STLectures
小孫學堂

Basic Coding

Python

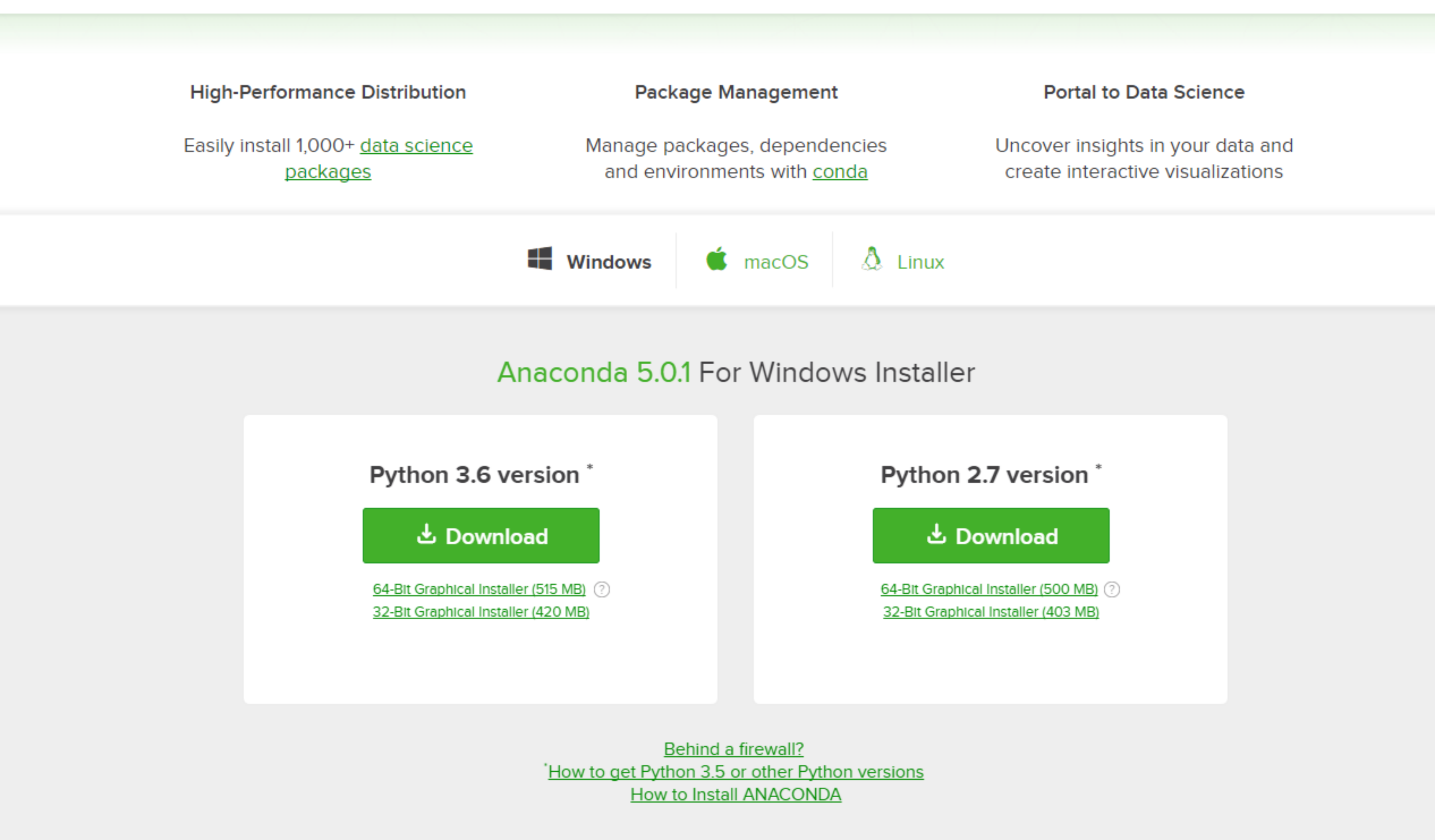
Created by 孫善堂 【小孫學堂】

Welcome to Python!

Python is an easy to learn programming language. You can use it to create web apps, games, even a search engine!

Install Anaconda

https://www.anaconda.com/download/



High-Performance Distribution

Easily install 1,000+ data science packages

Package Management

Manage packages, dependencies and environments with conda

Portal to Data Science

Uncover insights in your data and create interactive visualizations

Anaconda 5.0.1 For Windows Installer

Python 3.6 version *

64-Bit Graphical Installer (515 MB)
32-Bit Graphical Installer (420 MB)

