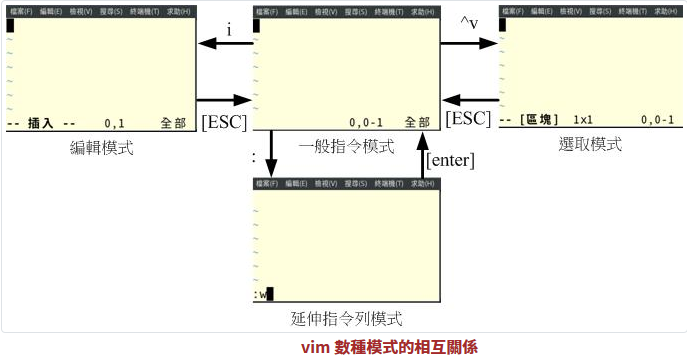
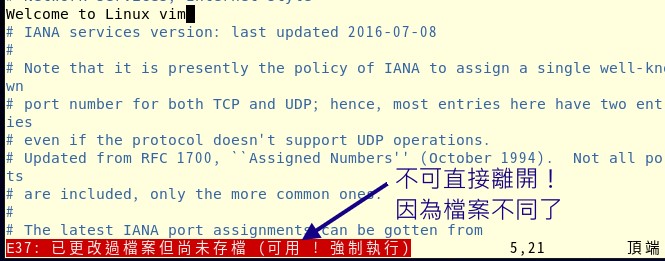
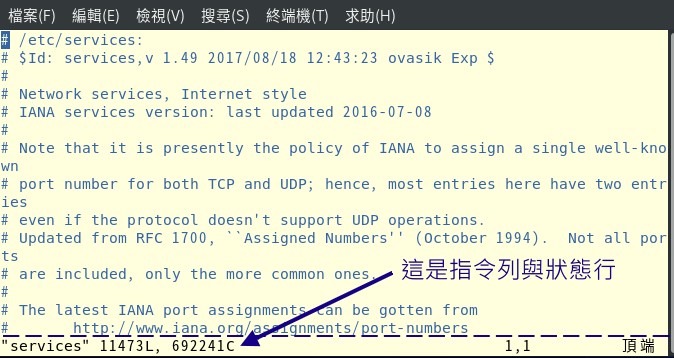
# vim

一般指令的y的測試

y+[space]: 複製游標所在字

y+[enter]: 複製游標所在行

y+↑:複製游標行及上一行

y+↓:複製游標行及下一行

y+←:複製游標所在前一字

y+→:複製游標所在字

很少在用

常用指令：

y: 複製1字元

yy: 複製1行

2yy: 複製2行

p: 貼下

P: 貼上

d: 刪除1字元 u: 復原

dd: 刪除1行

2dd: 刪除2行

代表檔名、行數、字元數

w: 寫入

q: 離開

q!: 強制離開

# 檔案管理

很重要，必考！！！

在 Linux 底下，所有的東西都以檔案來呈現，不同的檔案特性會有不同的結果。讀者可以常見的兩種檔案格式為：

* 一般檔案：實際放置資料的檔案
* 目錄檔案：重點在放置『檔名』，並沒有實際的資料

## 目錄的建立與刪除

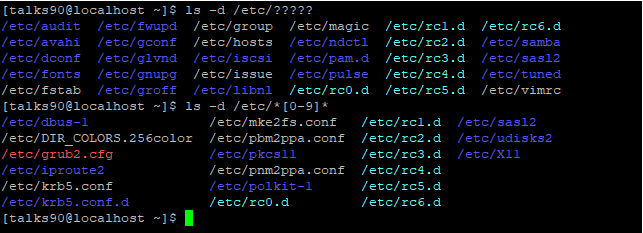
目錄的建立主要使用 mkdir 這個指令，這個指令將建立一個『空目錄』。所謂的『空目錄』意指該目錄內並沒有其他檔案的存在。 至於刪除目錄則使用 rmdir 這個指令，但同理，這個指令僅能『刪除空目錄』而已。

