

2-1-2 限以 read 輸入指令，test 辨別是否輸入指令，然後執行，執行結果輸出於螢幕上，錯誤訊息則不出現，並將指令紀錄於 commRecorder.txt(累加)

程式碼 由 read 來給予變數值再來用 test 判斷是否非空值後執行使用者輸入的命令

把錯誤的訊息丟到/dev/null 的黑洞

然後由 tee 把正確訊息輸出在螢幕與 commRecorder.txt 檔案裡

```
1 #!/bin/bash
2 #Description: read test & >
3 #Written by Rico
4 #Version 1.00
5
6 read -p "give me command: " myComm
7
8 if [ -n $myComm ]
9 then
10     $myComm 2>> /dev/null | tee -a /home/iot/Rico/cc101/SS/homework/commRecorder.txt
11
12 fi
13
14 exit 0
```

一開始還沒有 commRecorder.txt 這個檔案

執行剛撰寫好的 script 後輸入想要執行的命令，正確訊息有輸出到螢幕上

用 ls 查看是否有檔案？

再用 cat 查看裡面的內容

```
iot@Xenial-Host:20:57:11Sat Apr 281188~/Rico/cc101/SS/homework$ls
06_陳騏瑄1-1.docx 06_陳騏瑄1-1.pdf 06_陳騏瑄1-2.docx 06_陳騏瑄1-2.pdf 06_陳騏瑄2-1.docx 2-1-2
iot@Xenial-Host:20:57:15Sat Apr 281189~/Rico/cc101/SS/homework$bash 2-1-2
give me command: date
Sat Apr 28 20:57:35 CST 2018
iot@Xenial-Host:20:57:35Sat Apr 281190~/Rico/cc101/SS/homework$ls
06_陳騏瑄1-1.docx 06_陳騏瑄1-1.pdf 06_陳騏瑄1-2.docx 06_陳騏瑄1-2.pdf 06_陳騏瑄2-1.docx 2-1-2 commRecorder.txt
iot@Xenial-Host:20:57:51Sat Apr 281191~/Rico/cc101/SS/homework$cat commRecorder.txt
Sat Apr 28 20:57:35 CST 2018
```

如果純粹用 fdisk 沒加任何參數是不能執行的，但錯誤訊息已經被丟到/dev/null 的黑洞裡，所以不會顯示任何東西

然後再輸入一次 date，可以發現 commRecorder.txt 的內容增加了

```
iot@Xenial-Host:21:03:43Sat Apr 281195~/Rico/cc101/SS/homework$bash 2-1-2
give me command: fdisk
iot@Xenial-Host:21:04:08Sat Apr 281196~/Rico/cc101/SS/homework$bash 2-1-2
give me command: date
Sat Apr 28 21:05:15 CST 2018
iot@Xenial-Host:21:05:15Sat Apr 281196~/Rico/cc101/SS/homework$cat commRecorder.txt
Sat Apr 28 20:57:35 CST 2018
Sat Apr 28 21:05:15 CST 2018
```

2-1-3 限以 `read` 與 `test` 設計一個程式讓使用者輸入一個檔名或目錄(均可含路徑)後，判斷是否存在，並判別是否有 `RWX` 屬性，並列出相關訊息。

一樣先以 `read` 指令來讓使用者設定檔名的變數

先用 `if` 後以 `test` 判斷是否為正規檔案 `-f` 來執行接下來的 3 個指令

以 3 個分開的 `if` 來個別判斷檔案是否有讀的權限(然後顯示可以讀的資訊給使用者)，寫的權限(然後顯示可以寫的資訊給使用者)，執行的權限(然後顯示可以執行的資訊給使用者)

```
1 #!/bin/bash
2 #Description : read test permission for read write exec
3 #Written by Rico
4 #Version 1.00
5
6 read -p "give me a filename: " myFile
7
8 if [ -f $myFile ]
9 then
10     if [ -r $myFile ]
11     then
12         echo "$myFile you can read"
13     fi
14
15     if [ -w $myFile ]
16     then
17         echo "$myFile you can write"
18     fi
19
20     if [ -x $myFile ]
21     then
22         echo "$myFile you can exec"
23     fi
24 fi
25
26 exit 0
27
```

