

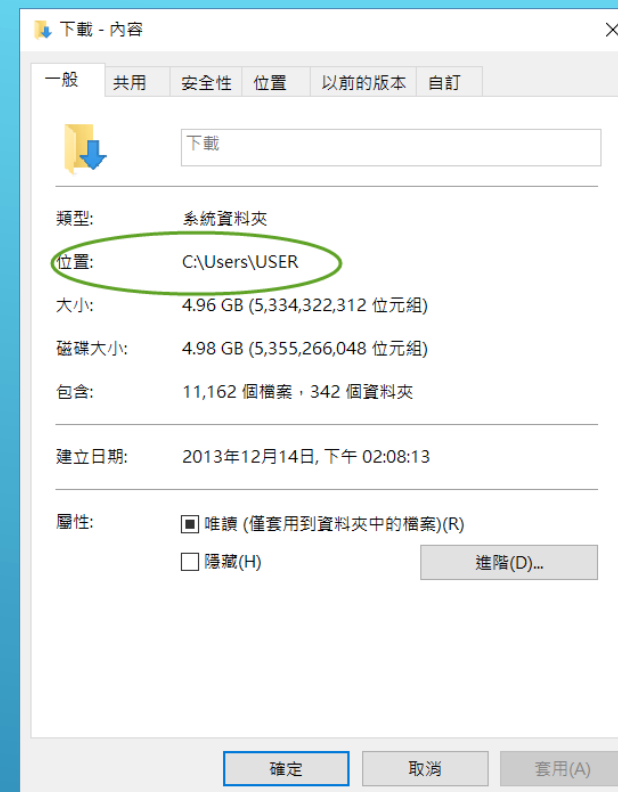
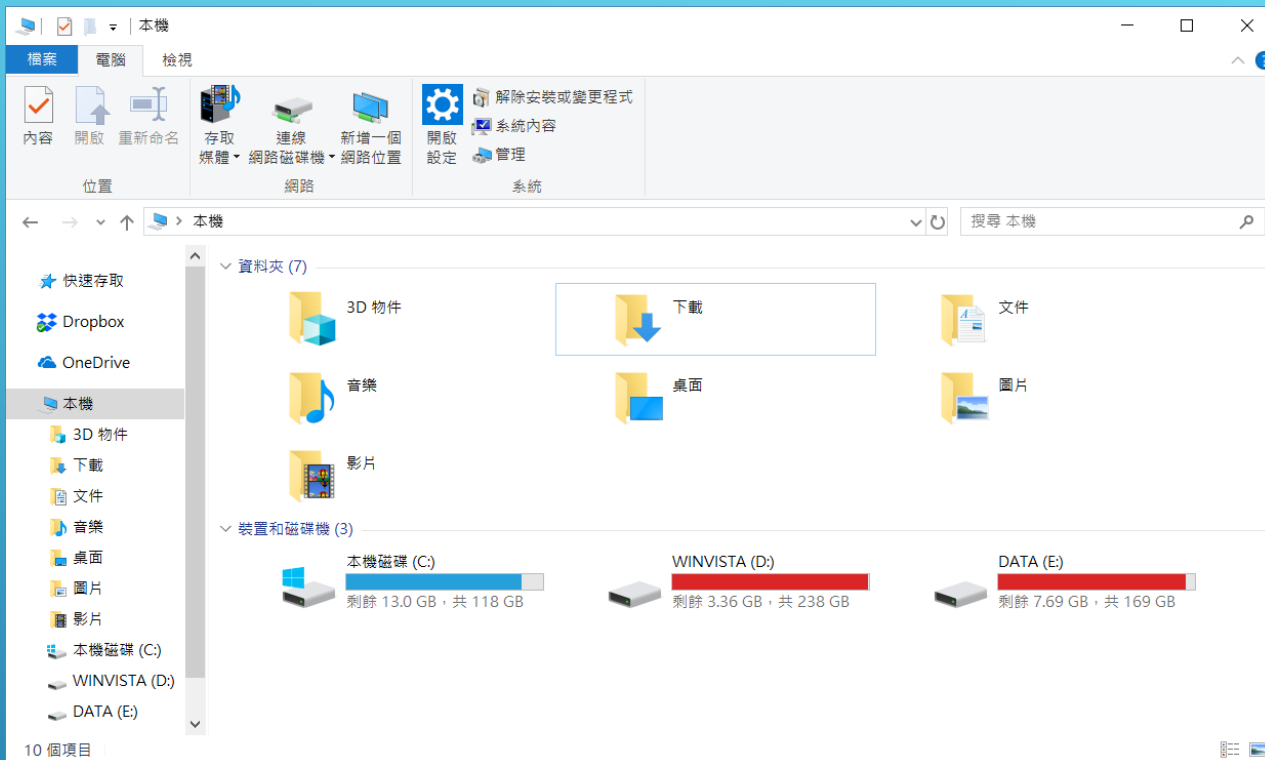
4

