



巨匠直播教學

APCS Python語法基礎班

www.pcschoolonline.com.tw

同學，歡迎你參加本課程

- ☑ 請關閉你的FB、Line等溝通工具，以免影響你上課。
- ☑ 考量頻寬、雜音，請預設關閉攝影機、麥克風，若有需要再打開。
- ☑ 隨時準備好，老師會呼叫你的名字進行互動，鼓勵用麥克風提問。
- ☑ 如果有緊急事情，你必需離開線上教室，請用聊天室私訊給老師，以免老師癡癡呼喚你的名字。
- ☑ 軟體安裝請在上課前安裝完成，未完成的同學，請盡快進行安裝。

課程檔案下載

The screenshot displays the homepage of the Juepin Computer Online Live website. The header features navigation links for '開課查詢', '免費體驗專區', '課程總覽', '專業師資', '學員專區', '講師專區', and '最新消息'. Social media icons for 360, Facebook, and YouTube are also present. A user login area shows '您好!' and a '登出' button. The main banner area contains a large text overlay: '程式語言好難學? 那是因為你還沒學過Python! (線上老師 LIVE 直播教學 · 搶先看)'. Below this, a sidebar menu is open, listing various user options. The '課程檔案下載' option is highlighted with an orange box and a callout bubble. The background of the banner features a blue and purple digital theme with circuit-like patterns and a clock face.

巨匠電腦線上真人 開課查詢 免費體驗專區 課程總覽 專業師資 學員專區 講師專區 最新消息

360 f YouTube

您好! 登出

點數卡產品兌換
APCS檢測專區
公告專區
我的課表
IT真人課程劃位
電腦分校課程劃位
外語真人課程劃位
美語分校課程劃位
取消劃位
課程檔案下載
上課權益查詢
教學平台測試
學習諮詢
常見問題
個資維護
忘記密碼
登出

課程檔案下載

程式語言好難學?
那是因為
你還沒學過Python!
(線上老師 LIVE 直播教學 · 搶先看)

巨匠電腦真人課程

ZOOM 學員操作說明

The screenshot shows the Zoom interface with several callouts:

- 5 查看選項/共同註記/筆 (連連看)**: Points to the '共同註記' (Co-Annotate) option in the top right menu.
- 2 共享螢幕 (指導演練; 點評作品)**: Points to the '共享螢幕' (Share Screen) button in the bottom toolbar. A sub-note says: '老師須先停止共享螢幕才能請學生共享螢幕' (The teacher must first stop sharing the screen before asking the student to share the screen).
- 1 聊天**: Points to the '聊天' (Chat) button in the bottom toolbar.
- 3 與會者/舉手**: Points to the '與會者' (Participants) button in the bottom toolbar. A sub-note says: '舉手' (Raise Hand).
- 4 解除靜音**: Points to the '解除靜音' (Unmute) button in the bottom toolbar.

The interface also shows a list of participants on the left, including '張齡月 (我)' and '婷婷', and a '解除靜音' button.



巨匠直播教學

APCS Python語法基礎班

Python 簡介及環境建立

www.pcschoolonline.com.tw

本堂教學重點

1. Python 語言簡介
2. Python 開發及執行環境安裝
3. Python 程式基本觀念
4. Python 輸出與輸入
5. Python 數值與文字運算

課程內容

1. Python 簡介

1-1. Python 語言簡介

1-2. Python 執行環境

1-3. Python 開發環境

2. Python 程式基本概念

2-1. 開發模式

2-2. 螢幕輸出

2-3. 註解

2-4. 變數宣告

3. Python 運算

3-1. 數值運算

3-2. 字串運算

3-3. 資料輸入

3-4. 格式化輸出

課程內容

1. Python 簡介

1-1. Python 語言簡介

1-2. Python 執行環境

1-3. Python 開發環境

2. Python 程式基本概念

2-1. 開發模式

2-2. 螢幕輸出

2-3. 註解

2-4. 變數宣告

3. Python 運算

3-1. 數值運算

3-2. 字串運算

3-3. 資料輸入

3-4. 格式化輸出

Python 程式語言簡介

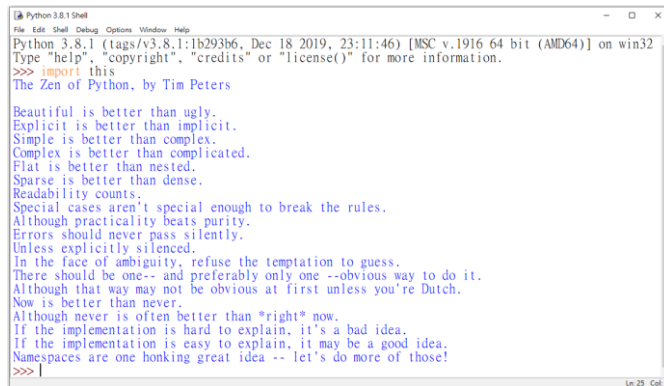
- ◆ 1989年由荷蘭的 Guido van Rossum 發明
 - ◇ 英文意思是蟒蛇 (作者是BBC電視劇 Monty Python's Flying Circus 的粉絲)
- ◆ 歷史沿革
 - ◇ 1991年2月20日首次公開，版本0.9.0
 - ◇ 1994年 1 月發佈Python 1.0 正式版
 - ◇ 2000年10月發佈Python 2.0 版本
 - ◇ 2008年12月發佈Python 3.0 版本
 - Python 2 與 Python 3 不相容，Python 2.x 支援至2020年
- ◆ 社群驅動的開放原始碼自由軟體
 - ◇ 由 Python 軟體基金會管理

Python 程式語言特性

- ◆ 跨平台直譯式的執行環境
- ◆ 物件導向程式語言
- ◆ 動態型別系統
- ◆ 語法簡潔，容易撰寫及閱讀
 - ◇ 直覺的語法
 - ◇ 縮排式腳本排列
- ◆ 擴充及整合性
 - ◇ 功能完整的程式庫

Python 之禪 (The Zen of Python)

- ◆ 優美優於醜陋 Beautiful is better than ugly.
- ◆ 明瞭優於隱晦 Explicit is better than implicit.
- ◆ 簡單優於複雜 Simple is better than complex.
- ◆ 複雜優於繁雜 Complex is better than complicated.
- ◆ 平坦優於巢狀 Flat is better than nested.
- ◆ 稀疏優於集中 Sparse is better than dense.
- ◆ 可讀性很重要 Readability counts.
- ◆ 實用性雖重於純粹性，但特例仍不該打破規則 Special cases aren't special enough to break the rules. Although practicality beats purity.
- ◆ 錯誤不能悄悄被忽略，除非你刻意為之 Errors should never pass silently. Unless explicitly silenced.
- ◆ 面對模糊不確定性時，拒絕猜測的誘惑 In the face of ambiguity, refuse the temptation to guess.
- ◆ 問題應該有一種，且最好只有一種，最明顯的解決方法 There should be one-- and preferably only one --obvious way to do it.
- ◆ 這個作法剛開始做時不是非常直觀，除非你是荷蘭人 Although that way may not be obvious at first unless you're Dutch.
- ◆ 現在做優於不做，但不假思索立刻做通常還不如不做 Now is better than never. Although never is often better than *right* now.
- ◆ 如果你的作法很難解釋，這必然是個壞方法 If the implementation is hard to explain, it's a bad idea.
- ◆ 如果你的作法很容易解釋，可能是好個方法 If the implementation is easy to explain, it may be a good idea.
- ◆ 命名空間是一個好主意，讓我們多多使用它 Namespaces are one honking great idea -- let's do more of those!



```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 23:11:46) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!
>>>
```

Python 開發及執行環境

◆ Python 官方環境

- ◆ Python 核心及標準函式庫(注意安裝的版本，有需要可安裝多個版本)
- ◆ pip 套件管理工具安裝其他第三方套件
- ◆ venv 虛擬環境工具

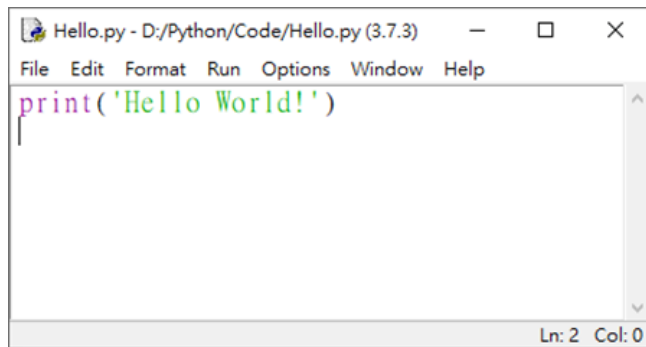
◆ Anaconda 環境 (森蚺，一種非常肥大的蟒蛇)

- ◆ 免費開源的 Python 發行版
 - 可同時包含多個版本Python執行環境
- ◆ conda 套件管理工具
- ◆ 包含虛擬工作環境

Python 開發及執行環境

◆ Python 官方執行環境

- ◆ 以C語言編寫而成，又稱為CPython
- ◆ Windows 作業系統需要另行安裝
- ◆ UNIX-Like (Mac / Linux) 作業系統有內建 Python 執行環境
 - 一般是 Python 2
- ◆ 內建簡易型開發工具 IDLE