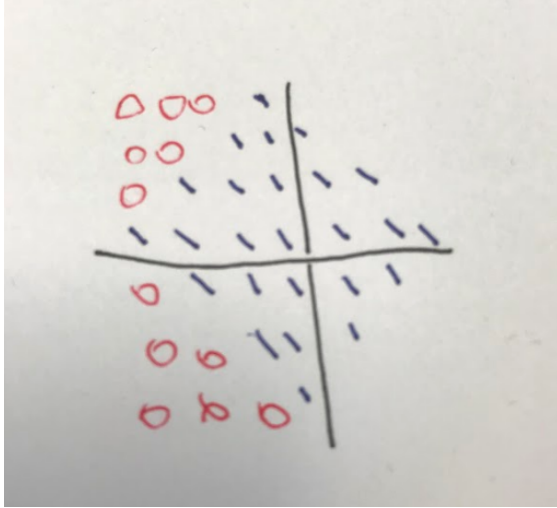
先測試 `/etc/passwd` 這個檔案，因為為一般使用者，`other` 的權限只有讀而已

再來查看上一題創建的 `commRecorder.txt`，顯示出我有讀跟寫的權限。

使用 `chmod u+x` 把擁有者的權限全部打開，再一次執行剛撰寫號的 `shell script` 可以看出所有權限都打開了。

```
iot@Xenial-Host:21:31:18Sat Apr 281204~/Rico/cc101/SS/homework$bash 2-1-3
give me a filename: /etc/passwd
/etc/passwd you can read
iot@Xenial-Host:21:31:33Sat Apr 281205~/Rico/cc101/SS/homework$ls -al /etc/passwd
-rw-r--r-- 1 root root 2594 Mar  7 22:26 /etc/passwd
iot@Xenial-Host:21:31:46Sat Apr 281206~/Rico/cc101/SS/homework$bash 2-1-3
give me a filename: ./commRecorder.txt
./commRecorder.txt you can read
./commRecorder.txt you can write
iot@Xenial-Host:21:32:24Sat Apr 281207~/Rico/cc101/SS/homework$ls -al commRecorder.txt
-rw-rw-r-- 1 iot iot 58 Apr 28 21:05 commRecorder.txt
iot@Xenial-Host:21:32:35Sat Apr 281208~/Rico/cc101/SS/homework$chmod u+x commRecorder.txt
iot@Xenial-Host:21:33:09Sat Apr 281209~/Rico/cc101/SS/homework$bash 2-1-3
give me a filename: ./commRecorder.txt
./commRecorder.txt you can read
./commRecorder.txt you can write
./commRecorder.txt you can exec
iot@Xenial-Host:21:33:22Sat Apr 281210~/Rico/cc101/SS/homework$ls -al commRecorder.txt
-rwxrw-r-- 1 iot iot 58 Apr 28 21:05 commRecorder.txt
iot@Xenial-Host:21:33:32Sat Apr 281211~/Rico/cc101/SS/homework$
```

2-1-16 請以 for 列印出如下格式



畫出菱形的概念上圖就很清楚了，把他劃分成 4 個有\*的三角形與空格的 2 個三角形，然後以上與下兩個部份來寫

```
1 #!/bin/bash
2 #Description : draw a diamond!!
3 #Written by Rico
4 #Version 1.00
5
6 #做上半部份的大三角形，先從行數開始
7 for (( x=1 ; x<=4 ; x=x+1 ))
8 do
9     #先畫每一行的空白處，為倒三角形
10    for (( y=3 ; y>=x ; y=y-1 ))
11    do
12        echo -n " "
13    done
14    #接下來畫每一行左半部份的*
15    for (( z=1 ; z<=x ; z=z+1 ))
16    do
17        echo -n "*"
18    done
19    #畫每一行裡右半部份的*
20    for (( a=1 ; a<=x ; a=a+1 ))
21    do
22        echo -n "*"
23    done
24    #每行畫完就換行
25    echo ""
26 done
```

27

## 最後呈現結果

```
iot@Xenial-Host:~/Rico/cc101/SS/homework$ bash 2-1-16
*
***
*****
*****
*****
*****
***
*
```