

Python 初學語法－IF，FOR

報告人 - 丁俊齊

指導老師 - 王文彥

Contents

1. 介紹
2. 平台安裝
3. 語法 – Python
 - IF
 - FOR

1.

介紹

Python

人生苦短，我用Python

- 直譯式、進階和通用的程式語言
- 支援多種程式設計範式，包括函數式、指令式、結構化、物件導向和反射式程式。
- 可以通過內建的擴充程式商店安裝擴充以拓展軟體功能。



我為什麼要學Python

1. Python的動態語法

範例:

別的語言

```
int number = 100;  
String str = "100";
```

Python

```
number = 100;  
string = "100";
```

2. Python的強制縮排

Python在運行時不縮排無法運行

Python的強制縮排讓新手可以慢慢地變成習慣

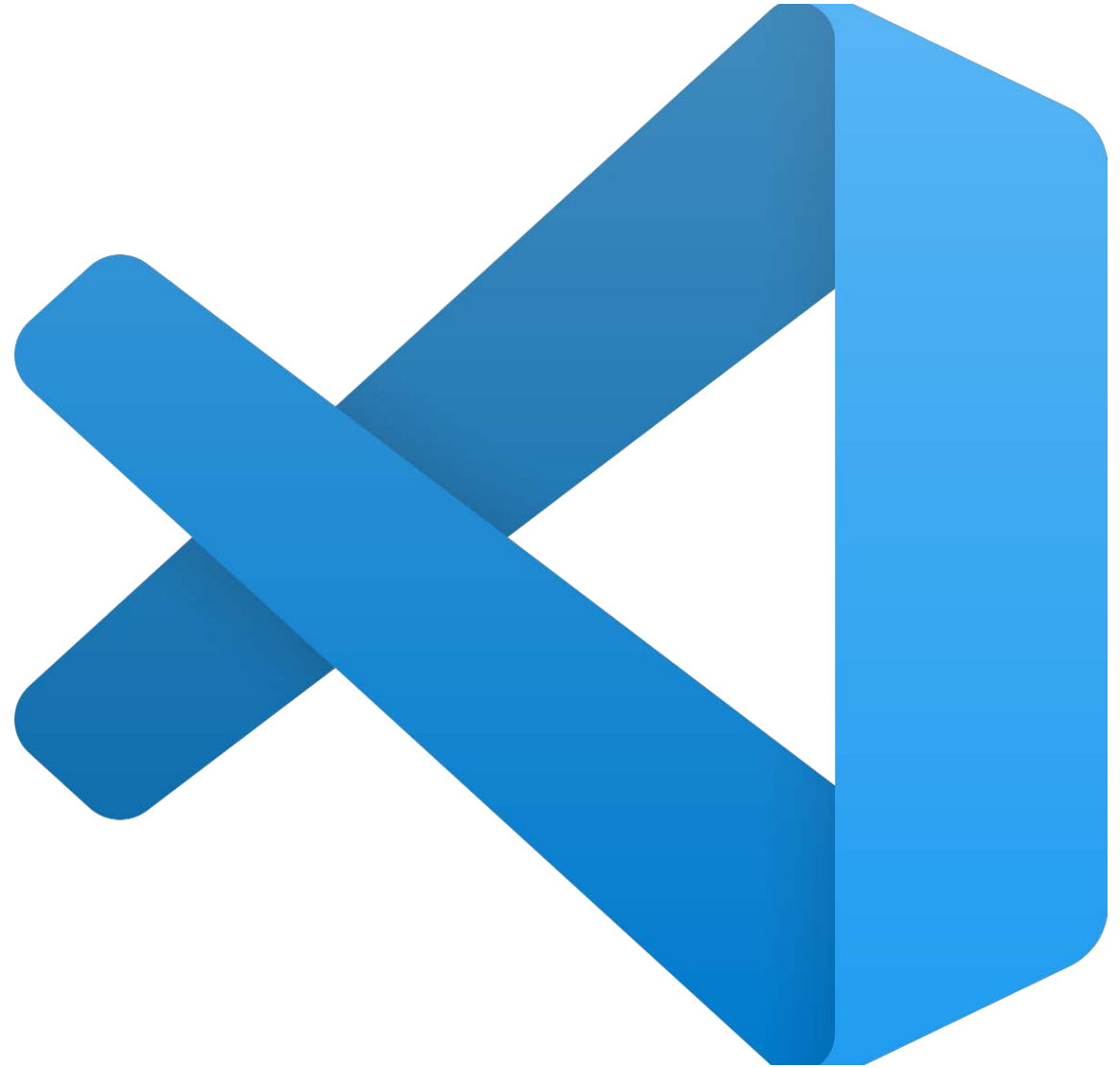
可使用平台

- google colab
- Visual Studio Code
- Pycharm
- Jupyter notebook
- Sublime Text
- Spider