檔案處理與操作

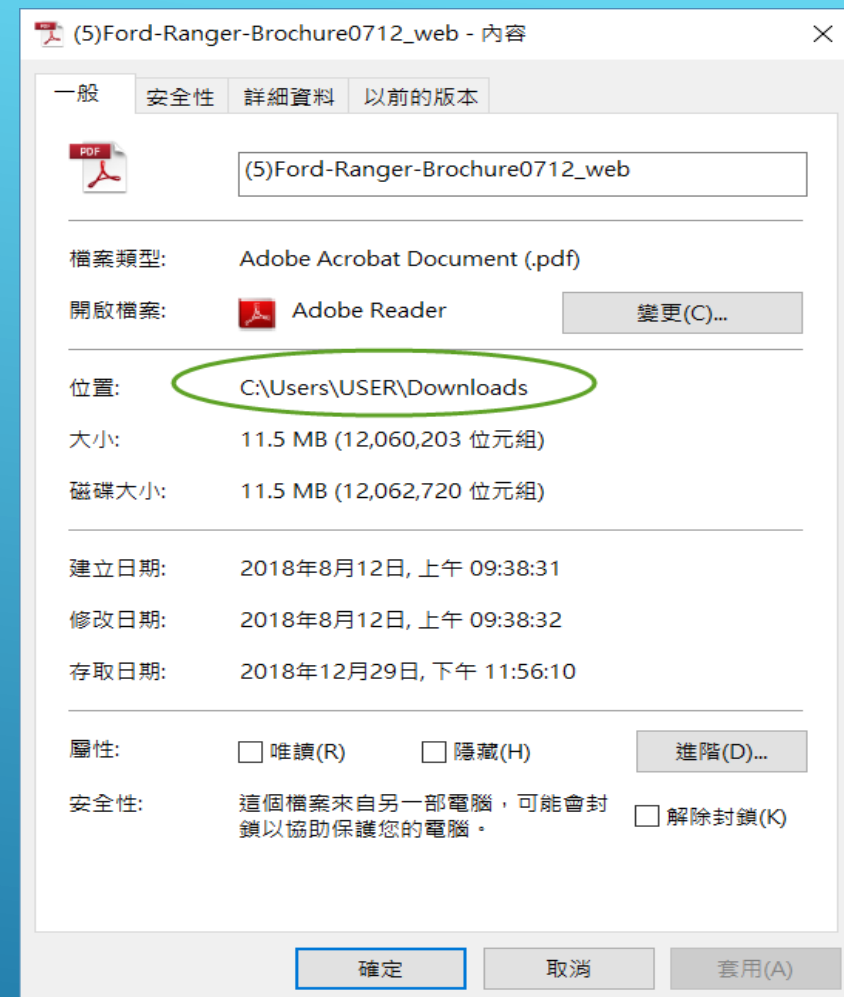
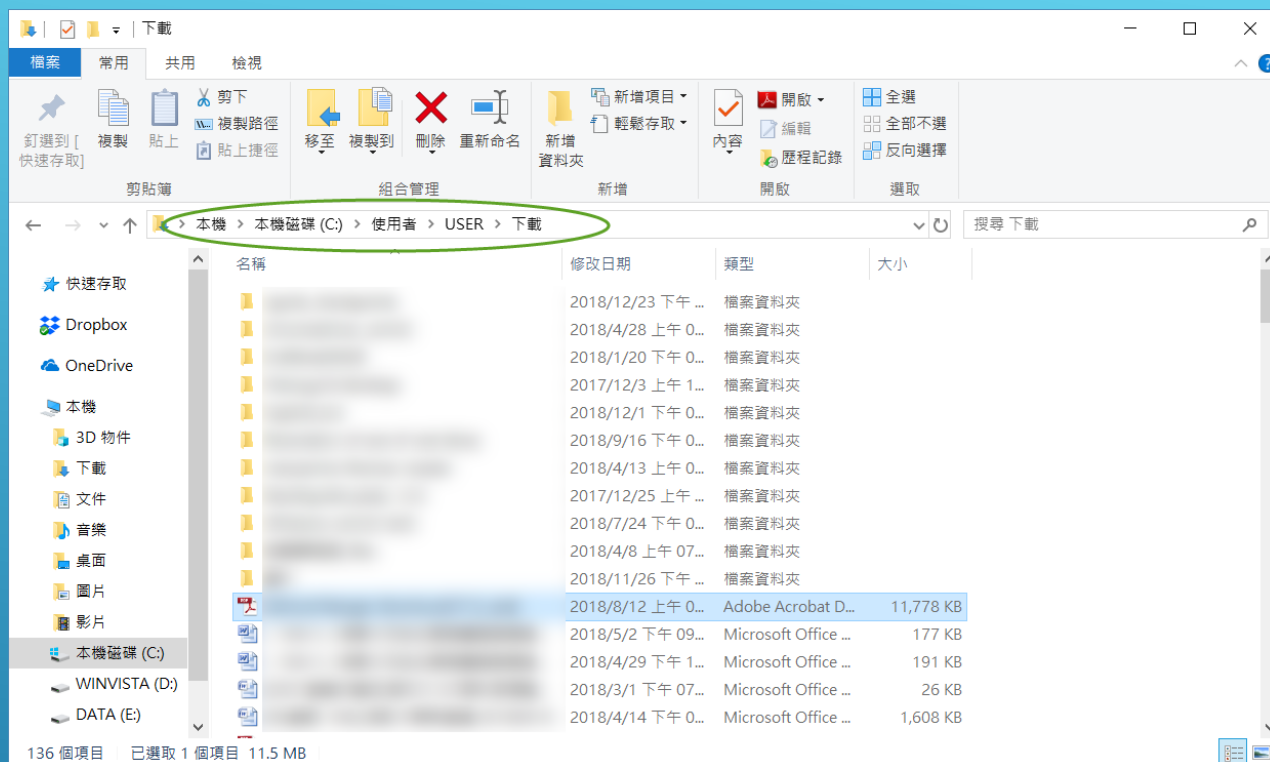
資訊社會必修的
12堂Python通識課

本堂課重點

- ▶ 路徑、資料夾、和檔案
- ▶ 寫入資料檔案
- ▶ 讀取資料檔案
- ▶ 例外處理
- ▶ 自製影像瀏覽網頁



WINDOWS作業系統檔案總管



WINDOWS作業系統檔案總管

```
import sys
print(sys.version_info)
print("---")
print(sys.platform)
print("---")
print(sys.argv)
print("---")
print(sys.path)
```