Python 開發及執行環境

◆ Anaconda 發行版

◇ 預安裝許多常用第三方套件

- 目前有 7500+個第三方套件，250個套件會預安裝
- 約需 3GB 硬碟空間

◇ 內含常用開發工具

- Spyder
- Jupyter Notebook 單機版

◆ Minicoda 發行版

◇ 不預先安裝第三方 Python 套件

◇ 約需600MB 硬碟空間

建立課程所需軟體與環境

◆ Windows 環境

◆ 官方環境下載安裝：<https://www.python.org/>

◆ Mac/Linux 作業系統

◆ 檢查內建Python版本

◆ 若為Python 2，需下載並安裝Python 3 環境

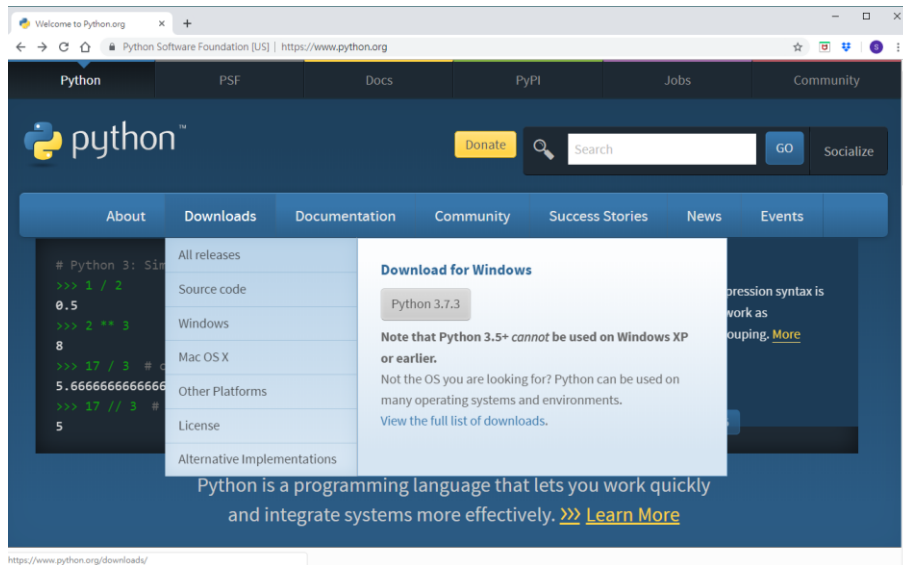
安裝 Python 官方環境

◆ Python 官方執行環境及IDLE開發工具

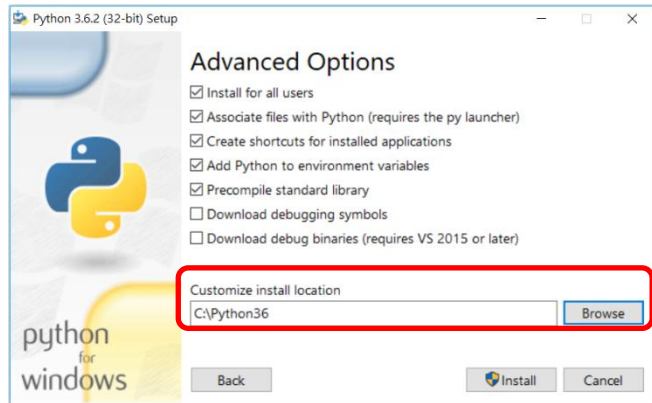
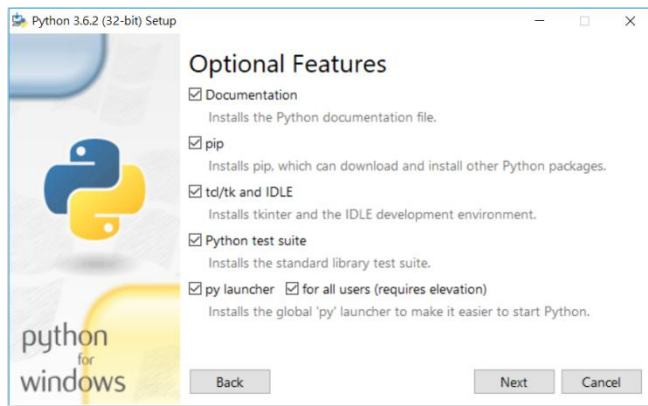
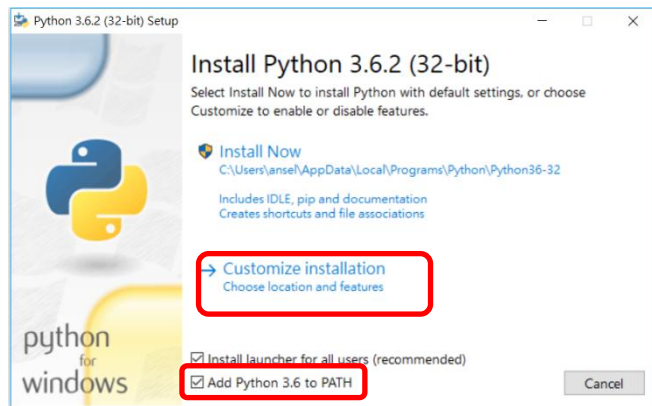
◆ <https://www.python.org/>

◆ python-3.x.x.exe

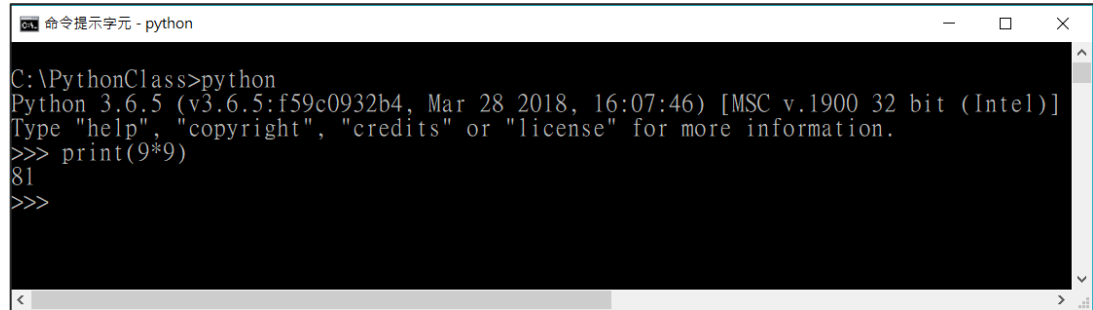
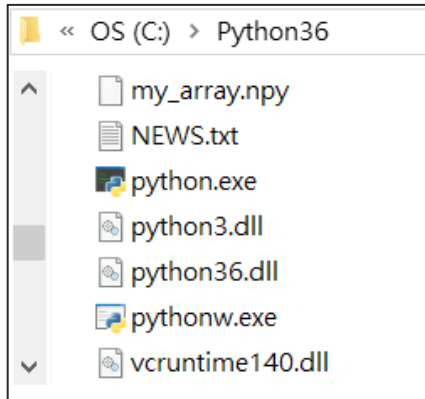
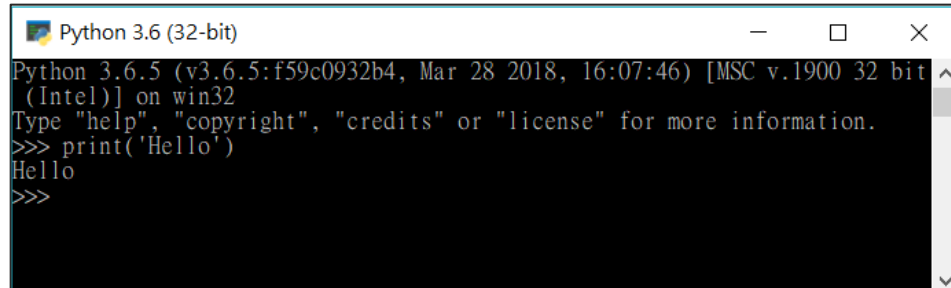
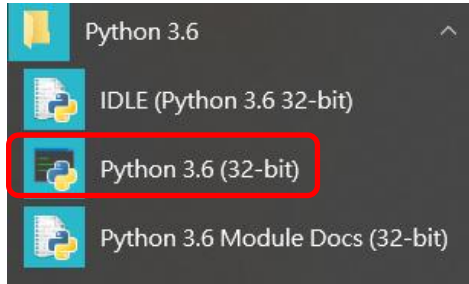
◆ 注意作業系統支援版本



