

計算機概論A班 實習課

W4
助教:王常在

課程內容

- 正則表達式講解與實作
- Linux 文字處理工具-2
- 問答時間
- HW
 - 正規表達法練習題

正則表達式 Regular Expression

正則表達式是透過一些特殊符號來比對字串的方法，並可對符合比對條件的字串進行搜尋、截取、替代、轉換等等

正則表達式練習網站

<https://regex101.com/>

教學：<https://www.minwt.com/webdesign-dev/html/20352.html>

正則表達式(特殊符號)

特殊符號	代表意義
^	搜尋規則前的「開頭」
\$	搜尋規則後的「結尾」
.	任意一個字元
*	任意字元或任意字串, 單一字元或群組出現任意次數
.*	一起使用代表任意字串
+	單一字元或群組出現至少一次
?	單一字元或群組出現至少0次或1次
{n,m}	比對前一個字元至少n次, 至多m次, m、n皆為正整數 EX: 'a{3,6}' 為三到六個 'a'
[]	比對範圍內的字元或字串, EX: '[a-z]' 為所有英文小寫字母
[^]	比對不再指定範圍內的字元
[-]	範圍;如[A-Z]及A,B,C一直到Z都符合要求
\	特別序列的起始字元

正則表達式(特定字元)

正規表達法的特定字元	說明	等效的正規表達法	符合的例子
\d	數字	[0-9]	123
\D	非數字	[^0-9]	abc或ABC
\w	數字、字母、底線	[a-zA-Z0-9_]	yes_123或YES123_
\W	非\d	[^a-zA-z0-9_]	, 或、或-
\s	空白字元	[\r\t\n\f]	
\S	非空白字元	[^\r\t\n\f]	123或yes123_或,

正則表達式

→ .【點】

點可代替所有可能的字元(字母、數字或符號)。

EX: .GC → UGC、OGC、PGC、2GC或是nGC等...

→ ?【問號】

比對前一個字串或是不比對。

EX: facebo?k → facebk、facebok

→ *【星號】

比對前一個字串零次或是多次。

EX: sky*blue → skblue(y出現0次)、skyblue(y出現1次)、skyyyblue(y出現多次)

→ -【破折號】

EX: product[A-K] → productA、productB、productC、productD...productK

正則表達式講解

→ +【加號】

跟星號類似，差別在於至少要與前一個字比對一次或以上。

EX: sky+blue → skyblue(y出現1次)、skyyyblue(y出現多次)

→ |【直線】

或者。

EX: 想找到Facebook、Instagram、Wordpress、Google相關的文章，可以使用

Facebook | Instagram | Wordpress | Google

→ ^【插入符號】和\$【錢字符號】

^插入符號是比對前開頭，\$錢字符號則是比對結尾。

EX: ^eat → eat、eaten

EX: eat\$ → creat、peat、leat

正則表達式講解

→ \【反斜線】

將任何特殊字元，恢復成一般字元。

EX: transbiz\.com → transbiz.com

→ ()【括號】

把想找的相關字詞放入括號內，可依照括號裡的字元排序找到可能的結果。

EX: (sym) → sympathy、symbol、assym等

→ []【中括號】

任意比對字串內的每個項目。

EX: product[DEFG] → productD、productE、productF、productG

正則表達式練習題

Q1: 身分證字號: 一個英文大寫字元搭配9個數字字元

A: $^[A-Z]\{9\}$

Q2: 西元出身年月日: 以oooo/oo/oo表示

A: $^\{4\}\{2\}\{2\}$

小試身手

Q1 : price: \$25.11 inc.VAT (取得價錢)

A : \d+\.\d+

Q2 : 3.7 out of 5 stars (取得評價)

A : \d\.\d

Q3 : date: 2000-08-18 (取得日期)