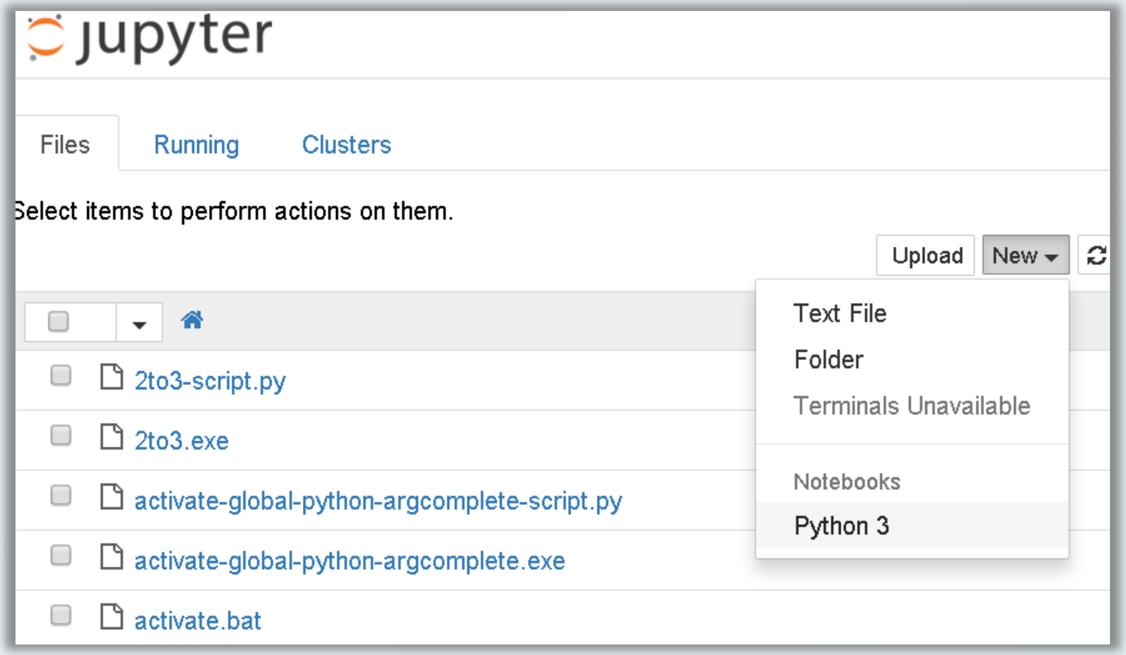
Python 2.7 version *

64-Bit Graphical Installer (500 MB)
32-Bit Graphical Installer (403 MB)

Behind a firewall?
*How to get Python 3.5 or other Python versions
How to Install ANACONDA

$ cd C:\Anaconda3\Scripts\ (if Anaconda was installed in C drive)

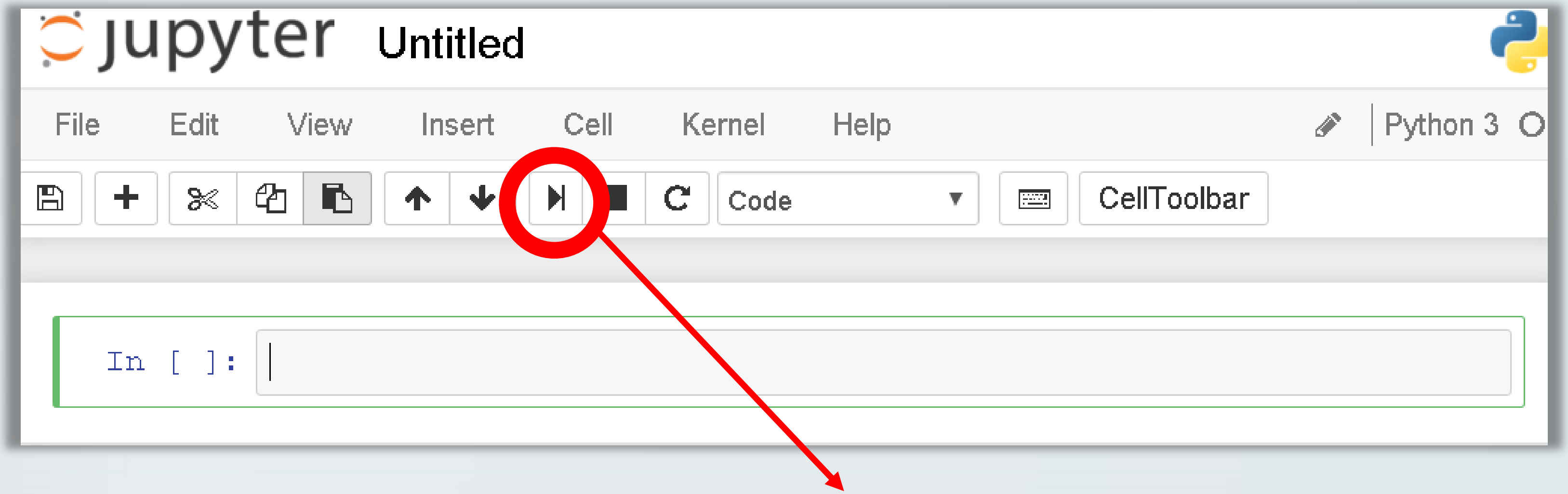
$ jupyter notebook (or python3 notebook)



Select items to perform actions on them.

