -n 19 12345
renice: failed to get priority for 12345 (process ID)
: No such process
```

```
Device: 810h/2064d    Inode: 182
top - 04:27:20 up 550 days, 23:37,  0 users,  load average: 0.37, 0.23, 0.19
Tasks:  2 total,   1 running,  1 sleeping,   0 stopped,   0 zombie
%Cpu(s):  1.3 us,   0.7 sy,   0.0 ni, 97.9 id,   0.0 wa,   0.0 hi,   0.1 si,   0.0 st
KiB Mem : 16399684 total, 2714168 free, 3491012 used, 10194504 buff/cache
KiB Swap:   0 total,    0 free,    0 used. 11617512 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1037	tooy0303	20	0	162000	4428	3844	R	0.3	0.0	0:00.06	top
28803	tooy0303	20	0	115688	3816	3128	S	0.0	0.0	0:00.13	sh


```
[1]+ Done nice -n 19 sleep 30
-sh-4.2$ pstree -p
?(1)---?(28784)---sh(28803)---pstree(1497)
-sh-4.2$ time ls -l
total 4
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 a
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 again,
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 an
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 change
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 command
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 create
drwxrwxr-x. 2 tooy0303 tooy0303 6 Nov 24 03:52 dir1
drwxrwxr-x. 3 tooy0303 tooy0303 18 Nov 24 04:20 dir2
drwxrwxr-x. 3 tooy0303 tooy0303 52 Nov 24 04:19 dir3
drwxrwxr-x. 2 tooy0303 tooy0303 23 Nov 24 04:21 dir50
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 existing
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 file
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 file.
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:10 file1.txt
-rw-rw-r--. 2 tooy0303 tooy0303 18 Nov 24 04:08 hello
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 new
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 Now
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 of
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 or
lrwxrwxrwx. 1 tooy0303 tooy0303 21 Nov 24 04:20 softlink -> dir2/dir3/dir4/hi.txt
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 stamp
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 time
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 try
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 will

real    0m0.060s
user    0m0.055s
sys     0m0.005s
```

รูปตัวอย่าง Lesson4

คำสั่ง	คำอธิบาย
Ps	ใช้ในการแสดงข้อมูลเกี่ยวกับกระบวนการ (processes) ที่กำลังทำงานในระบบ. คำสั่งนี้ไม่แสดงกระบวนการของทุกๆ ผู้ใช้, แต่แสดงกระบวนการของผู้ใช้ปัจจุบัน.
sleep 60 &	ใช้ในการให้ระบบหยุดการทำงาน (sleep) เป็นเวลา 60 วินาที และใช้สัญลักษณ์ & เพื่อให้กระบวนการทำงานที่สร้างขึ้น (ในที่นี้คือ sleep) ทำงานในพื้นหลัง (background).
kill 12345	ใช้ในการส่งสัญญาณการสิ้นสุด (terminate signal) ไปยังกระบวนการที่มี Process ID (PID) เท่ากับ 12345. สัญญาณการสิ้นสุดทำให้กระบวนการหยุดการทำงานหากคำสั่งทำงานได้สำเร็จ, กระบวนการที่มี PID เท่ากับ 12345 จะถูกสิ้นสุด. หากไม่มีข้อผิดพลาด, kill จะไม่ส่งข้อความยืนยันกลับ.

killall sleep	ใช้ในการส่งสัญญาณการสิ้นสุดไปยังกระบวนการทั้งหมดที่ชื่อ "sleep". คำสั่งนี้มักใช้เพื่อยุติการทำงานของกระบวนการทั้งหมดที่ชื่อ "sleep" ที่กำลังทำงานในระบบ.
killall -u webminal	ใช้ในการส่งสัญญาณการสิ้นสุดไปยังกระบวนการทั้งหมดที่ถูกเริ่มขึ้นโดยผู้ใช้ที่มีชื่อ "webminal". คำสั่งนี้จะหยุดการทำงานของกระบวนการทั้งหมดที่เป็นของผู้ใช้ "webminal".
killall -w find	ใช้ในการส่งสัญญาณการสิ้นสุดไปยังกระบวนการทั้งหมดที่มีชื่อ "find" และรอให้กระบวนการนั้นจบการทำงานก่อนที่จะสิ้นสุด. การใช้ตัวเลือก -w (wait) ทำให้คำสั่งรอให้กระบวนการ "find" ทุกรายการที่กำลังทำงานเสร็จสิ้นก่อนที่จะส่งสัญญาณการสิ้นสุด. ในกรณีที่มีย่อยกระบวนการ "find" ทำงานพร้อมกัน, -w จะทำให้ killall รอทุกรายการจนกว่ากระบวนการทั้งหมดจะเสร็จสิ้นก่อนที่จะทำการสิ้นสุด. ควรใช้ -w ในกรณีที่คุณต้องการให้การสิ้นสุดกระบวนการเสร็จสมบูรณ์ก่อนที่จะสั่ง killall จะถูกตั้งตาม.
pidof bash	ใช้ในการแสดง Process ID (PID) ของกระบวนการ (process) ที่ชื่อ "bash"
pidof -s bash	ใช้ในการแสดง Process ID (PID) ของกระบวนการ (process) ที่ชื่อ "bash" ที่กำลังทำงานในระบบปฏิบัติการ Linux และให้ผลลัพธ์เป็น PID เดียวที่ตรงกับกระบวนการแรกที่พบ.
nice -n 19 sleep 30 &	ใช้ในการเริ่มกระบวนการ sleep ที่มีความล่าช้า (niceness) ที่สูงมาก (19) ในลักษณะ background ในระบบปฏิบัติการ Linux nice: คำสั่งที่ใช้ในการเปลี่ยนความล่าช้าของกระบวนการ, ทำให้กระบวนการมีความล่าช้าต่ำหรือสูงขึ้น. -n 19: ตัวเลือก -n ใช้ในการกำหนดความล่าช้า (niceness) ของกระบวนการ. ค่า niceness อยู่ในช่วง -20 (สูงสุด) ถึง 19 (ต่ำสุด). ค่า niceness สูงแสดงถึงการทำงานที่น้อยที่สุดของ CPU. ในที่นี้, nice -n 19 ทำให้กระบวนการ sleep มีความล่าช้าสูงมาก (จำนวนความล่าช้าต่ำ) และ & ทำให้กระบวนการทำงานในพื้นหลัง. กระบวนการ sleep 30 จะหยุดการทำงานเป็นเวลา 30 วินาที โดยไม่มีการใช้ทรัพยากร CPU มากนัก, เนื่องจากมีความล่าช้าสูง.
renice -n 19 12345	ใช้ในการเปลี่ยนความล่าช้า (niceness) ของกระบวนการที่มี Process ID (PID) เท่ากับ 12345 ในระบบ
top	ใช้ในการแสดงข้อมูลเกี่ยวกับประสิทธิภาพของระบบและกระบวนการที่กำลังทำงานในระบบปฏิบัติการ Linux. เมื่อคุณป้อน top ในหน้าต่างคำสั่ง, มันจะแสดงตารางที่ประกอบด้วยข้อมูลต่าง ๆ เกี่ยวกับ CPU, หน่วยความจำ (Memory), และกระบวนการที่กำลังทำงาน.
ps tree -p	ใช้ในการแสดงโครงสร้างของกระบวนการ (process tree) ในระบบปฏิบัติการ Linux พร้อมกับ Process ID (PID) ของแต่ละกระบวนการ.

