

準備工作

○○○○○○○○○○○○○○○○○○○○ 本章大綱 ○○○○○○○○○○○○○○○○○○○○○

☆ 1-1 Python 優點

☆ 1-2 體驗 Python 整合開發環境

1-1 Python 優點

以前因為電腦硬體貴、記憶體少、執行效率差，所以程式語言的語法規定就要一大堆，這樣才能讓寫出的程式有效率的執行，例如，Visual Basic、C/C++、C#、Java 等語法規定就一大堆，尤其 C/C++ 規定更是多，所以其執行效率也特別高。但是目前硬體的記憶體便宜了，執行速度也提高了，小小的一支智慧型手機，其功能與記憶體都超過 10 年前的桌上型電腦。Python 之所以能異軍突起，主要就是硬體功能與效率提高了，所以可以再使用解譯式語言、將語法的包容性提高、資料型態簡化、及撰寫更多的高階函式。分別說明如下：

解譯式語言

C/C++、C#、Java 等都是編譯式語言，要等到全部程式全對，才有執行結果，其次，還要有類似連續劇片頭與片尾的標頭檔，才

能構成完整程式，這樣才能提高程式執行效率。但是 Python 就神了，每次都是開門見山，絕不浪費時間播放片頭與片尾，所以程式特別簡潔，其次，編譯式語言，要程式全對才能執行，解譯程式則可以一面測試、一面除錯、所以較方便程式除錯。例如，Python 一行也是程式，如以下敘述：

```
print (3+5)
```

資料型態簡潔

C 等（本書以『C 等』表示 C/C++/C#/java 語言）的資料型態宣告都要很明確，不但要分整數或實數，且不管是整數、還是實數，其精密度都還要再細分兩三種。Python 就神了，只要給個初值，其精密度通通接近無限，一定可以滿足您的需求。例如，請鍵入以下程式，並觀察執行結果。

```
a=2 #整數
print(2**100) #2的100次方
b=3.4 #實數
print(b**100) #b的100次方
```

其次 C 的函式還要依精密度挑選恰當的函式，例如，abs() 與 fabs() 傳回值的精密度就不同，這樣常常讓初學者哭不完，找不到錯誤。例如：

```
x=3.4
abs(x)
```

傳回整數：

```
x=3.4
fabs(x)
```

傳回實數。誤用了，真的沒有錯誤信息，結果就是錯。

語法很人性

C 等爲了追求執行效率，它們的規定多如牛毛。例如，陣列不只要規定型態、整個陣列型態也要一致，那是以前硬體技術不佳，所以凡事斤斤計較。既然目前硬體進步了，Python 串列 (list) 內的資料可以是任何資料型態、甚至 list 內還可有長度不一的 list。例如：

```
a=[3,"aa",[4,5,"bb"],["國勝"]]
print(a)
```

運算子的取用也是盡量接近人類的習慣。例如，

```
a=3
b="abcd"
print (a*3) #數值3倍
print(b*3) #輸出字串3次
```

and、or、xor 等邏輯運算子就直接用英文了，這樣更好記。例如：

```
x>3 and y<0    (C等都是x>3 && y<0)
```

內部方法或函式特別多且實用

C 等在大批資料處理，都要使用者事必躬親，例如求極大、極小、排序、搜尋、集合等都要自己寫程式。Python 就神了，諸如以上使用者常用的需求，Python 內部都有現成的方法或函式，例如，極大、極小、排序、搜尋等功能，都是一行就可解決。例如：

```
a=[3,8,4]
print(max(a)) #8
a.sort() #排序
print(a) # [3,4,8]
```

但以上程式，C 等都是要寫十幾行，才能完成以上功能。

協力廠商套件源源不絕

因為 Python 開源，所以熱心的人士與協力廠商就源源不絕開發實用套件。以股市分析程式為例（可能是股票買賣公司，爲了鼓勵大家投資股市，就幫忙寫一些股票看盤套件），例如，以下程式，就可輸出近三日『大立光』盤價。

```
import twstock as a
b=a.Stock("3008")#大立光
print(b.price)#近三日盤價
```