Visual Studio Code

- 微軟開發且跨平台的免費原始碼編輯器。
- 該軟體支援語法突顯、代碼自動補全、代碼重構、檢視定義功能。
- 可以通過內建的擴充程式商店安裝擴充以拓展軟體功能。

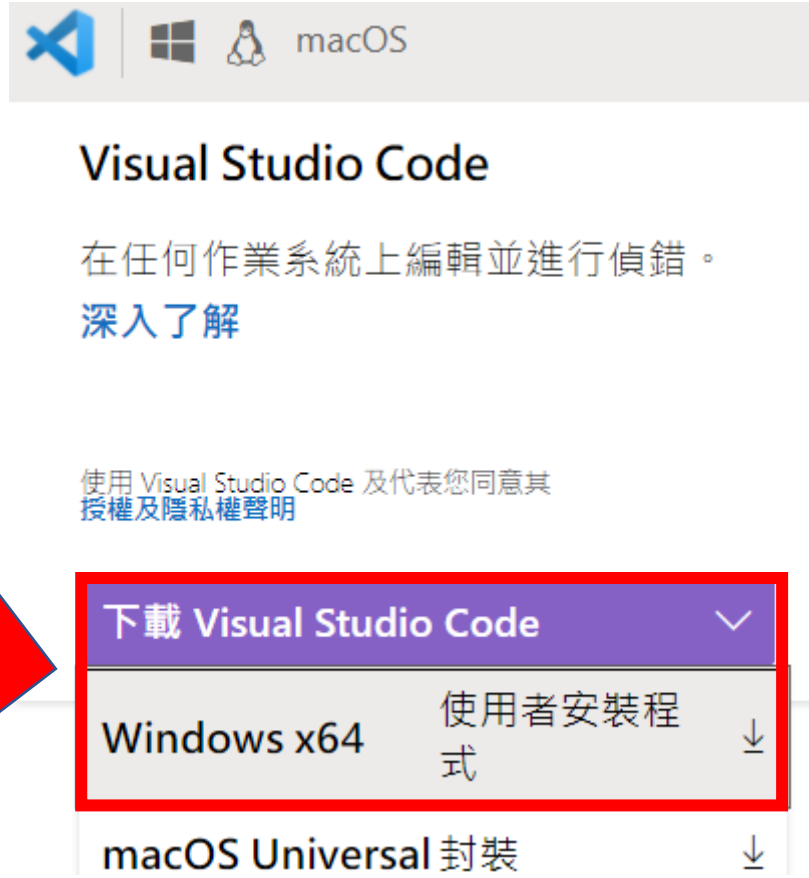


2.

