

# 計算機概論A班 實習課

W3

# 課程內容

→ Linux背景介紹

- ◆ 壓縮檔案

- ◆ 資料傳輸介紹

→ 基本指令實作

- ◆ 壓縮檔案

- ◆ 檔案搜尋

- ◆ 檔案&文字相關

- ◆ 網路相關指令

→ HW

# Linux 背景知識 - 壓縮檔案

為什麼要壓縮檔案呢？

1. 備份資料的時候，方便整理。
2. 將檔案變小，節省電腦硬碟的空間。(但圖片、音訊、視訊等多媒體檔案壓縮率低，並不能有效節省空間)
3. 將無數個散亂的檔案打包成一個較小的檔案，亦方便資訊在網路上流通。
4. 壓縮檔案時，可以視情況進行加密。

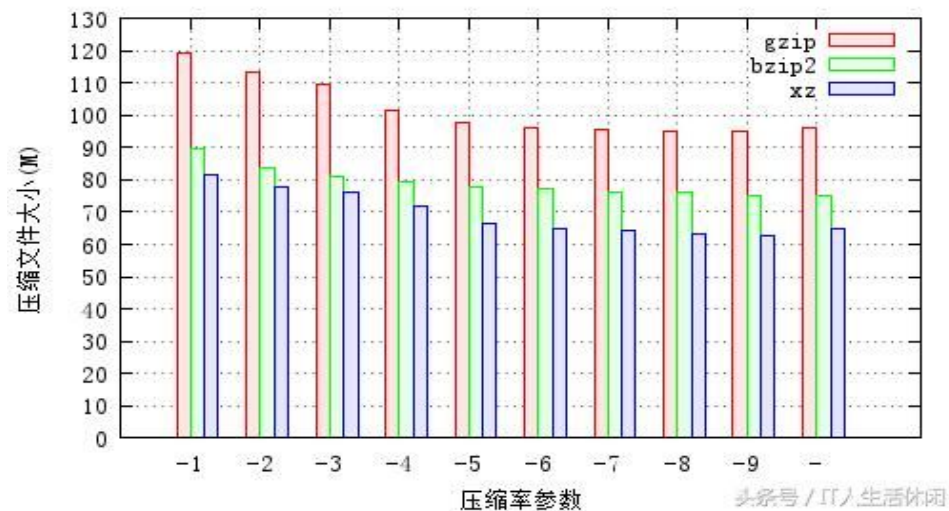
# Linux背景知識-各壓縮的差別

## → 考慮的因素

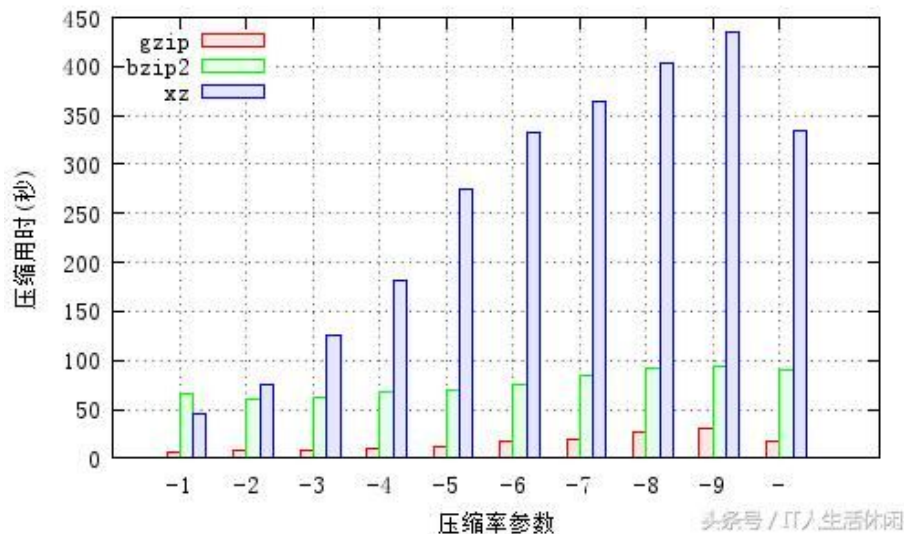
- ◆ 壓縮率 (compression ratio), 能夠將檔案壓到多小。
- ◆ 解壓縮所需的時間, 也就是需要的 CPU 計算量。
- ◆ 解壓縮所需的記憶體空間。
- ◆ 相容性 (compatibility), 即解壓縮程式的普遍性, 是不是大部分人都  
有辦法解壓縮這種格式?