## 萬用字元

|  |  |
| --- | --- |
| 符號 | 意義 |
| \* | 代表『 0 個到無窮多個』任意字元 |
| ? | 代表『一定有一個』任意字元 |
| [ ] | 同樣代表『一定有一個在括號內』的字元(非任意字元)。例如 [abcd] 代表『一定有一個字元， 可能是 a, b, c, d 這四個任何一個』 |
| [ - ] | 若有減號在中括號內時，代表『在編碼順序內的所有字元』。例如 [0-9] 代表 0 到 9 之間的所有數字，因為數字的語系編碼是連續的！ |
| [^ ] | 若中括號內的第一個字元為指數符號 (^) ，那表示『反向選擇』，例如 [^abc] 代表 一定有一個字元，只要是非 a, b, c 的其他字元就接受的意思。 |

### 使用萬用字元搭配，來找出某些特殊的檔名

1. 列出 /etc/ 底下含有 5 個字元的檔名
2. 列出 /etc/ 底下含有數字在內的檔名



加上 –d 把目錄內容過濾掉，否則畫面就太亂了。

## 檔案及目錄的複製與刪除

### 複製目錄時

一般來說，複製目錄需要加上 -r 或 -a，兩者的差異如下：

|  |
| --- |
| # 先理解複製連結檔需要注意的事項：  [student@localhost ~]$ cd /dev/shm  [student@localhost shm]$ cp -r /etc/init.d .  [student@localhost shm]$ ll  lrwxrwxrwx. 1 student student 11 3月 10 09:12 init.d -> rc.d/init.d  # 由於這個 /etc/init.d 連結到的是一個目錄檔，因此，當你複製時，若結尾沒有加上 / 的話，  # 就會像這個案例一樣，你只是複製連結檔本身，而不是原本實際的資料。  [student@localhost shm]$ rm init.d  [student@localhost shm]$ cp -r /etc/init.d/ . <==結尾一定要加上斜線 /  [student@localhost shm]$ ll  drwxr-xr-x. 2 student student 80 3月 10 09:13 init.d <==這樣就複製過來了。  # 查看使用 -r 複製時，兩個目錄間的差異：  [student@localhost shm]$ ll init.d /etc/init.d/  /etc/init.d/:  總計 24  -rw-r--r--. 1 root root 18440 8月 23 2019 functions  -rw-r--r--. 1 root root 1161 11月 9 07:13 README  init.d:  總計 24  -rw-r--r--. 1 student student 18440 3月 10 09:13 functions  -rw-r--r--. 1 student student 1161 3月 10 09:13 README  # 因為我們用一般帳號嘗試複製，所以權限不同還 OK。問題是，怎麼連檔案的時間也不同？  # 測試 -a 複製時，嘗試複製權限的行為狀態：  [student@localhost shm]$ cp -a /etc/init.d/ init.d2  [student@localhost shm]$ ll init.d2  -rw-r--r--. 1 student student 18440 8月 23 2019 functions  -rw-r--r--. 1 student student 1161 11月 9 07:13 README  # 看起來時間參數就跟原本的相同了。 |

此外，如果以 root 的身份來執行上述指令時， 則連同權限 (前面的 root 變成 student) 也會跟原始檔案相同！這就是 -r 與 -a 的差異。因此，當系統備份時，還是建議使用 -a 的。

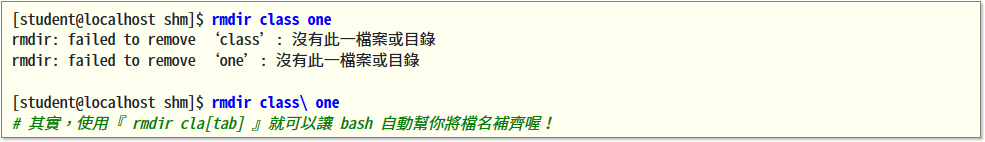
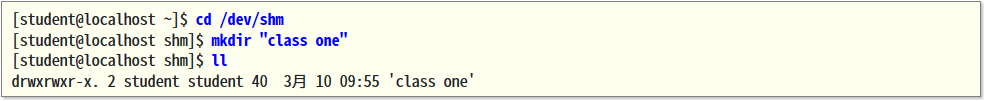
### 刪除檔案