time ls -l	<p>ใช้ในการวัดเวลาที่ใช้ในการทำงานของคำสั่ง ls -l ซึ่งในที่นี้เป็นการแสดงรายการไฟล์และไดเรกทอรีในรูปแบบละเอียด</p> <p>ผลลัพธ์ที่ได้จะแบ่งเป็นสามส่วน:</p> <p>real: เวลาที่ใช้ทั้งหมดตั้งแต่เริ่มต้นคำสั่งจนถึงสิ้นสุด.</p> <p>user: เวลาที่ CPU ใช้ในการประมวลผลคำสั่ง.</p> <p>sys: เวลาที่ CPU ใช้ในการประมวลผลงานที่เกี่ยวข้องกับระบบ (system-related tasks).</p>
------------	---

Lesson5 - Manipulate or parse file contents

```
-sh-4.2$ grep "linux" hello
linux
-sh-4.2$ grep -r 'Hello' .
-sh-4.2$
hello
^Z
[1]+  Stopped(SIGTSTP)      grep --color=auto -i 'LINUX'
-sh-4.2$ grep -i 'LINUX' hello
linux
-sh-4.2$ grep -n 'linux' hello
2:linux
-sh-4.2$ grep -v 'world' hello
hello
linux
-sh-4.2$ wc -L hello
5 hello
-sh-4.2$ cut -f1 -d' ' new.txt
cut: new.txt: No such file or directory
-sh-4.2$ echo -e "col1 col2 r1\ncol5 col6 r2\ncol3 col4 r3" >> new.txt
-sh-4.2$ echo -e "Hello\nlinux\nProgrammers paradise" >> linux.txt
-sh-4.2$ cut -f1 -d' ' new.txt
col1
col5
col3
-sh-4.2$ วางส่วสัดใหม่.txt
hello  col1 col2 r1
linux  col5 col6 r2
world  col3 col4 r3
-sh-4.2$ วาง -s ส่วสัดใหม่
hello  linux  world
col1 col2 r1      col5 col6 r2      col3 col4 r3
-sh-4.2$ sort new.txt
col1 col2 r1
col3 col4 r3
col5 col6 r2
-sh-4.2$ diff hello linux.txt
1c1
< hello
---
> Hello
3c3
```

```

< world
---
> Programmers paradise
-sh-4.2$ diff hello linux.txt
1c1
< hello
---
> Hello
3c3
< world
---
> Programmers paradise
-sh-4.2$ diff3 hello new.txt linux.txt
====
1:1,3c
    hello
    linux
    world
2:1,3c
    col1 col2 r1
    col5 col6 r2
    col3 col4 r3
3:1,3c
    Hello
    linux
    Programmers paradise

```

รูปตัวอย่าง Lesson5

คำสั่ง	คำอธิบาย
grep "linux" hello	ใช้ในการค้นหาข้อความ "linux" ในไฟล์ที่ชื่อ "hello" grep: เป็นคำสั่งที่ใช้ในการค้นหาข้อความในไฟล์. "linux": เป็นข้อความที่คุณกำลังค้นหา. hello: เป็นชื่อของไฟล์ที่คุณต้องการค้นหาใน.
grep -r 'Hello' .	ใช้ในการค้นหาข้อความ "Hello" ในไฟล์ทั้งหมดในไดเรกทอรีปัจจุบัน (รวมถึงไดเรกทอรีย่อยทุกๆ ระดับ) ในระบบปฏิบัติการ Linux. grep: เป็นคำสั่งที่ใช้ในการค้นหาข้อความในไฟล์. -r: ตัวเลือกที่ใช้ในการค้นหาไฟล์ทั้งหมดในไดเรกทอรีและไดเรกทอรีย่อย. 'Hello': เป็นข้อความที่คุณกำลังค้นหา. .: เป็นพารามิเตอร์ว่าคุณต้องการค้นหาในไดเรกทอรีปัจจุบัน.
grep -i 'linux'	ใช้ในการค้นหาข้อความ "linux" (โดยไม่สนใจตัวพิมพ์ใหญ่-เล็ก) ในไฟล์ ที่ชื่อ "hello". grep: เป็นคำสั่งที่ใช้ในการค้นหาข้อความในไฟล์. -i: ตัวเลือกที่ใช้ในการทำให้ค้นหาเป็น case-insensitive, ซึ่งหมายความว่าไม่สนใจตัวพิมพ์ใหญ่-เล็ก.

grep -n 'linux' hello	<p>ใช้ในการค้นหาข้อความ "linux" ในไฟล์ที่ชื่อ "hello" และแสดงหมายเลขบรรทัดที่พบข้อความนั้นด้วย.</p> <p>grep: เป็นคำสั่งที่ใช้ในการค้นหาข้อความในไฟล์.</p> <p>-n: ตัวเลือกที่ใช้ในการแสดงหมายเลขบรรทัดที่พบข้อความ.</p>
grep -v 'world' hello	<p>ใช้ในการแสดงบรรทัดที่ไม่มีข้อความ "world" ในไฟล์ที่ชื่อ "hello".</p> <p>grep: เป็นคำสั่งที่ใช้ในการค้นหาข้อความในไฟล์.</p> <p>-v: ตัวเลือกที่ใช้ในการแสดงบรรทัดที่ไม่ตรงกับข้อความที่กำลังค้นหา.</p>
wc -L hello	<p>ใช้ในการแสดงความยาวของบรรทัดที่ยาวที่สุดในไฟล์ที่ชื่อ "hello".</p> <p>wc: เป็นคำสั่งที่ใช้ในการนับจำนวนบรรทัด, คำ, และตัวอักษรในไฟล์.</p> <p>-L: ตัวเลือกที่ใช้ในการแสดงความยาวของบรรทัดที่ยาวที่สุด.</p>
cut -f1 -d' ' new.txt	<p>ใช้ในการแสดงเนื้อหาที่อยู่ในคอลัมน์ที่หนึ่ง (-f1) ของไฟล์ที่ชื่อ "new.txt" โดยใช้เครื่องหมายขีดว่างเป็นตัวคั่น (-d' ').</p> <p>cut: เป็นคำสั่งที่ใช้ในการแสดงหรือตัดข้อมูลจากไฟล์.</p> <p>-f1: ตัวเลือกที่กำหนดว่าควรแสดงคอลัมน์ที่เท่าไร. ในที่นี้คือคอลัมน์ที่หนึ่ง.</p> <p>-d' ': ตัวเลือกที่กำหนดตัวคั่น (delimiter) ที่ใช้ในการแบ่งข้อมูล. ในที่นี้คือเครื่องหมายขีดว่าง.</p>
paste hello new.txt	<p>ใช้ในการนำเนื้อหาของไฟล์ "hello" และ "new.txt" มาต่อกันแนวตั้ง (vertically).</p>
sort new.txt	<p>ใช้ในการเรียงลำดับข้อมูลในไฟล์ "new.txt" ตามลำดับที่มีค่าน้อยไปที่มีค่ามาก.</p>
diff hello linux.txt	<p>ใช้ในการเปรียบเทียบเนื้อหาของไฟล์และแสดงการแตกต่างระหว่างไฟล์ทั้งสอง.</p>