# Linux背景知識-各壓縮的差別

在不同壓縮級別下，壓縮同一文件後的大小：

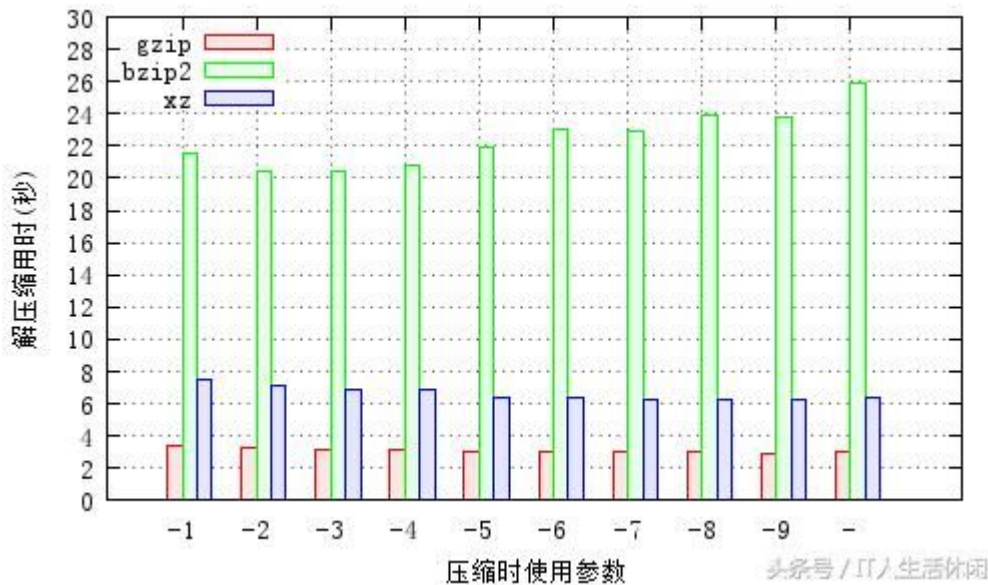


在不同壓縮級別下，壓縮同一文件用的時間對比：



# Linux背景知識-各壓縮的差別

在不同壓縮級別下，解壓同一文件用的時間對比：



## Linux背景知識-各壓縮的差別

- 需要在記憶體很小的機器(如小於 128MB)上解壓縮時, 則選擇 `gzip` 格式。
- 需要在很簡單、沒有什麼工具可用的機器解壓縮時, 則選擇 `gzip` 格式。
- 需要節省網路頻寬、縮短下載所需要的時間時, 則選擇 `xz` 格式。
- 需要有最好的壓縮率時, 則選擇 `tar.xz` 格式。

# Linux 指令 - 壓縮檔案

→ gzip

- ◆ 壓縮: `gzip FileName`
- ◆ 解壓縮:
  - `gunzip FileName.gz`
  - `gzip -d FileName.gz`

→ xz

- ◆ 壓縮: `xz -z FileName`
- ◆ 解壓縮: `xz -d FileName.xz`



# Linux 指令 - 壓縮檔案

➔ tar.gz

- ◆ 壓縮: `tar -zcvf FileName.tar.gz DirName`
- ◆ 解壓縮: `tar -zxvf FileName.tar.gz`

# Linux 指令 - 檔案搜尋

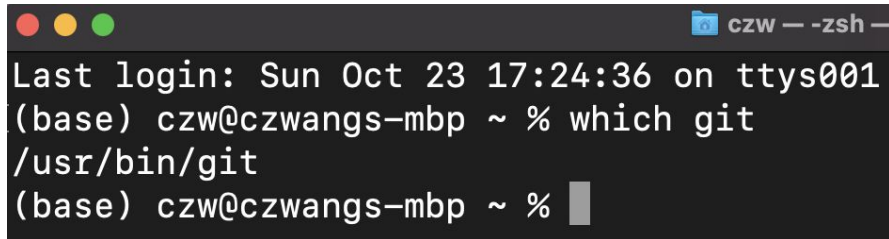
→ `find [path] [option] [action] filename`

## ◆ option

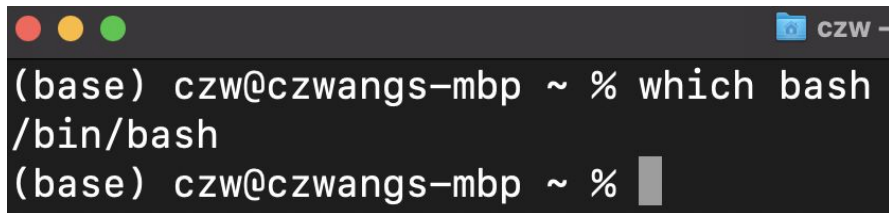
- `-size EX`: 找出大於500M的檔案 → `-size +500M`
- `-name EX`: 找出為照片的檔案 → `-name "*.jpg"`
- `-type EX`: `-type f` → 一般檔案; `-type d` → 一般目錄
- `-user EX`: 同時找兩個擁有者的檔案  
→ `-user user1 -o -user user2`

# Linux 指令 - 檔案搜尋

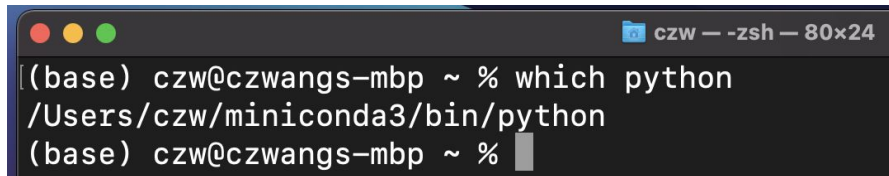
- which filename
  - -a:系統會顯示所有被找到的命令執行檔之完整路徑
  - -n<文件名長度>指定文件名長度, 指定的長度必須大於或等於所有文件中最長的文件名。
  - -p<文件名長度>與-n參數相同, 但此處的<文件名長度>包括了文件的路徑。
  - -w:指定輸出欄位的寬度。
  - -V:顯示版本訊息。



```
czw — -zsh —
Last login: Sun Oct 23 17:24:36 on ttys001
(base) czw@czwangs-mbp ~ % which git
/usr/bin/git
(base) czw@czwangs-mbp ~ %
```



```
czw —
(base) czw@czwangs-mbp ~ % which bash
/bin/bash
(base) czw@czwangs-mbp ~ %
```