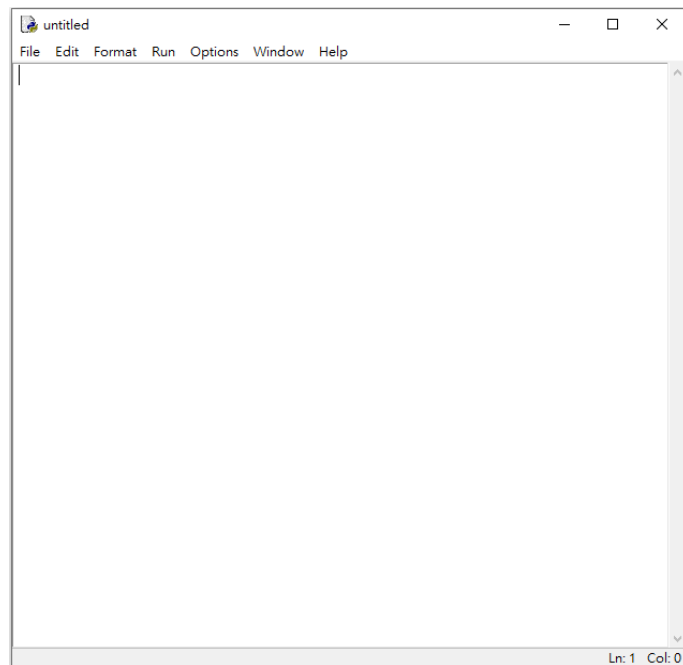
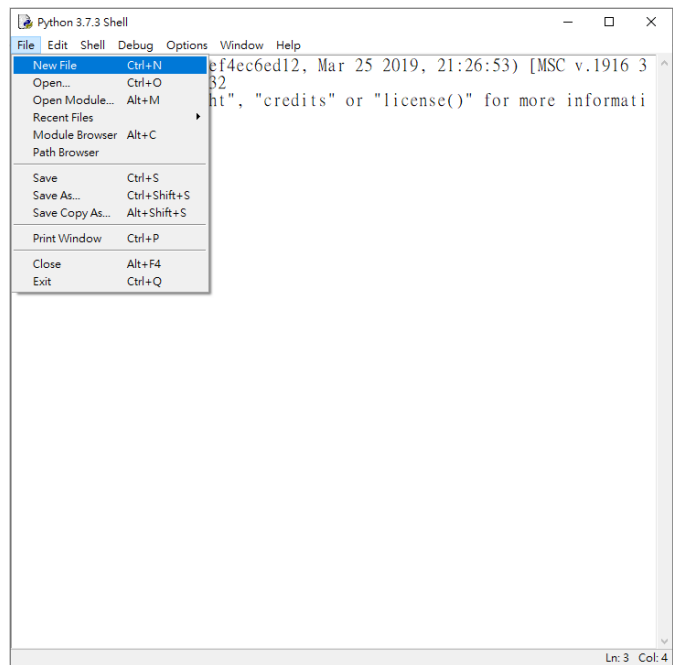
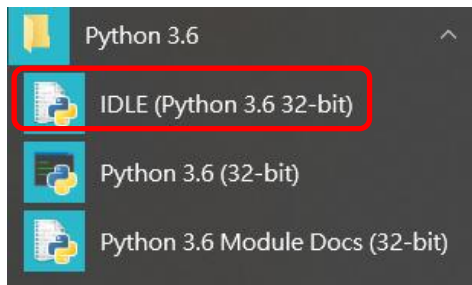
安裝 Python 官方環境



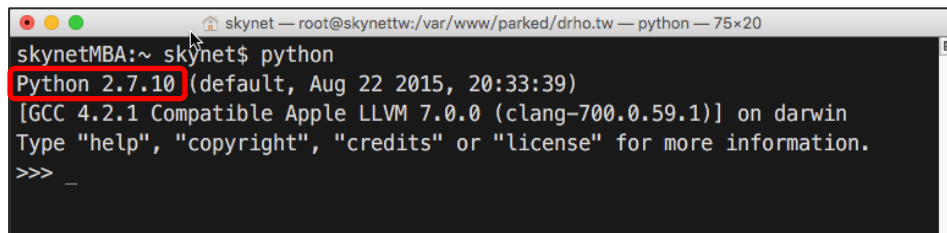
Python Shell

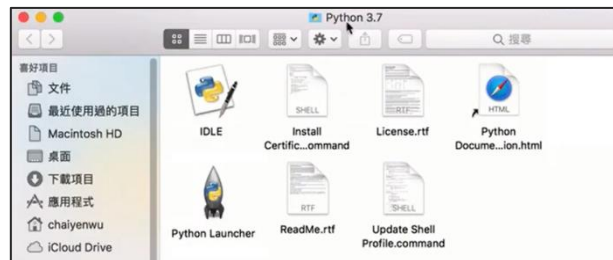
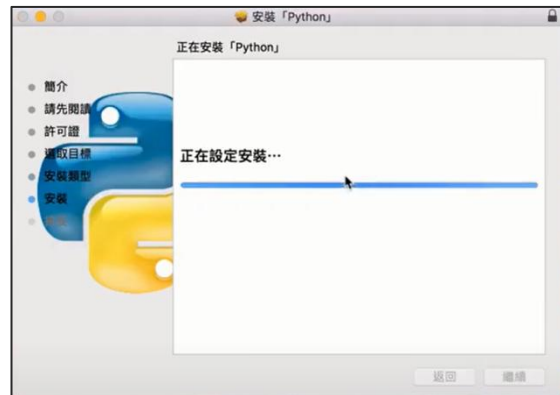
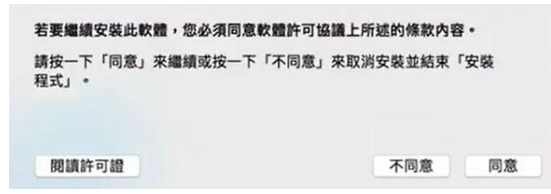
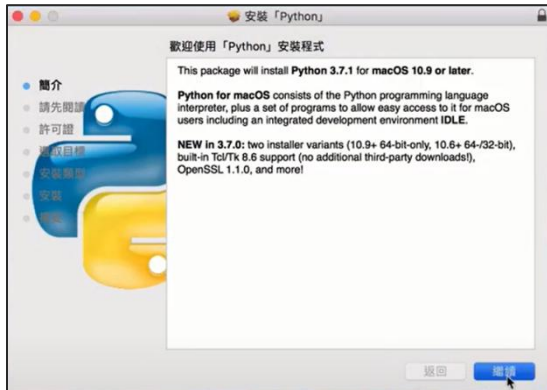
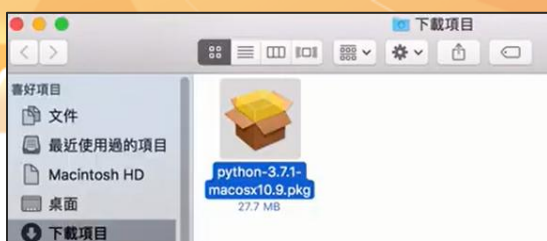


Python IDLE 工具



Mac 檢查內建Python版本





Python 開發環境

◆ Python 開發環境

- ◆ 純文字編輯器 + Python 執行環境

- ◆ 官方IDLE編輯器

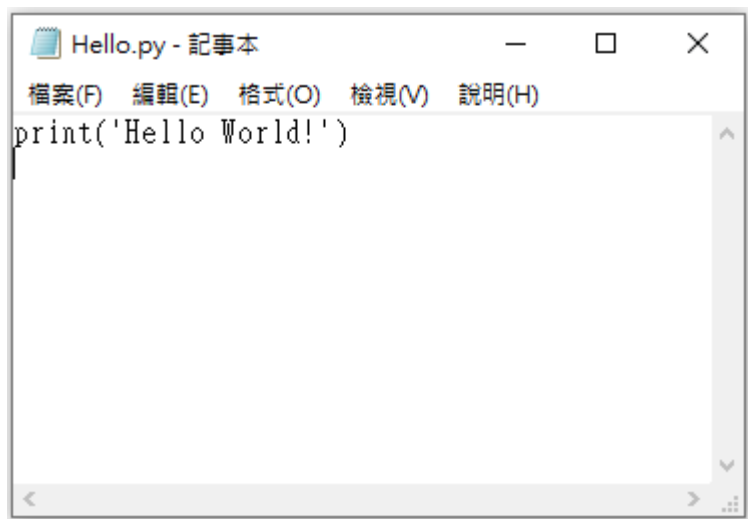
- ◆ 網頁形式編輯器

 - 單機Jupyter Notebook

 - 網路線上編輯器

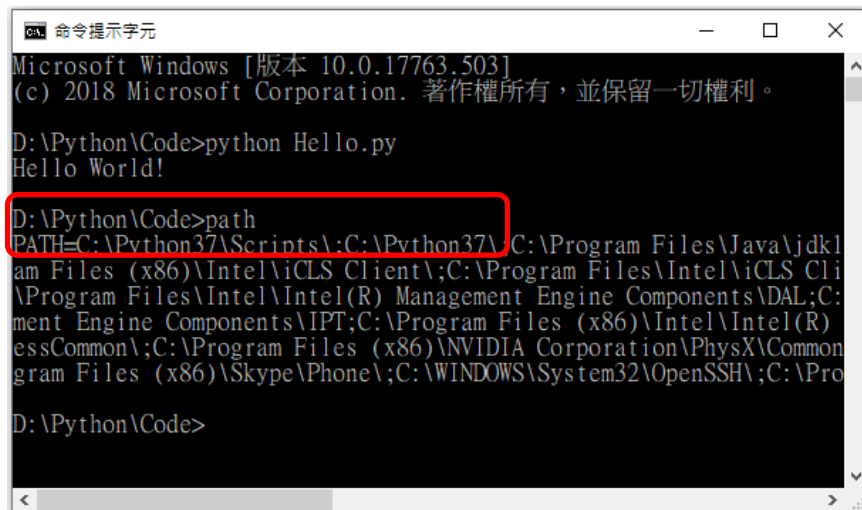
- ◆ 整合式開發環境IDE

純文字編輯器 + Python 執行環境



Notepad window titled "Hello.py - 記事本" showing the code:

```
print('Hello World!')
```