使用SYS模組檢查系統參數

```
import sys  
print(sys.argv)
```

存成 argtest.py

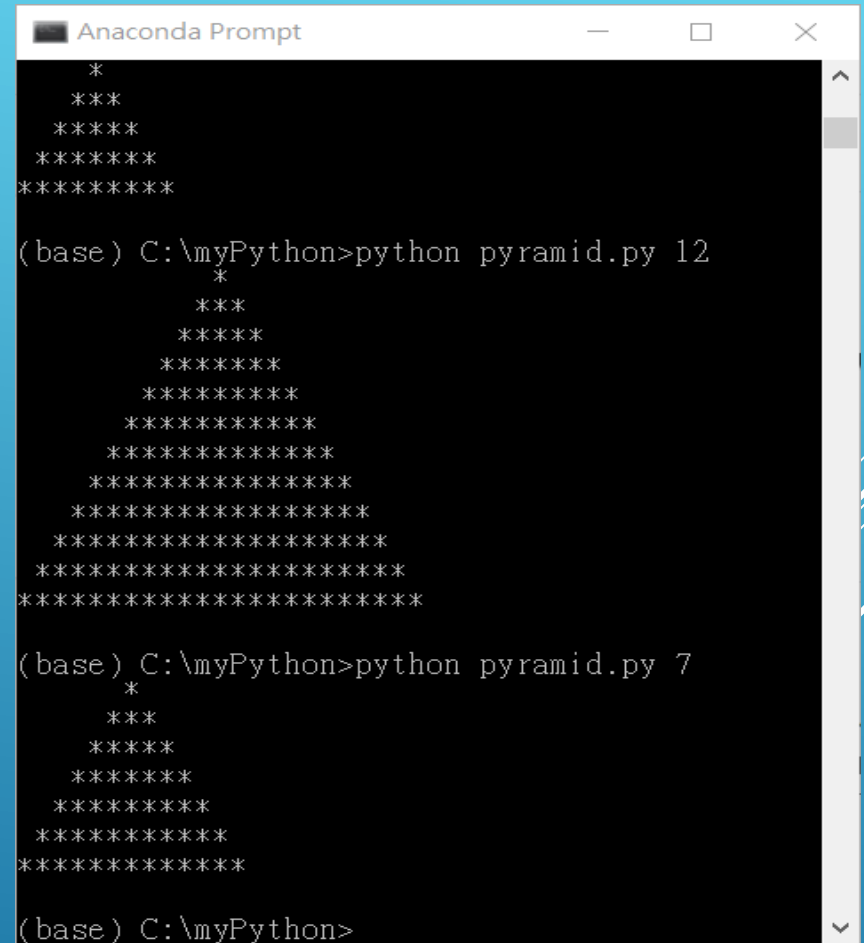
```
(base) C:\myPython>python argtest.py a1 a2 a3  
['argtest.py', 'a1', 'a2', 'a3']
```

在Anaconda Prompt中的
執行結果

在程式中存取命令列參數

```
import sys
if len(sys.argv)>1:
    n = int(sys.argv[1])
else:
    print("You need to specify a valid number!")
    exit(1)

lst = [i*2+1 for i in range(n)]
for i in lst:
    print(" "*((lst[-1]-i)//2), end="")
    for k in range(i):
        print("*", end="")
    print()
```



```
Anaconda Prompt

*
***
*****
*****
*****

(base) C:\myPython>python pyramid.py 12
*
***
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****

(base) C:\myPython>python pyramid.py 7
*
***
*****
*****
*****
*****
*****

(base) C:\myPython>
```

星號金字塔程式

```
1: import os
2: items = os.listdir()
3: print(os.path.exists('myprime.py'))
4: for item in items:
5:     print(os.path.abspath(item))
```

True

/Users/minhuang/Dropbox/books_project/2018books/Pythonbasics/codes/twstock.ipynb
/Users/minhuang/Dropbox/books_project/2018books/Pythonbasics/codes/ch03.ipynb
/Users/minhuang/Dropbox/books_project/2018books/Pythonbasics/codes/Untitled1.ipynb
/Users/minhuang/Dropbox/books_project/2018books/Pythonbasics/codes/Untitled.ipynb

找出檔案絕對路徑的程式碼 (MACOS)

內建函數名稱	說明
<code>os.path.abspath()</code>	傳回檔案的絕對路徑
<code>os.path.basename()</code>	傳回不包含路徑的檔案名稱
<code>os.path.dirname()</code>	傳回不包含檔案名稱的路徑
<code>os.path.exists()</code>	檢查該檔案是否存在
<code>os.path.getatime()</code>	傳回上次存取此檔案的時間，以Unix epoch時間為計算單位
<code>os.path.getmtime()</code>	傳回上次修改此檔案的時間，以Unix epoch時間為計算單位
<code>os.path.getctime()</code>	傳回建立此檔案的時間，以Unix epoch時間為計算單位
<code>os.path.getsize()</code>	以位元組為單位傳回檔案的大小
<code>os.path.isabs()</code>	檢查是否為絕對路徑
<code>os.path.isfile()</code>	檢查是否為檔案
<code>os.path.isdir()</code>	檢查是否為目錄（資料夾）
<code>os.path.join()</code>	進行路徑串接
<code>os.path.split()</code>	把檔案名稱和路徑名稱拆解成串列項目
<code>os.path.splitdrive()</code>	把路徑和磁碟機拆解成串列項目
<code>os.path.splitext()</code>	把副檔名拆解成串列項目

```
import shutil
import os
fullpath = os.path.abspath('myprime.py')
path, filename = os.path.split(fullpath)
filename, extname = os.path.splitext(filename)
if not os.path.exists("test-dir"):
    os.mkdir("test-dir")
targetfullpath = os.path.join(path, os.path.join("test-dir", "00"+extname))
shutil.copy(fullpath, targetfullpath)
```