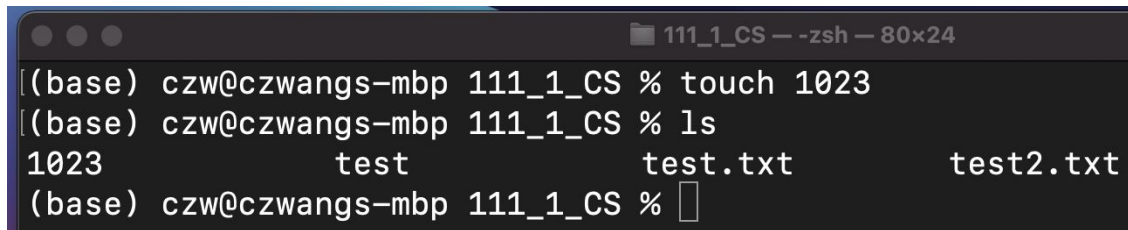


```
czw — -zsh — 80x24
(base) czw@czwangs-mbp ~ % which python
/Users/czw/miniconda3/bin/python
(base) czw@czwangs-mbp ~ %
```

# Linux 指令 - 檔案&文字相關

→ touch

- ◆ 效果:建立空白檔案
- ◆ 用法:touch filename

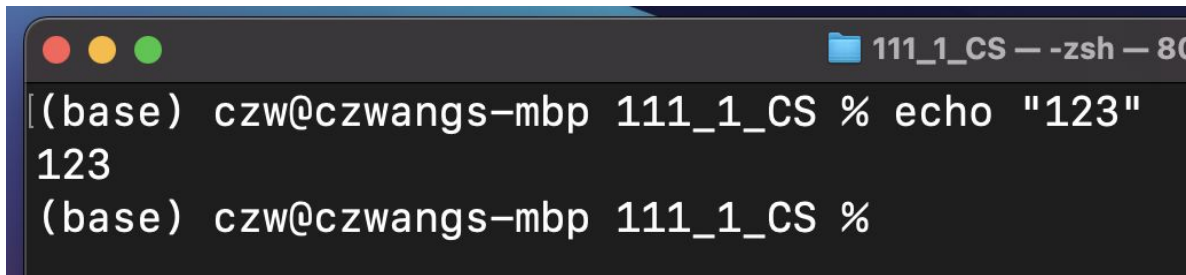
A terminal window titled '111\_1\_CS - zsh - 80x24' showing a sequence of commands and their output. The user runs 'touch 1023', then 'ls', which lists '1023', 'test', 'test.txt', and 'test2.txt'. The prompt then returns to the shell.

```
(base) czw@czwangs-mbp 111_1_CS % touch 1023
(base) czw@czwangs-mbp 111_1_CS % ls
1023          test          test.txt      test2.txt
(base) czw@czwangs-mbp 111_1_CS %
```

# Linux 指令 - 檔案&文字相關

→ echo

- ◆ 效果:印出文字
- ◆ 用法:echo “text”

A screenshot of a macOS terminal window. The title bar at the top shows three colored window control buttons (red, yellow, green) on the left and a blue folder icon followed by the text '111\_1\_CS - -zsh - 80' on the right. The terminal content shows a shell prompt '(base) czw@czwangs-mbp 111\_1\_CS %' followed by the command 'echo "123"'. The output '123' is printed on the next line. The prompt is then shown again as '(base) czw@czwangs-mbp 111\_1\_CS %' on the following line.

```
(base) czw@czwangs-mbp 111_1_CS % echo "123"
123
(base) czw@czwangs-mbp 111_1_CS %
```

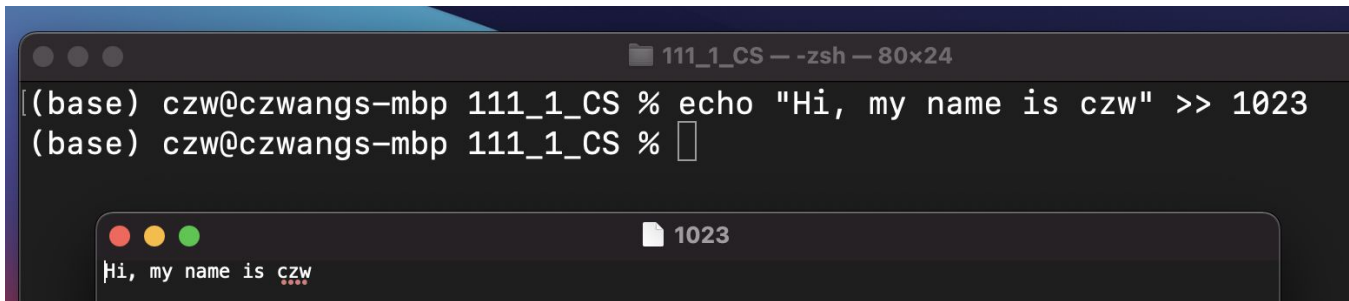
# Linux 指令 - 檔案&文字相關

→ echo 進階用法

◆ 用法: `echo "text" >> filename`

`echo "text" > filename`

◆ 效果: 將文字寫入 >> or > 後的檔案



```
111_1_CS — zsh — 80x24
(base) czw@czwangs-mbp 111_1_CS % echo "Hi, my name is czw" >> 1023
(base) czw@czwangs-mbp 111_1_CS %
```

The screenshot shows a terminal window with a dark background. The title bar indicates the window is titled "111\_1\_CS" and is running "zsh" with a size of "80x24". The prompt is "(base) czw@czwangs-mbp 111\_1\_CS %". The user has entered the command "echo 'Hi, my name is czw' >> 1023". Below the command, there is a cursor. At the bottom of the terminal, there is a preview of the file "1023" which contains the text "Hi, my name is czw".