Lesson6 - Changing file attributes

```
#####
-sh-4.2$ dirname dir2/dir3/dir4/hi.txt
dir2/dir3/dir4
-sh-4.2$ dir2/dir3/dir4
-sh: dir2/dir3/dir4: Is a directory
-sh-4.2$ basename dir2/dir3/dir4/hi.txt
hi.txt
-sh-4.2$ hi.txt
-sh: hi.txt: command not found

chmod: missing operand after 'a+r'
Try 'chmod --help' for more information.
-sh-4.2$
-sh-4.2$
-sh-4.2$ chmod a+rw file1.txt
-sh-4.2$ chmod -v 666 file1.txt
mode of 'file1.txt' retained as 0666 (rw-rw-rw-)
-sh-4.2$ chmod a-rw file1.txt
-sh-4.2$ chmod -R 644 ~/chmod_dir
chmod: cannot access '/home/tooy0303/chmod_dir': No such file or directory
-sh-4.2$ chmod -v 666 file1.txt
mode of 'file1.txt' changed from 0000 (-----) to 0666 (rw-rw-rw-)
-sh-4.2$ chmod a+rw file1.txt
-sh-4.2$ chmod -R 644 ~/chmod_dir
chmod: cannot access '/home/tooy0303/chmod_dir': No such file or directory
-sh-4.2$ chmod -R 644 ~/chmod_dir
chmod: cannot access '/home/tooy0303/chmod_dir': No such file or directory
-sh-4.2$ chmod mkdir chmod_dir
chmod: invalid mode: 'mkdir'
Try 'chmod --help' for more information.
-sh-4.2 $
-sh-4.2$
-sh-4.2$
-sh-4.2$
-sh-4.2$
-sh-4.2 $ CHDI chmod_dir
-sh: mmkdir: command not found
-sh-4.2$ mkdir chmod_dir
-sh-4.2$ chmod -R 644 ~/chmod_dir
-sh-4.2$ chown root:staff -R ~/dir2
chown: changing ownership of '/home/tooy0303/dir2/dir3/dir4/hi.txt': Operation not permitted
chown: changing ownership of '/home/tooy0303/dir2/dir3/dir4': Operation not permitted
chown: changing ownership of '/home/tooy0303/dir2/dir3/file2.txt': Operation not permitted
chown: changing ownership of '/home/tooy0303/dir2/dir3/hello.txt': Operation not permitted
chown: changing ownership of '/home/tooy0303/dir2/dir3': Operation not permitted
chown: changing ownership of '/home/tooy0303/dir2': Operation not permitted
-sh-4.2$ chown --from=webminal:webminal root:staff -R ~/dir2
chown: invalid user: 'webminal:webminal'
-sh-4.2$ chgrp root file1.txt
chgrp: changing group of 'file1.txt': Operation not permitted
-sh-4.2$ chgrp -hR root dir2
chgrp: changing group of 'dir2/dir3/dir4/hi.txt': Operation not permitted
chgrp: changing group of 'dir2/dir3/dir4': Operation not permitted
chgrp: changing group of 'dir2/dir3/file2.txt': Operation not permitted
chgrp: changing group of 'dir2/dir3/hello.txt': Operation not permitted
chgrp: changing group of 'dir2/dir3': Operation not permitted
chgrp: changing group of 'dir2': Operation not permitted
```

รูปตัวอย่าง Lesson6

คำสั่ง	คำอธิบาย
dirname dir2/dir3/dir4/hi.txt	ใช้ในการแสดง directory ที่บรรจุ path ของไฟล์หรือ directory ที่ระบุ.
basename dir2/dir3/dir4/hi.txt	ใช้ในการแสดงชื่อของไฟล์หรือ directory ที่ถูกระบุใน path

chmod -v 666 file1.txt	<p>ใช้ในการเปลี่ยนสิทธิ์การเข้าถึงของไฟล์ "file1.txt" เป็น 666 ซึ่งหมายถึงทุกคน (owner, group, others) สามารถอ่านและเขียนไฟล์นี้ได้.</p> <p>chmod: เป็นคำสั่งที่ใช้ในการเปลี่ยนแปลงสิทธิ์การเข้าถึงของไฟล์หรือไดเรกทอรี.</p> <p>-v: ตัวเลือกที่ใช้ในการแสดงผลการทำงานของคำสั่ง (verbose mode).</p> <p>666: ค่าสิทธิ์ที่กำหนด 666 ให้ทุกคนมีสิทธิ์อ่านและเขียนไฟล์.</p> <p>file1.txt: ชื่อของไฟล์ที่คุณต้องการเปลี่ยนแปลงสิทธิ์.</p>
chmod a+rw file1.txt	<p>ใช้ในการเปลี่ยนสิทธิ์การเข้าถึงของไฟล์ "file1.txt" ให้ทุกคน (owner, group, others) มีสิทธิ์อ่านและเขียนไฟล์.</p> <p>chmod: เป็นคำสั่งที่ใช้ในการเปลี่ยนแปลงสิทธิ์การเข้าถึงของไฟล์หรือไดเรกทอรี.</p> <p>a: คือ "all" หรือทุกคน (owner, group, others).</p> <p>+rw: เพิ่มสิทธิ์ให้ทุกคนอ่าน (read) และเขียน (write).</p> <p>file1.txt: ชื่อของไฟล์ที่คุณต้องการเปลี่ยนแปลงสิทธิ์</p>
chmod -R 644 ~/chmod_dir	<p>ใช้ในการเปลี่ยนสิทธิ์การเข้าถึงของไดเรกทอรี "chmod_dir" และไฟล์ทั้งหมดที่อยู่ในไดเรกทอรีนี้ให้มีสิทธิ์อ่าน (read) เฉพาะสำหรับเจ้าของไฟล์และสิทธิ์อ่าน (read) เฉพาะสำหรับกลุ่มและสิทธิ์ไม่มีสิทธิ์ในการเขียน (write) สำหรับทุกคนอื่น ๆ.</p> <p>chmod: เป็นคำสั่งที่ใช้ในการเปลี่ยนแปลงสิทธิ์การเข้าถึงของไฟล์หรือไดเรกทอรี.</p> <p>-R: ตัวเลือกที่ใช้ในการทำการเปลี่ยนแปลงทั้งหมดในโครงสร้างไดเรกทอรี (recursive).</p> <p>644: ค่าสิทธิ์ที่กำหนด 644 ให้เจ้าของไฟล์มีสิทธิ์อ่าน (read) และเขียน (write) และกลุ่มและทุกคนมีสิทธิ์อ่าน (read) เท่านั้น.</p> <p>~/chmod_dir: ที่อยู่ของไดเรกทอรีที่คุณต้องการเปลี่ยนแปลงสิทธิ์.</p>
chown root:staff -R ~/dir2	<p>ใช้ในการเปลี่ยนเจ้าของและกลุ่มของไดเรกทอรี "dir2" และไฟล์ทั้งหมดที่อยู่ใน "dir2" เป็น root เป็นเจ้าของ และ staff เป็นกลุ่ม.</p> <p>chown: เป็นคำสั่งที่ใช้ในการเปลี่ยนเจ้าของและกลุ่มของไฟล์หรือไดเรกทอรี.</p> <p>root:staff: ระบุเจ้าของและกลุ่มที่คุณต้องการกำหนด (root เป็นเจ้าของ, staff เป็นกลุ่ม).</p> <p>-R: ตัวเลือกที่ใช้ในการทำการเปลี่ยนแปลงทั้งหมดในโครงสร้างไดเรกทอรี (recursive).</p> <p>~/dir2: ที่อยู่ของไดเรกทอรีที่คุณต้องการเปลี่ยนแปลงเจ้าของและกลุ่ม.</p>
chgrp root file1.txt	<p>ใช้ในการเปลี่ยนกลุ่มของไฟล์ "file1.txt" เป็นกลุ่ม "root".</p> <p>chgrp: เป็นคำสั่งที่ใช้ในการเปลี่ยนกลุ่มของไฟล์หรือไดเรกทอรี.</p> <p>root: ระบุกลุ่มที่คุณต้องการกำหนดให้กับไฟล์ (ในที่นี้คือ "root").</p> <p>file1.txt: ชื่อของไฟล์ที่คุณต้องการเปลี่ยนกลุ่ม.</p>