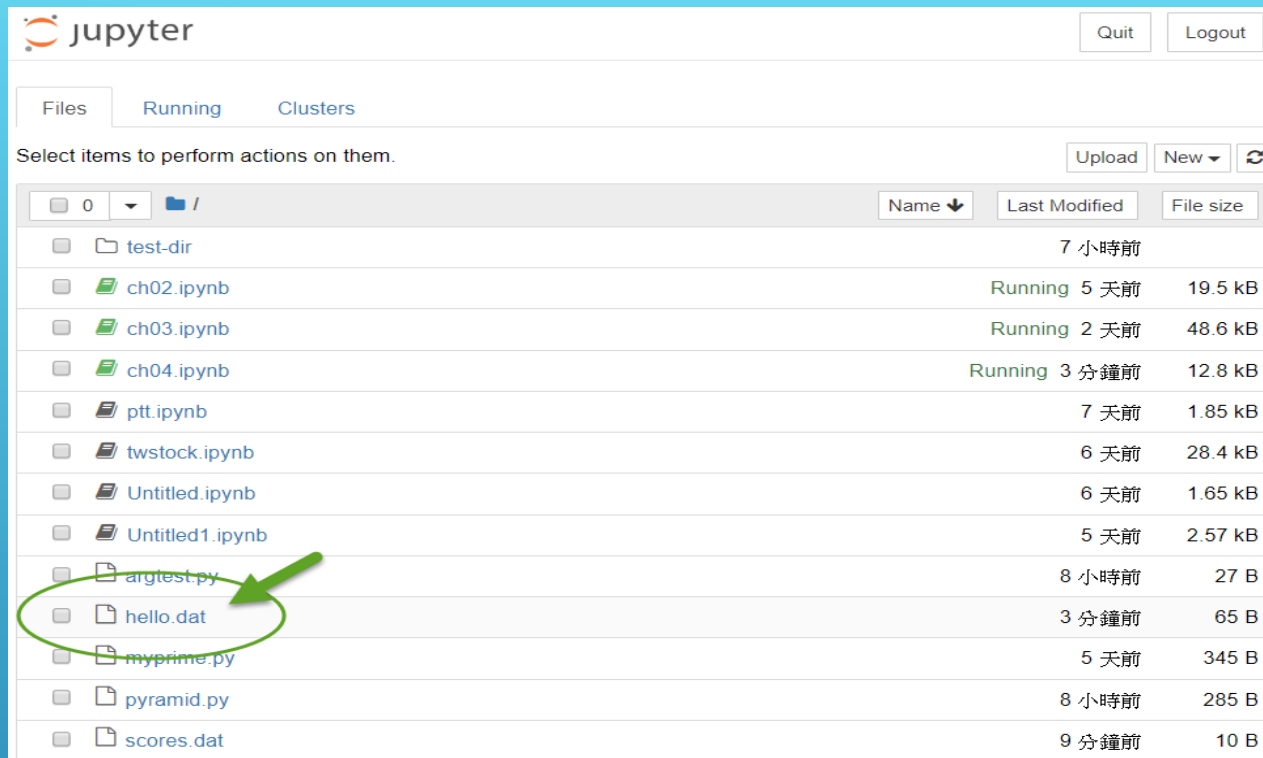
更名並複製檔案

檔案物件 = open("檔案名稱", "開檔模式")

內建函數名稱	說明
r	讀取模式。
w	全新寫入模式，如果檔案不存在的話就建立一個新的檔案，如果檔案存在的話，會刪除之前的資料。
a	如果檔案不存在的話就建立一個新的檔案，如果檔案存在的話，會以附加的方式寫入資料檔案。
r+	開啟為讀寫模式，如果檔案不存在的話，則會出現錯誤。
w+	開啟為讀寫模式，如果檔案不存在的話，就新增該檔案。

開啟檔案

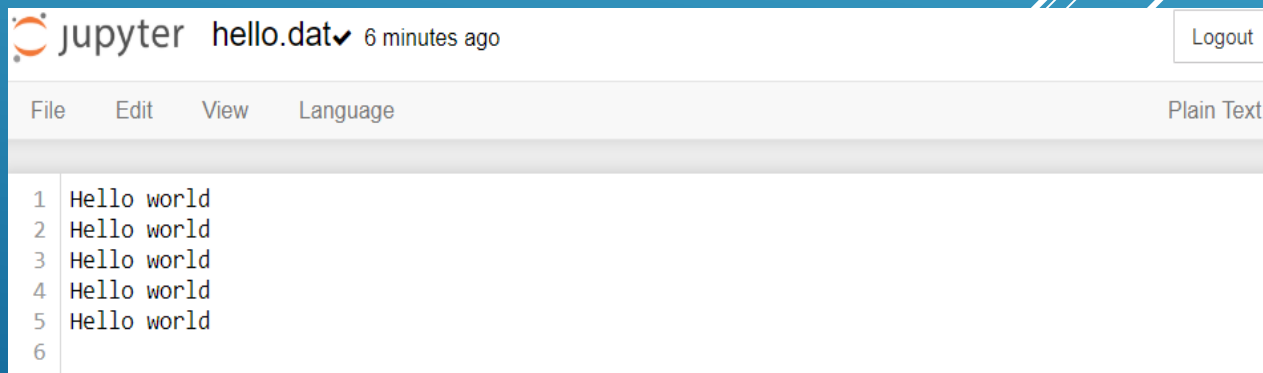
```
f = open("hello.dat", "w")
for i in range(5):
    f.write("Hello world\n")
f.close()
```



The JupyterLab interface shows a file browser with a list of files and folders. The file 'hello.dat' is highlighted with a green circle and a green arrow pointing to it. The table below shows the details of the files:

Name	Last Modified	File size
test-dir	7 小時前	
ch02.ipynb	Running 5 天前	19.5 kB
ch03.ipynb	Running 2 天前	48.6 kB
ch04.ipynb	Running 3 分鐘前	12.8 kB
ptt.ipynb	7 天前	1.85 kB
twstock.ipynb	6 天前	28.4 kB
Untitled.ipynb	6 天前	1.65 kB
Untitled1.ipynb	5 天前	2.57 kB
argtest.py	8 小時前	27 B
hello.dat	3 分鐘前	65 B
myprime.py	5 天前	345 B
pyramid.py	8 小時前	285 B
scores.dat	9 分鐘前	10 B

開啟檔案並寫入資料



The JupyterLab interface shows the code editor for the file 'hello.dat'. The text 'Hello world' is written five times, one on each line. The line numbers 1 through 6 are visible on the left side of the editor.

```
1 Hello world
2 Hello world
3 Hello world
4 Hello world
5 Hello world
6
```

```
score = int(input("請輸入國文成績 ( -1結束 ) : "))  
fp = open("scores.dat", "w")  
while score != -1:  
    fp.write("{} ".format(score))  
    score = int(input("請輸入國文成績 ( -1結束 ) : "))  
fp.close()
```

56 24 68 89 90

寫入成績資料的方法

```
username = "何敏煌"
hello = ["Hello", "BONJOUR", "HOLA", "こんにちは", "안녕하세요", "你好"]
f = open("hello.html", "w", encoding='utf-8')
f.write("<html>")
f.write("<head><meta charset='utf-8' /></head>")
f.write("<body>")
for index, msg in enumerate(hello):
    f.write("<h{0}>{1}, {2}</h{0}>".format(len(hello)-index, msg,
username))
f.write("</body></html>")
f.close()
```

利用PYTHON程式建立HTML檔案

```
<html><head><meta charset='utf-8' /></head><body><h6>Hello, 何 敏 煌</h6><h5>BONJOUR, 何敏煌</h5><h4>HOLA, 何 敏 煌</h4><h3>こんにちは, 何敏煌</h3><h2>안녕하세요, 何敏煌</h2><h1>你好, 何敏煌</h1></body></html>
```