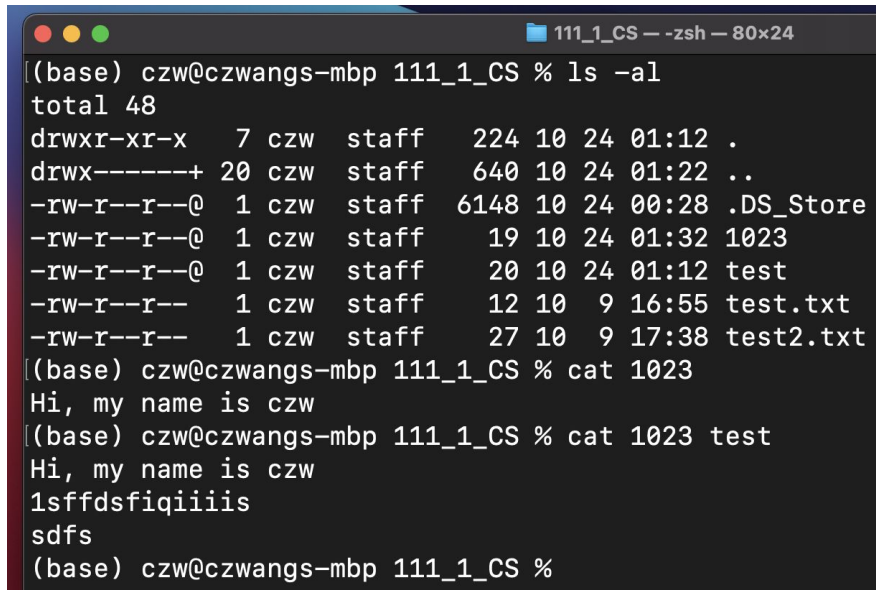
# Linux 指令 - 檔案&文字相關

→ cat (concatenate)

◆ 效果: 將文件內容串接後顯示

(較常用於顯示單一檔案的文字 內容)

◆ 用法: cat file1 file2

A terminal window titled '111\_1\_CS - zsh - 80x24' showing a series of commands and their outputs. The first command is 'ls -al', which lists the contents of the current directory with detailed permissions, owner, group, size, and timestamps. The output shows files like '.', '..', '.DS\_Store', '1023', 'test', 'test.txt', and 'test2.txt'. The second command is 'cat 1023', which displays the content of the file '1023', which is 'Hi, my name is czw'. The third command is 'cat 1023 test', which concatenates the content of '1023' and 'test', resulting in 'Hi, my name is czw' followed by '1sffdsfiqiiiiis' and 'sdfs' on separate lines.

```
(base) czw@czwangs-mbp 111_1_CS % ls -al
total 48
drwxr-xr-x  7 czw  staff   224 10 24 01:12 .
drwx-----+ 20 czw  staff   640 10 24 01:22 ..
-rw-r--r--@  1 czw  staff  6148 10 24 00:28 .DS_Store
-rw-r--r--@  1 czw  staff    19 10 24 01:32 1023
-rw-r--r--@  1 czw  staff    20 10 24 01:12 test
-rw-r--r--  1 czw  staff    12 10  9 16:55 test.txt
-rw-r--r--  1 czw  staff    27 10  9 17:38 test2.txt
(base) czw@czwangs-mbp 111_1_CS % cat 1023
Hi, my name is czw
(base) czw@czwangs-mbp 111_1_CS % cat 1023 test
Hi, my name is czw
1sffdsfiqiiiiis
sdfs
(base) czw@czwangs-mbp 111_1_CS %
```

# 討論時間

Q: 檔案怎麼可以沒有副檔名？

A: Linux 不依賴副檔名來判斷檔案的類型！



# 練習時間：新建、寫入、顯示

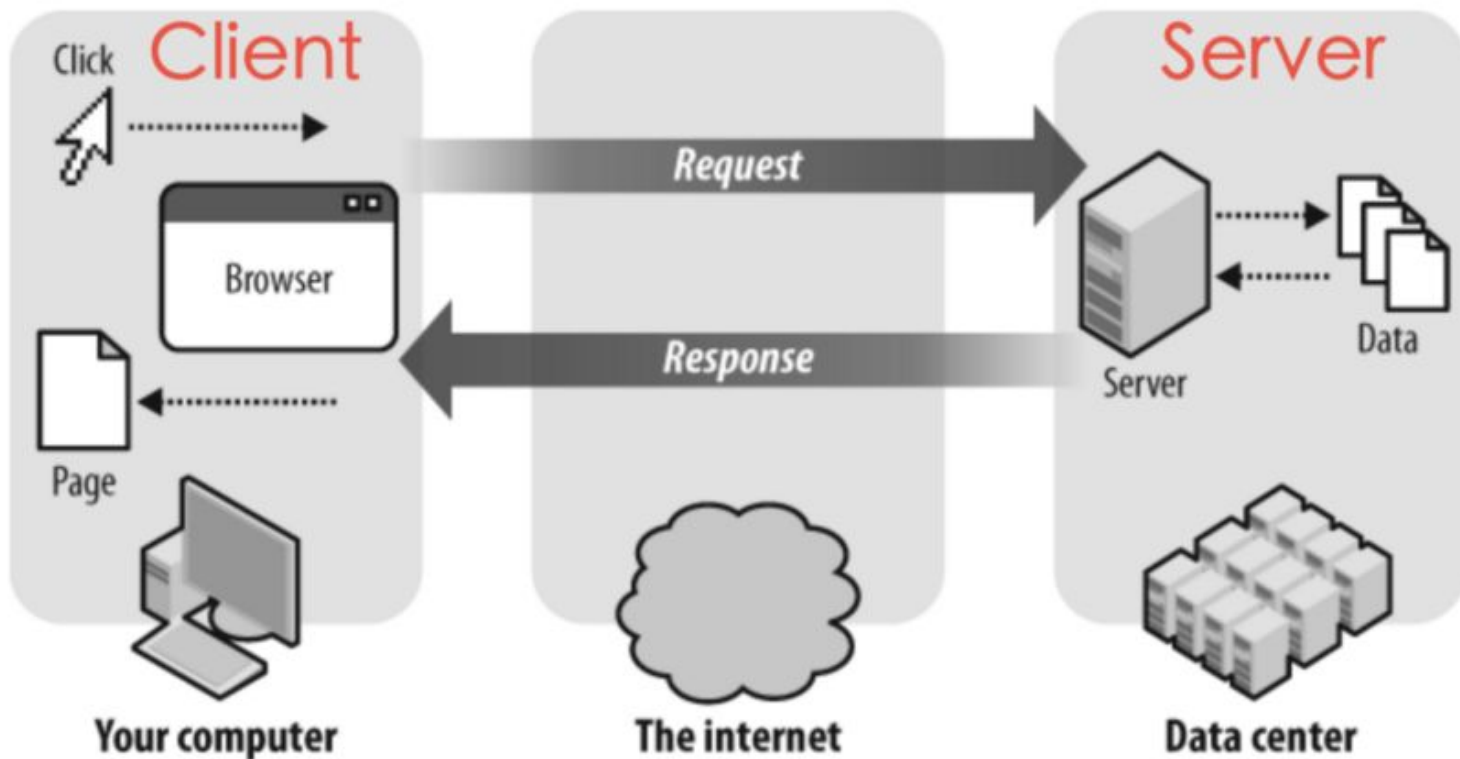
1. 建立一個叫做“all-day-long”的檔案
2. 寫入「歸剛欸」到“all-day-long”的檔案
3. 顯示“all-day-long”的文字內容