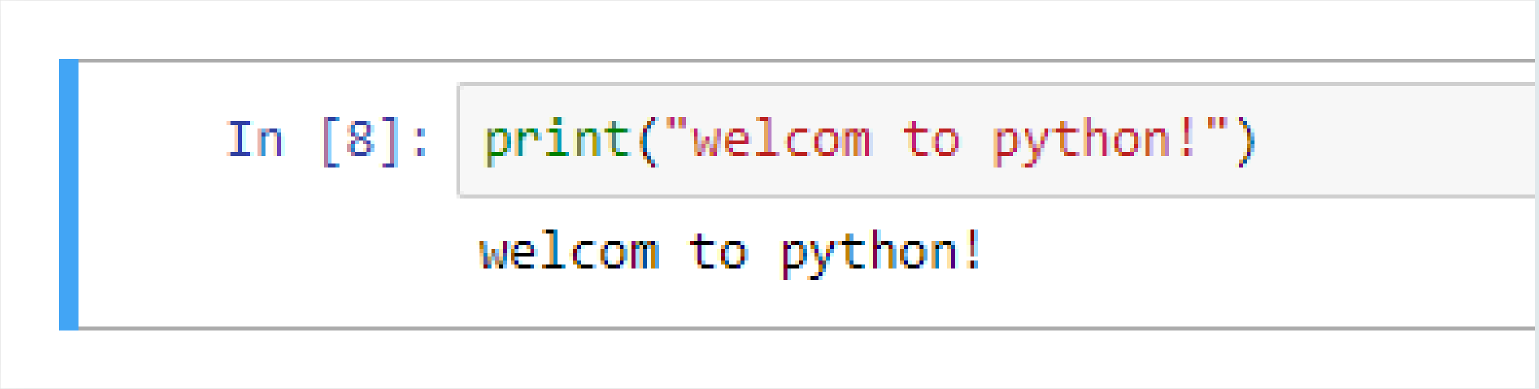
2to3-script.py
2to3.exe
activate-global-python-argcomplete-script.py
activate-global-python-argcomplete.exe
activate.bat

START Coding !!



Run your program!!

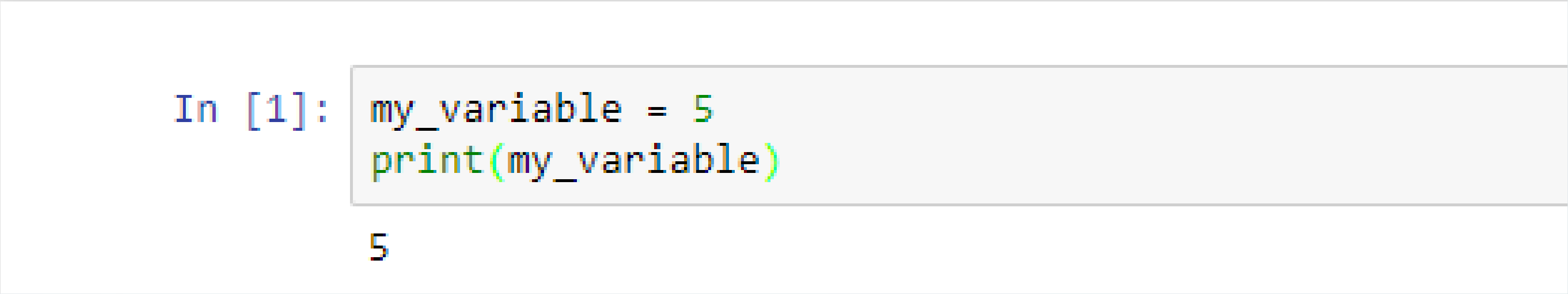
Print (顯示此項目)



```
In [8]: print("welcom to python!")

        welcom to python!
```

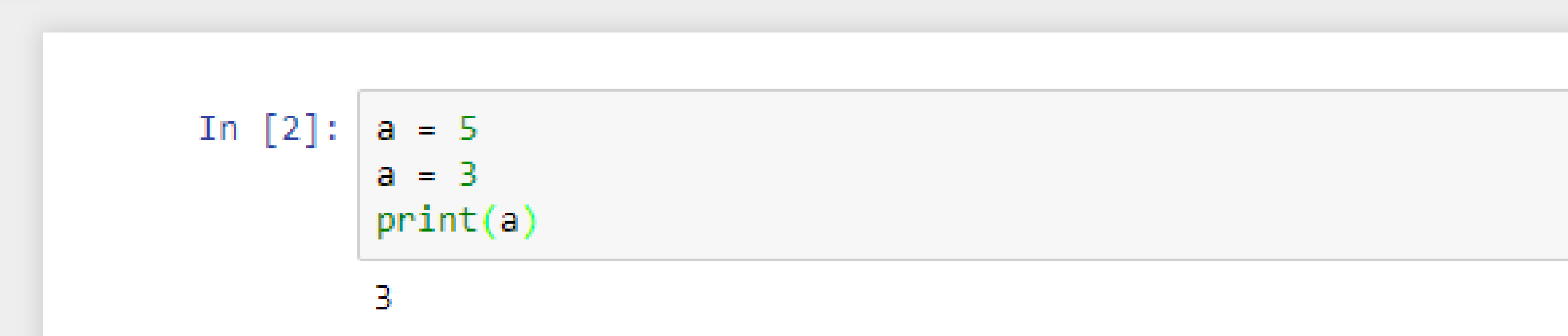
設定變數(Variables)