<pre>chgrp -hR root dir2</pre>	<p>ใช้ในการเปลี่ยนกลุ่มของไดเรกทอรี "dir2" และไฟล์ทั้งหมดที่อยู่ใน "dir2" และซัพไดเรกทอรีทั้งหมด เป็นกลุ่ม "root" โดยรวมกับตัวเลือก -R ทำให้มีการเปลี่ยนแปลงทั้งหมดในโครงสร้างไดเรกทอรี (recursive).</p> <p>chgrp: เป็นคำสั่งที่ใช้ในการเปลี่ยนกลุ่มของไฟล์หรือไดเรกทอรี.</p> <p>-h: ตัวเลือกที่ใช้ในกรณีที่กำหนดกลุ่มบน symbolic link ไฟล์หรือไดเรกทอรี จะเปลี่ยนกลุ่มของ symbolic link ไม่ได้ แต่จะเปลี่ยนกลุ่มของไฟล์หรือไดเรกทอรีที่ symbolic link ชี้ไป.</p> <p>-R: ตัวเลือกที่ใช้ในการทำการเปลี่ยนแปลงทั้งหมดในโครงสร้างไดเรกทอรี (recursive).</p> <p>root: ระบุกลุ่มที่คุณต้องการกำหนดให้กับไฟล์และไดเรกทอรี (ในที่นี้คือ "root").</p> <p>dir2: ที่อยู่ของไดเรกทอรีที่คุณต้องการเปลี่ยนกลุ่ม.</p>
--------------------------------	--

Lesson7 - Locate file and its type

```
# WE WANT TO LOCATE YOUR LINUX FILE ON THE LINUX #
-sh-4.2$ file linux.txt
linux.txt: ASCII text
-sh-4.2$ file /dev/null
/dev/null: character special
-sh-4.2$ file -s /dev/sda2
/dev/sda2: cannot open (No such file or directory)
-sh-4.2$ whereis ls
ls: /usr/bin/ls /usr/share/man/man1/ls.1.gz
-sh-4.2$ ls: /bin/ls /usr/share/man/man1/ls.1p.gz /usr/share/man/man1/ls.1.gz
-sh: ls:: command not found
-sh-4.2$ whereis stdio.h
stdio: /usr/include/stdio.h
-sh-4.2$ php
-sh: php: command not found
-sh-4.2$ which php
/usr/bin/which: no php in (/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin)
-sh-4.2$ find ~ -name "linux.txt"
/home/tooy0303/linux.txt
-sh-4.2$ find . -type f -exec file '{}' \;
./bashrc: ASCII text
./bash_profile: ASCII text
./bash_logout: ASCII text
./magic_string.txt: ASCII text
./dir2/dir3/dir4/hi.txt: ASCII text
./dir2/dir3/file2.txt: ASCII text
./dir2/dir3/hello.txt: ASCII text
./file1.txt: empty
./command: empty
./will: empty
./create: empty
./a: empty
./new: empty
./file: empty
./or: empty
./change: empty
./time: empty
./stamp: empty
./of: empty
./an: empty
./existing: empty
./file.: empty
./Now: empty
./try: empty
./again,: empty
./bash_history: ASCII text
./dir3/file2.txt: ASCII text
./dir3/hello.txt: ASCII text
./hello: ASCII text
./new.txt: ASCII text
./linux.txt: ASCII text
-sh-4.2$ find . -type f -exec ls -l '{}' \;
-rw-r--r--. 1 tooy0303 tooy0303 231 Nov 24 2021 ./bashrc
-rw-r--r--. 1 tooy0303 tooy0303 193 Nov 24 2021 ./bash_profile
-rw-r--r--. 1 tooy0303 tooy0303 18 Nov 24 2021 ./bash_logout
```



```

-rw-r--r--. 1 root root 9 Nov 24 03:43 ./magic_string.txt
-rw-rw-r--. 2 tooy0303 tooy0303 18 Nov 24 04:08 ./dir2/dir3/dir4/hi.txt
-rw-rw-r--. 1 tooy0303 tooy0303 18 Nov 24 04:19 ./dir2/dir3/file2.txt
-rw-rw-r--. 1 tooy0303 tooy0303 18 Nov 24 04:19 ./dir2/dir3/hello.txt
-rw-rw-rw-. 1 tooy0303 tooy0303 0 Nov 24 04:10 ./file1.txt
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./command
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./will
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./create
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./a
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./new
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./file
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./or
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./change
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./time
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./stamp
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./of
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./an
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./existing
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./file.
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./Now
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./try
-rw-rw-r--. 1 tooy0303 tooy0303 0 Nov 24 04:05 ./again,
-rw-----. 1 tooy0303 tooy0303 4354 Nov 24 05:01 ./bash_history
-rw-rw-r--. 1 tooy0303 tooy0303 18 Nov 24 04:19 ./dir3/file2.txt
-rw-rw-r--. 1 tooy0303 tooy0303 18 Nov 24 04:19 ./dir3/hello.txt
-rw-rw-r--. 2 tooy0303 tooy0303 18 Nov 24 04:08 ./hello
-rw-rw-r--. 1 tooy0303 tooy0303 40 Nov 24 04:41 ./new.txt
-rw-rw-r--. 1 tooy0303 tooy0303 33 Nov 24 04:41 ./linux.txt
-sh-4.2$ find ~ -type f -size +20c -exec ls -hl {} \;
-rw-r--r--. 1 tooy0303 tooy0303 231 Nov 24 2021 /home/tooy0303/.bashrc
-rw-r--r--. 1 tooy0303 tooy0303 193 Nov 24 2021 /home/tooy0303/.bash_profile
-rw-----. 1 tooy0303 tooy0303 4.3K Nov 24 05:01 /home/tooy0303/.bash_history
-rw-rw-r--. 1 tooy0303 tooy0303 40 Nov 24 04:41 /home/tooy0303/new.txt
-rw-rw-r--. 1 tooy0303 tooy0303 33 Nov 24 04:41 /home/tooy0303/linux.txt
-sh-4.2$ find ~ -type f -size +20c -exec cp dir1 {} \;
cp: omitting directory 'dir1'
cp: omitting directory 'dir1'
cp: omitting directory 'dir1'
cp: omitting directory 'dir1'
cp: omitting directory 'dir1'
-sh-4.2$