# Linux 背景知識 - 資料傳輸

OSI模型階層	OSI模型	TCP/IP	協定
第七層	應用層(Application Layer)	應用層(Application Layer)	HTTP、FTP、DNS
第六層	展現層(Presentation Layer)		
第五層	會談層(Session Layer)		
第四層	傳輸層(Transport Layer)	傳輸層(Transport Layer)	TCP、UDP
第三層	網路層(Network Layer)	網際網路層(Internet Layer)	IP、ICMP、IPX
第二層	資料連接層(Data Link Layer)	網路存取層(Network Access Layer)	乙太網、Wi-Fi、PPP
第一層	實體層(Physical Layer)		

# Linux 背景知識 - 資料傳輸



# Linux 背景知識 - 資料傳輸

<https://youtu.be/zvKadd9Cflc>



# Linux 背景知識 - 資料傳輸

溝通案例:開瀏覽器(客戶端)並輸入Youtube首頁網址(伺服器端)

1. 瀏覽器(客戶端)向YouTube的遠端主機(伺服器端)發出一個請求
2. 該請求透過網路被傳遞到YouTube的遠端主機(伺服器端)
3. 位於YouTube首頁的遠端主機(伺服器端)收到一個請求
4. 遠端主機(伺服器端)會根據請求內容, 找到對應的資料
5. 取出對應資的資料, 伺服器將其回傳至瀏覽器
6. 瀏覽器(客戶端)收到回傳內容, 開始解析資源, 顯示於瀏覽器上

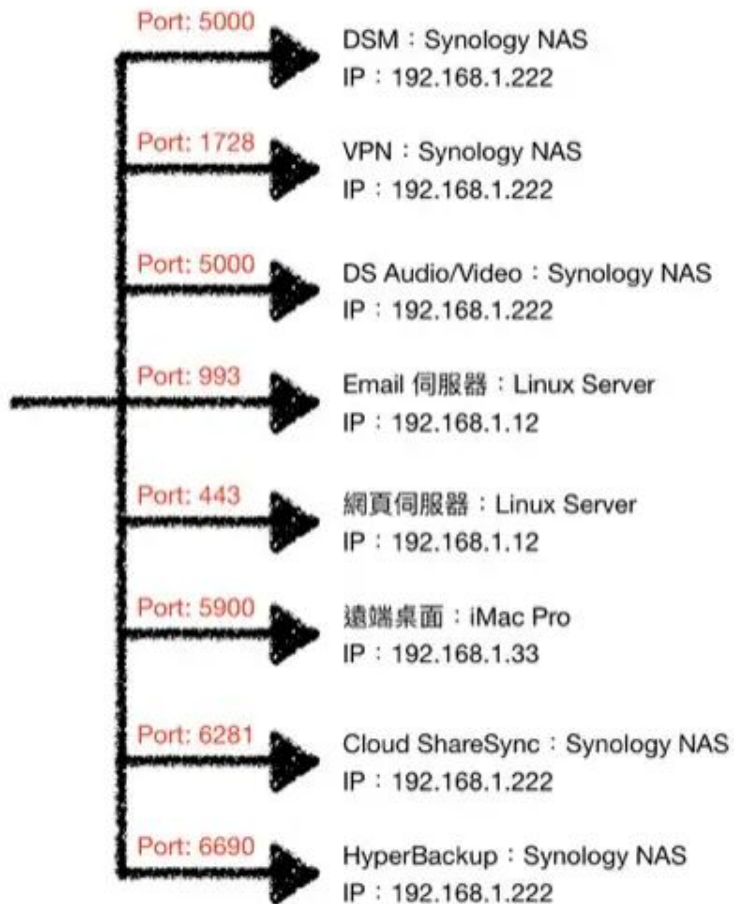
# Linux背景知識-資料傳輸

## Port

通訊埠號是 TCP/UDP 與上層通訊的通道，當 TCP/UDP 要傳送訊息時，會指定要由哪一個通訊埠號來接收。一些常用的服務會使用特定的埠號來等待要求的訊息。埠號是由 16 個位元所組成的號碼，0 ~ 255 為保留號碼，256 ~ 65535 則可自行設定。