變數 = 值



```
In [1]: my_variable = 5
        print(my_variable)

        5
```

改變變數數值



```
a = 5
a = 3
print(a)
```

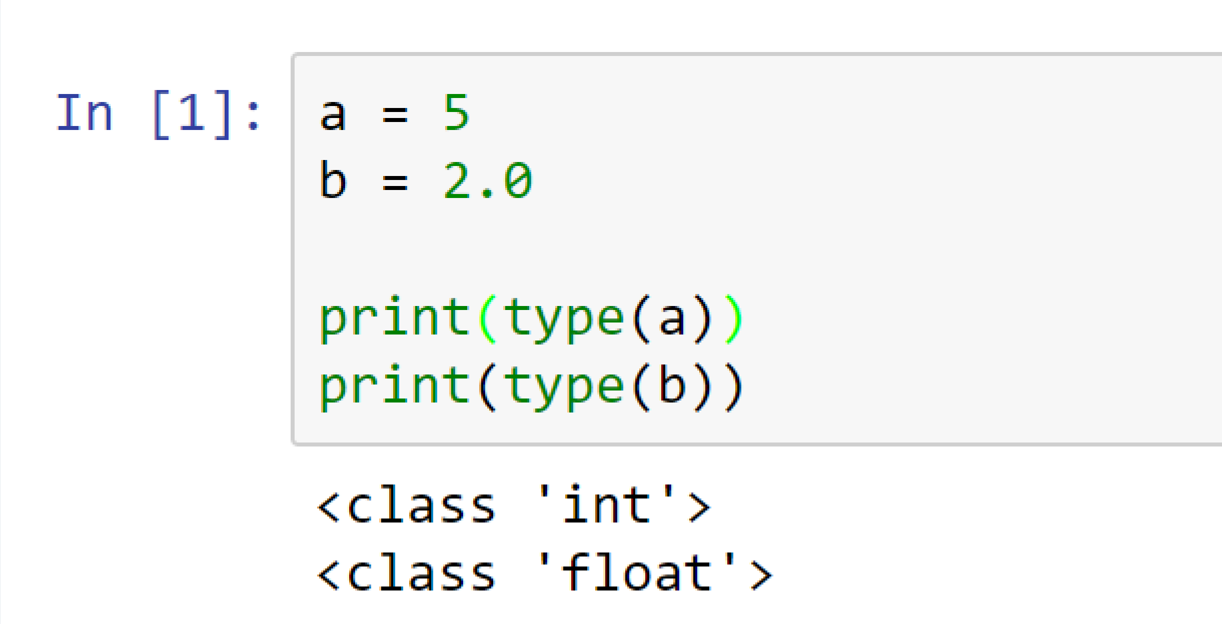
```
3
```

常用變數型態

格式	格式代碼	定義方式
值(value)	Int、float、bool	a = 5
字串(string)	str	a = ‘cat’ or “ball”
list	list	a = [1,’a’,{‘x’:1}]
dictionary	dict	a = {1,’a’,[1,2,3]}

顯示變數型態

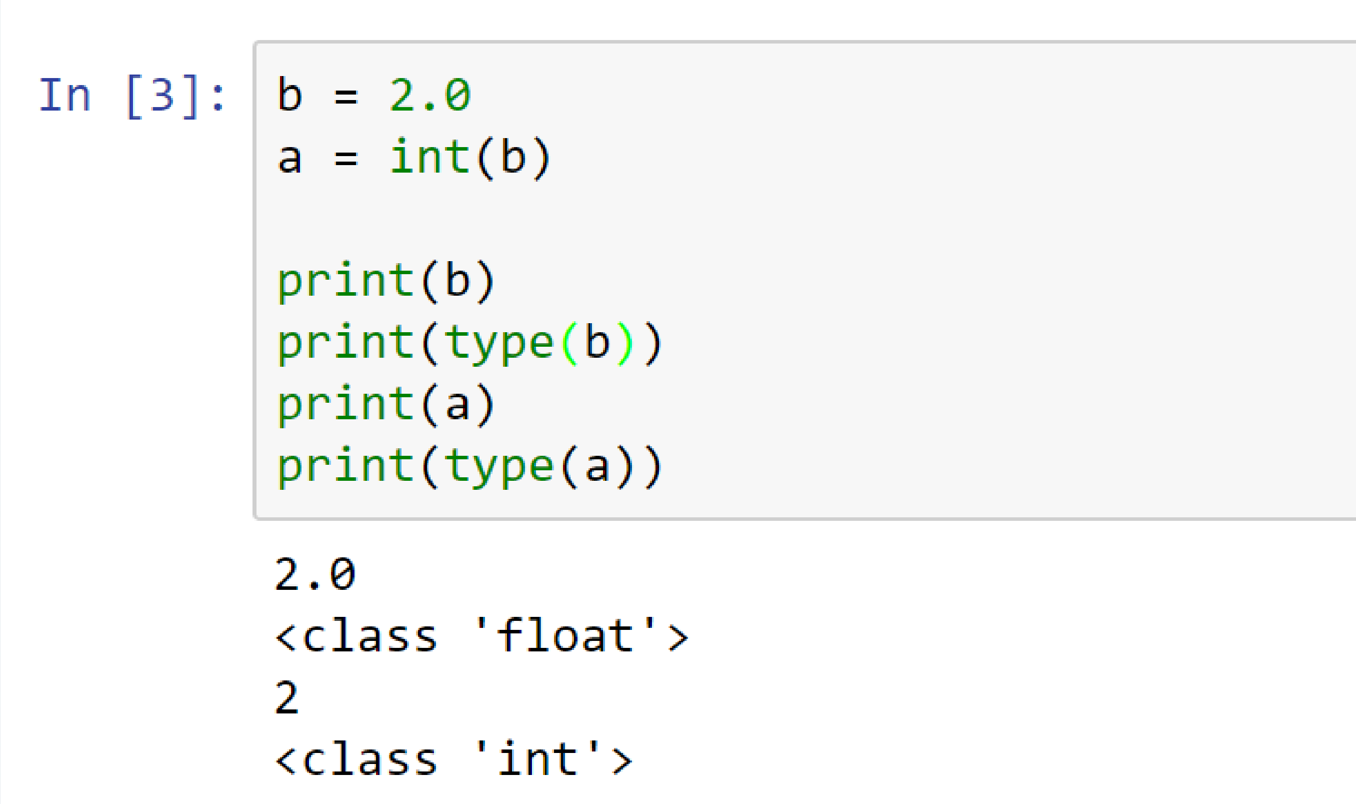
Print(type(變數))



```
In [1]: a = 5
        b = 2.0

        print(type(a))
        print(type(b))

        <class 'int'>
        <class 'float'>
```