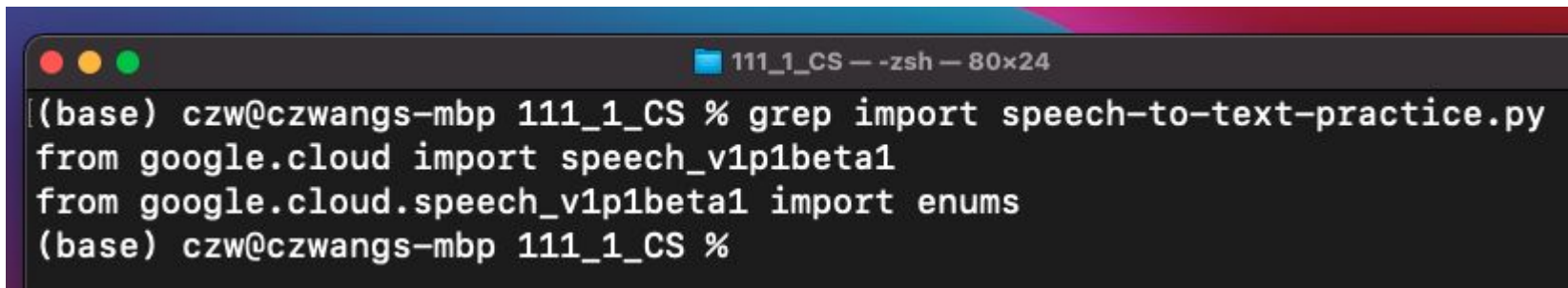
A : \d{4}-\d{2}-\d{2}

Linux 文字處理工具-2

grep

可從資料或檔案中，使用關鍵字或正規表達法(Regex)找出想要的內容

```
grep [option] filename
```

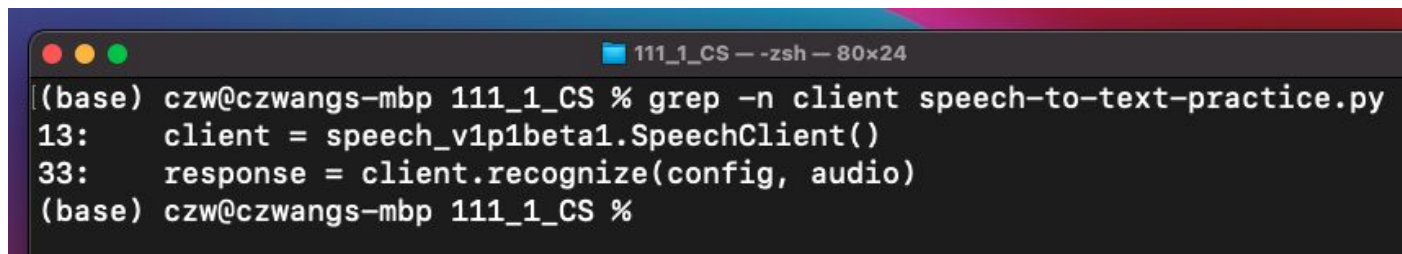
A terminal window titled '111_1_CS — -zsh — 80x24' showing a user running a grep command. The command is 'grep import speech-to-text-practice.py'. The output shows two lines of Python code: 'from google.cloud import speech_v1p1beta1' and 'from google.cloud.speech_v1p1beta1 import enums'. The prompt '(base) czw@czwangs-mbp 111_1_CS %' is visible at the end of the command line and the output lines.

```
(base) czw@czwangs-mbp 111_1_CS % grep import speech-to-text-practice.py
from google.cloud import speech_v1p1beta1
from google.cloud.speech_v1p1beta1 import enums
(base) czw@czwangs-mbp 111_1_CS %
```

grep 參數

→ -i: 忽略大小寫

→ -n: 顯示匹配行及行號

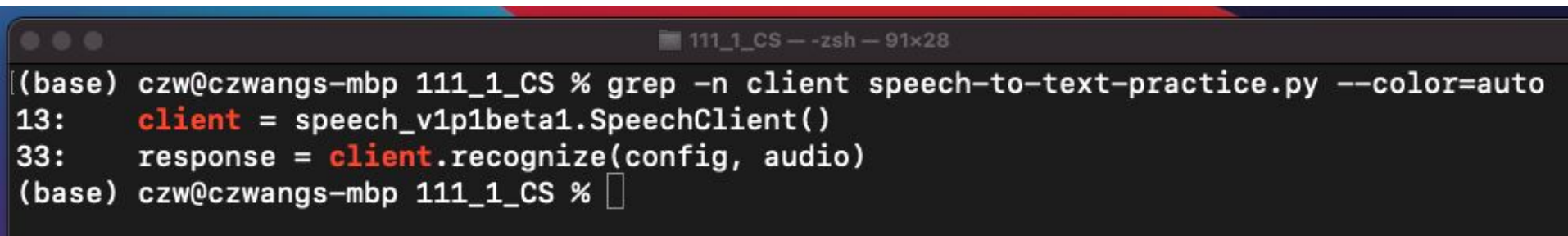
A terminal window titled '111_1_CS - zsh - 80x24' showing a user running the command 'grep -n client speech-to-text-practice.py'. The output shows two lines: '13: client = speech_v1p1beta1.SpeechClient()' and '33: response = client.recognize(config, audio)'. The prompt is '(base) czw@czwangs-mbp 111_1_CS %'.