平台安裝

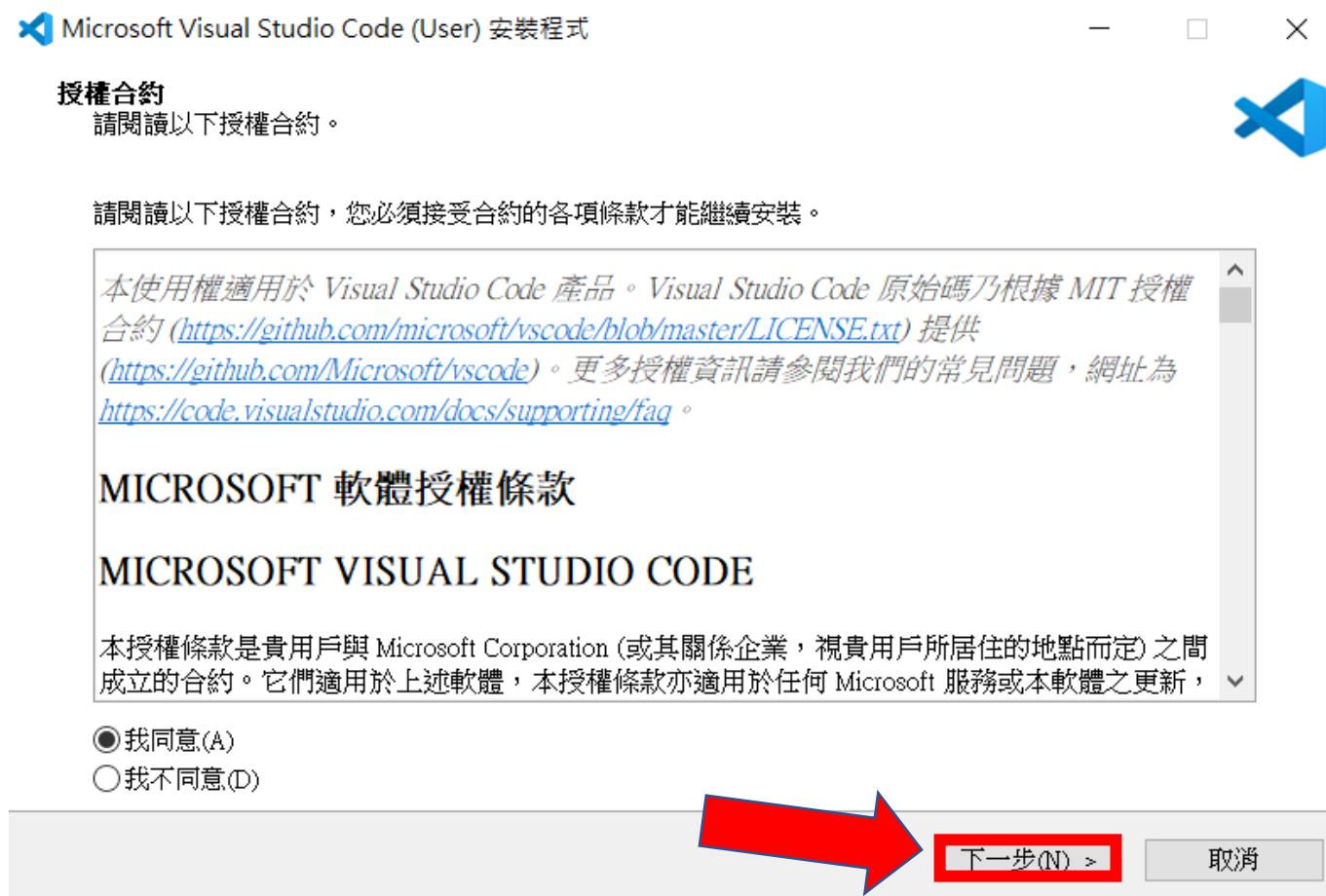
Visual Studio Code 安裝

進入官網下載 <https://visualstudio.microsoft.com/zh-hant/> 程式



Visual Studio Code 安裝

安裝程式-1



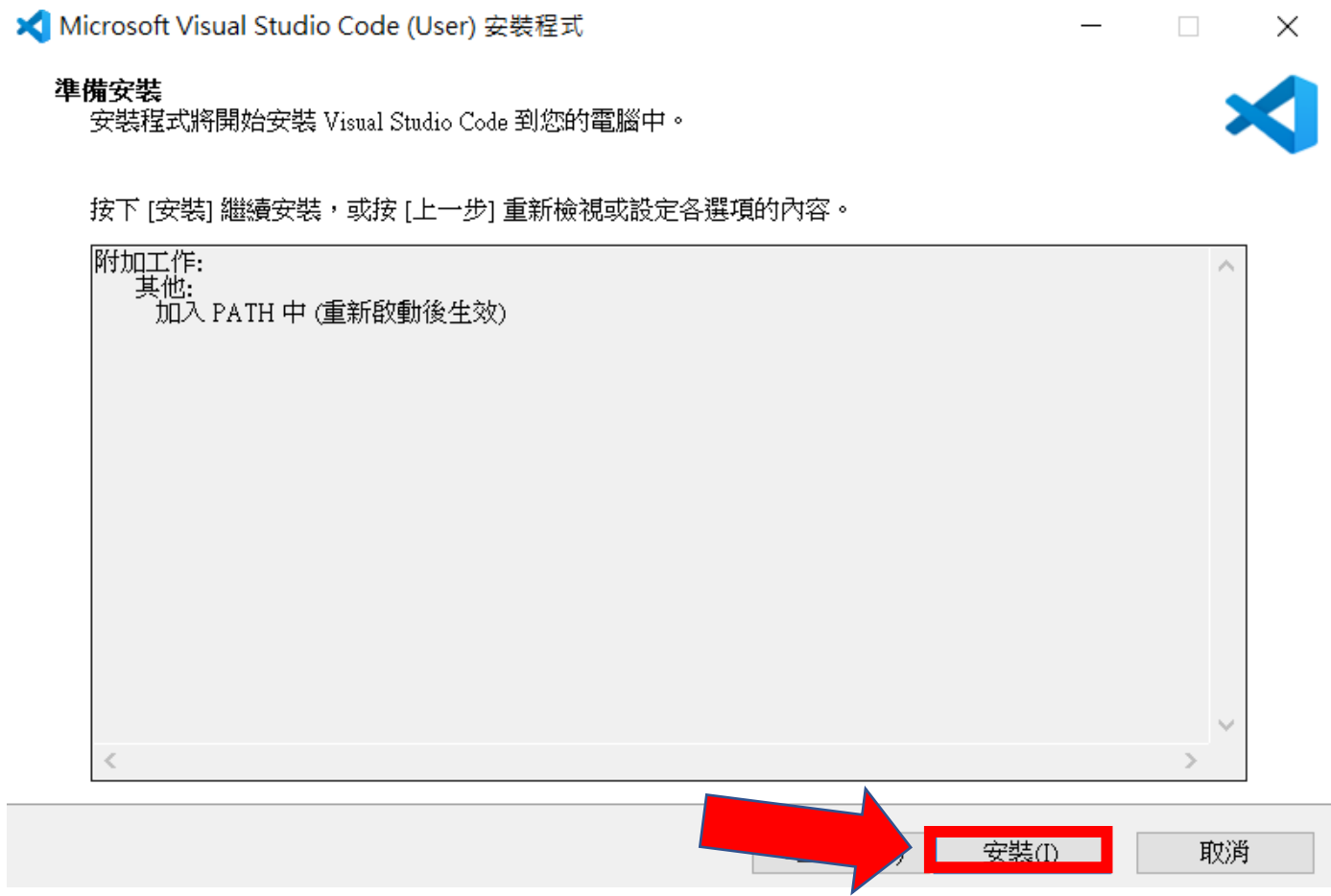
Visual Studio Code 安裝

安裝程式-2



Visual Studio Code 安裝

安裝程式-3