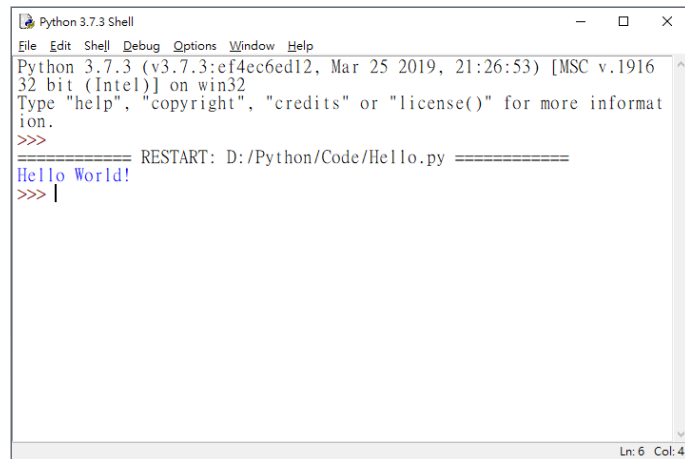
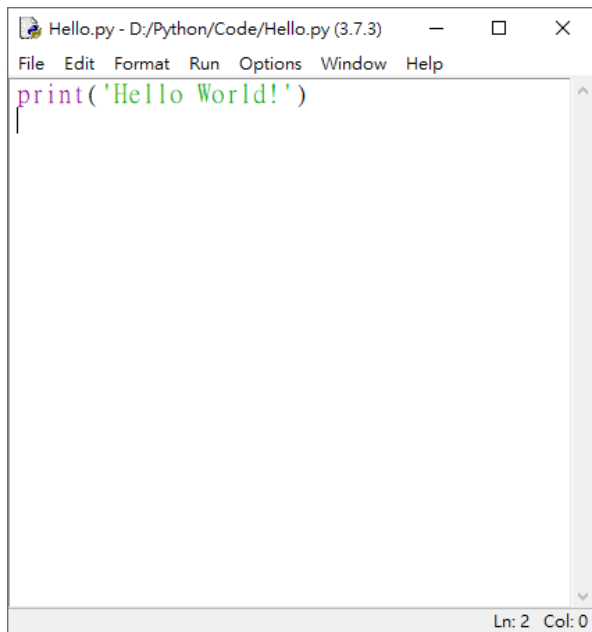


Windows Command Prompt window showing the execution of the Python script:

```
D:\Python\Code>python Hello.py
Hello World!

D:\Python\Code>path
PATH=C:\Python37\Scripts\;C:\Python37\;C:\Program Files\Java\jdk1
am Files (x86)\Intel\iCLS Client\;C:\Program Files\Intel\iCLS Cli
\Program Files\Intel\Intel(R) Management Engine Components\DAL;C:
ment Engine Components\IPT;C:\Program Files (x86)\Intel\Intel(R)
essCommon\;C:\Program Files (x86)\NVIDIA Corporation\PhysX\Comm
gram Files (x86)\Skype\Phone\;C:\WINDOWS\System32\OpenSSH\;C:\Pro
```

Python IDLE 編輯器



網頁形式編輯器

- ◆ 單機版Jupyter Notebook

- ◆ 線上程式碼編輯器

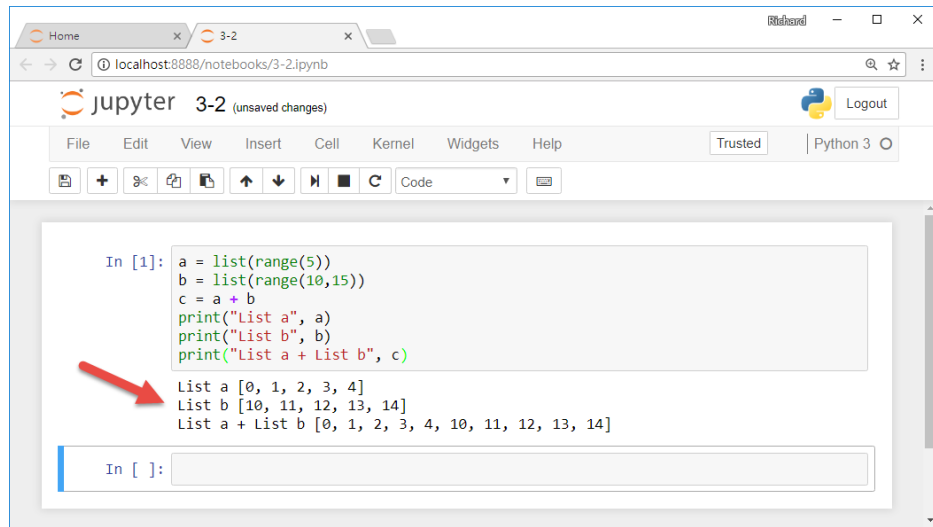
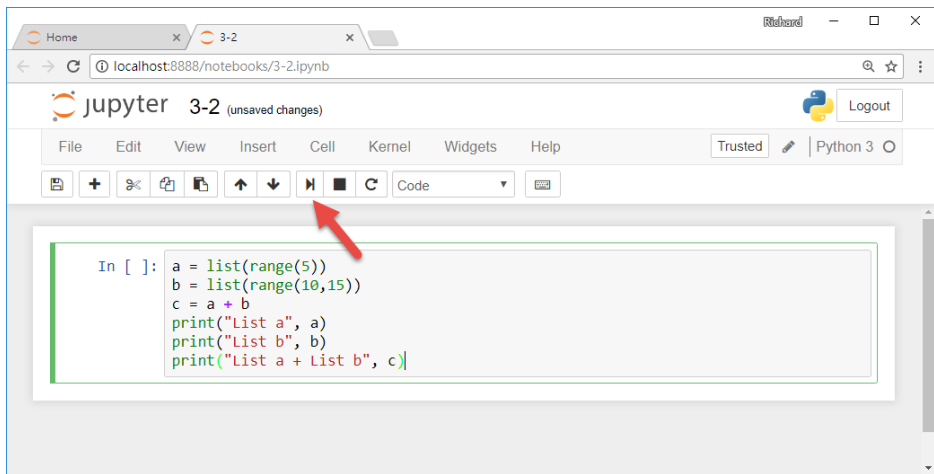
 - ◆ <https://repl.it>

 - ◆ 微軟 Azure Notebook <https://notebooks.azure.com/>

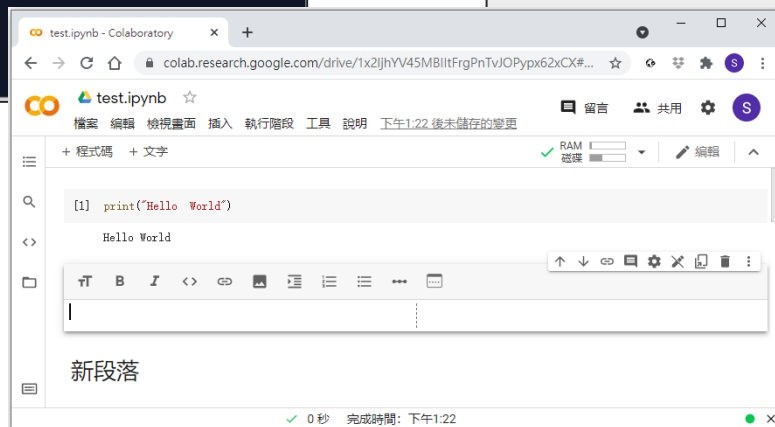
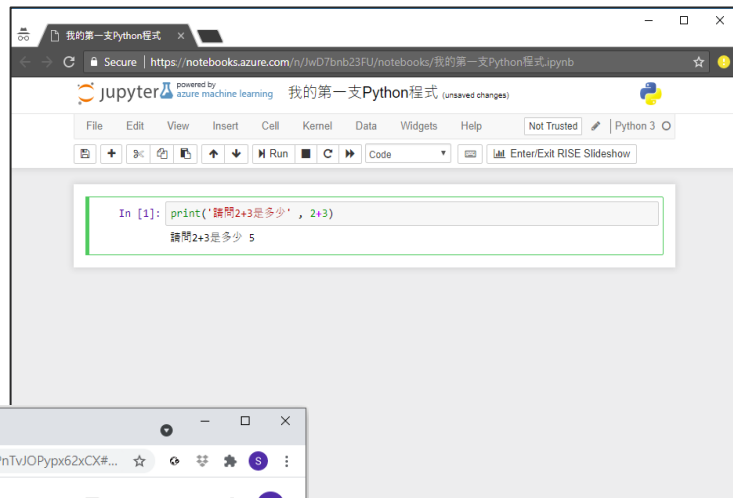
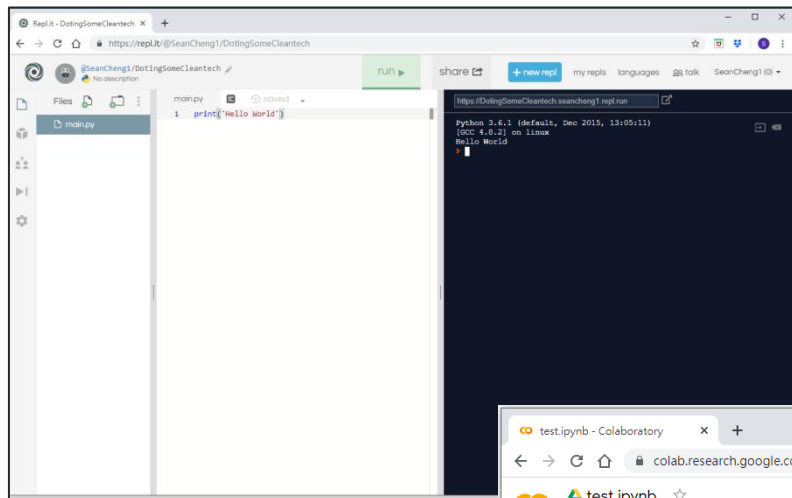
 - ◆ Google 雲端編輯器Colaboratory <https://colab.research.google.com>