對外固定 IP : 220.112.121.22



# Linux 背景知識 - 資料傳輸

Port	Service	Explanation
22	ssh	一種加密的網路傳輸協定，可在不安全的網路中及公安全的傳輸環境
23	telnet	一種很傳統的連線程式，可用來診斷各種伺服器與網路連線問題
80	http	超文本傳輸協定
20/21	ftp	一個傳遞客戶端與伺服器之間的命令(21, 命令通訊埠);另一個傳遞資料(20, 資料通訊埠)
443	https	超級文字傳輸安全協定
25	smtp	簡單郵件傳輸協定，是一個在網際網路上傳輸電子郵件的標準
53	DNS	網域名稱系統，將域名和IP位址相互對應的一個分散式資料庫，能夠使人更方便的存取網際網路



# Linux 背景知識 - 資料傳輸

## TCP/IP

TCP/IP提供了點對點的連結機制，將資料應該如何封裝、定址、傳輸、路由以及在目的地如何接收，都加以標準化。它將軟體通信過程抽象化為四個抽象層，採取協議堆疊的方式，分別實作出不同通信協定。協定套組下的各種協議，依其功能不同，被分別歸屬到這四個階層之中，常被視為是簡化的七層OSI模型。

# Linux 背景知識 - 資料傳輸

當瀏覽器輸入網址，發出一個 GET 請求時，過程中發生了哪些事情？

## 1. TCP三向交握

在瀏覽器送出請求之後，瀏覽器和伺服器就會開始初步溝通，確定雙方的溝通管道順暢，以便後續請求的執行

## 2. 瀏覽器請求、資料傳輸、渲染畫面

如同前面所提，當三向交握結束後，瀏覽器和伺服器便會開始執行請求、資料傳輸與渲染畫面的過程

## 3. TCP四次揮手，結束連線

在網頁成功渲染之後，瀏覽器就會和伺服器進行最後的溝通，確認傳輸過程已完成，準備結束連線

# Linux 指令 - 網路相關

→ route

→ netstat -r  
(mac)

```
yasmine@DESKTOP-1S8M9P8:/mnt/c/Users/mian1$ route
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref    Use Iface
192.168.92.0     0.0.0.0         255.255.255.0   U        256    0      0 eth2
192.168.92.1     0.0.0.0         255.255.255.255 U        256    0      0 eth2
192.168.92.255   0.0.0.0         255.255.255.255 U        256    0      0 eth2
224.0.0.0        0.0.0.0         240.0.0.0       U        256    0      0 eth2
255.255.255.255  0.0.0.0         255.255.255.255 U        256    0      0 eth2
192.168.30.0     0.0.0.0         255.255.255.0   U        256    0      0 eth3
192.168.30.1     0.0.0.0         255.255.255.255 U        256    0      0 eth3
```

→ Destination, Genmask: 完整的網域

→ Gateway: 這個網域是從哪個Gateway連接出去的, 0.0.0.0表示這個路由是從本機傳送, 有IP的話, 代表需要經過路由器才能傳出去

→ Iface: 網路卡介面

# Linux 指令 - 網路相關

## → route

add : 新增一條路由規則

del : 刪除一條路由規則

-net : 目的地址是一個網路

-host : 目的地址是一個主機

target : 目的網路或主機

netmask : 目的地址的網路掩碼

gw : 路由資料包通過的閘道器

dev : 為路由指定的網路介面

Destination	目標網段或者主機
Gateway	閘道器地址, " *" 表示目標是本主機所屬的網路, 不需要路由
Genmask	網路掩碼
Flags	標記。一些可能的標記如下:
	U — 路由是活動的
	H — 目標是一個主機
	G — 路由指向閘道器
	R — 恢復動態路由產生的表項
	D — 由路由的後臺程式動態地安裝
	M — 由路由的後臺程式修改
	! — 拒絕路由
Metric	路由距離, 到達指定網路所需的中轉數 (linux 核心中沒有使用)
Ref	路由項引用次數 (linux 核心中沒有使用)
Use	此路由項被路由軟體查詢的次數
Iface	該路由表項對應的輸出介面

# Linux 指令 - 網路相關

→ ping

```
C:\Users\mian1  
λ ping facebook.com
```

```
Ping facebook.com [2a03:2880:f117:83:face:b00c:0:25de] (使用 32 位元組的資料):
```

```
回覆自 2a03:2880:f117:83:face:b00c:0:25de: 時間=32ms
```

```
回覆自 2a03:2880:f117:83:face:b00c:0:25de: 時間=30ms
```

```
回覆自 2a03:2880:f117:83:face:b00c:0:25de: 時間=38ms
```

```
回覆自 2a03:2880:f117:83:face:b00c:0:25de: 時間=27ms
```

```
2a03:2880:f117:83:face:b00c:0:25de 的 Ping 統計資料:
```

```
封包: 已傳送 = 4, 已收到 = 4, 已遺失 = 0 (0% 遺失),
```

```
大約的來回時間 (毫秒):
```

```
最小值 = 27ms, 最大值 = 38ms, 平均 = 31ms
```

# Linux 指令 - 網路相關

## → ping

常用網路檢測工具，可藉由發送封包，檢查自己與特定設備之間的網路是否暢通  
同時測量網路連線的來回通訊延遲時間(round-trip delay time)。

- ◆ -n: 參數指定封包數→EX: `ping -n 10 blog.gtwang.org`
- ◆ -t: 持續監看網路是否正常→EX: `ping -t blog.gtwang.org`
- ◆ -4/-6: IPv4/IPv6
- ◆ -c: 指定Ping次數→EX: `ping -c 4 blog.gtwang.org`
- ◆ -s: 指定發送的數據字結數→EX: `ping -s 1024 facebook.com`
- ◆ -i: 指定收發資訊間隔時間→EX: `ping -i 0.4 facebook.com`