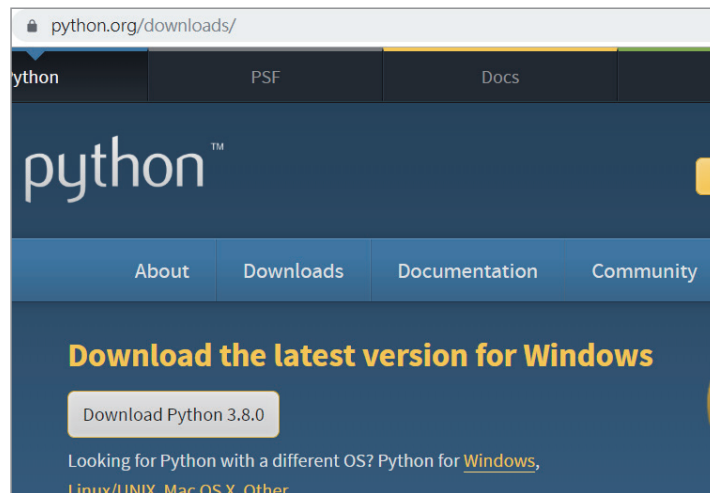
所以，使用 Python，就如同站在巨人的肩膀上，您可以立即看的更高更遠，就如同現在的照相初學者使用手機拍照，竟然也能拍出過去十年僅有專業攝影師使用高檔相機拍出的效果。所以有點是跨世代的智慧程式語言，目標也是希望寫程式就如同用手機拍照，就可以取代傳統高檔專業相機一樣，只要簡單操作，就能寫出專業程式設計師的成果。所以現在程式設計有點要分流了，入門者可先學這種智慧型程式語言，例如，Python，等有興趣者再鑽研其他專業程式語言。

1-2 Python開發環境

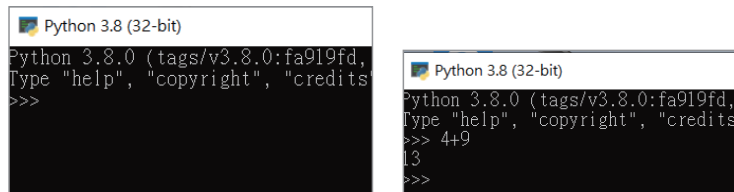
Python 的整合開發學習環境（Integrated Development Learning Environment，簡稱 IDLE）很多，通通各有優缺點，本書僅介紹 Python 官網的 IDLE 與 Anaconda 的 Spyder(也是整合開發環境)，分別說明如下：

Python IDLE

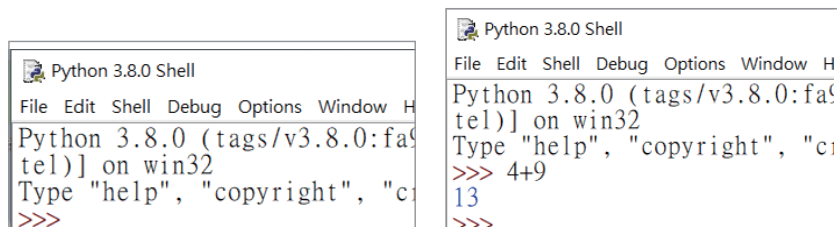
請開啓 Python 官網(www.python.org)，畫面如下圖，點選 Downloads 即可下載執行檔，並完成安裝。



請留意開始功能表已經出現『Python』資料夾，點擊後可出現三個子選項。下圖左是開啓子選項『Python』，電腦出現的即時運算視窗畫面，您可在此進行簡單計算機功能運算。例如，下圖右是輸入 $4+9$ ，Python 回應 9 的畫面。



下圖左是開啓『IDLE』，電腦出現 Python 整合編輯視窗的畫面，同樣可在此畫面進行計算機運算功能，進行即時運算。例如，下圖右是輸入 $4+9$ ，Python 回應 9 的畫面。