```

รูปตัวอย่าง Lesson7

คำสั่ง	คำอธิบาย
file linux.txt	ใช้ในการแสดงประเภทของไฟล์ "linux.txt" โดยตรวจสอบความหลากหลายของไฟล์ตามลายลักษณ์บนไฟล์.
file /dev/null	ใช้ในการแสดงประเภทของไฟล์ที่ตำแหน่ง "/dev/null".
whereis ls	ใช้ในการค้นหาตำแหน่งของไบนารี (binary), ไฟล์ความช่วยเหลือ (man page), และไฟล์ที่ทำให้โปรแกรมรันได้ (source code) ของคำสั่ง "ls" ในระบบของคุณ.
whereis stdio.h	ใช้ในการค้นหาตำแหน่งของไฟล์ "stdio.h" ในระบบของคุณ.
php	ใช้เรียกใช้ PHP interpreter ในโปรแกรมคอมพิวเตอร์หรือ command line interface (CLI). การเรียกใช้คำสั่ง php โดยอยู่ในโหมดแบบ

	โต้ตอบ (interactive mode) จะเปิดหน้าต่างสำหรับป้อนคำสั่ง PHP ต่อมา.
which php	<p>ใช้ในการแสดงตำแหน่งของไฟล์ทำงาน (executable) สำหรับโปรแกรม PHP ในระบบของคุณ.</p> <p><code>find ~ -name "linux.txt"</code> คือ ใช้ในการค้นหาไฟล์ที่มีชื่อ "linux.txt" ในโฮมไดเรกทอรีของผู้ใช้ปัจจุบันและตำแหน่งย่อยทั้งหมด.</p> <p><code>~</code>: แทนโฮมไดเรกทอรีของผู้ใช้ปัจจุบัน.</p> <p><code>-name "linux.txt"</code>: กำหนดเงื่อนไขในการค้นหา ในที่นี้คือ การค้นหาไฟล์ที่มีชื่อเป็น "linux.txt"</p>

Lesson8 - System and user details

```
-sh-4.2$ uptime
06:55:29 up 551 days, 2:06, 0 users, load average: 0.16, 0.20, 0.21

-sh-4.2$ date
Fri Nov 24 06:55:36 UTC 2023

-sh-4.2$ who
-sh-4.2$ who -a
-sh-4.2$ w
06:55:57 up 551 days, 2:06, 0 users, load average: 0.09, 0.18, 0.20
USER      TTY      FROM          LOGIN@      IDLE   JCPU   PCPU   WHAT

-sh-4.2$ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime,hidepid=2)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=8188224k,nr_inodes=2047056,mode=755)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,mode=755)
tmpfs on /sys/fs/cgroup type tmpfs (ro,nosuid,nodev,noexec,seclabel,mode=755)
cgroup on /sys/fs/cgroup/systemd type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,xattr,release_agent=/usr/lib/systemd/systemd-cgroups-agent,name=systemd)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
cgroup on /sys/fs/cgroup/pids type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,pids)
cgroup on /sys/fs/cgroup/perf event type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,perf event)
cgroup on /sys/fs/cgroup/net_cls,net_prio type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,net_cls,net_prio)
cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,blkio)
cgroup on /sys/fs/cgroup/cpu,cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,cpu,cpuacct)
cgroup on /sys/fs/cgroup/memory type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,memory)
cgroup on /sys/fs/cgroup/rdma type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,rdma)
cgroup on /sys/fs/cgroup/devices type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,devices)
cgroup on /sys/fs/cgroup/hugetlb type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,hugetlb)
cgroup on /sys/fs/cgroup/freezer type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,freezer)
cgroup on /sys/fs/cgroup/cpuset type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,cpuset)
configs on /sys/kernel/config type configs (rw,relatime)
/dev/sda on / type ext4 (rw,relatime,seclabel)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,relatime)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,relatime,seclabel)
queue on /dev/queue type queue (rw,relatime,seclabel)
/dev/sdb on /home type xfs (rw,nosuid,nodev,noexec,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,usrquota,grpquota)
tmpfs on /run/user/0 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700)
tmpfs on /run/user/230165 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=230165,gid=230224)
tmpfs on /run/user/230291 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=230291,gid=230350)
tmpfs on /run/user/230347 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=230347,gid=230406)
tmpfs on /run/user/212599 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=212599,gid=212658)
tmpfs on /run/user/199548 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=199548,gid=199607)
tmpfs on /run/user/219058 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=219058,gid=219117)
tmpfs on /run/user/220923 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=220923,gid=220982)
tmpfs on /run/user/223554 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=223554,gid=223613)
tmpfs on /run/user/231909 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=231909,gid=231968)
tmpfs on /run/user/231900 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=231900,gid=231959)
tmpfs on /run/user/232285 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=232285,gid=232344)
tmpfs on /run/user/214148 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=214148,gid=214207)
tmpfs on /run/user/232491 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=232491,gid=232550)
tmpfs on /run/user/233083 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=233083,gid=233142)
tmpfs on /run/user/233137 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=233137,gid=233196)
tmpfs on /run/user/234183 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=234183,gid=234242)
tmpfs on /run/user/234240 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=1639972k,mode=700,uid=234240,gid=234299)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=26,grp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=96379412)
binfmt_misc on /proc/sys/fs/binfmt_misc type binfmt_misc (rw,relatime)
tracefs on /sys/kernel/debug/tracing type tracefs (rw,relatime,seclabel)
/dev/sdc on /common_pool type btrfs (rw,nosuid,nodev,relatime,seclabel,ssd,space_cache,subvolid=5,subvol=)
```

```

-sh-4.2$ mount -t ext4
/dev/sda on / type ext4 (rw,relatime,seclabel)
-sh-4.2$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs         7.9G   0  7.9G   0% /dev
tmpfs            7.9G 201M  7.7G   3% /dev/shm
tmpfs            7.9G 774M  7.1G  10% /run
tmpfs            7.9G   0  7.9G   0% /sys/fs/cgroup
/dev/sda        116G  31G   79G  29% /
/dev/sdb        118G  51G   67G  43% /home
tmpfs           1.6G   0  1.6G   0% /run/user/0
tmpfs           1.6G   0  1.6G   0% /run/user/230165
tmpfs           1.6G   0  1.6G   0% /run/user/230291
tmpfs           1.6G   0  1.6G   0% /run/user/230347
tmpfs           1.6G   0  1.6G   0% /run/user/212599
tmpfs           1.6G   0  1.6G   0% /run/user/199548
tmpfs           1.6G   0  1.6G   0% /run/user/219058
tmpfs           1.6G   0  1.6G   0% /run/user/220923
tmpfs           1.6G   0  1.6G   0% /run/user/223554
tmpfs           1.6G   0  1.6G   0% /run/user/231909
tmpfs           1.6G   0  1.6G   0% /run/user/231900
tmpfs           1.6G   0  1.6G   0% /run/user/232285
tmpfs           1.6G   0  1.6G   0% /run/user/214148
tmpfs           1.6G   0  1.6G   0% /run/user/232491
tmpfs           1.6G   0  1.6G   0% /run/user/233083
tmpfs           1.6G   0  1.6G   0% /run/user/233137
tmpfs           1.6G   0  1.6G   0% /run/user/234183
tmpfs           1.6G   0  1.6G   0% /run/user/234240
/dev/sdc         79G  2.1G   75G   3% /common_pool
-sh-4.2$ free -m
              total        used        free      shared  buff/cache   available
Mem:          16015         3622         2345          974        10046        11084
Swap:           0              0              0