刪除檔案使用 rm，其中需要特別注意，不要隨便使用 rm - rf 這樣的選項，因為 -r 為刪除目錄， -f 為不詢問直接刪除， 因此若後續的檔名寫錯時，將會有相當大的影響 (一般來說，檔案刪除是無法救援回來的)。

## 特殊檔名

### 空白字元的檔名

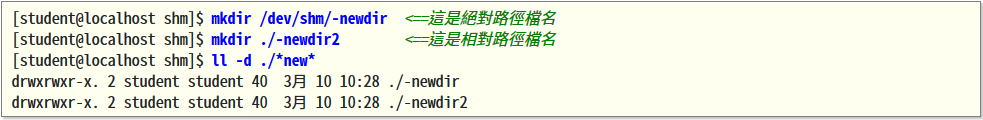
可以使用單引號或雙引號或反斜線 (\) 來處理這樣的檔名。CentOS8 很特別，在有特殊檔名的狀態下，會主動的在檔名左右兩側加上單引號～不過，這只是顯示的狀態而已， 事實上，這個檔名其實是沒有單引號的喔！



### 遇到開頭是+或-的檔名



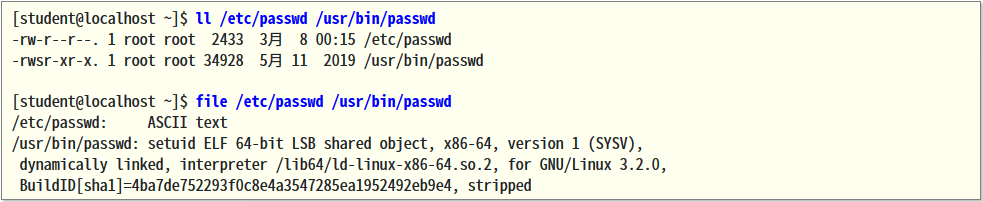
使用單引號來處理時，同樣回報錯誤！使用反斜線，同樣回報錯誤。



透過『絕對/相對路徑』的作法來處理就可以建立開頭為 + 或 - 的檔名。刪除同樣得要使用這樣的檔名撰寫方式來處理。

## 觀察檔案的類型

使用 file 這個指令可以觀察檔案的類型與型態。



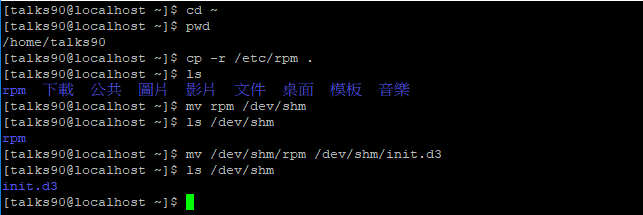
文字檔 (ASCII text)

執行檔 (ELF 64-bit LSB...)

## 檔名的移動與更名

### 如何進行檔案的更名

1. 讓 student 回到家目錄
2. 將 /etc/rpm 複製到本目錄
3. 該目錄移動錯誤，請將本目錄的 rpm 移動到 /dev/shm
4. 檔名依舊錯誤，請將 /dev/shm 底下的 rpm 更名為 init.d3



-r 連同目錄下面的東西都一起複製

## 大量建置空白檔案的方式

有時候為了測試系統，管理員可能需要建立許多的檔名來測試，此時可以透過 touch 這個指令來處理。如果需要建立較多的檔名，例如 test1, test2, test3, test4 時，可以透過大括號的方式來處理。