# Linux 指令 - 網路相關

影響網路速度的因素：

- 延遲 (Latency) : 封包從來源端至目的端中間所花的時間
- 頻寬 (Bandwidth) : 傳輸媒介的最大吞吐量

網路延遲 (Latency) 的組成元素:

- **propagation delay**: 封包在網路線上傳輸所花費的時間，與網路線上電子訊號跑的速度有關，這個時間就是距離除以訊號傳送速度所得到的數值。
- **transmission delay**: 網路卡將資料傳送到網路線上所花的時間，與網路設備的傳送速度有關。
- **nodal processing delay**: 路由器處理封包表頭、檢查位元資料錯誤與尋找配送路徑等所花費的時間。
- **queuing delay**: 路由器因為某些因素無法立刻將封包傳送到網路上，造成封包暫存在佇列中等待的時間。



## 補充小知識

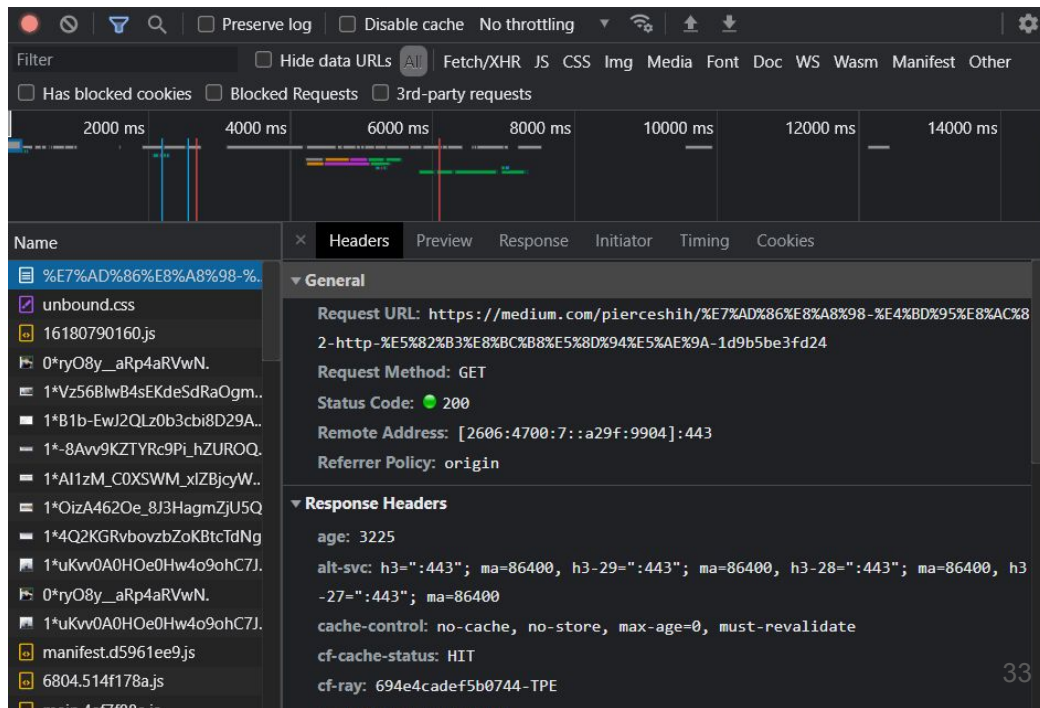


# Linux 背景知識 - 資料傳輸

## HTTP超文本傳輸協定(HyperText Transfer Protocol)

協定內容：

1. 標準化內容格式
2. 分為 header 跟 body
3. 用狀態碼標準化結果  
(Http status code)
4. 用動詞標準化請求方法  
(Http Request Method)



# Linux 背景知識 - 資料傳輸

## Http Request Method

序號	方法	描述
1	GET	請求指定的頁面信息，並返回實體主體。
2	HEAD	類似於get請求，只不過返回的響應中沒有具體的內容，用於獲取報頭
3	POST	向指定資源提交數據進行處理請求（例如提交表單或者上傳文件）。數據被包含在請求體中。POST請求可能會導致新的資源的建立和/或已有資源的修改。
4	PUT	從客戶端向服務器傳送的數據取代指定的文檔的內容。
5	DELETE	請求服務器刪除指定的頁面。
6	CONNECT	HTTP/1.1協議中預留給能夠將連接改為管道方式的代理服務器。
7	OPTIONS	允許客戶端查看服務器的性能。
8	TRACE	回顯服務器收到的請求，主要用於測試或診斷。