```

รูปตัวอย่าง Lesson8

คำสั่ง	คำอธิบาย
uptime	ใช้เพื่อแสดงข้อมูลเกี่ยวกับระยะเวลาที่ระบบได้ทำงาน, จำนวนผู้ใช้ที่เข้าสู่ระบบ, และโหลดของระบบ (system load).
date	ใช้เพื่อแสดงเวลาและวันที่ปัจจุบันของระบบ.
who -a	ใช้ในการแสดงข้อมูลเกี่ยวกับผู้ใช้ที่เข้าสู่ระบบ รวมถึงข้อมูลเพิ่มเติมเกี่ยวกับการเชื่อมต่อ (login sessions).
w	ใช้ในการแสดงข้อมูลเกี่ยวกับผู้ใช้ที่เข้าสู่ระบบและกำลังทำงานในระบบ รวมถึงข้อมูลที่เกี่ยวข้องกับโหลดของระบบ (system load).
mount	ใช้ในการแสดงรายการของพาร์ติชันที่มีการติดตั้งและใช้งานในระบบไฟล์ของ Linux. ตำแหน่งการติดตั้ง, พาร์ติชัน, และประเภทของระบบไฟล์ที่กำลังใช้งานจะถูกแสดงในผลลัพธ์.
mount -t ext4	ใช้ในการแสดงรายการของพาร์ติชันที่ใช้ระบบไฟล์ประเภท ext4 ที่ถูกติดตั้งและใช้งานในระบบ Linux.
df -h	ใช้ในการแสดงข้อมูลเกี่ยวกับการใช้พื้นที่ดิสก์ (disk space) ในระบบไฟล์. ตัวอักษร -h ในที่นี้หมายถึง "human-readable," ทำให้ข้อมูลแสดงในรูปแบบที่อ่านง่ายสำหรับมนุษย์.

free -m	ใช้ในการแสดงข้อมูลเกี่ยวกับการใช้งานหน่วยความจำ (RAM) ในระบบ. ตัวอักษร -m หมายถึง "megabytes," ทำให้ข้อมูลแสดงในหน่วยเป็นเมกะไบต์.
---------	---

Lesson9 - Linux Process Basic commands 1

```

-sh-4.2$ hostname
139-162-5-218
-sh-4.2$ file /bin/hostname
/bin/hostname: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked (uses shared libs), for GNU/Linux 2.6.32, BuildID[sha1]=93633698bd11eeb4bee21a388c191a5656990d8e, stripped
-sh-4.2$ cat /bin/hostname
-----
-sh-4.2$ ps
  PID TTY          TIME CMD
29330 pts/126    00:00:00 sh
30039 pts/126    00:00:00 ps
-sh-4.2$ ps -o ppid 31400
  PPID
-sh-4.2$ ps -o ppid,cmd 31400
  PPID CMD
-sh-4.2$ ps -o ppid,cmd 27447
  PPID CMD

```

รูปตัวอย่าง Lesson9

คำสั่ง	คำอธิบาย
hostname	ใช้ในการแสดงชื่อของเครื่อง (hostname) ที่กำลังใช้งาน
file /bin/hostname	ใช้ในการแสดงประเภทของไฟล์ที่ตำแหน่ง "/bin/hostname"
cat /bin/hostname	<p>เรามีอักขระที่อ่านได้และอ่านไม่ได้รวมกัน ที่ใดที่หนึ่งในเอาต์พุตเหล่านั้น เรากำลังบอกให้ Kernel อ่านชื่อไฟล์ '/etc/hosts' และค้นหารายการ '127.0.0.1' (ซึ่งโดยทั่วไปจะชี้ไปที่ชื่อเครื่องโฮสต์) และแสดงเนื้อหานั้น</p> <p>เมื่อเราพิมพ์ไฟล์ปฏิบัติการ '/bin/hostname' ใน bash prompt เนื้อหาใดก็ตามที่คุณเห็นสองสามบรรทัดด้านบนจะถูกโหลดลงใน RAM เคอร์เนลปฏิบัติตามคำสั่งเหล่านั้นและดำเนินการ</p> <p>สุดท้ายนี้ ต่อไปนี้เป็นคำจำกัดความของกระบวนการ</p> <p>'กระบวนการไม่มีอะไรนอกจากเนื้อหาไฟล์ที่อยู่ใน RAM'</p>
ps	ใช้ในการแสดงข้อมูลเกี่ยวกับกระบวนการ (processes) ที่กำลังทำงานในระบบ.