 - ◆ Azure Notebook 及 Colaboratory 均為網路版Jupyter Notebook

單機版 Jupyter Notebook



線上程式碼編輯器

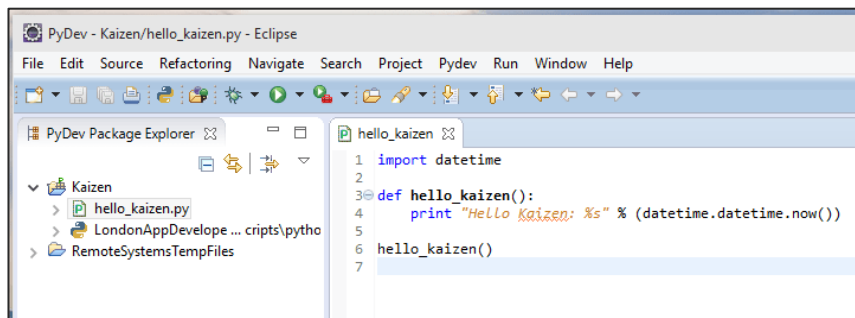
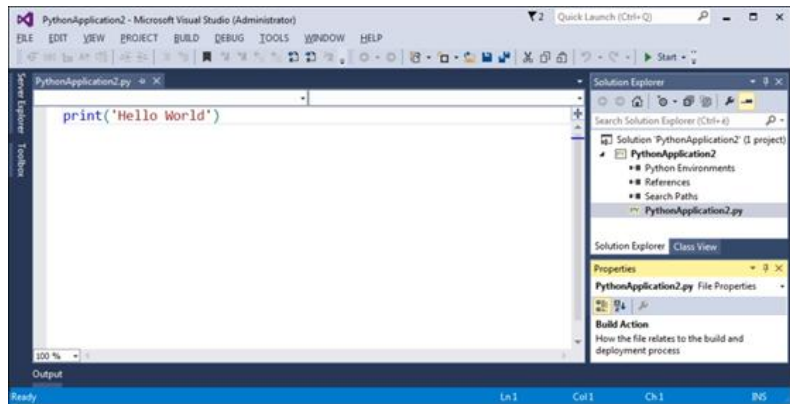
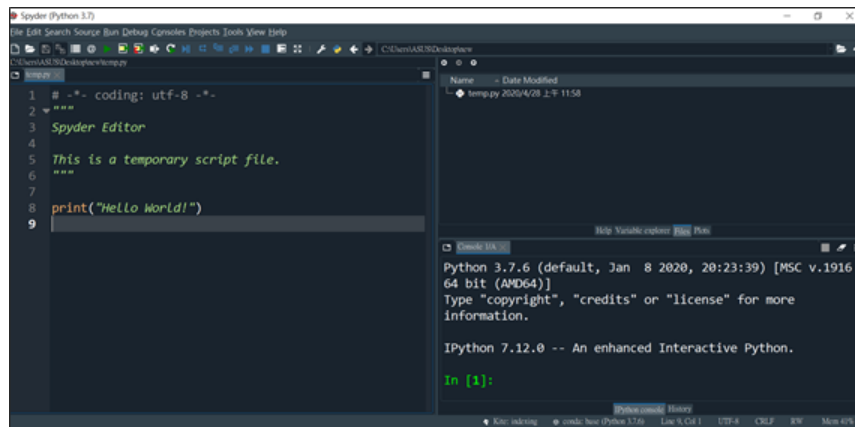
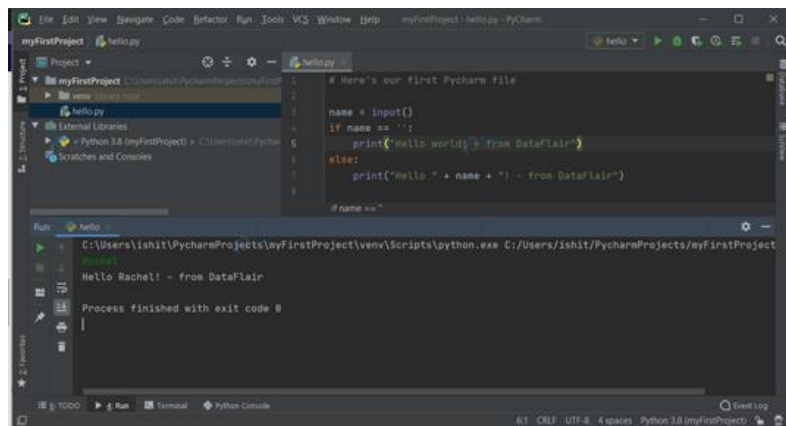


整合式開發環境IDE

◆ 整合式開發環境IDE

- ◆ 專門用於程式開發的商業軟體
- ◆ 一般需付費，有些有免費的Community 版本
- ◆ 友善的開發功能：語法上色、程式碼自動補齊、自動縮排、程式碼摺疊
- ◆ 常見Python IDE軟體
 - Pycharm
 - Spyder
 - Eclipse 外掛 PyDev
 - Python Tools for Visual Studio

常見Python IDE軟體





Q：下列關於Python語言的描述何者錯誤？

- a) Python 語言寫的程式需透過直譯器轉換成機器語言在電腦上執行
- b) Python 程式可被嵌入C/C++，作為C/C++的延伸程序或模組
- c) Python 2.x 版開發的程式碼可在Python3.x環境中執行
- d) Python 語言常被應用在大數據統計分析及人工智慧的機器學習



Q：下列何者不是線上編輯的開發工具？

- a) Jupyter Notebook
- b) Repl.it
- c) Python IDLE
- d) Google Colab

課程內容

1. Python 簡介

1-1. Python 語言簡介

1-2. Python 執行環境

1-3. Python 開發環境

2. Python 程式基本概念

2-1. 開發模式

2-2. 螢幕輸出

2-3. 註解

2-4. 變數宣告

3. Python 運算

3-1. 數值運算

3-2. 字串運算

3-3. 資料輸入

3-4. 格式化輸出

Python 程式開發模式

◆ Python Shell 視窗互動模式


- ◆ 直接在命令列下指令互動
- ◆ `exit()` 離開 Python Shell

◆ 檔案開發模式

- ◆ 程式儲存於純文字檔案中
 - 存檔為 `xxx.py`
- ◆ 以 `python` 直譯器讀取檔案執行

Python Shell 視窗互動模式

- ◆ 螢幕資料輸出
- ◆ 數學運算
- ◆ 適合作程式片段測試



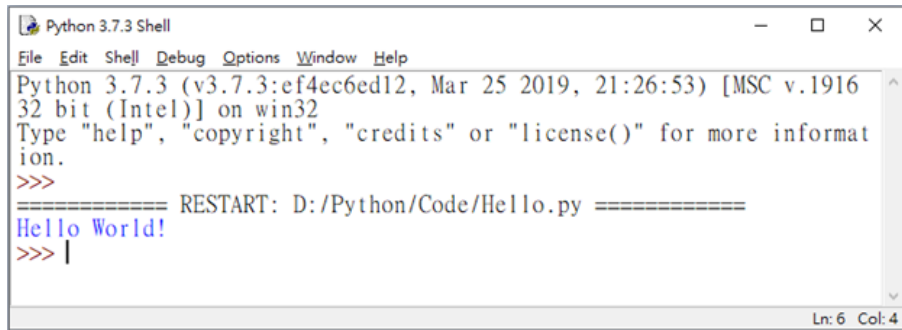
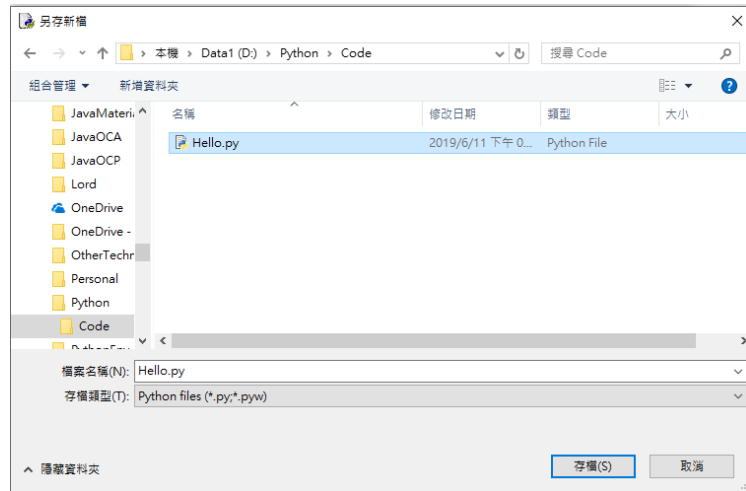
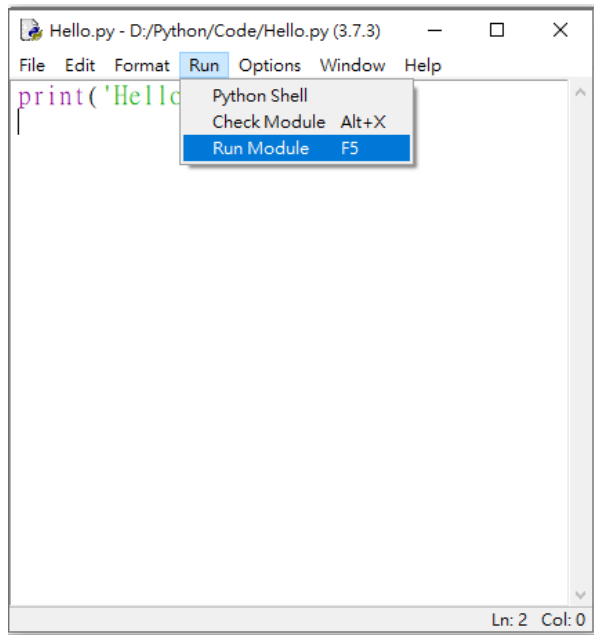
```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 2
3:11:46) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()
" for more information.
>>> print('Hello World')
Hello World
>>> 3 * 5 + 12
27
>>> |
```