→ -r: 顯示匹配文字的檔案路徑

→ -c: 只輸出匹配行的計數

grep 參數

- -v: 只列出不符合的內容
- --color = never | always | auto: 顏色標示



A terminal window titled "111_1_CS --zsh-- 91x28" showing a command prompt. The user enters the command `grep -n client speech-to-text-practice.py --color=auto`. The output shows two lines from the file `speech-to-text-practice.py`, with the word `client` highlighted in red in both lines. The first line is `13: client = speech_v1p1beta1.SpeechClient()` and the second line is `33: response = client.recognize(config, audio)`. The prompt then returns to the user.

```
(base) czw@czwangs-mbp 111_1_CS % grep -n client speech-to-text-practice.py --color=auto
13:  client = speech_v1p1beta1.SpeechClient()
33:  response = client.recognize(config, audio)
(base) czw@czwangs-mbp 111_1_CS %
```

grep + 正規表達法

→ 開頭結尾

- ◆ a開頭 → `ls | grep "^a"`
- ◆ a或b結尾 → `ls | grep "[ab]$"`

→ 出現次數

- ◆ a開頭, 出現0次以上 → `ls | grep "^a*"`
- ◆ a開頭, 出現零次或一次 → `ls | grep "^a?"`
- ◆ a開頭, 出現一次以上 → `ls | grep "^a+"`

→ 多種組合

- ◆ 含有ab或cd → `ls | grep "ab|cd"`
- ◆ 含有ab開頭或cd結尾 → `ls | grep "^ab|cd$"`

WC 文字處理工具

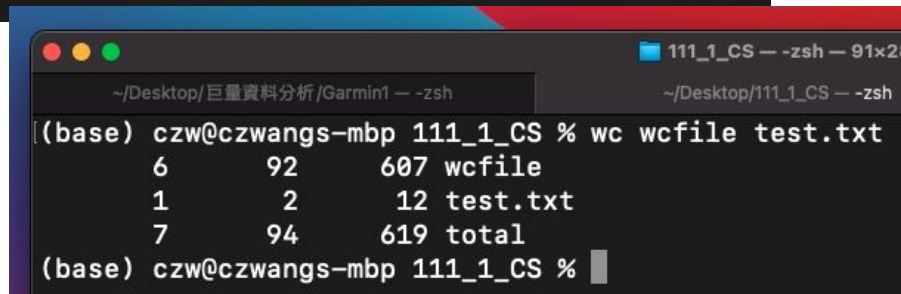
計算指定檔案內容的換行數、字數與位元組數

`wc [option] filename`

- `-l` → 只計算換行數
- `-w` → 只計算字數
- `-c` → 只計算位元組數
- `-m` → 只計算字元數
- `-L` → 計算最常行的長度



```
(base) czw@czwangs-mbp 111_1_CS % wc wcfile
6      92      607 wcfile
(base) czw@czwangs-mbp 111_1_CS %
```



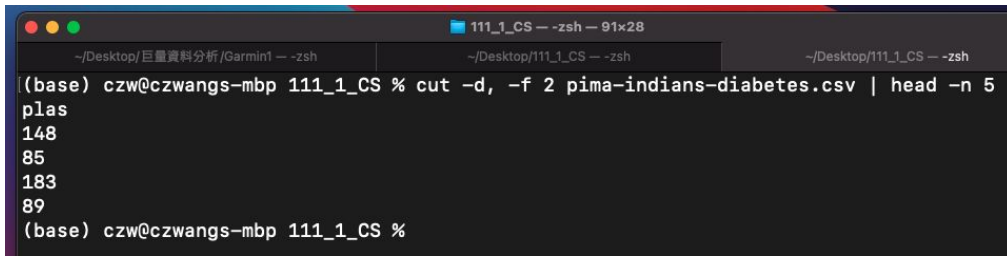
```
(base) czw@czwangs-mbp 111_1_CS % wc wcfile test.txt
6      92      607 wcfile
1       2       12 test.txt
7      94      619 total
(base) czw@czwangs-mbp 111_1_CS %
```