改變變數型態

目標變數型態(要改變的變數)

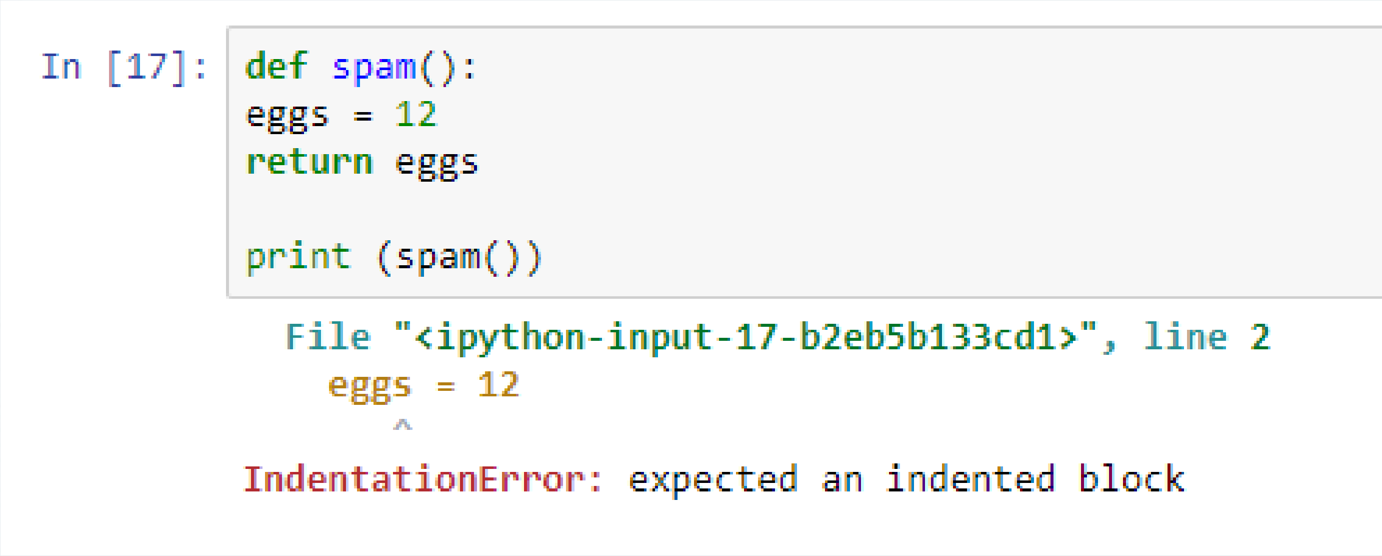


In [3]:

```
b = 2.0
a = int(b)

print(b)
print(type(b))
print(a)
print(type(a))
```

```
2.0
<class 'float'>
2
<class 'int'>
```

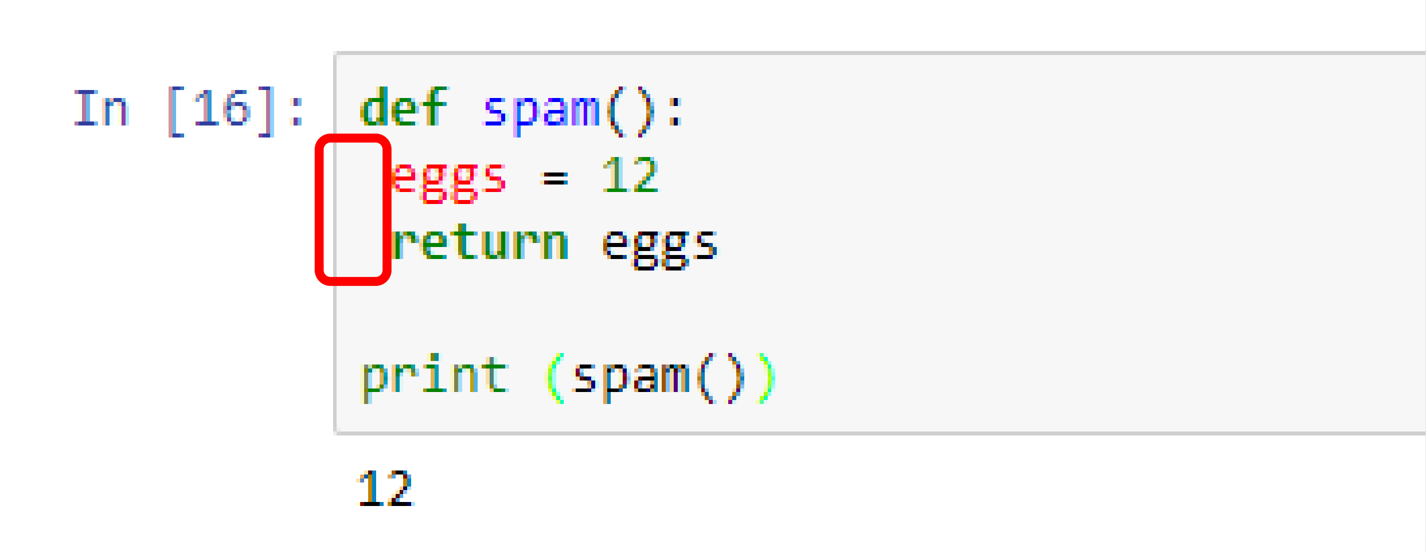
空白鍵很重要!!



```
In [17]: def spam():
         eggs = 12
         return eggs

         print (spam())
```

```
  File "<ipython-input-17-b2eb5b133cd1>", line 2
    eggs = 12
       ^
IndentationError: expected an indented block
```