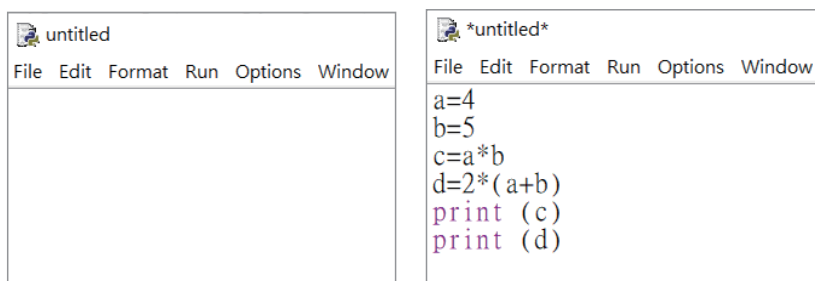


開始程式設計

以下是一個很簡單的 Python 程式，它可以計算一個長方形的面積與周長。

```
a=4
b=5
c=a*b
d=2*(a+b)
print (c)
print (d)
```

下圖左是點選『File/New File』的畫面，下圖右是輸入以下程式的畫面。

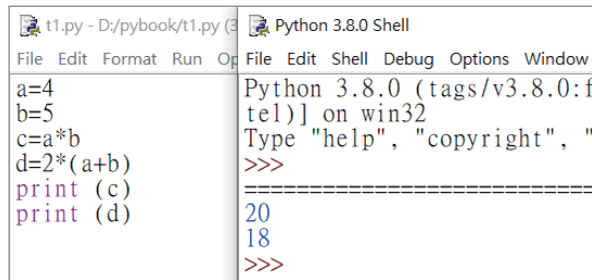


存檔

每一次寫完、或修改程式都要先存檔。第一次請先點選『File/Save as』存檔，請自行選擇資料夾與檔名，本例以『t1』為檔名，副檔名預設為『py』，請不用鍵入。

執行

點選功能表『Run/Run Module』，結果如下圖右。目前共有兩個視窗，一個是 Python Shell，如下圖右，可出現執行結束；一個是程式編輯視窗，如下圖左，您可以用以上功能表的『Window』切換。



```

t1.py - D:/pybook/t1.py (3) Python 3.8.0 Shell
File Edit Format Run Op File Edit Shell Debug Options Window
a=4
b=5
c=a*b
d=2*(a+b)
print (c)
print (d)
Python 3.8.0 (tags/v3.8.0:fa
tel)] on win32
Type "help", "copyright", "
>>>
=====
20
18
>>>

```

自我練習

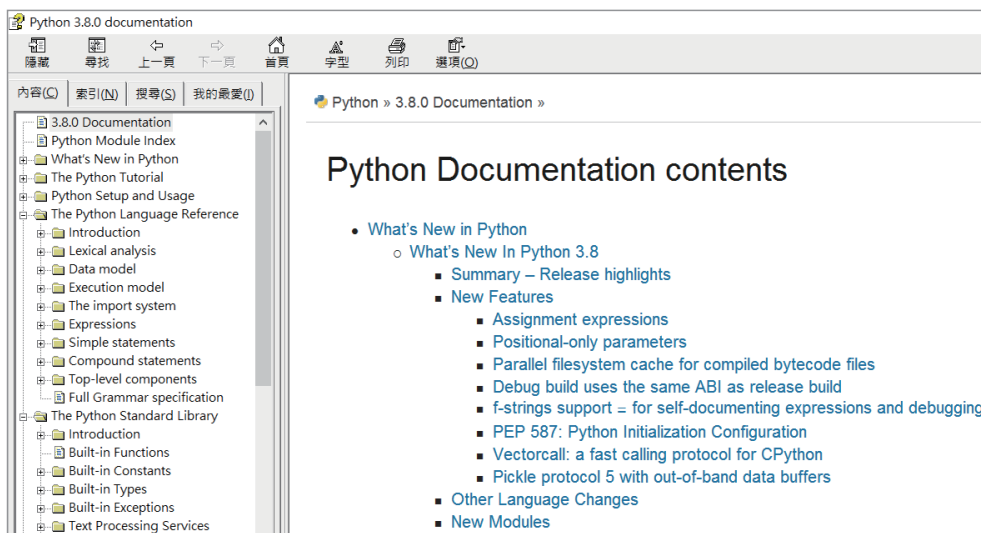
請指派長方體的長、寬、高，指令是 `a=1;b=2;c=3`，寫程式計算其表面積與體積。