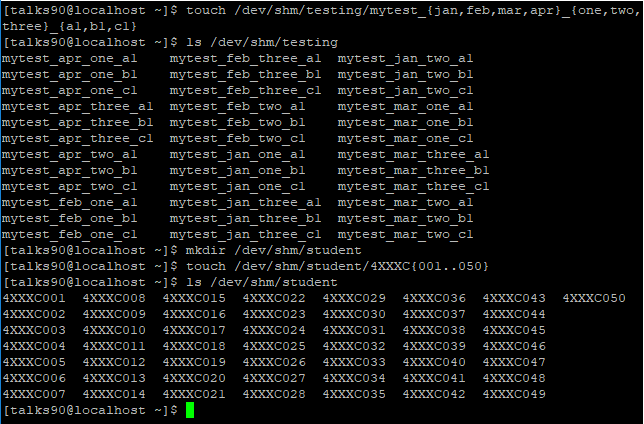


除了建空白檔案之外，還可以用來變更檔案修改時間

或用 {1..4} 來取代上述的輸出。若需要輸出 01, 02 這樣的字樣 {01..10} 來處理。

### 大量建置檔案的方法，透過 {} 的輔助

1. 我需要在 /dev/shm/testing 目錄下建立名為 mytest\_XX\_YY\_ZZ 的檔案，其中 XX 為 jan, feb, mar, apr 四個資料， YY 為 one, two, three 三個資料，而 ZZ 為 a1, b1, c1 三個資料，如何使用一個指令就建立出上述的 36 個檔案？
2. 我需要在 /dev/shm/student/ 目錄下，建立檔名為 4XXXC001 到 4XXXC050 的檔案，如何使用一個指令來完成這 50 個檔案的建置？



# 查詢檔案的內容

除了前面教過的 more, less 之外，還可以用 cat, tac, head, tail 等指令。

cat 一次把所有內容從頭到尾列出

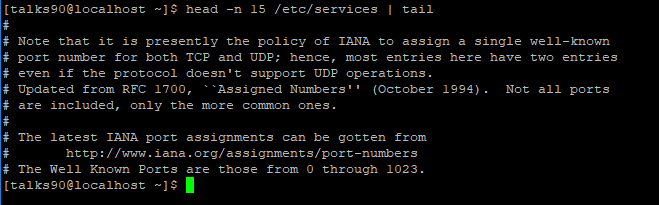
tac 一次把所有內容從尾到頭列出

head *n* 列出前面 n 行，預設是10

tail *n* 列出後面 n 行，預設是10

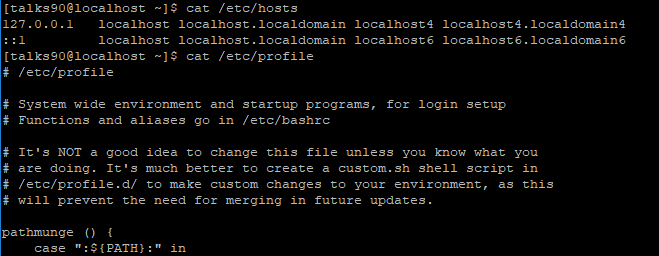
-n 可以列出及取用行號

### 怎麼查 5 - 15 行的資料？

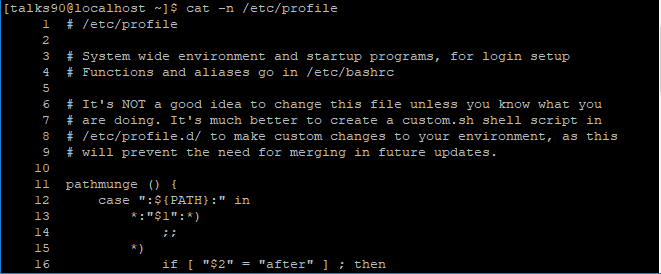


### 用不同的指令查詢檔案的內容

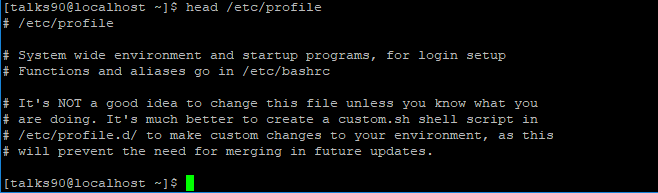
1. 列出 /etc/hosts 檔案的內容
2. 列出 /etc/profile 檔案的內容