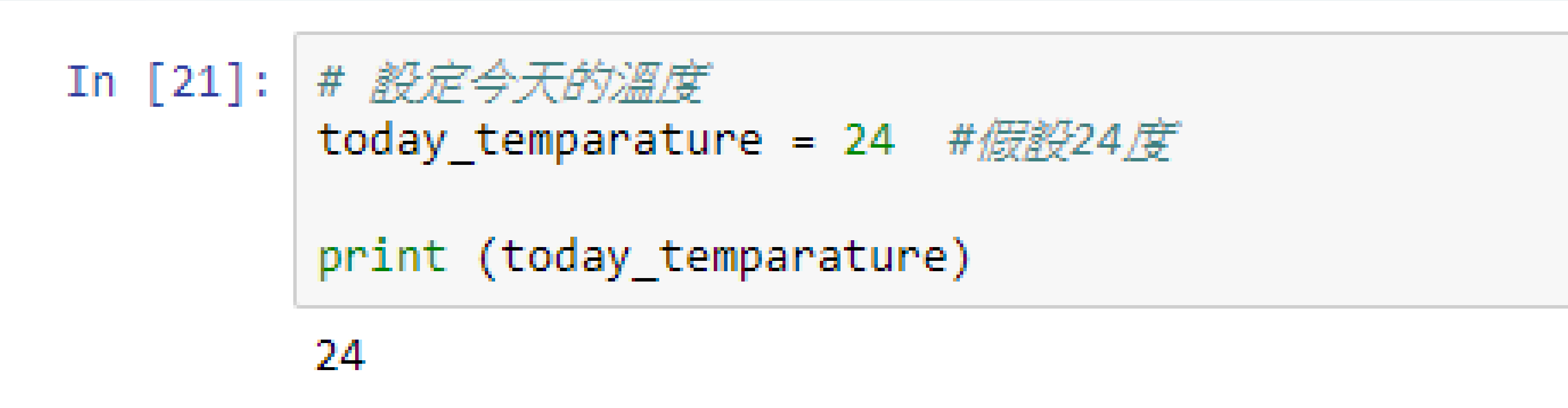
空白鍵很重要!!



```
In [16]: def spam():
          eggs = 12
          return eggs

         print (spam())

         12
```

單行註記

# 此行不影響程式運行



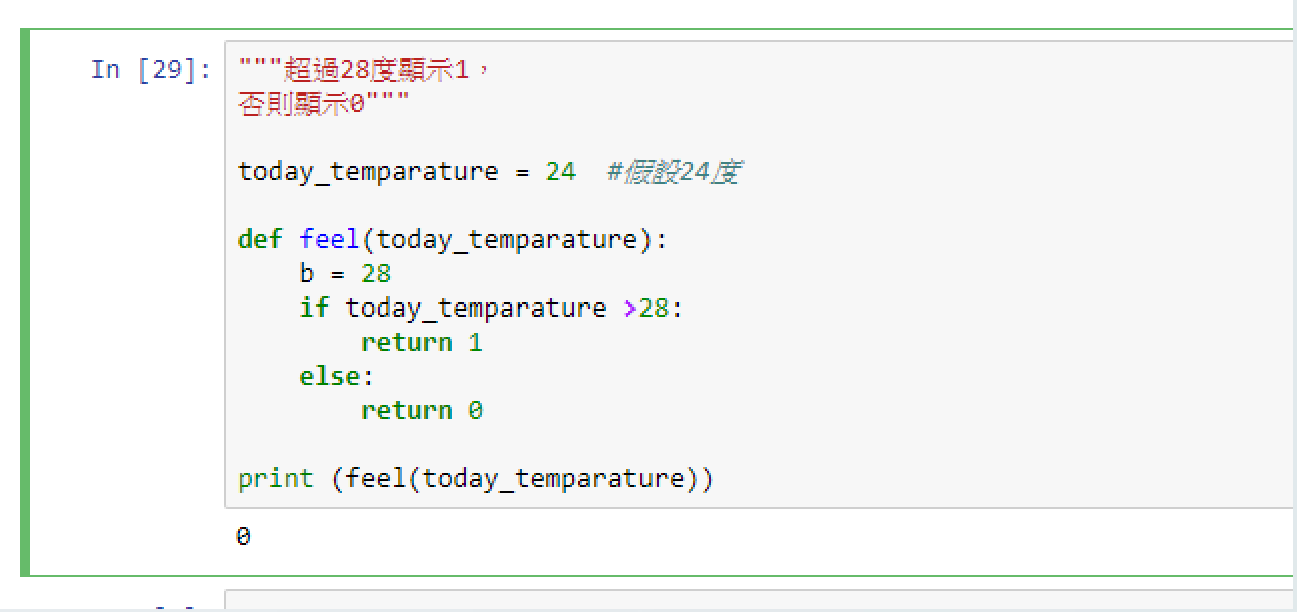
```
In [21]: # 設定今天的溫度
         today_temparature = 24  #假設24度

         print (today_temparature)

         24
```