使用手冊

Python 的使用手冊有語言使用手冊與模組說明文件，分別說明如下：

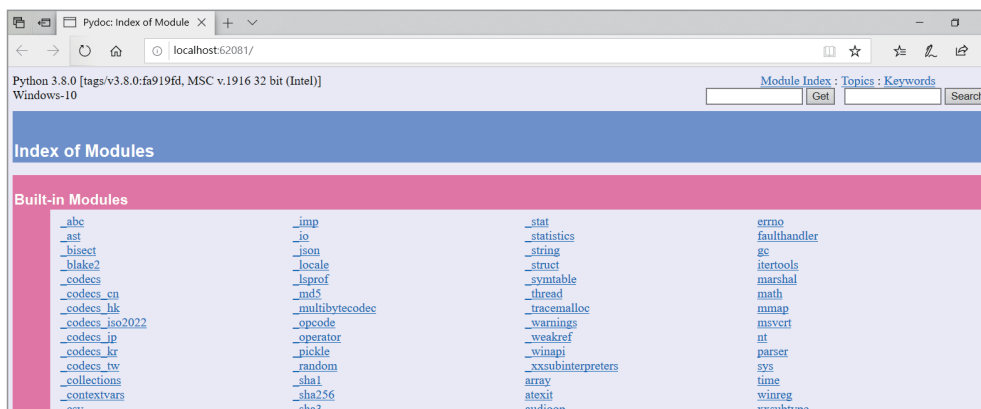
語言使用手冊

下圖是開啓 Windows 作業系統『開始』功能表，『Python/Python 3.8 Manuals』的畫面，請自行探索『The Python Language Reference』與『The Python Standard Library』。



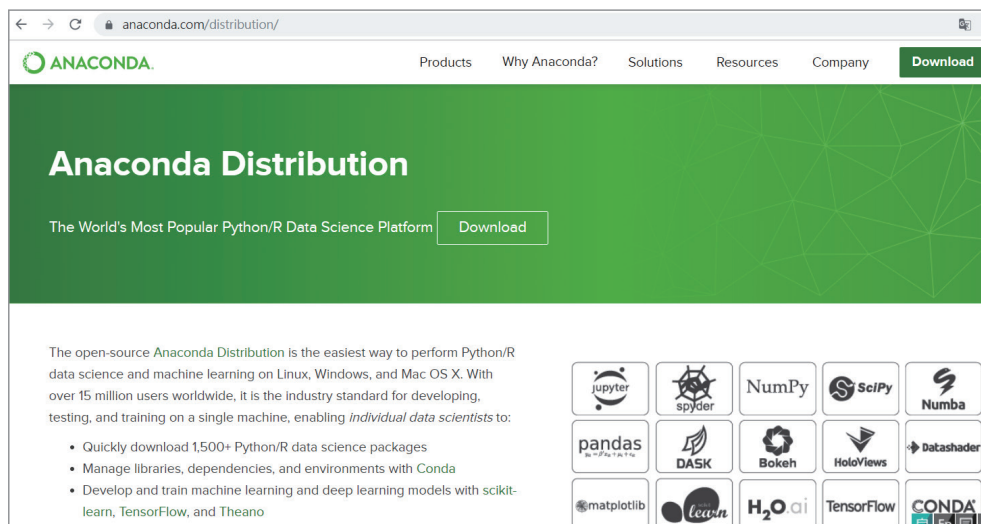
模組說明文件

下圖是開啓 Windows 作業系統『開始』功能表『Python/Python 3.7 Moudule Docs』的畫面，這裡有很實用模組使用文件說明，在此推薦。



Anaconda

因為 Python 很實用，就有熱心人士也自行開發 Python 整合開發環境。我是認為 Anaconda 是目前最實用、最強的 Python 外掛整合開發環境，就由 Anaconda 公司開發，官網如下圖，請於下圖下載安裝執行檔。