Visual Studio Code 安裝

安裝程式-4

Microsoft Visual Studio Code (User) 安裝程式

安裝完成

安裝程式已經將 Visual Studio Code 安裝在您的電腦中，您可以選擇程式的圖示來執行該應用程式。

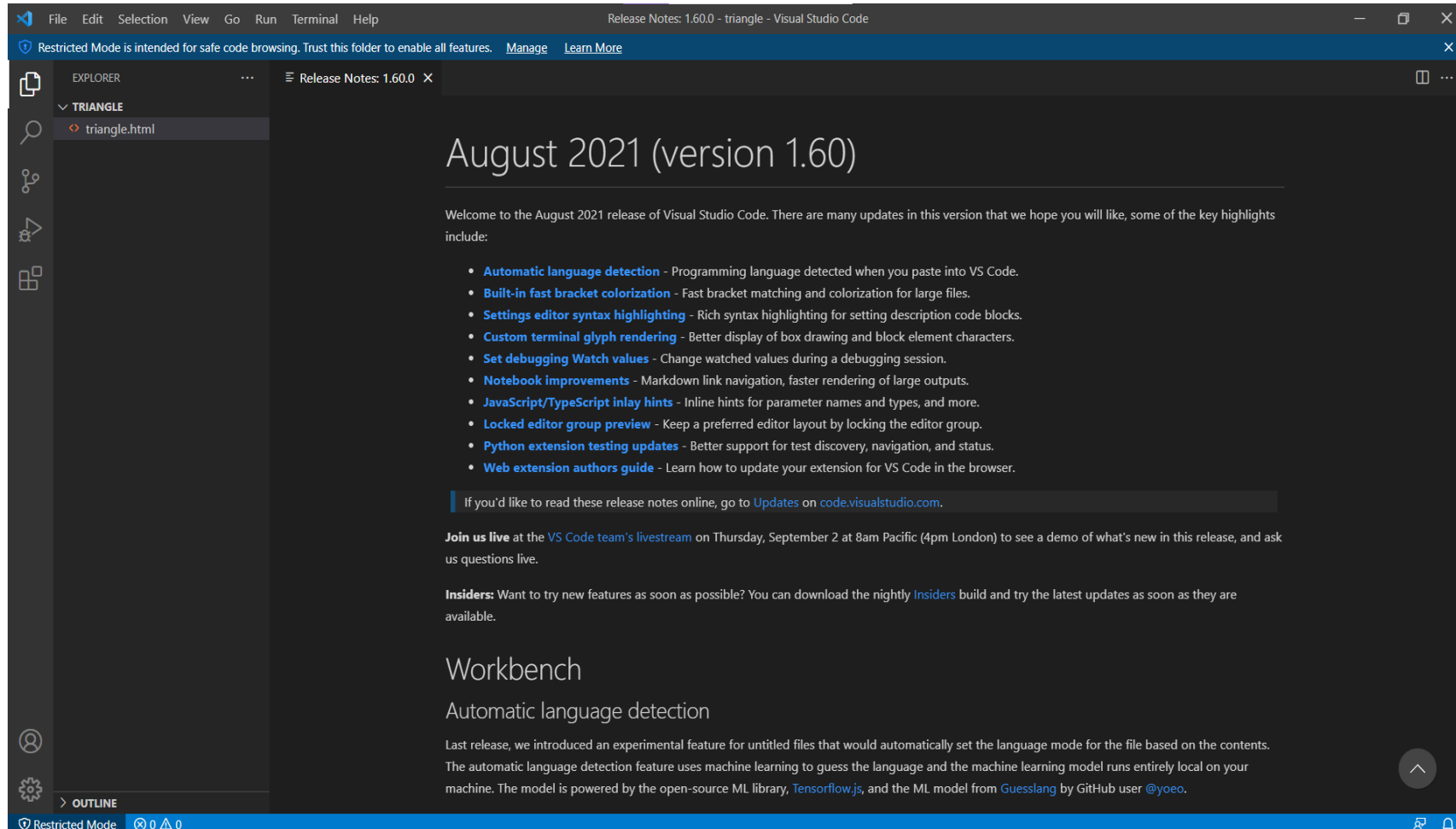
按 [完成] 以結束安裝程式。

☒ 啟動 Visual Studio Code



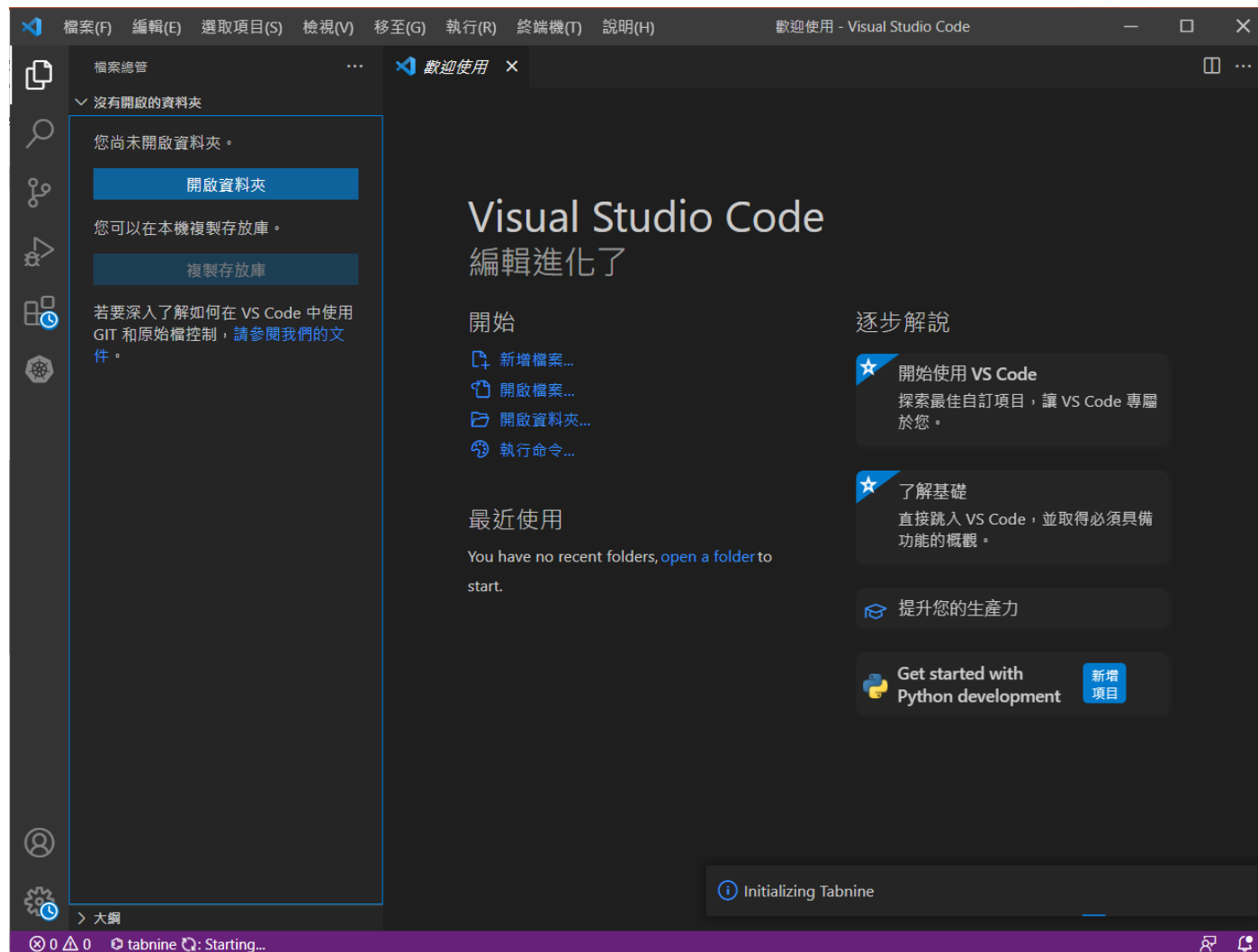
完成(F)