Hello, 何敏煌

BONJOUR, 何敏煌

HOLA, 何敏煌

こんにちは, 何敏煌

안녕하세요, 何敏煌

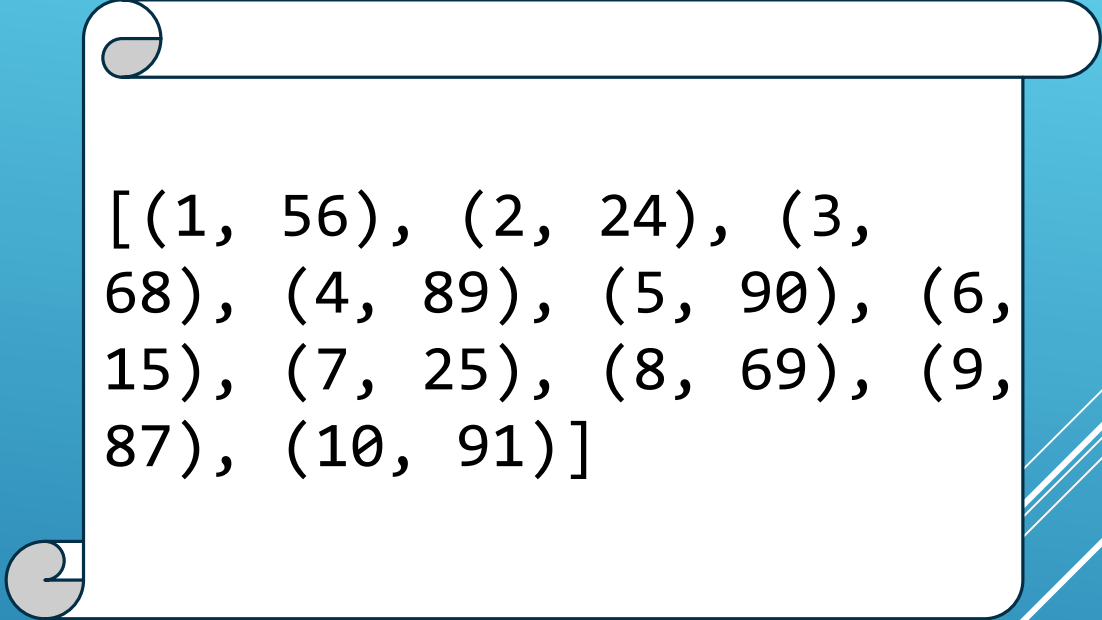
你好, 何敏煌

執行結果

```
scores = list()
for i in range(1,11):
    score = int(input("請輸入{}號同學的國文成績：".format(i)))
    scores.append((i, score))
fp = open("scores.dat", "w")
fp.write(str(scores))
fp.close()
```

輸入成績並儲存成串列型式文字檔

請輸入1號同學的國文成績：56
請輸入2號同學的國文成績：24
請輸入3號同學的國文成績：68
請輸入4號同學的國文成績：89
請輸入5號同學的國文成績：90
請輸入6號同學的國文成績：15
請輸入7號同學的國文成績：25
請輸入8號同學的國文成績：69
請輸入9號同學的國文成績：87
請輸入10號同學的國文成績：91



$[(1, 56), (2, 24), (3, 68), (4, 89), (5, 90), (6, 15), (7, 25), (8, 69), (9, 87), (10, 91)]$

執行過程與結果

```
scores = list()
for i in range(1,6):
    chi = int(input("請輸入{}號同學的國文成績：".format(i)))
    eng = int(input("請輸入{}號同學的英文成績：".format(i)))
    mat = int(input("請輸入{}號同學的數學成績：".format(i)))
    scores.append((i, chi, eng, mat))
fp = open("scores.dat", "w")
fp.write(str(scores))
fp.close()
```

```
[(1, 85, 64, 85), (2, 21,
85, 96), (3, 26, 45, 78),
(4, 26, 85, 98), (5, 80,
95, 65)]
```

輸入三科成績並儲存成元組串列型式文字檔

```
f = open("mydata.txt", "r", encoding="utf-8")  
data = f.read()  
f.close()  
print(data)
```

讀取資料檔

文字檔的名稱是"mydata.txt"，
必需放在執行程式的同一個資料夾當中。

```
f = open("mydata.txt", "r", encoding="utf-8")  
data = f.readlines()  
f.close()  
print(data)
```

['李白\n', '將進酒\n', '君不見\n', '黃河之水
天上來，奔流到海不復回？\n', '君不見\n', '
高堂明鏡悲白髮，朝如青絲暮成雪？\n', '人
生得意須盡歡，莫使金樽空對月。\n', '天生
我材必有用，千金散盡還復來。\n', '烹羊宰
牛且為樂，會須一飲三百杯。\n', '岑夫子，
丹丘生，\n', '將進酒，君莫停。\n', '與君歌
一曲，請君為我側耳聽。\n', '鐘鼓饌玉不足
貴，但願長醉不願醒。\n', '古來聖賢皆寂寞，
惟有飲者留其名。\n', '陳王昔時宴平樂，斗
酒十千恣歡譁。\n', '主人何為言少錢，徑須
沽取對君酌。\n', '五花馬，千金裘，\n', '呼
兒將出換美酒，與爾同銷萬古愁！']

以分列的方式讀取文字檔案

```
with open("mydata.txt", "r", encoding="utf-8") as f:  
    data = [line.strip() for line in f.readlines()]  
print(data)
```

以分列的方式並去除符號
讀取文字檔案

['李白', '將進酒', '君不見', '黃河之水天上來，
奔流到海不復回？', '君不見', '高堂明鏡悲白
髮，朝如青絲暮成雪？', '人生得意須盡歡，
莫使金樽空對月。', '天生我材必有用，千金
散盡還復來。', '烹羊宰牛且為樂，會須一飲
三百杯。', '岑夫子，丹丘生，', '將進酒，君
莫停。', '與君歌一曲，請君為我側耳聽。',
鐘鼓饌玉不足貴，但願長醉不願醒。', '古來
聖賢皆寂寞，惟有飲者留其名。', '陳王昔時
宴平樂，斗酒十千恣歡譔。', '主人何為言少
錢，徑須沽取對君酌。', '五花馬，千金裘，',
'呼兒將出換美酒，與爾同銷萬古愁！']

```
import ast
with open("scores.dat", "r") as f:
    scores = ast.literal_eval(f.read())
print(scores)
```