Ln: 7 Col: 4

檔案開發模式

- ◆ 將程式內容保留下來
- ◆ 本課程使用官方**IDLE**編輯器新增並儲存檔案
 - ◇ 可以使用任一開發編輯器
- ◆ 編輯器提供內建工具
 - ◇ 撰寫程式
 - ◇ 測試
 - ◇ 執行

Python IDLE 編輯器



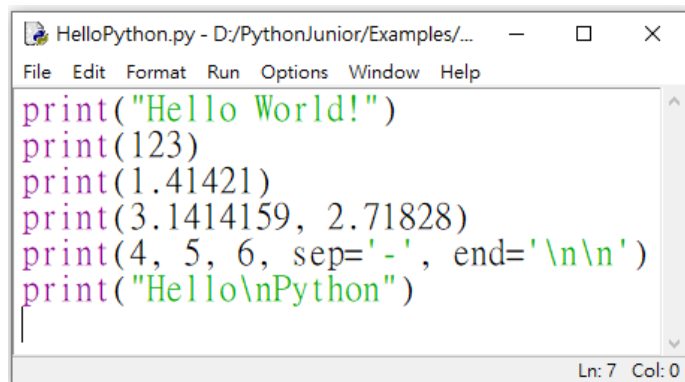
Python 螢幕輸出

- ◆ print() 函式可將括號中的的資料輸出在螢幕上
 - ◇ 資料可為整數、浮點數(含小數點)、文字字串
 - 文字字串以單引號'或雙引號"包夾
 - ◇ 列印結尾
 - python預設在輸出資料後自動換行，結尾預設設定為 "\n"
 - ◇ 列印間隔
 - 一次輸出多個資料用逗號隔開
 - 資料間預設設定為空一格 " "
 - ◇ Python 命令結束的時候不需要加上;

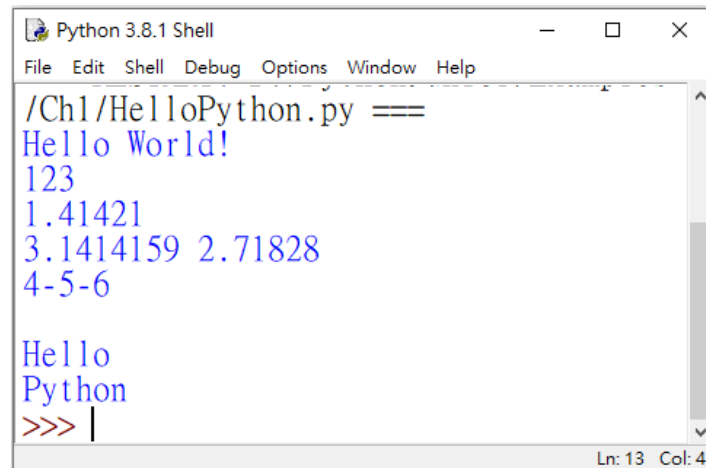
Python 螢幕輸出

- ◆ `print()` 函式參數進階設定
 - ◆ `sep = ' ... '` : 設定資料間的分隔符號
 - ◆ `end = ' ... '` : 設定句尾符號
 - ◆ 輸出後不想自動換行
 - 加上參數 `end = ''`

程式範例



```
File Edit Format Run Options Window Help
print("Hello World!")
print(123)
print(1.41421)
print(3.1414159, 2.71828)
print(4, 5, 6, sep='-', end='\n\n')
print("Hello\nPython")
Ln: 7 Col: 0
```



```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
/Ch1/HelloPython.py ==
Hello World!
123
1.41421
3.1414159 2.71828
4-5-6

Hello
Python
>>> |
Ln: 13 Col: 4
```

Python 註解

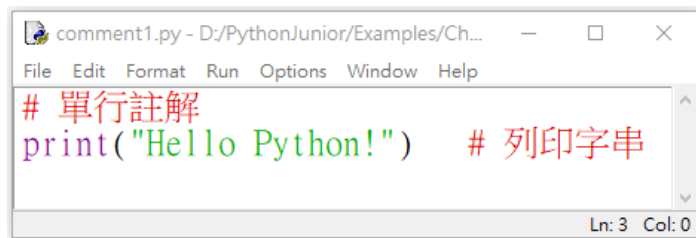
- ◆ 程式中附加說明文字
- ◆ 單行註解：
 - ◇ 使用 # 符號到該行結束
- ◆ 多行註解：
 - ◇ 使用一對三個引號將註解包起來
 - ◇ 可使用單引號或雙引號

```
# 這是單行註解  
# Python 不會處理
```

```
"""  
    這些是多行註解  
    Python 不會處理  
"""
```

```
"""  
    這也是多行註解  
    Python 不會處理  
"""
```


程式範例



```
comment1.py - D:/PythonJunior/Examples/Ch...  
File Edit Format Run Options Window Help  
# 單行註解  
print("Hello Python!") # 列印字串  
Ln: 3 Col: 0
```



```
comment2.py - D:/PythonJunior/Examples/Ch...  
File Edit Format Run Options Window Help  
'''  
程式範例comment2.py  
使用三個單引號當作註解  
'''  
print("Hello Python!") # 列印字串  
Ln: 6 Col: 0
```



```
comment3.py - D:\PythonJunior\Examples\Ch1...  
File Edit Format Run Options Window Help  
"""  
程式範例comment3.py  
使用三個單引號當作註解  
"""  
print("Hello Python!") # 列印字串  
Ln: 6 Col: 0
```

變數宣告

- ◆ Variable (變數)

- ◇ 定義程式運算時所需儲存的資料

- 變數名稱 = 初始值

- ◆ Python 是動態語言，宣告變數時不需指定資料型態資訊

- ◇ 型態是在執行時期的系統判斷建立物件

- ◆ Python 的變數必須先指定賦值才能作讀取使用

變數命名規則

- ◆ 可使用大小寫文字、數字或「_」
 - ◇ 開頭第一個字元不能是數字
 - ◇ Python 3 可以使用中文
- ◆ 不可與內建關鍵字同名
- ◆ 大小寫不同為不同的變數

and	as	assert
break	class	continue
def	del	except
else	elif	finally
for	from	False
global	if	import
in	is	lambda
not	None	nonlocal
or	pass	return
raise	try	True
with	while	yield

變數範圍

◆ 變數範圍

◇ 全域變數 Global Variable

- 變數未宣告在函式、方法或類別結構中
- 在整個程式檔案中有效

◇ 區域變數 Local Variable

◇ 類別欄位變數 Class Variable

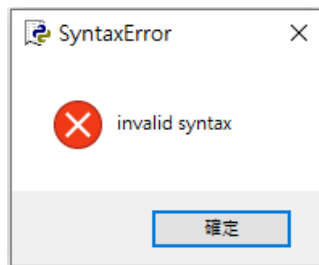
◆ 刪除變數

◇ del 變數名稱

程式範例

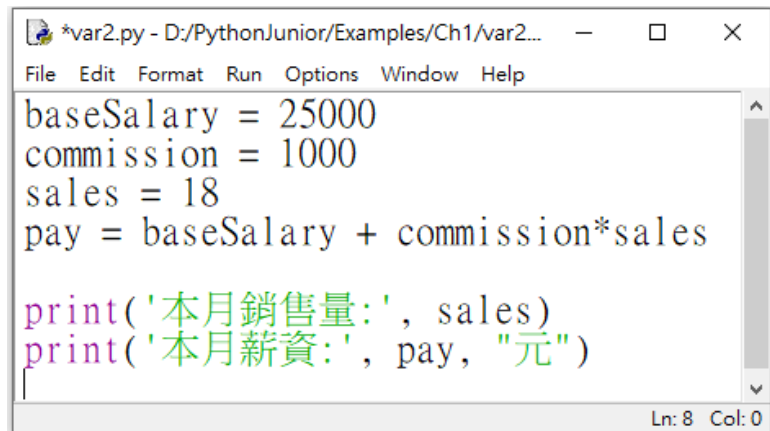
```
var1.py - D:/PythonJunior/Examples/Ch1/var1.py (3.8.1)
File Edit Format Run Options Window Help
age1 = 18
#2age = 15    #變數開頭第一個字元不能是數字
年齡3 = 12
#class = 602 #class是內建關鍵字

print(age1)
#print(2age)
print(年齡3)
#print(class)
Ln: 10 Col: 0
```



```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Examples/Ch1/var1.py
18
12
(>>> |
Ln: 17 Col: 4
```

程式範例

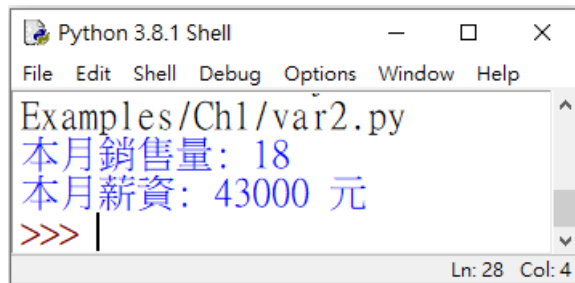


A screenshot of a Python IDE window titled '*var2.py - D:/PythonJunior/Examples/Ch1/var2...'. The window contains the following Python code:

```
baseSalary = 25000
commission = 1000
sales = 18
pay = baseSalary + commission*sales

print('本月銷售量:', sales)
print('本月薪資:', pay, "元")
```

The status bar at the bottom right indicates 'Ln: 8 Col: 0'.