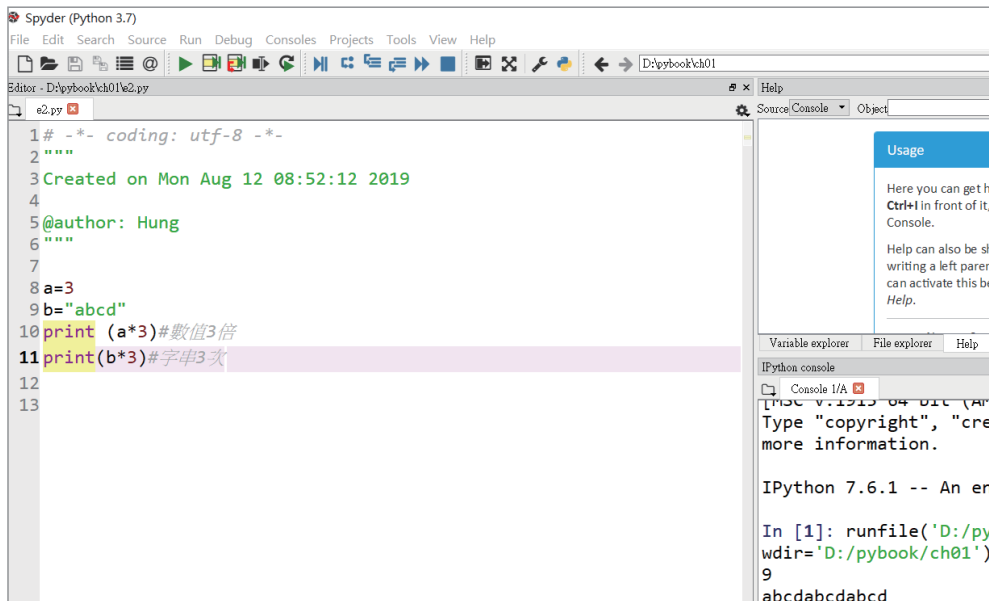


特色

Anaconda 與 Python 官網的整合編輯環境相較，其優點如下：

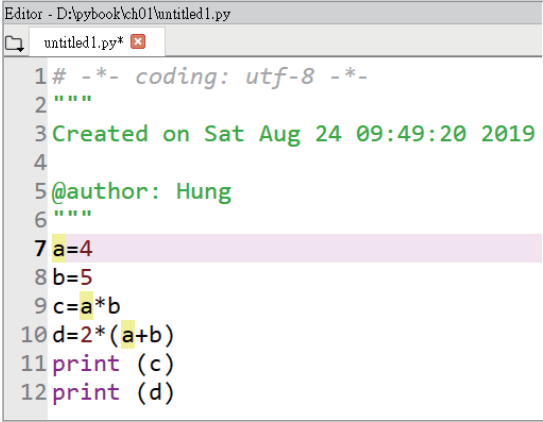
1. 程式、執行結果或錯誤信息都在同一視窗出現，這也是所有整合編輯環境的趨勢。
2. 打完物件後，有出現可用方法提示，然後就可用『Enter』鍵點選所要方法。
3. 打完函式或方法，還會出現該函式或方法的參式格式。
4. 執行前還會自動存檔。
5. 常用模組或套件已經下載完畢，使用者不用再額外下載。

以下是 Spyder 開新檔案（點選功能表的『File/NewFile』）的畫面。左邊窗格是程式編寫區，右上方窗格是輔助說明區，右下方窗格是程式執行輸出區。左邊程式編寫區，第一行的井號『#』與兩個『三雙引號（"'''）』之間都是註解，僅給人看，電腦都不解譯。