Lesson10 - Linux Process Basic commands 2

```
-sh-4.2$ ps
  PID TTY          TIME CMD
 29330 pts/126    00:00:00 sh
 30039 pts/126    00:00:00 ps
-sh-4.2$ ps -o ppid 31400
  PPID
-sh-4.2$ ps -o ppid,cmd 31400
  PPID CMD
-sh-4.2$ ps -o ppid,cmd 27447
  PPID CMD
-sh-4.2$ ps
  PID TTY          TIME CMD
 29330 pts/126    00:00:00 sh
 30381 pts/126    00:00:00 ps
-sh-4.2$ sleep 5
-sh-4.2$ sleep 2
-sh-4.2$ sleep 5 &
[1] 30596
-sh-4.2$ sleep 2
[1]+  Done                  sleep 5
-sh-4.2$ sleep 5 &
[1] 30621
-sh-4.2$ ps
  PID TTY          TIME CMD
 29330 pts/126    00:00:00 sh
 30621 pts/126    00:00:00 sleep
 30626 pts/126    00:00:00 ps
-sh-4.2$ pstree 5254
[1]+  Done                  sleep 5
-sh-4.2$ sleep 45 &
[1] 30719
-sh-4.2$ sleep 45 &
[2] 30733
-sh-4.2$ sleep 45 &
[3] 30736
-sh-4.2$ sleep 45 &
[4] 30744
-sh-4.2$ sleep 45 &
[5] 30746
-sh-4.2$ pstree -p 5254
-sh-4.2$ jobs
[1]  Done                  sleep 45
[2]  Running              sleep 45 &
[3]  Running              sleep 45 &
[4]- Running              sleep 45 &
[5]+ Running              sleep 45 &
-sh-4.2$ fg 5
sleep 45
[2]  Done                  sleep 45
[3]  Done                  sleep 45
[4]- Done                  sleep 45
-sh-4.2$ jobs
-sh-4.2$ bg 5
-sh: bg: 5: no such job
-sh-4.2$ jobs
```

รูปตัวอย่าง Lesson10

เราว่ามีกระบวนการแรกที่เราเชื่อว่า 'init' พร้อมด้วย pid นี้คือพาเรนต์ของกระบวนการทั้งหมดในระบบ และกระบวนการชื่อ 'bash' ได้ตอบกับเคอร์เนลในนามของคำขอหรือคำสั่งของผู้ใช้

ตอนนี้เมื่อนั้นเข้าสู่ระบบและพิมพ์ ps ทุกครั้งที่คุณเข้าสู่ระบบ ระบบจะสร้าง parent bash ใหม่ ในกรณีที่เราทำงาน Manager Bash ของเราคือ 5254 คุณรู้ไหมว่าแต่ละคำสั่งนั้นเป็นกระบวนการใช้ไหม? เรามาสร้างกระบวนการเล็กๆ น้อยๆ สำหรับเซชันนี้กัน

Background process

พิมพ์ `sleep 5` มันค้างเป็นเวลา 5 วินาทีแล้วแจ้ง `bash prompt` ให้คุณอีกครั้ง

วิธีที่ `bash` ของเราสร้างกระบวนการลูกกำลังบล็อกการโทร หมายความว่ารันกระบวนการย่อยและรอให้เสร็จสิ้น แล้วกลับมาหาฉันอีกครั้ง กระบวนการเบื้องหลัง ดังนั้นเมื่อเรารันกระบวนการลูก (สลิป) เซลล์ 'Bash' จะรอให้มันเสร็จสิ้น

เราสามารถวิ่ง `sleep 2` โดยไม่ต้องห้อยเป็นเวลา 5 วินาทีได้! และหมายเลขที่คุณเห็น 5781 คือรหัสกระบวนการลูกพื้นหลัง มันเป็นการอ้างอิงของเรา จะตรวจสอบได้อย่างไรว่าเป็น pid ของเด็กจริงๆ? ง่ายๆ เพียงเริ่มกระบวนการลูกในพื้นหลังและดำเนินการคำสั่ง `'ps'`

`sleep 5` pid ของเราตรงกับเอาต์พุต `ps` เรามีวิธีที่ดีกว่าในการแสดงภาพนี้ไหม? ใช่ทำ เรามีคำสั่งชื่อ `'pstree'` ซึ่งจะบอกคุณถึงการแมประหว่างกระบวนการลูกและกระบวนการหลัก

มันบอกว่าเรามี Bash และโพรเซสลูกหนึ่งตัวชื่อ `'pstree'` คอยดูแลลูกของเรา

ตอนนี้มันบอกว่า 'Bash' มีลูกสองคนชื่อ `pstree` และกระบวนการเบื้องหลัง `'sleep'` `pstree` มีตัวเลือกให้พุดถึง `-p` ซึ่งแสดง pid ถัดจากชื่อกระบวนการ

pid ที่เราได้รับขณะเริ่มกระบวนการพื้นหลังเหมือนกับที่เราได้รับจากเอาต์พุต `pstree`

สมมติว่าเราได้เริ่มต้นงานเบื้องหลังที่ทำงานระยะยาว 4 งาน แต่ละงานใช้เวลา 145 วินาที และ 1 กระบวนการที่ยาวมากเป็นเวลา 3000 วินาที

List background jobs

`ptree` ให้ข้อมูลเกี่ยวกับงานทั้งหมด เราไม่ต้องการ `pstree(6410)` เพราะสนใจเฉพาะงานพื้นหลังเท่านั้น วิธีดูเฉพาะงานเหล่านั้น เพื่อจุดประสงค์นี้ เรามีคำสั่ง `'jobs'` ที่จะให้ผลลัพธ์เช่น

Foreground process

กระบวนการของเรา-[5] ทำงานเป็นเวลา 3,000 วินาที ซึ่งใช้เวลานานจึงจะเสร็จสมบูรณ์ พื้นหลังจะใช้เวลา CPU น้อยกว่าเมื่อเทียบกับกระบวนการที่ไม่ใช่พื้นหลัง กระบวนการเบื้องหน้า เราเข้าสู่กระบวนการเบื้องหน้ากันดีกว่า พิมพ์ `fg` ในกรณีของเรา เราต้องนำงานพื้นหลัง-5 มาด้วย

Switch between foreground to background

ไม่ต้องกังวล Linux มีความยืดหยุ่นมาก เราก็สามารถทำได้เช่นกัน เพียงกด `'ctrl+z'` คุณจะได้รับเอาต์พุตแจ้งว่างานหยุดแล้ว

Lesson11 - Linux Process states

-sh-4.2\$ ps S				
PID	TTY	STAT	TIME	COMMAND
26443	pts/107	Ss	0:00	-sh
26751	pts/107	T	0:00	sleep 100
26796	pts/107	T	0:00	seq 1 500000
26908	pts/107	R+	0:00	ps S
-sh-4.2\$ man ps				
-sh-4.2\$ ps -S				
PID	TTY	STAT	TIME	COMMAND
26443	pts/107	Ss	0:00	-sh
26751	pts/107	T	0:00	sleep 100
26796	pts/107	T	0:00	seq 1 500000
26942	pts/107	R+	0:00	ps -S
-sh-4.2\$ ps S				
PID	TTY	STAT	TIME	COMMAND
26443	pts/107	Ss	0:00	-sh
26751	pts/107	T	0:00	sleep 100
26796	pts/107	T	0:00	seq 1 500000
26961	pts/107	R+	0:00	ps S
-sh-4.2\$ T				
-sh: T: command not found				
-sh-4.2\$ ps S fg				
PID	TTY	STAT	TIME	COMMAND
26443	pts/107	Ss	0:00	-sh
26751	pts/107	T	0:00	_ sleep 100
26796	pts/107	T	0:00	_ seq 1 500000
27054	pts/107	R+	0:00	_ ps S fg
-sh-4.2\$ ps				
PID	TTY	TIME	CMD	
26443	pts/107	00:00:00	sh	
26751	pts/107	00:00:00	sleep	
26796	pts/107	00:00:00	seq	
27063	pts/107	00:00:00	ps	
-sh-4.2\$ bash				
[tooy0303@139-162-5-218 ~]\$ ps				
PID	TTY	TIME	CMD	
26443	pts/107	00:00:00	sh	
26751	pts/107	00:00:00	sleep	
26796	pts/107	00:00:00	seq	
27607	pts/107	00:00:00	bash	
27641	pts/107	00:00:00	ps	
[tooy0303@139-162-5-218 ~]\$ ((kill -STOP 2498))				
bash: kill: (2498) - No such process				
[tooy0303@139-162-5-218 ~]\$ ps S				
PID	TTY	STAT	TIME	COMMAND
26443	pts/107	Ss	0:00	-sh
26751	pts/107	T	0:00	sleep 100
26796	pts/107	T	0:00	seq 1 500000
27607	pts/107	S	0:00	bash
27785	pts/107	R+	0:00	ps S
[tooy0303@139-162-5-218 ~]\$ fg				
bash: fg: current: no such job				
[tooy0303@139-162-5-218 ~]\$ bash				
[tooy0303@139-162-5-218 ~]\$ killed				
bash: killed: command not found				
[tooy0303@139-162-5-218 ~]\$ ps -o ppid 3376				
PPID				
[tooy0303@139-162-5-218 ~]\$				
bash: 1: command not found				

