

Lab: >

```
#cd /home/max
```

```
#ls /
```

```
#echo $?
```

<\$?為回覆值,通常 0 代表成功, 故 ls / 為 STDOUT>

```
#ls /123
```

```
#echo $?
```

<\$?為回覆值,通常 0 代表成功, 故 ls /123 為 STDERR>

```
#ls / > ls.txt
```

<將 ls / 的結果導向到 ls.txt ">" 預設針對 STDOUT >

```
#/bin/ls
```

```
#cat ls.txt
```

<請觀察內容>

```
#ls /123 > ls.txt
```

<ls /123 訊息為標準錯誤, 故無法導向到 ls.txt>

```
#cat ls.txt
```

<請觀察內容, 空白,因 ">" 新的內容會覆蓋舊的 >

```
#ls /123 2> ls.txt
```

<將 ls /123 的 STDERR 導向到 ls.txt, 2> 指定 STDERR>

```
#cat ls.txt
```

Q: 若執行#ls / /123 會同時出現 STDOUT and STDERR

試問要如何將 STDOUT and STDERR 導向 ls.txt?

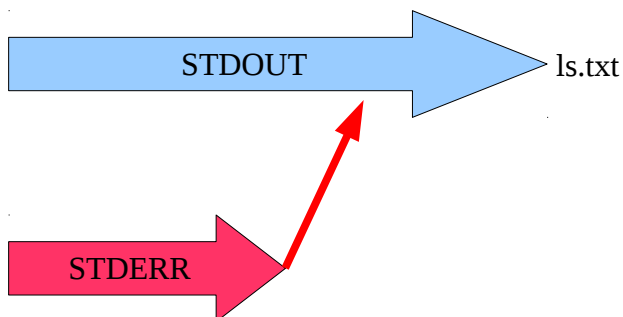
Hint: 2>&1

Hint: 1>&2

Ans:

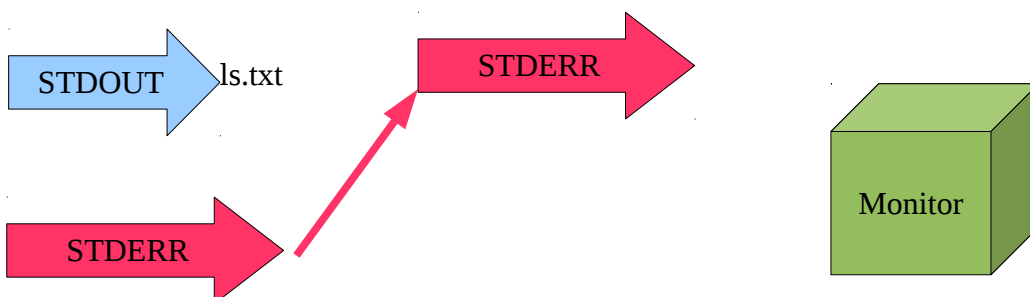
正確

```
#ls / /123 > ls.txt 2>&1
```



錯誤

```
#ls / /123 2>&1 > ls.txt
```



Lab: >> , <

```
#date
```

```
#date > df.txt
```

<將 date 結果導向 df.txt>

```
#cat df.txt
```

```
#df -h > df.txt
```

<將 df -h 結果導向 df.txt,因使用">" 故新的會覆蓋舊的>

```
#cat df.txt
```

```
#date >> df.txt
```

<使用附加的方式將 date 結果導向 df.txt >

```
#cat df.txt
```

<使用 ">>" 新的訊息會附加在舊的後面 >

```
#mail root -s "disk free" < df.txt
```

<以 df.txt 為信件內容寫信給 root,主旨為 disk free >

```
#mail
```

<收信,請按信件編號觀察,按 q 離開>

Lab: &&, ||

```
#ls / && echo "novell"
```

```
#ls /123 && echo "novell"
```

```
#ls / || echo "novell"
```

```
#ls /123 || echo "novell"
```