透過 `eval()` 函數的計算，
把讀入的字串內容恢復成
它應用的型態。

解讀以串列型式存檔的文字資料檔

```
score = int(input("請輸入成績："))
```

例外處理的重要性

請輸入成績：o

```
-----  
ValueError                                Traceback (most recent call last)  
<ipython-input-21-4c5052fa4d69> in <module>()  
----> 1 score = int(input("請輸入成績："))
```

```
ValueError: invalid literal for int() with base 10: 'o'
```

```
try:
```

```
    score = int(input("請輸入成績："))
```

```
except:
```

```
    print("你必須輸入一個數字才行")
```

```
try:
    print("實際上預期可能會有例外的程式碼寫在這裡！")
    # 10 / 0
    print("在可能發生例外的指令之後的程式碼放在這邊！")
except Exception as e:
    print("發生錯誤了，錯誤訊息如下：")
    print(e)
else:
    print("沒有發生任何錯誤。")
finally:
    print("不管如何，都要執行這裡")
```

例外處理基本型式

例外種類	說明
Exception	一般性的錯誤，全部的內建例外都算是這個。
IndexError	索引值超出範圍錯誤
KeyboardInterrupt	使用者使用中斷按鍵（Ctrl + C）
NameError	找不到要使用的變數名稱
OverflowError	溢位錯誤
SyntaxError	語法錯誤
IndentationError	縮排錯誤
TypeError	型態錯誤
ValueError	給值錯誤
ZeroDivisionError	除以零錯誤
IOError	輸出入裝置錯誤
FileExistsError	要建立一個已經存在的檔案時發生的錯誤
FileNotFoundError	找不到檔案錯誤
PermissionError	權限不足錯誤

```
import os
classname = input("請輸入想要使用的班級名稱：")
filename = classname + ".dat"
try:
    f = open(filename, "w")
except Exception as e:
    print("你所使用的班級名稱{}：發生錯誤".format(classname))
    print(e)
finally:
    f.close()
```

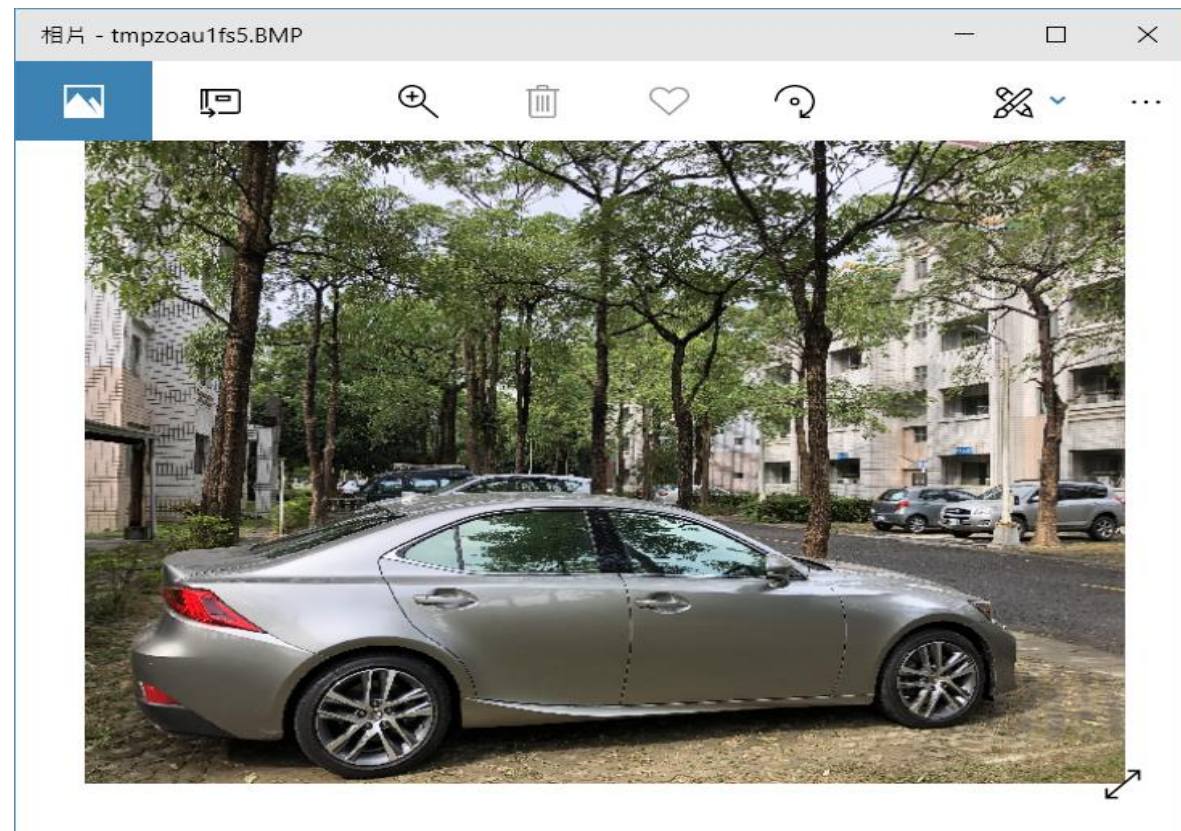
操作檔案時使用的例外處理程式碼

```
from PIL import Image  
im = Image.open("myIS300.jpg")  
print(im.format, im.size, im.mode)  
im.close()
```

如果發現不能執行的話，
請先安裝pillow模組。

開啟圖形檔資訊程式碼

```
from PIL import Image
im = Image.open("myIS300.jpg")
smaller = im.resize((320,200))
smaller.show()
smaller.save("myIS300s.jpg")
im.close()
```



縮放圖形檔案大小之程式碼

```
import os
from PIL import Image

source = input("請輸入來源資料夾：")
if os.path.exists(source):
    target = input("請輸入目標資料夾：")
    if not os.path.exists(target):
        os.mkdir(target)
        allfiles = os.listdir(source)
        << 處理每一個檔案的程式碼，請看下一張投影片 >>
    else:
        print("目標資料夾已存在，無法進行。")
else:
    print("找不到來源資料夾。")
```

批次轉換圖檔尺寸程式 PART 1

```
for file in allfiles:
    filename, ext = os.path.splitext(file)
    filename = filename + "_s"
    targetfile = filename + ext
    im = Image.open(os.path.join(source, file))
    thumbnail = im.resize((320,200))
    thumbnail.save(os.path.join(target, targetfile))
    im.close()
    thumbnail.close()
    print("{}-->{}".format(file, targetfile))
```