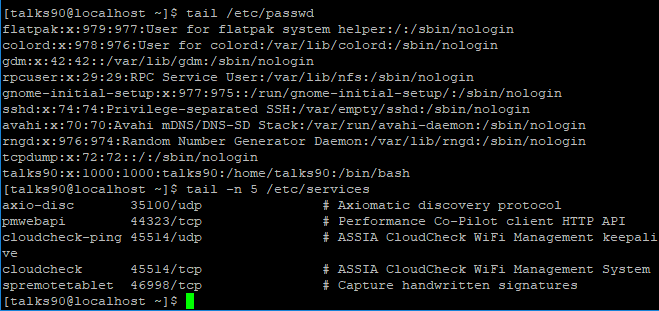
1. 承上，第二次列出 /etc/profile 時加上行號輸出



1. 讀者僅須列出 /etc/profile 的最前面 10 行



1. 讀者僅須列出 /etc/passwd 最後面 10 行的內容
2. 讀者僅須列出 /etc/services 最後 5 行的內容



# 指令表

|  |  |  |
| --- | --- | --- |
|  | 檔案 | 目錄 |
| 新增 | Touch  vim / nono | mkdir |
| 修改 | cp, mv  vim | mv, cp |
| 刪除 | Rm | rm –r（非空目錄）  rmdir（空目錄） |
| 查詢 | more, less  cat, tac  head, tail | ls, ll |
| 找尋內容 | grep, find | find |

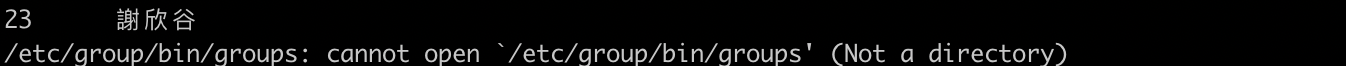
# 課後練習

1. 關於 vim 建立檔案，以及資料的轉存情況：
   1. 先用 vim 建立一個名為 /dev/shm/mycheck.txt 的檔案，裡面填寫你的姓名與學號，一行一個

一張含有 文字 的圖片

自動產生的描述

* 1. 使用 file 去檢查 /etc/group /bin/groups 的檔案類型，並將查詢的資料轉存到 /dev/shm/mycheck.txt 當中

* 1. 查詢出 /etc/ 底下，使用 s 開頭以及 s 結尾的檔名 (不含子目錄下的檔案)，並將找到的檔名轉存一份到 /dev/shm/mycheck.txt 當中