รูปตัวอย่าง Lesson10

คำสั่ง	คำอธิบาย
D	การนอนหลับอย่างต่อเนื่อง (โดยปกติคือ IO)
R	กำลังรันหรือรันได้ (บนคิวรัน)
S	การนอนหลับแบบขัดจังหวะ (รอให้เหตุการณ์เสร็จสิ้น)
T	หยุดทำงานโดยสัญญาณควบคุมงาน
X	ตายแล้ว (ไม่ควรเห็น)
ps -S	STAT S ย่อมาจาก Interruptible sleep ที่นี่ bash/shell ของเรากำลังรอให้ลูก 28682 ดำเนินการให้เสร็จสิ้น 28682 อยู่ในสถานะกำลังทำงาน s - ย่อมาจาก session leader และ + บอกว่ามันจะทำงานในเบื้องหน้า
ps S	PID (Process ID): เลขระบุกระบวนการ ซึ่งเป็นตัวเลขที่ไม่ซ้ำกันที่ใช้ระบุแต่ละกระบวนการ. TTY (Terminal): บอกถึงที่สัญญาณเอาต์พุตของกระบวนการ หรือ terminal ที่กระบวนการนั้นๆ กำลังทำงานอยู่. STAT (Status): แสดงสถานะปัจจุบันของกระบวนการ เช่น R (Running), S (Sleeping), Z (Zombie) ฯลฯ TIME: เวลาที่ CPU ใช้ในการประมวลผลกระบวนการนั้นๆ. COMMAND: คำสั่งที่ใช้เริ่มต้นกระบวนการนั้นๆ.
seq 1 500000	ใช้ในการสร้างลำดับตัวเลขที่เริ่มต้นที่ 1 และสิ้นสุดที่ 500,000 แล้วแสดงผลลัพธ์บนหน้าจอ. โดยปกติ, ผลลัพธ์จะมีจำนวนมากเพื่อแสดงลำดับตัวเลขทั้งหมด.
Zombie process	Zombie เป็นกระบวนการที่ถูกยกเลิกแต่ไม่ได้ถูกเก็บเกี่ยวโดยผู้ปกครอง เมื่อเราบอกว่าพ่อแม่ไม่ได้เก็บเกี่ยว เราหมายถึง 'ผู้ปกครองยังไม่ได้รับสถานะการออกจากลูก' ลูกดำเนินการเสร็จสิ้นพร้อมทั้งสถานะออกและรอให้ผู้ปกครองขอ นี่เป็นกรณีที่ยุงยากในการทำซ้ำ ให้ลอง: เรามีผู้นำเซชันของเรา 2249
bash	ชื่อของ Shell (Bourne Again SHell) ซึ่งเป็นโปรแกรมที่ให้เราสามารถใช้ในการประมวลผลคำสั่งที่คุณป้อนในระบบ Unix-like. ถ้าคุณพิมพ์ "bash" และกด Enter, นั่นจะทำให้คุณเข้าสู่โหมด Bash Shell ที่ให้คุณป้อนคำสั่งโดยตรง. คุณจะเห็นพร้อมประสบการณ์คำสั่งคำสั่งที่สามารถใช้งานได้ Bash พิมพ์ ps

Orphaned process

เด็กกำพร้ากำลังดำเนินการโดยไม่มีผู้ปกครอง บิดามารดาเสียชีวิตโดยไม่ได้เก็บเกี่ยวบุตร (ได้รับสถานะออกจากบุตร) กระบวนการลูกที่ไม่มีพารেন্টเหล่านี้เรียกว่ากระบวนการกำพร้า กระบวนการเหล่านี้ถูกนำมาใช้โดย init ซึ่งมี pid 1 มาลองสร้างกระบวนการที่ถูกกลืนซึ่งคล้ายกับขั้นตอนข้างต้น:

คำสั่ง	คำอธิบาย
(sleep 100 & (kill -9 3329))	<p>ให้เราแยกแยะที่ละส่วน:</p> <p>sleep 100 &: คำสั่ง sleep นี้จะทำให้กระบวนการหยุดการทำงาน (sleep) นาน 100 วินาที และ & ทำให้กระบวนการนี้ทำงานใน background.</p> <p>(kill -9 3329): คำสั่ง kill นี้จะใช้สั่งการจบการทำงานของกระบวนการที่มี PID คือ 3329 โดยใช้สัญลักษณ์ -9 ซึ่งหมายถึงใช้สั่งการของการทำลาย (forceful kill). การใช้ () รวมคำสั่ง kill ใน subshell นี้ทำให้ทั้งคำสั่งทำงานในกระบวนการย่อยแยกต่างหาก.</p> <p>การรวมทั้งหมดเข้าด้วยกัน, กระบวนการ sleep 100 จะทำงานใน background นาน 100 วินาที และพร้อมกันนั้น คำสั่ง kill -9 3329 จะถูกเรียกใช้เพื่อทำลายกระบวนการที่มี PID เท่ากับ 3329 โดยใช้การทำลายแบบแรงค่วน</p>
ps S	จะเห็นได้ว่า PID 3329 ถูก kill แล้ว

สรุปผลการทดลอง

การฝึกใช้คำสั่งสำหรับการใช้งานเพื่อให้รู้ถึงการใช้งานของคำสั่งต่างๆ และให้สามารถนำไปใช้งานได้มีประสิทธิภาพ อย่างเช่นการใช้คำสั่งในการสร้างไฟล์ โดยการสร้างแบบหลายๆ โฟลเดอร์ mkdir หรือการสร้างไฟล์แบบเลือกโฟลเดอร์ทำให้เราไม่จำเป็นต้อง cd เข้าไปสร้างไฟล์แบบปกติจึงทำให้ช่วยประหยัดเวลาในการทำงานมากยิ่งขึ้น