(給程式設計師閱讀的註記)

多行註記

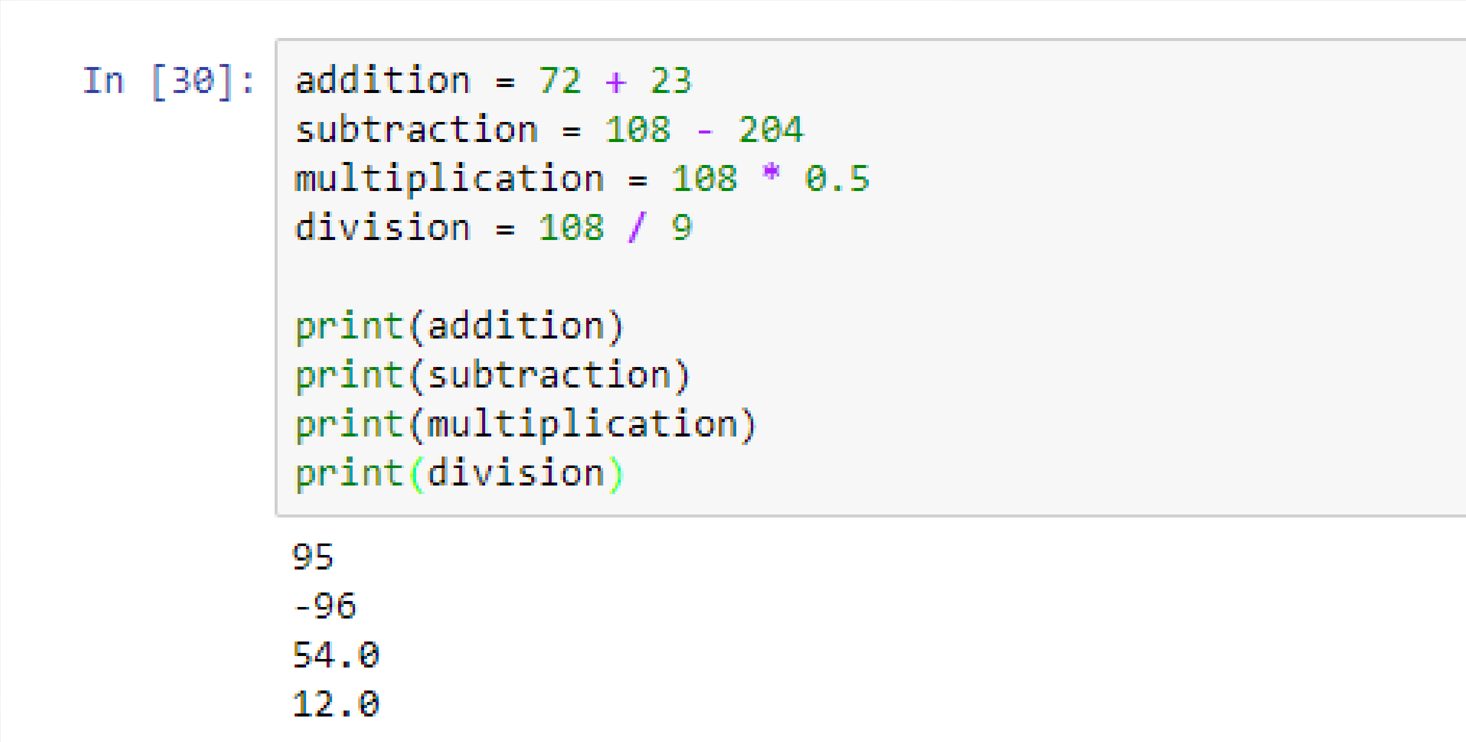
“”“此段文字不管多長、幾行，

都不影響程式運行"""



```
In [29]: """超過28度顯示1，
         否則顯示0"""

         today_temparature = 24  #假設24度

         def feel(today_temparature):
             b = 28
             if today_temparature >28:
                 return 1
             else:
                 return 0

         print (feel(today_temparature))

         0
```