 一張含有 文字 的圖片

自動產生的描述

* 1. 找出最近有使用過的 rm 的指令，並將該執行的指令轉存到 /dev/shm/mycheck.txt 當中

 一張含有 文字 的圖片

自動產生的描述

* 1. 將 /etc/passwd 前 5 行與最後 5 行抓出來，並且附上行號之後，轉存到 /dev/shm/mycheck.txt 當中

一張含有 文字 的圖片

自動產生的描述

1. 關於檔案管理：
   1. 嘗試在 /dev/shm/unit03/files/ 目錄下，新增 ksu001 ~ ksu100 這 100 個空檔案

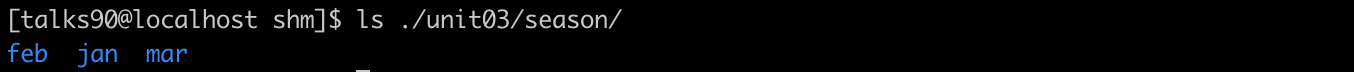
一張含有 文字 的圖片

自動產生的描述

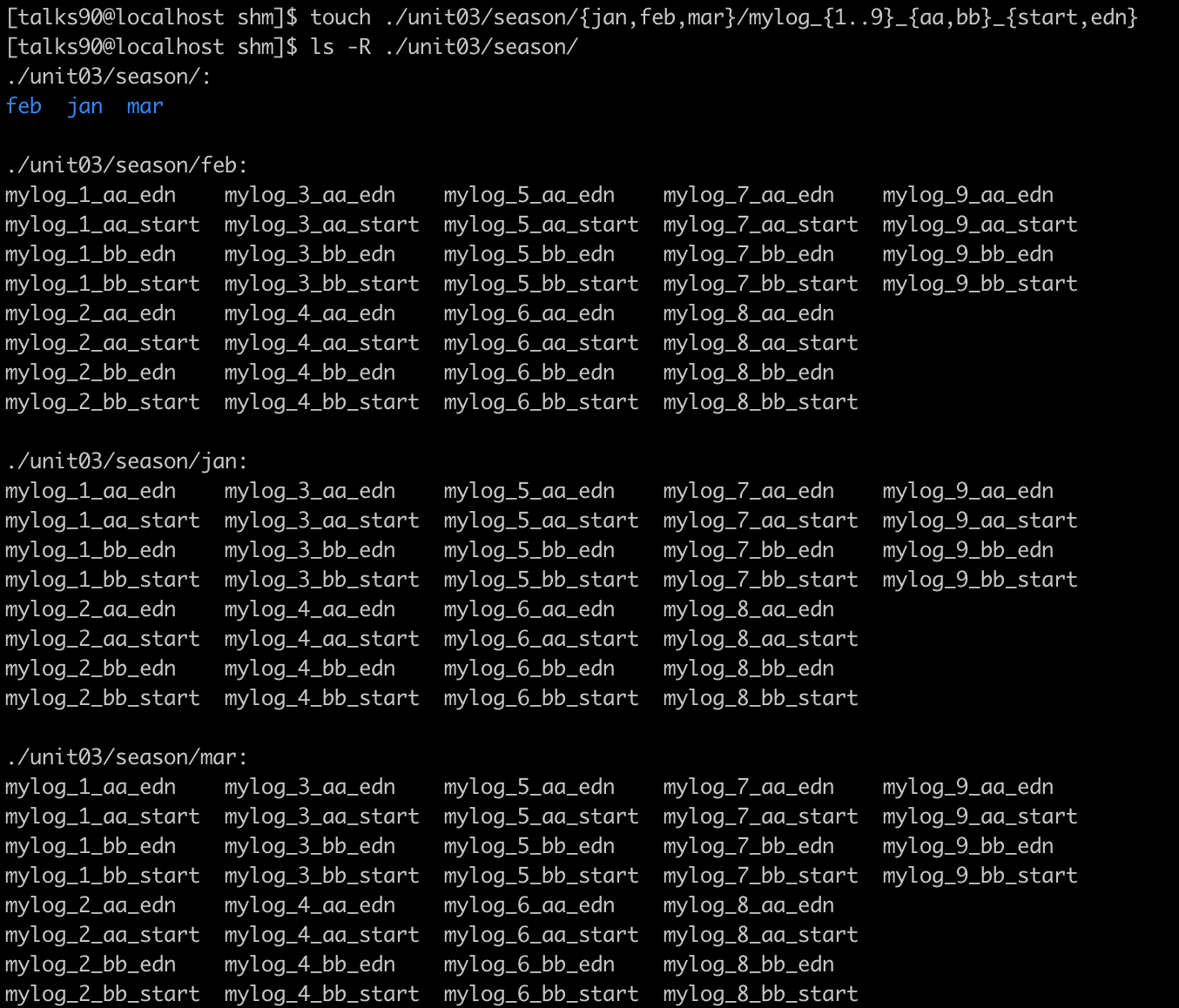
* 1. 嘗試在 /dev/shm/unit03/dirs/ 目錄下，新增 dir01 ~ dir10 這 10 個空目錄



* 1. 嘗試在 /dev/shm/unit03/season/ 目錄下，新增 jan, feb, mar 這三個空目錄



* 1. 嘗試在 /dev/shm/unit03/season/ 目錄下的 jan, feb, mar 子目錄內，建立 mylog\_XX\_YY\_ZZ.txt 空檔案，其中：『XX 為 1~9』, 『YY 為 aa 與 bb』，『ZZ 為 start 與 end』。



* 1. 嘗試在 /dev/shm/unit03/spc/ 目錄下，建立名為 -check.txt 的目錄