Visual Studio Code 安裝 完成



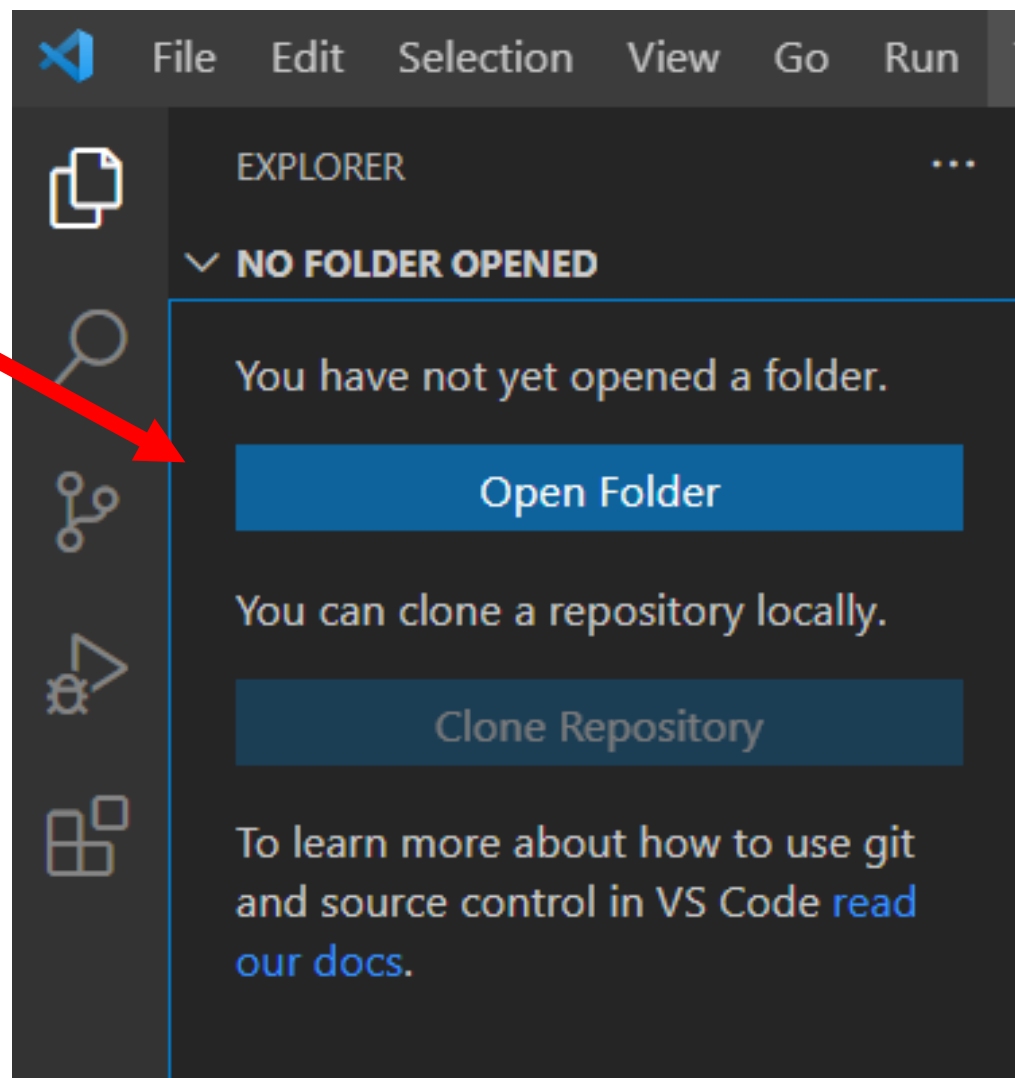
功能介紹

開啟檔案資料夾
搜索檔案內指令
平常用不到
偵錯
裝擴充套件
平常用不到



開啟檔案

選取所在資料夾 不然python預設資料夾為C槽

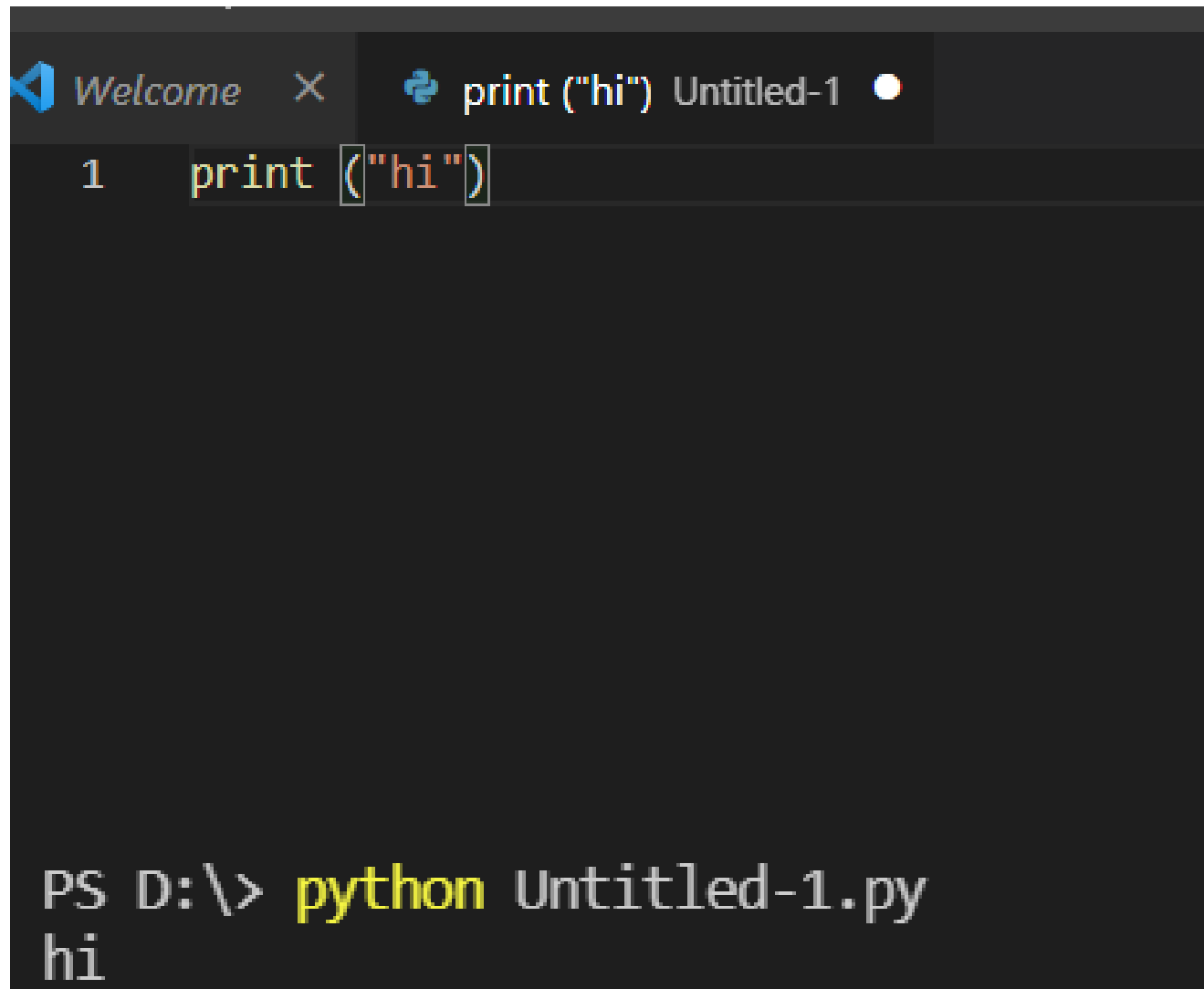


執行方式

先開啟一個終端

執行方式

- Python 檔名
執行區要在檔案所在資料夾



The image shows a code editor window with a dark theme. The top bar has two tabs: 'Welcome' and 'print ("hi") Untitled-1'. The code editor contains a single line of Python code: `1 print ("hi")`. Below the code editor is a terminal window. The terminal shows the command `PS D:\> python Untitled-1.py` being executed, followed by the output `hi`.

```
Welcome X print ("hi") Untitled-1
1 print ("hi")

PS D:\> python Untitled-1.py
hi
```