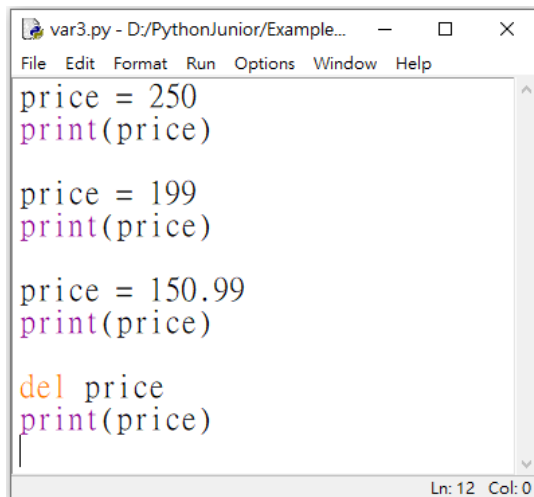


A screenshot of a Python 3.8.1 Shell window titled 'Python 3.8.1 Shell'. The window shows the output of the script executed in the previous image:

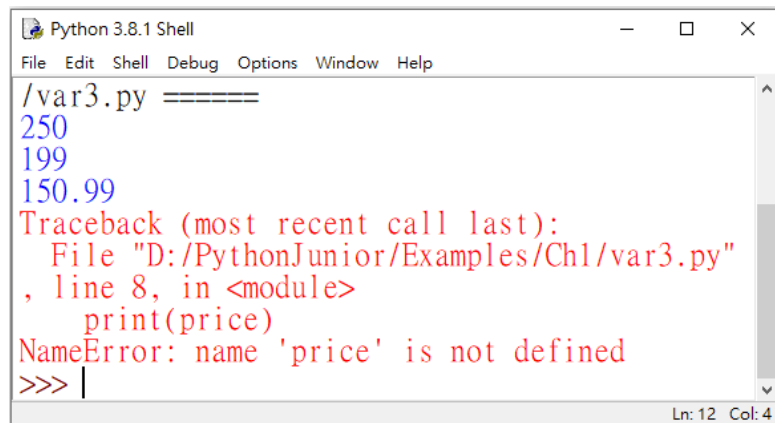
```
Examples/Ch1/var2.py
本月銷售量: 18
本月薪資: 43000 元
>>> |
```

The status bar at the bottom right indicates 'Ln: 28 Col: 4'.

程式範例



```
var3.py - D:/PythonJunior/Example...  
File Edit Format Run Options Window Help  
price = 250  
print(price)  
  
price = 199  
print(price)  
  
price = 150.99  
print(price)  
  
del price  
print(price)  
|  
Ln: 12 Col: 0
```



```
Python 3.8.1 Shell  
File Edit Shell Debug Options Window Help  
/var3.py =====  
250  
199  
150.99  
Traceback (most recent call last):  
  File "D:/PythonJunior/Examples/Ch1/var3.py"  
    , line 8, in <module>  
      print(price)  
NameError: name 'price' is not defined  
>>> |  
Ln: 12 Col: 4
```



Q : Python IDLE 使用哪一個工具作互動式開發？

- a) Shell
- b) Editor
- c) 以上皆可
- d) 以上皆非



Q : Python 程式存檔後檔案副檔名為何？

- a) .script
- b) .pyscript
- c) .py
- d) .python



Q : 下列哪一個是合法的變數名稱？

- a) pass
- b) _x5
- c) 9x
- d) x-1

課程內容

1. Python 簡介

1-1. Python 語言簡介

1-2. Python 執行環境

1-3. Python 開發環境

2. Python 程式基本概念

2-1. 開發模式

2-2. 螢幕輸出

2-3. 註解

2-4. 變數宣告

3. Python 運算

3-1. 數值運算

3-2. 字串運算

3-3. 資料輸入

3-4. 格式化輸出

數值運算

運算符號	說明	範例	
+	加法	9+2	=> 11
-	減法	9-2	=> 7
*	乘法	9*2	=> 18
/	真除法 (結果為浮點數)	9/2	=> 4.5
		9.0/2.0	=> 4.5
//	整數除法	9//2	=> 4
		9.0//2.0	=> 4.0
%	取餘數	9%2	=> 1
**	指數	9**2	=> 81
<<	二進位左移	9<<2	=> 36
>>	二進位右移	9>>2	=> 2

Python 除法

◆ Python 除法

◆ 整數相除(/ 或 //)

- / 運算結果為浮點數

```
10 / 3    >> 3.3333333333333335
```

- //運算結果為整數

```
10 // 3   >> 3
```

◆ 浮點數相除

- / 會保留小數部份

```
10 / 3.0  >> 3.3333333333333335
```

- // 則會截去小數部份

```
10 // 3.0 >> 3.0
```

◆ 浮點數是2進位近似值

數值運算優先順序

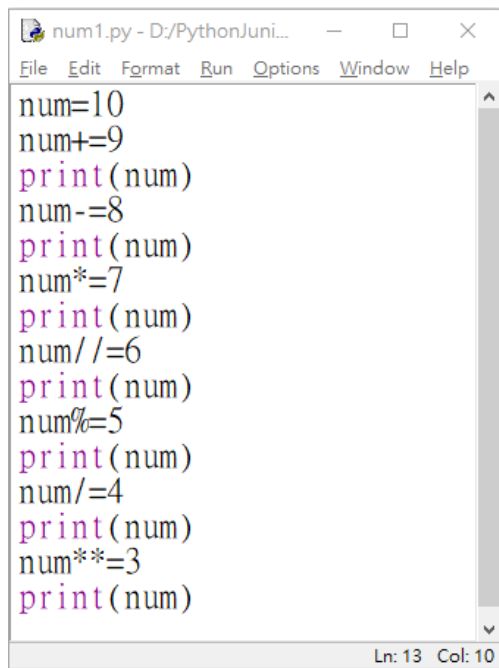
◆ 數值運算計算順序

1. 小括弧優先
2. 次方
3. 先乘除後加減
4. 由左至右進行

賦值運算(Assignment Operator)

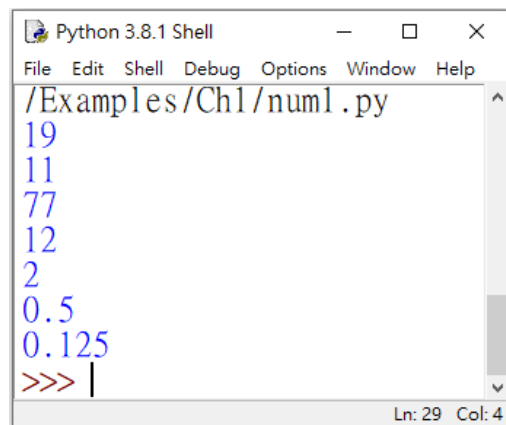
運算子	說明	範例
=	基本指定	$a = b$
+=	加法指定	$a += b \rightarrow a = a + b$
-=	減法指定	$a -= b \rightarrow a = a - b$
*=	乘法指定	$a *= b \rightarrow a = a * b$
/=	除法指定	$a /= b \rightarrow a = a / b$
//=	整數除法指定	$a //= b \rightarrow a = a // b$
%=	餘數指定	$a \% = b \rightarrow a = a \% b$
**=	指數指定	$a ** = b \rightarrow a = a ** b$

程式範例



```
num=10
num+=9
print(num)
num-=8
print(num)
num*=7
print(num)
num//=6
print(num)
num%=5
print(num)
num/=4
print(num)
num**=3
print(num)
```

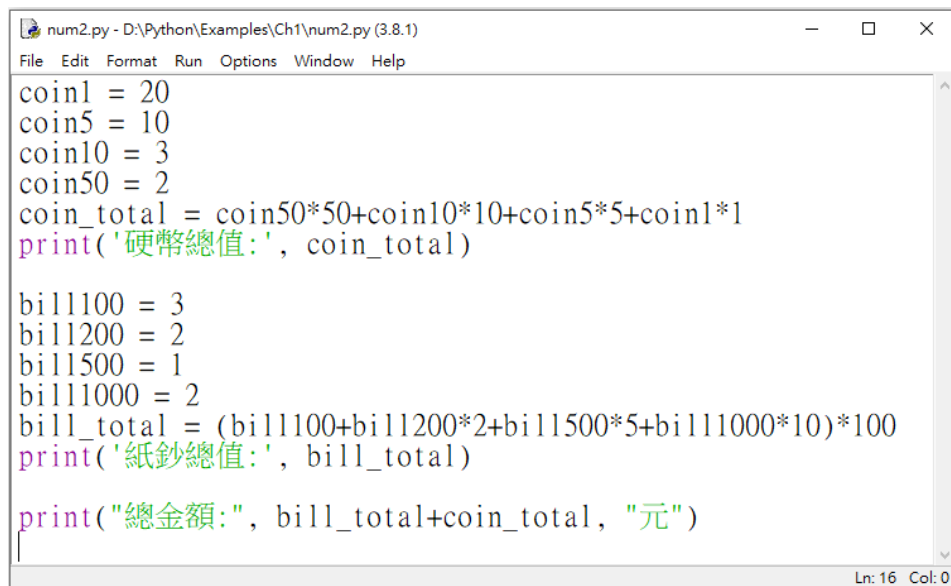
Ln: 13 Col: 10



```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
/Examples/Ch1/num1.py
19
11
77
12
2
0.5
0.125
>>> |
```