Cut 文字處理工具

逐行擷取部分字元或欄位資料

`cut [option] filename`

- -b: 輸出的指定範圍以bytes作為單位
- -c: 輸出的指定範圍以字元數量作為單位
- -d: 指定分隔字元, default為tab作為分隔
- -f: 輸出的指定範圍(每筆data的第幾column作為區分)
- -s: 若該行無分隔字元則不顯示



```
(base) czw@czwangs-mbp 111_1_CS % cut -d, -f 2 pima-indians-diabetes.csv | head -n 5
plas
148
85
183
89
(base) czw@czwangs-mbp 111_1_CS %
```

#表示篩選第二欄且僅寫顯示前五筆

Paste 文字處理工具

將每個文件以列對列的方式進行合併

paste [option] filename

→ -s: 合併為多行呈現

```
(base) czw@czwangs-mbp 111_1_CS % cat 123.txt
1
2
3%
(base) czw@czwangs-mbp 111_1_CS % cat name.txt
Michael
David
Amy%
(base) czw@czwangs-mbp 111_1_CS % cat age.txt
27
34
23%
(base) czw@czwangs-mbp 111_1_CS %
```

```
(base) czw@czwangs-mbp 111_1_CS % paste 123.txt name.txt age.txt
1      Michael 27
2      David   34
3      Amy     23
(base) czw@czwangs-mbp 111_1_CS % paste -s 123.txt name.txt age.txt
1      2      3Michael      David      Amy27      34      23%
(base) czw@czwangs-mbp 111_1_CS %
```

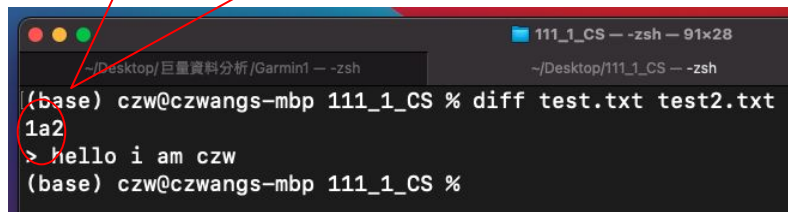
Diff 文字處理工具

比較文件的內容，特別是兩版本不同的同份文件

```
diff [option] filename1 filename2
```

- -y → 以並列方式顯示文件的異同之處
- -W → 使用-y參數時，指定欄寬
- -C → 前後輸出格式
- -u → 統一格式輸出

a -add | c -change | d -delete



```
(base) czw@czwangs-mbp 111_1_CS % diff test.txt test2.txt
1a2
> Hello i am czw
(base) czw@czwangs-mbp 111_1_CS %
```

Lines 1 in test.txt need to add line 2 to match file2.

文字處理工具

- >: 覆蓋原有檔案
- >>: 追加內容, 不覆蓋繼續寫

```
mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ cat test1.txt
1
2
3

mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ echo 'abc'>test1.txt

mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ cat test1.txt
abc
```

echo相當python的print, 可以打印字串

```
mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ echo 'def'>>test1.txt

mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ cat test1.txt
abc
def
```

Sort 文字處理工具

處理文字的排序問題

`sort [option] filename`

- `-f`: 忽略大小寫
- `-u`: 去除重複資料
- `-r`: 反向排序
- `-t`: 指定欄位的分隔字元(default=blank or tab)
- `-k`: 指定欄位的編號
- `-n`: 依照實際數值的大小排序
- `-h`: 對有單位的數值排序
- `-M` → 依照月份排序

LC_ALL=C (調整系統語言)
在非英文語系的系統上操作
月份資料排序時, 需先將語言
設定為英文, 方可操作。

```
mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ cat test3.txt
Feb
Aug
May
Sep
Jan

mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ sort test3.txt
Aug
Feb
Jan
May
Sep

mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ LC_ALL=C sort -M test3.txt
Jan
Feb
May
Aug
Sep
```