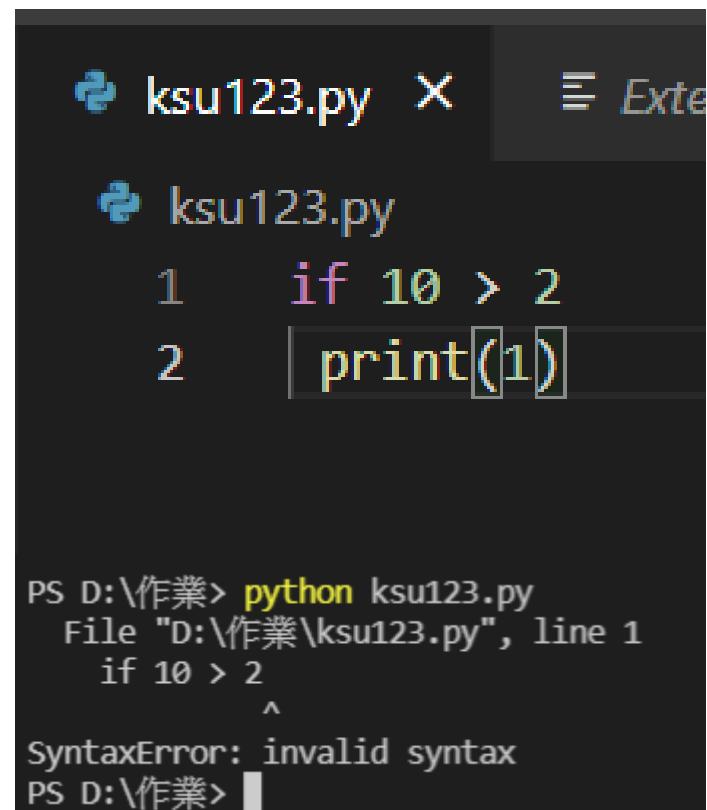
簡易除錯

- 如果有錯可看執行欄位錯誤資訊

例:右邊沒加上冒號

下方為沒縮排

```
SyntaxError: invalid syntax
PS D:\作業> python ksu123.py
File "D:\作業\ksu123.py", line 2
    print(1)
    ^
IndentationError: expected an indented block
```



```
ksu123.py X  Ext
ksu123.py
1  if 10 > 2
2  print(1)

PS D:\作業> python ksu123.py
File "D:\作業\ksu123.py", line 1
    if 10 > 2
            ^
SyntaxError: invalid syntax
PS D:\作業> 
```

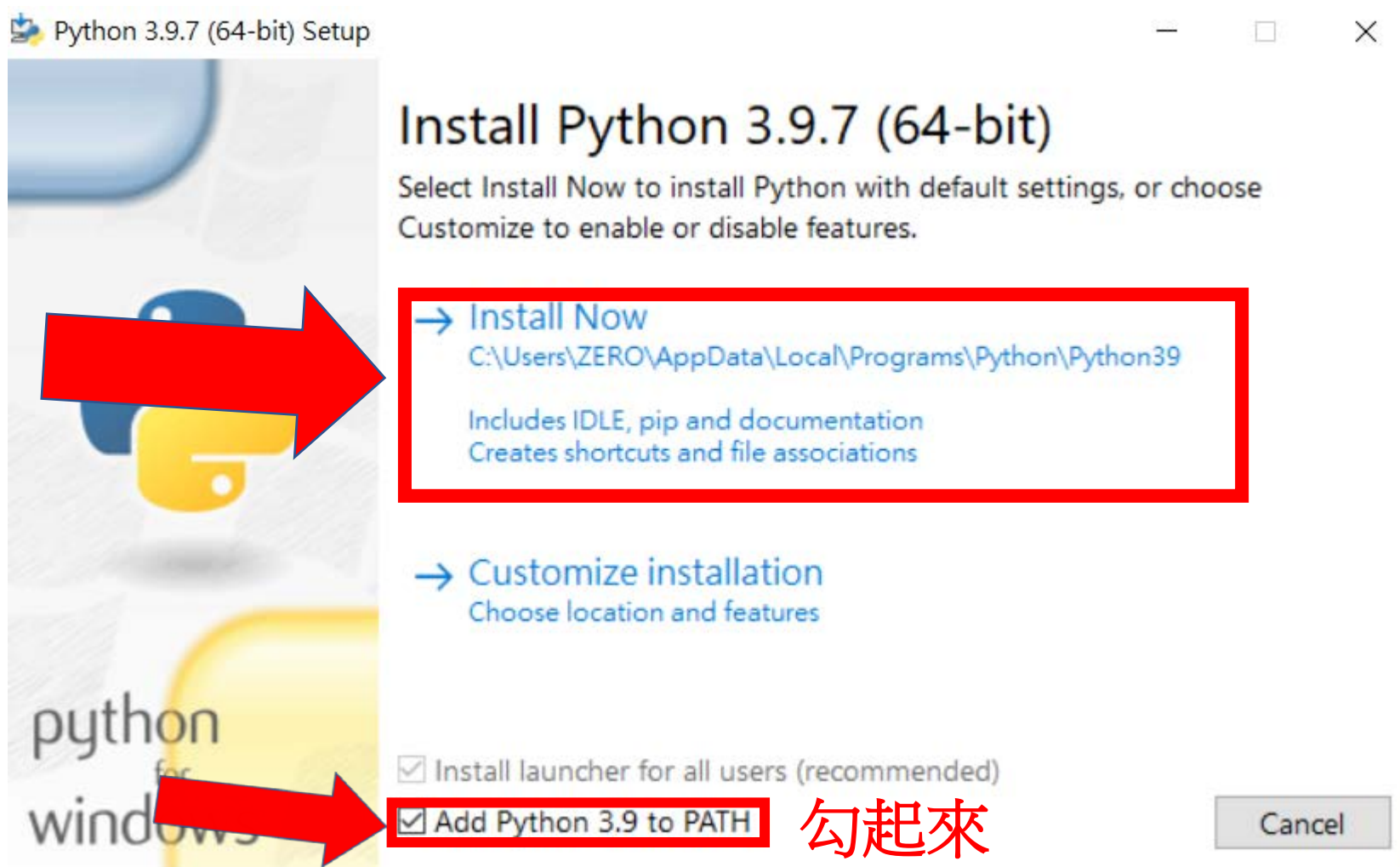
Python安裝

Python官網(<https://www.python.org/downloads/>)