Ln: 29 Col: 4

程式範例



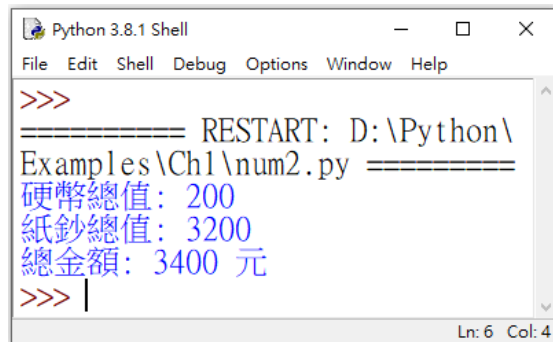
```
num2.py - D:\Python\Examples\Ch1\num2.py (3.8.1)
File Edit Format Run Options Window Help

coin1 = 20
coin5 = 10
coin10 = 3
coin50 = 2
coin_total = coin50*50+coin10*10+coin5*5+coin1*1
print('硬幣總值:', coin_total)

bill100 = 3
bill200 = 2
bill500 = 1
bill1000 = 2
bill_total = (bill100+bill200*2+bill500*5+bill1000*10)*100
print('紙鈔總值:', bill_total)

print("總金額:", bill_total+coin_total, "元")

Ln: 16 Col: 0
```



```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help

>>>
===== RESTART: D:\Python\
Examples\Ch1\num2.py =====
硬幣總值: 200
紙鈔總值: 3200
總金額: 3400 元
>>> |

Ln: 6 Col: 4
```



Q : 下列程式運算後的輸出為何？

```
>>> x = 10
```

a) 10

```
>>> x //=10
```

b) 90

```
>>> print(x)
```

c) 1

d) 100



Q : 下列程式運算後的輸出為何？

```
>>> x = 9 + 3 * 2 ** 2 - 1
```

```
>>> print(x)
```

a) 575

b) 44

c) 20

d) 15



練習：數值運算練習

- ◆ 請撰寫一個程式，計算攝氏0、25、98度，分別等於華氏幾度？
 - ◇ 攝氏溫度 (C) 轉換為華氏溫度 (F) 的公式為： $F=9/5*C+32$ 。
 - ◇ 使用 / 與 // 的差別

文字資料運算

- ◆ 字串可用「+」符號進行串接運算

'Hello' + ' World' >> 'Hello World'

- ◆ 字串可用「*」符號進行複製運算

'Hello ' * 3 >> 'Hello Hello Hello '

Python 加法比較

◆ 數字加法

`2 + 3` `>> 5`

◆ 運算結果為數字

◆ 字串加法

`'2' + '3'` `>> '23'`

◆ 運算結果為字串串接

◆ 數字加字串

`2 + '3'` `>> TypeError`

◆ 會發生錯誤

◆ `int()`, `float()`：字串轉換為數字

`2 + int('3')` `>> 5`

◆ `str()`：數字轉換為字串

`str(2) + '3'` `>> '23'`

變數資料交換

◆ Python 允許直覺的變數交換語法

```
int a = 3;  
int b = 4;  
  
int temp = a;  
a = b;  
b = temp;
```

C / Java

```
a = 3  
b = 4  
  
a, b = b, a
```

Python

程式範例

```
str1.py - D:/PythonJunior/Examples... - □ ×
File Edit Format Run Options Window Help

title='先生'
lastName='鄭'
greet='您好'
print(lastName+title+greet)

thank="感謝"
very="非常"
end="!"
print(very*3+thank+end)

title='Ms. '
lastName='Louis'
greet='Hello, '
print(greet+title+lastName)

thank="Thanks, "
very="very "
end="much!"
print(thank+very*3+end)

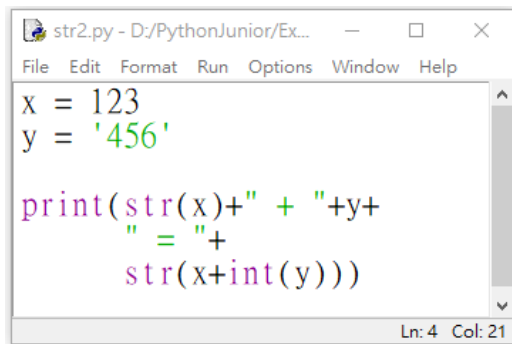
Ln: 20 Col: 0
```

```
Python 3.8.1 Shell - □ ×
File Edit Shell Debug Options Window Help

>>>
= RESTART: D:/PythonJunior/Exam
ples/Ch1/str1.py
鄭先生您好
非常非常非常感謝!
Hello, Ms. Louis
Thanks, very very very much!
>>> |

Ln: 44 Col: 4
```


程式範例

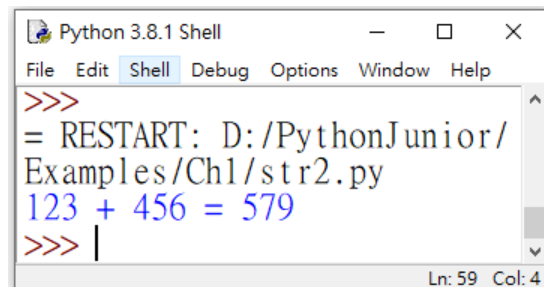


A screenshot of a Python IDE window titled "str2.py - D:/PythonJunior/Ex...". The window contains the following Python code:

```
x = 123
y = '456'

print(str(x)+" + "+y+
      " = "+
      str(x+int(y)))
```

The status bar at the bottom indicates "Ln: 4 Col: 21".



A screenshot of a "Python 3.8.1 Shell" window. The window shows the output of running the script from the previous image:

```
>>>
= RESTART: D:/PythonJunior/
Examples/Ch1/str2.py
123 + 456 = 579
>>> |
```

The status bar at the bottom indicates "Ln: 59 Col: 4".

取得使用者資料輸入

- ◆ `input('提示字串')` 函式可取得使用者資料輸入
- ◆ 取得的資料型態為字串
 - ◇ 查詢變數的資料型態
 - `type(變數)` 函式
- ◆ 資料轉型
 - ◇ `int()` 函式將字串轉換為整數
 - ◇ `float()` 函式將字串轉換為浮點數



練習：輸入成績與取得總分

- ◆ 設計一個輸入國文英文數學成績後再計算總分的程式。

```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Solutions\Ch1\score.py
請輸入國文成績:90
請輸入數學成績:70
請輸入英文成績:80
總成績: 240
平均分數: 80.0
>>> |
Ln: 31 Col: 4
```

取得使用者輸入練習

- ◆ 修改攝氏溫度轉換華氏溫度程式
 - ◇ 讀取使用者輸入的攝氏溫度，顯示對應的華氏溫度
 - ◇ 攝氏溫度 (C) 轉換為華氏溫度 (F) 的公式為： $F=9/5*C+32$ 。

練習：BMI 計算

- ◆ 請設計一個 BMI 計算題目
 - ◆ 公式為「體重 (公斤) 除以身高 (公尺) 的平方」。
 - ◆ 顯示BMI數值

資料格式化輸出

◆ 資料格式化輸出：

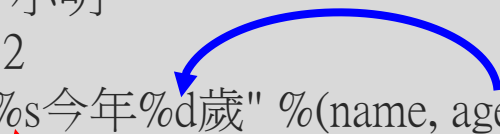
◆ 指定輸出資料型態及格式

◆ `print("輸出字串與格式代碼" % (變數))`

- 多個變數則用逗點隔開。
- 資料型態可區分為整數、浮點數與文字三種

```
name="小明"  
age = 12  
print(name, "今年", age, "歲", sep="")
```

```
name="小明"  
age = 12  
print("%s今年%d歲" %(name, age))
```



資料格式化輸出

◆ 整數資料格式代碼：

◇ 十進位、八進位與十六進位

整數資料格式代碼	輸出方式
%d、%i	以十進位方式輸出
%o	以八進位方式輸出
%x、%X	以十六進位方式輸出

資料格式化輸出

◆ 文字資料的格式代碼：

文字資料格式代碼	輸出方式
%s	以字串方式輸出
%c	以字元方式輸出

整數與字元與字串的輸出

- ◆ %d、%c 與 %s 的 % 符號與後面格式代碼之間可以加入一個整數值，代表資料輸出位數：
 - ◇ 若整數的絕對值大於實際資料位數：
 - 若是正數則向右對齊。
 - 若是負數則向左對齊。
 - 加上0則數字向右對齊，左邊補0
 - ◇ 若整數的絕對值小於實際資料位數，則資料仍依照實際位數輸出。

資料格式化輸出

◆ 浮點數的格式代碼：

格式代碼	說明
%f、%F	輸出浮點數
%e、%E	科學記數法
%g、%G	自己判斷 %f 或著 %e 輸出

資料格式化輸出

◆ 浮點數代表有小數值的數字

◇ 有一般顯示與科學計數法兩種方式：

- 一般顯示只會顯示小數位六位。
- 科學計數法可顯示極大或極小的數值。

◇ 例如 34.5678912345

- 一般顯示：34.567891
- 科學計數法：3.456789e+01

浮點數資料輸出

◆ %f 的 % 符號與後面代碼之間可以加入「n.m」兩個整數值

◇ m 代表小數位數：

- 大於實際小數位數，補0。
- 小於實際小數位數，依下一位數的值四捨五入。

◇ n 的絕對值代表整數加上小數加上「.»的總和：

- 若 n 的絕對值大於資料輸出位數：
 - ▶ 正數則向右對齊，負數則向左對齊。
- 若 n 的絕對值小於實際資料位數，資料依照實際位數輸出。



練習：輸入成績與取得總分

- ◆ 修改成績計算程式
 - ◇ 平均分數顯示小數點後兩位

```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Solutions\Ch1\score.py
請輸入國文成績:90
請輸入數學成績:72
請輸入英文成績:80
總成績: 242
平均分數: 80.67
>>> |
Ln: 52 Col: 4
```

練習：BMI 計算

◆ 修改 BMI 計算程式

- ◆ 公式為「體重 (公斤) 除以身高 (公尺) 的平方」。
- ◆ BMI數值顯示小數點後兩位



巨匠直播教學

www.pcschoolonline.com.tw