批次轉換圖檔尺寸程式 PART 2

請輸入來源資料夾：a

請輸入目標資料夾：c

2018-11-02 09.11.50.jpg-->2018-11-02 09.11.50_s.jpg

2018-11-07 11.53.40.jpg-->2018-11-07 11.53.40_s.jpg

2018-11-07 11.55.52.jpg-->2018-11-07 11.55.52_s.jpg

2018-11-07 17.25.37.jpg-->2018-11-07 17.25.37_s.jpg

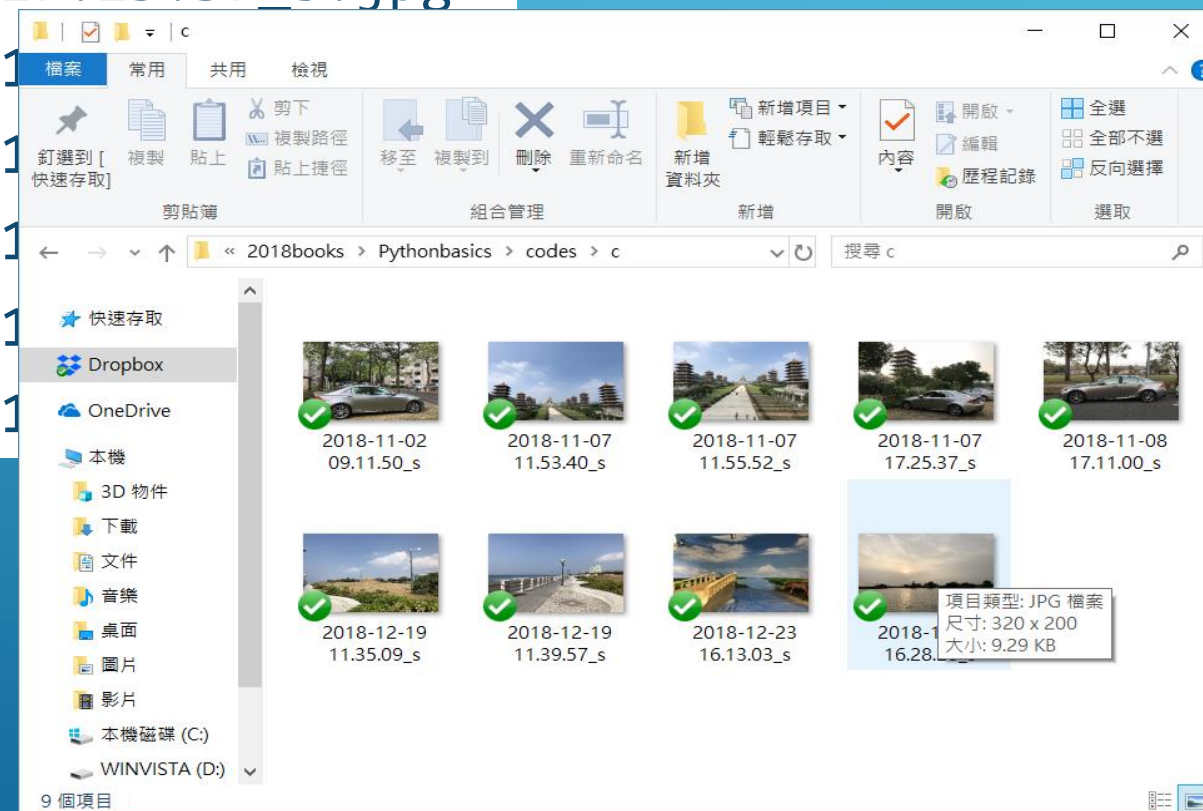
2018-11-08 17.11.00.jpg-->2018-11-08 17.11.00_s.jpg

2018-12-19 11.35.09.jpg-->2018-12-19 11.35.09_s.jpg

2018-12-19 11.39.57.jpg-->2018-12-19 11.39.57_s.jpg

2018-12-23 16.13.03.jpg-->2018-12-23 16.13.03_s.jpg

2018-12-23 16.28.20.jpg-->2018-12-23 16.28.20_s.jpg



執行過程以及結果

```
pre_html = '''
<!DOCTYPE html>
<head>
<meta charset='utf-8'/>
</head>
<body>
<table>
'''

post_html = '''
</table>
</body>
</html>
'''
```

把HTML前後程式碼放在變數中


```
<table border=1>
<tr>
<td>第一列第一欄</td><td>第一列第二欄</td><td>第一列第三欄</td>
</tr>
<tr>
<td>第二列第一欄</td><td>第二列第二欄</td><td>第二列第三欄</td>
</tr>
<tr>
<td>第三列第一欄</td><td>第三列第二欄</td><td>第三列第三欄</td>
</tr>
</table>
```

第一列第一欄	第一列第二欄	第一列第三欄
第二列第一欄	第二列第二欄	第二列第三欄
第三列第一欄	第三列第二欄	第三列第三欄

HTML表格的標準格式碼

被連結的文字內容或是其它標記

圖形檔HTML格式與超連結格式的結合

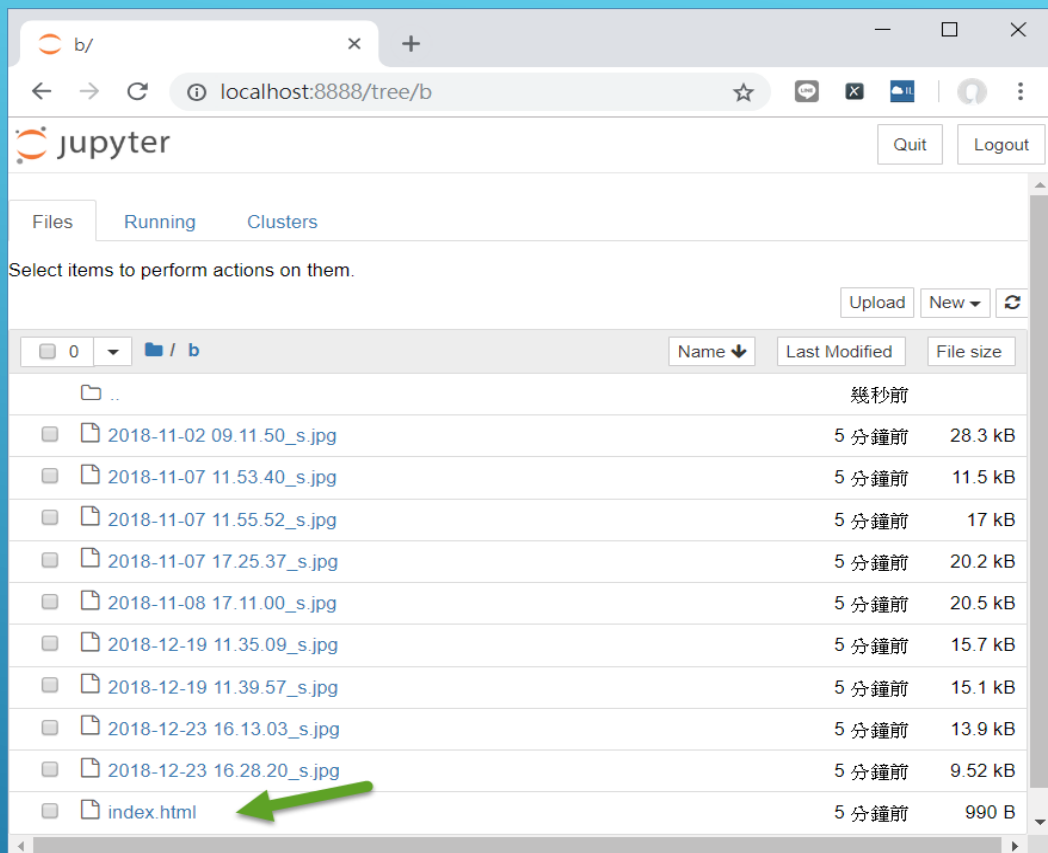
```
table_html += "<tr><td><a href='{0}'><img src='{0}'></a></td></tr>".format(
    os.path.join("../", os.path.join(source, file)),
    targetfile)
```

```
html = pre_html + table_html + post_html
```