Python安裝

安裝程式-1



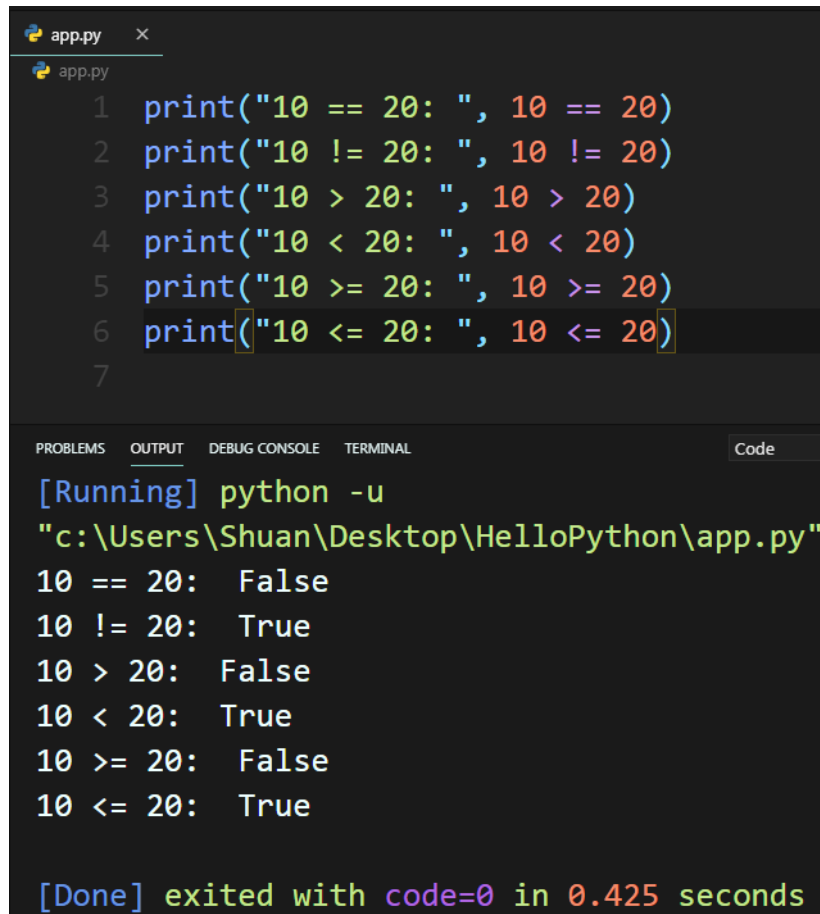
3.

語法

Python條件判斷 - IF

比較運算子 能比較但不能用來執行

需使用IF搭配



The screenshot shows a Python IDE with a file named 'app.py'. The code in the editor consists of seven lines, each using a print statement to output a comparison between the number 10 and 20 using different operators. The output window below the code shows the results of these comparisons: 10 == 20 is False, 10 != 20 is True, 10 > 20 is False, 10 < 20 is True, 10 >= 20 is False, and 10 <= 20 is True. The IDE also shows the command 'python -u' being run and the file path 'c:\Users\Shuan\Desktop\HelloPython\app.py'.

```
app.py x
app.py
1 print("10 == 20: ", 10 == 20)
2 print("10 != 20: ", 10 != 20)
3 print("10 > 20: ", 10 > 20)
4 print("10 < 20: ", 10 < 20)
5 print("10 >= 20: ", 10 >= 20)
6 print("10 <= 20: ", 10 <= 20)
7

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Code
[Running] python -u
"c:\Users\Shuan\Desktop\HelloPython\app.py"
10 == 20: False
10 != 20: True
10 > 20: False
10 < 20: True
10 >= 20: False
10 <= 20: True

[Done] exited with code=0 in 0.425 seconds
```

Python條件判斷 - IF

輸入一個變數，讓他可以進行判斷

如果 $m > 90$ 執行下面動作

記得冒號一定要打

縮排一定要縮

```
1  m = 100
2  if m > 90:
3      print("不買，好貴")
4  else:
5      print("買")
```

不買，好貴

Python條件判斷 - IF

else ?

else是如果不符合的情況才會觸發
如果不填else 不符合時就不會有動作

```
1 m = 20
2 if m > 90:
3     print("不買, 好貴")
4 else:
5     print("買")
```

買

```
1 m = 20
2 if m > 90:
3     print("不買, 好貴")
4
```

Python條件判斷 - IF

如果想判斷更多？

如果判斷不只是兩個
就用elif

elif = 不是 但如果

```
1  m = 70
2  if m > 90:
3      print("不買，好貴")
4  elif m > 60:
5      print("還是有點貴")
6  else:
7      print("買")
8
```

還是有點貴

Python條件判斷 - IF

如果想判斷別的？

```
c = True
k = True
if c and k:
    print("通過")
elif c or k:
    print("還行")
```

```
1  if not 10 == 20:
2      print("不通過")
3  else:
4      print("通過")
5
```

不通過

當然後面也可以處理不只一個指令

當a沒超過20 就直接輸出

如果a超過20，但沒超過50，+10輸出文字跟值

如果都超過，+30輸出文字跟值

```
a = 100
✓ if a < 20:
    print(a,"元")
✓ elif a < 50 :
    a = a + 10
    print("含運費",a,"元")
✓ else:
    a = a + 30
    print("超重")
    print("含運費",a,"元")
```

```
PS D:\作業> python ksu123.py
```

```
超重
```

```
含運費 130 元
```

loop

可以幫你執行類似的重複事情。

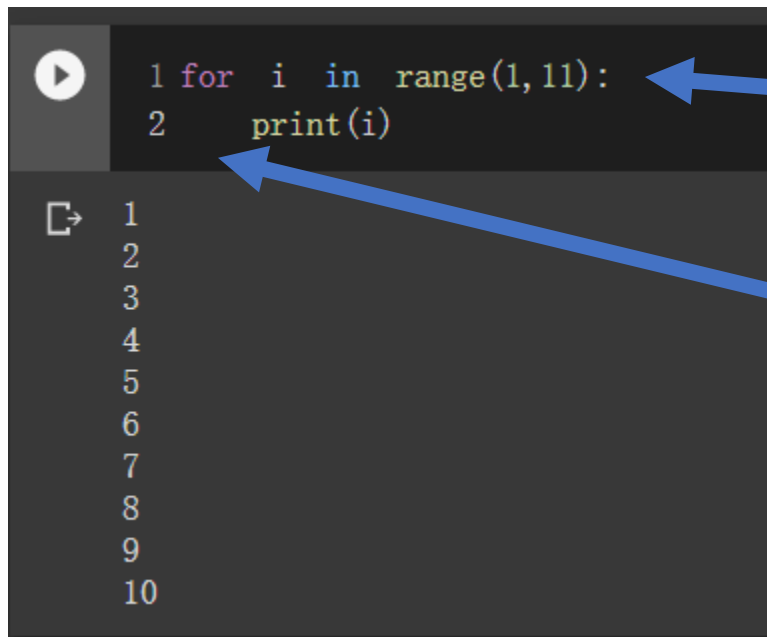
結果一樣

```
>>> print(1)
>>> print(2)
>>> print(3)
>>> print(4)
>>> print(5)
>>> print(6)
>>> print(7)
>>> print(8)
>>> print(9)
>>> print(10)
```

```
>>> for i in range(1,11):
>>>     print(i)
```

```
>>> count = 1
>>> while count < 11:
>>>     print(count)
>>>     count += 1
```

for loop



```
1 for i in range(1, 11):  
2     print(i)
```

Output:

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```

The screenshot shows a code editor with a dark background. The first part shows two lines of Python code: a for loop that iterates over the range from 1 to 11, and a print statement inside the loop. The second part shows the output of the code, which is the numbers 1 through 10, each on a new line. Two blue arrows point from the Chinese text annotations to the code: one points to the colon at the end of the first line, and the other points to the indentation of the second line.

Loop後面要加冒號

記得縮排

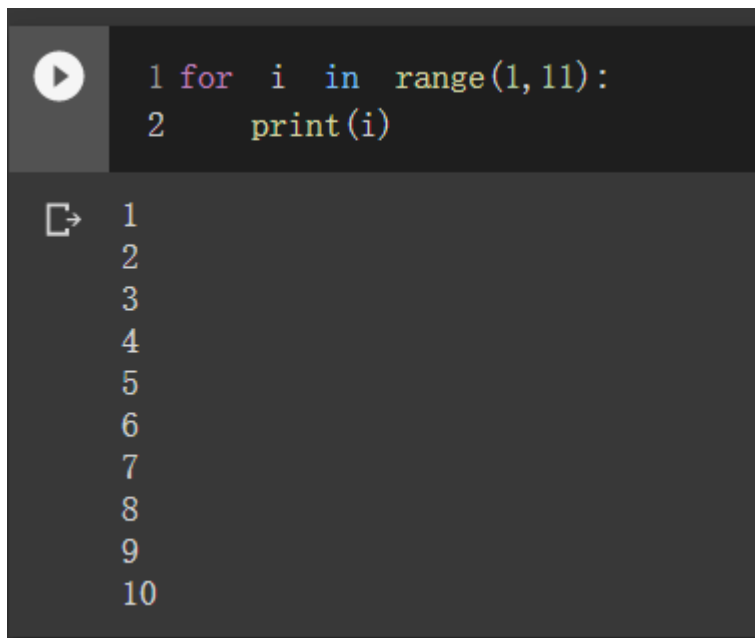
建立一個變數 *i*，依序塞入 1 - 11 塞入一個列印一個

for loop

In ?

in 是程式語言中常見的運算子
基本上使用 **for** 陳述句都會與 **in** 運算子搭配

in 運算子的工作是幫助 **for** 陳述句界定迭代範圍
in 就代表在什麼範圍內的意思。

A screenshot of a code editor showing a Python for loop. The code is: 1 for i in range(1, 11): 2 print(i). The output of the code is displayed below the code block, showing the numbers 1 through 10, each on a new line. The code is written in a dark-themed editor with syntax highlighting: 'for' is blue, 'in' is blue, 'range' is green, '1, 11' is green, 'print' is green, and 'i' is blue. The output is in a plain white font on a dark background.

```
1 for i in range(1, 11):  
2     print(i)
```

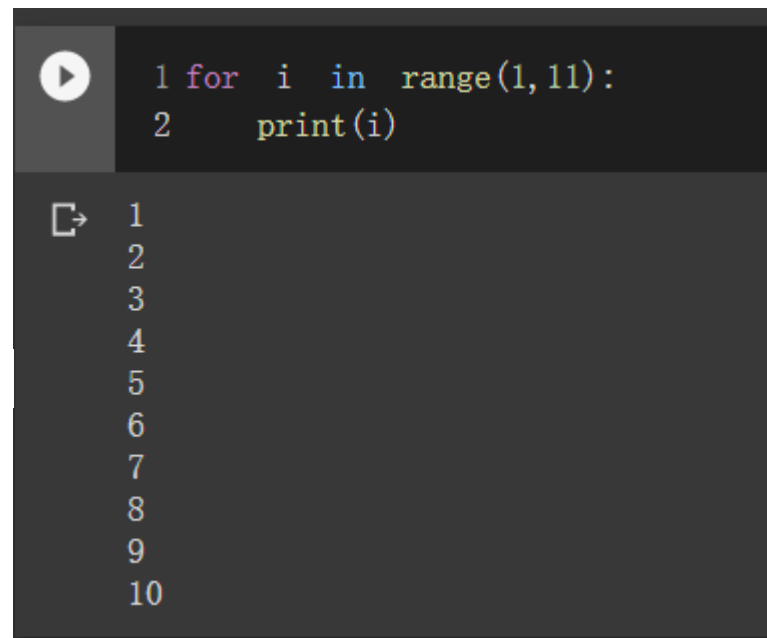
1
2
3
4
5
6
7
8
9
10

練習 1: for loop

迭代範圍?

`range()` 是 `for` 陳述句常用的迭代範圍。
`in` 代表在什麼範圍內，
而迭代的範圍就接在 `in` 的後方，
這裡的例子以 `range(1, 11)` 表示。

迭代簡單來說就是 重複進行 反覆運算

A screenshot of a Python code editor with a dark background. The top section shows two lines of code: '1 for i in range(1, 11):' and '2 print(i)'. The first line is highlighted in blue. Below the code, there is a list of numbers from 1 to 10, each on a new line, representing the output of the loop. To the left of the numbers is a small icon of a square with an arrow pointing right.

```
1 for i in range(1, 11):  
2     print(i)
```

1
2
3
4
5
6
7
8
9
10

import

如果要使用不只python擴充功能
基本都要用import匯入指令
才能使用

```
#!/usr/bin/env python
"""Django's command-line utility for administrative tasks."""
import os
import sys

def main():
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'mylinebot.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)

if __name__ == '__main__':
    main()
```

input

Input 用於使用者自己輸入
輸入的值會被帶入處理

```
1 name = input("輸入名稱")
2 print("你好",name)
```

輸入名稱11
你好 11

練習 2

使用**FOR**迴圈，輸入整數值n，程式計算 $1+2+.....+n$