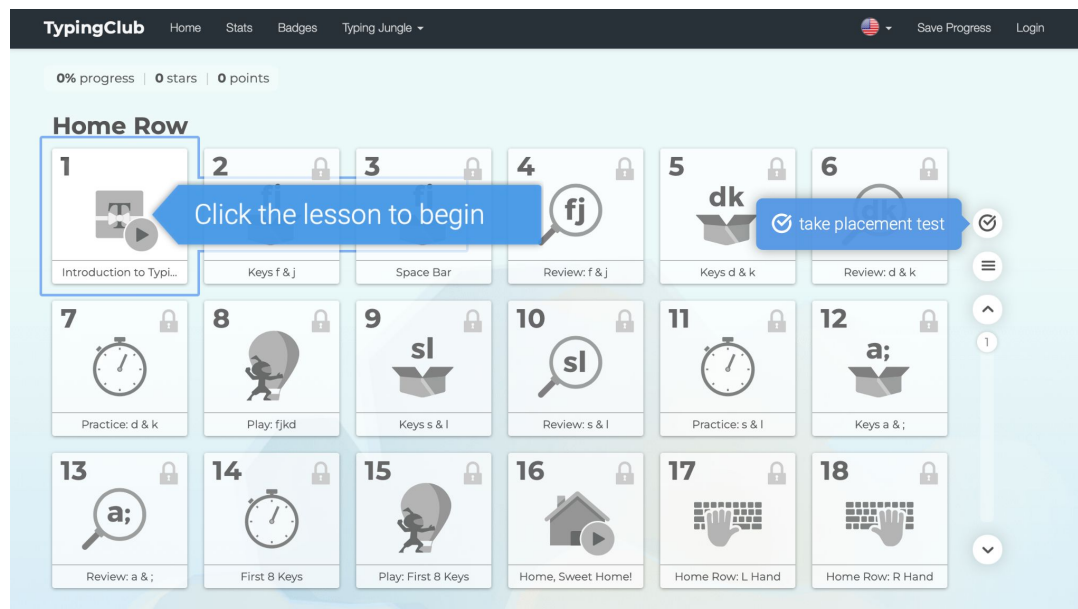
# Linux 背景知識 - 資料傳輸

1xx: 稍等	100Continue	Server成功接收、但Client還要進行一些處理
2xx: 成功	<b>200OK</b>	<b>成功</b>
	204No Content	成功但沒有回傳的內容(當你發出Delete的Request時)
3xx: 重新導向, 用戶端瀏覽器必須採取更多動作才能完成請求	301Moved Permanently	資源「永久」移到其他位置, 在下次發出Request時, 瀏覽器直接到新位置
	302Found(Moved Temporarily)	資源「暫時」移到其他位置
	<b>304Not Modified</b>	<b>東西跟之前長一樣, 可以快取拿就好</b>
4xx: Client端錯誤	400Bad Request	請求語法錯誤、或資源太大...等等
	401Unauthorized	未認證, 可能需要登入或Token
	403Forbidden	沒有權限
	<b>404Not Found</b>	<b>找不到資源</b>
5xx: Service端錯誤	<b>500Internal Server Error</b>	<b>伺服器出錯, 搶票時很可能發生</b>
	501Not Implemented	
	502Bad Gateway	通常是伺服器的某個服務沒有正確執行

手速不夠快？

# 英打練習1


## TypingClub: Learn Touch Typing Free



# 英打練習2

## LiveChat typing-speed-test

See our other products: [ChatBot](#) - Automate customer service with AI [HelpDesk](#) - Support customers by tickets [KnowledgeBase](#) - Guide and educate

 [Product](#) [Pricing](#) [Marketplace](#) [Community](#) [Customers](#) [Support](#) [Try LiveChat](#)

[Continue with Facebook](#)


TYPING SPEED TEST

# Test your typing skills

<b>60</b> seconds	<b>0</b> words/min	<b>0</b> chars/min	<b>0</b> % accuracy
----------------------	-----------------------	-----------------------	------------------------

[Start typing](#)

began typing of stand one ground

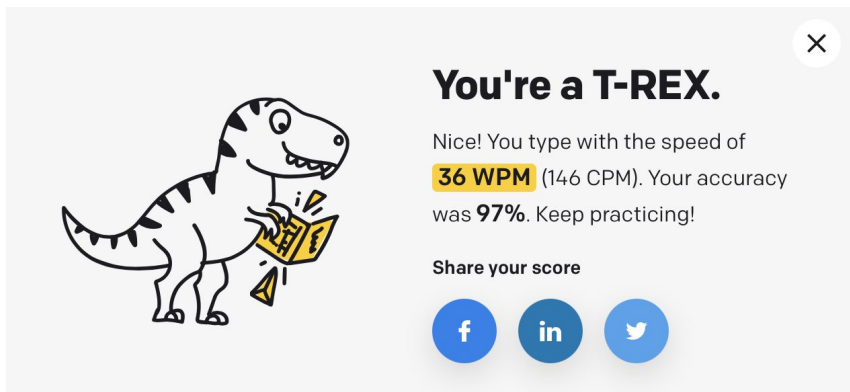


# 作業二

擇一執行：

1. [TypingClub](#) 完成前23關(兩頁)，截圖上傳 tronclass
2. [LiveChat](#) 達到 30 WPM 以上，截圖上傳 tronclass

期限:10/31 23:59



Try again

# 參考資料

- ★ <https://medium.com/pierceshih/%E7%AD%86%E8%A8%98-%E7%B6%B2%E8%B7%AF%E6%BA%9D%E9%80%9A%E6%A6%82%E5%BF%B5%E8%88%87%E9%81%8E%E7%A8%8B-3ece34903ae1>
- ★ <https://medium.com/pierceshih/%E7%AD%86%E8%A8%98-%E5%89%8D%E7%AB%AF%E8%A9%B2%E7%90%86%E8%A7%A3%E7%9A%84-tcp-ip-%E8%A7%80%E5%BF%B5-71ff859cb6eb>
- ★ <https://blog.gtwang.org/linux/windows-linux-ping-command-tutorial/>
- ★ <https://ithelp.ithome.com.tw/articles/10210473?sc=rss.iron>
- ★ <https://miahsuwork.medium.com/%E7%AC%AC%E5%85%AD%E9%80%B1-%E7%B6%B2%E8%B7%AF%E5%9F%BA%E7%A4%8E-http-request-response-7d7e0cb88ed8>