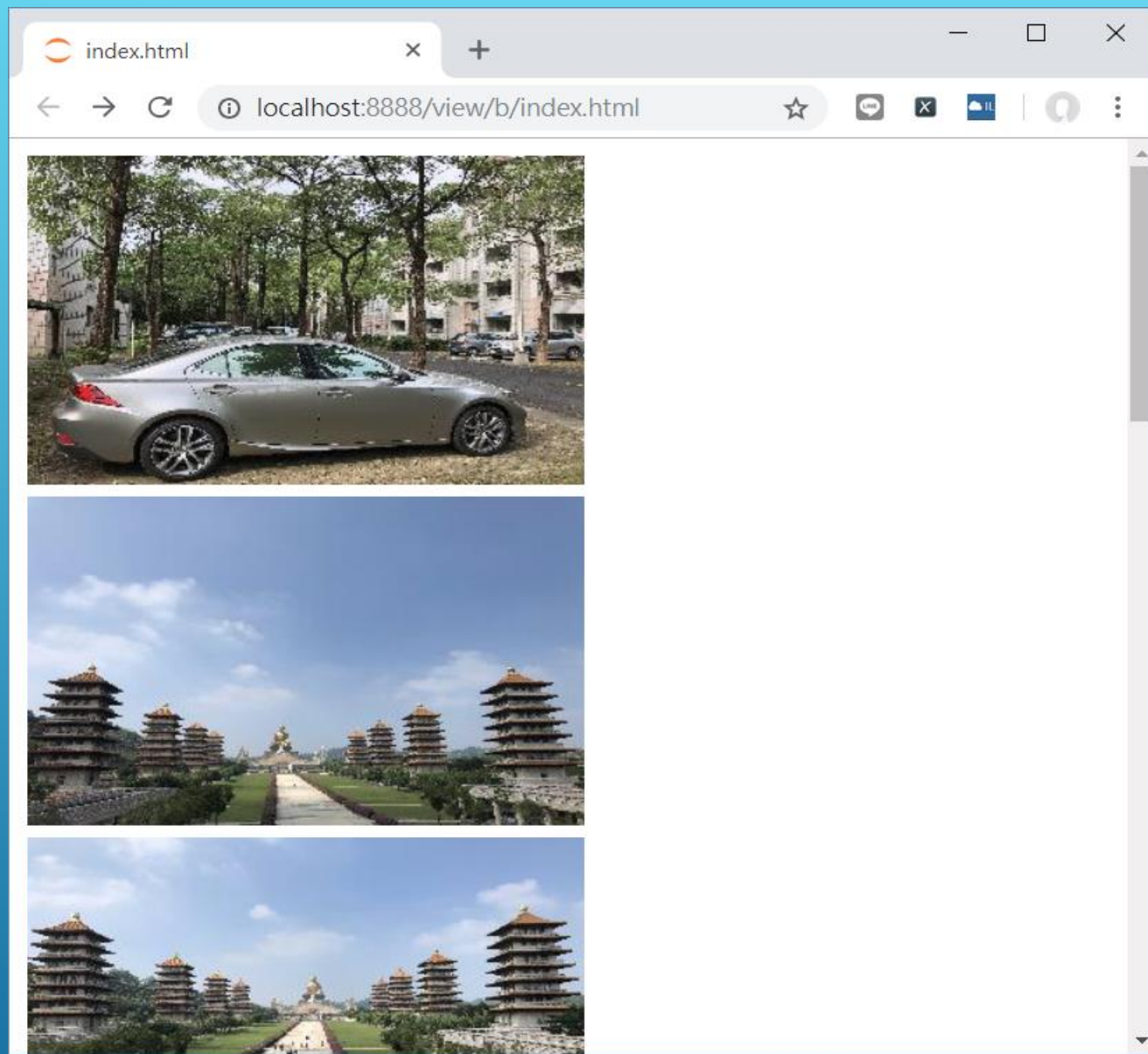
建立表格的PYTHON程式碼

```
for file in allfiles:
    filename, ext = os.path.splitext(file)
    filename = filename + "_s"
    targetfile = filename + ext
    im = Image.open(os.path.join(source, file))
    thumbnail = im.resize((320,200))
    thumbnail.save(os.path.join(target, targetfile))
    im.close()
    thumbnail.close()
    print("{}-->{}".format(file, targetfile))
#以下的程式碼用來建立HTML索引檔的表格內容
    table_html += "<tr><td><a href='{}'><img src='{}'></a></td></tr>".format(
        os.path.join("../", os.path.join(source, file)),
        targetfile)
#以上的程式碼用來建立HTML索引檔的表格內容
```

加到迴圈中的主程式片段



產生之索引檔位置及
瀏覽之效果



在適當的地方加上以下的</tr><tr>對，
就可以強制換列（其實就是新增表格中
的一列）

```
if (index+1) % 3 == 0:  
    table_html += "</tr><tr>"
```

改一下圖形檔排列的版面