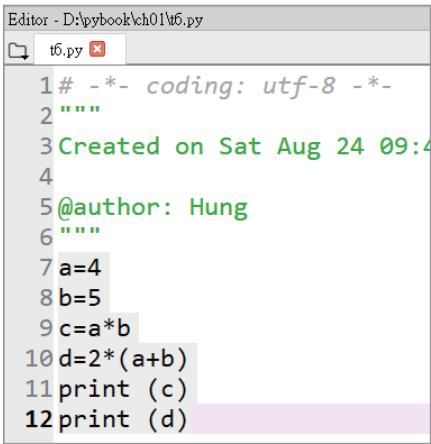
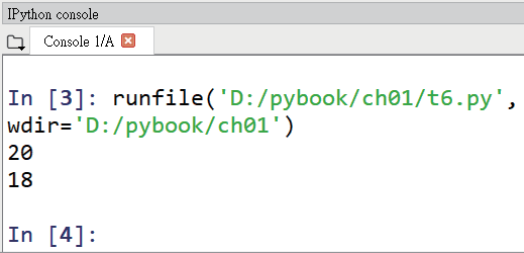
編寫執行程式

同上一範例周長的計算，程式如下圖左，下圖右是寫入程式的畫面。

<pre>a=4 b=5 c=a*b d=2*(a+b) print (c) print (d)</pre>	
--	--

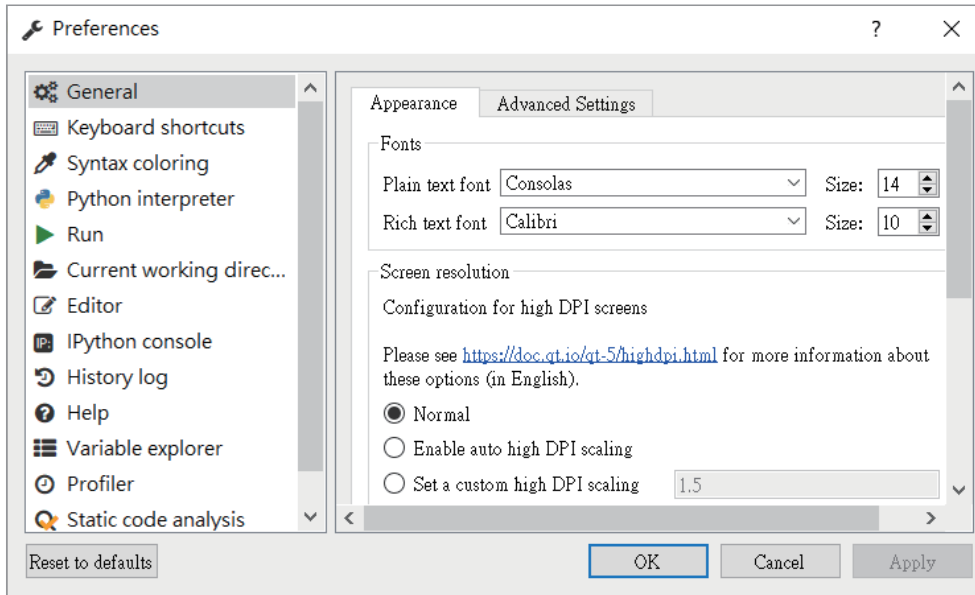
下圖左是點選工具列的『Run File』的畫面，因為是第一次執行，電腦會出現存檔對話框，要求先存檔，請點選資料夾，填入主檔名就好，附檔名預設為『.py』，也是不用鍵入。

下圖右是執行結果畫面。

	
---	--

修改編輯環境

點選功能表的『Tools/Preference』，畫面如下圖，可在此修改字型大小。



補充說明

Anaconda 雖然比較好用，但經過長時間測試，發現其穩定度不是很好，有時候明明沒有錯誤，但就是沒有結果，若讀者遇到這一問題，有時候重開機、有時候嘗試複製到官方版 Python，竟然是好的。

自我練習

以下是解一元二次方程式 $x^2+4x-5=0$ 的程式，請自行選用 Python 解譯器輸入，並觀察輸出結果是否為 1 與 -5。

```
a=1;b=4;c=-5
d=(b**2-4*a*c)**(1/2)
x1=(-b+d)/(2*a)
x2=(-b-d)/(2*a)
print(x1)
print(x2)
```

MEMO