Uniq 文字處理工具

將連續重複文字刪除

uniq [option] filename

- -c: 計算文字行重複次數
- -d: 將重複行刪掉
- -u: 只輸出沒有重複的文字行
- -f: 指定要跳過的欄位
- -s: 跳過每一行開頭幾個字元
- -w: 只比較每一行開頭幾個字元
- -i: 忽略大小寫

```
111_1_CS -- zsh -- 91x28
~/Desktop/巨量資料分析/Garmin1 -- zsh
~/Desktop/111_1_CS -- zsh

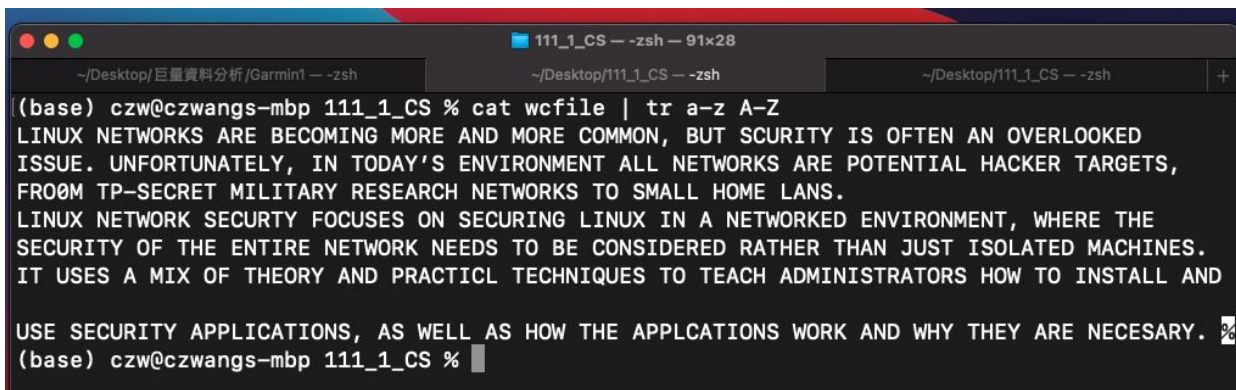
(base) czw@czwangs-mbp 111_1_CS % cat uniqfile
test 30
test 30
test 30
Hello 95
Hello 95
Hello 95
Hello 95
Linux 85
Linux 85
(base) czw@czwangs-mbp 111_1_CS % uniq uniqfile
test 30
Hello 95
Linux 85
(base) czw@czwangs-mbp 111_1_CS % uniq -c uniqfile
  3 test 30
  4 Hello 95
  2 Linux 85
(base) czw@czwangs-mbp 111_1_CS % sort uniqfile| uniq -d
Hello 95
Linux 85
test 30
(base) czw@czwangs-mbp 111_1_CS %
```

Tr 文字處理工具

字串替換或刪除

`tr [option] set1 set2`

- `-c` → 用set1中字符集的補集替換此字符集
- `-d` → 刪除檔案中所有在set1中出現的字元
- `-s` → 刪除檔案中重複且set1中出現的字元，只保留一個



```
(base) czw@czwangs-mbp 111_1_CS % cat wcfile | tr a-z A-Z
LINUX NETWORKS ARE BECOMING MORE AND MORE COMMON, BUT SCURITY IS OFTEN AN OVERLOOKED
ISSUE. UNFORTUNATELY, IN TODAY'S ENVIRONMENT ALL NETWORKS ARE POTENTIAL HACKER TARGETS,
FROM TP-SECRET MILITARY RESEARCH NETWORKS TO SMALL HOME LANS.
LINUX NETWORK SECURITY FOCUSES ON SECURING LINUX IN A NETWORKED ENVIRONMENT, WHERE THE
SECURITY OF THE ENTIRE NETWORK NEEDS TO BE CONSIDERED RATHER THAN JUST ISOLATED MACHINES.
IT USES A MIX OF THEORY AND PRACTICE TECHNIQUES TO TEACH ADMINISTRATORS HOW TO INSTALL AND
USE SECURITY APPLICATIONS, AS WELL AS HOW THE APPLICATIONS WORK AND WHY THEY ARE NECESSARY.
(base) czw@czwangs-mbp 111_1_CS %
```


Join 文字處理工具

- join: 將兩個文件中, 指定欄位內容相同的行連接起來

join [option] filename1 filename2

- -1: 連接filename1指定的欄位
- -2: 連接filename2指定的欄位
- -t: 使用欄位的分隔符號
- -i: 忽略大小寫
- -o: 按指定的格式顯示結果
- -a: 除顯示結果, 原檔案的其他行也顯示

```
mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ cat test1.txt
1 a
2 b
3 c

mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ cat test2.txt
1 aaa
2 bbb
3 ccc

mian1@DESKTOP-1S8M9P8 MINGW64 ~/desktop
$ join test1.txt test2.txt
1 a aaa
2 b bbb
3 c ccc
```

作業三

正規表達練習題*5

- 繳交word檔至tronclass作業區
- 期限:11/14 23:59前