簡單數學：四則運算

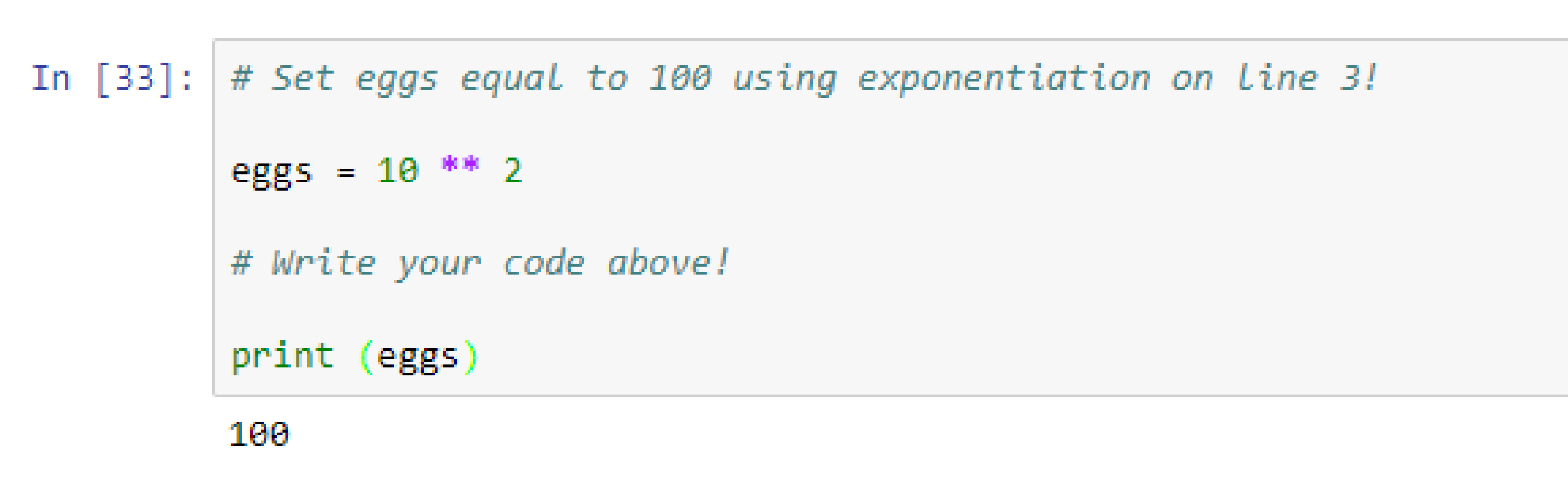


```
In [30]: addition = 72 + 23
         subtraction = 108 - 204
         multiplication = 108 * 0.5
         division = 108 / 9

         print(addition)
         print(subtraction)
         print(multiplication)
         print(division)
```

```
95
-96
54.0
12.0
```

簡單數學：次方

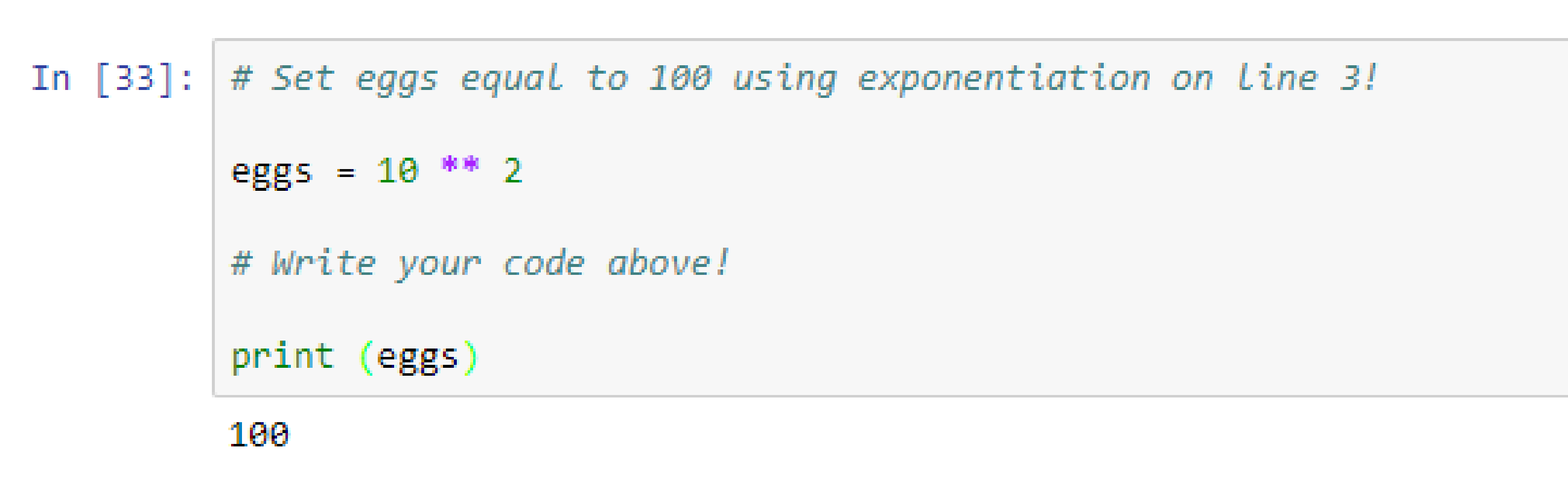
10 ** 2 = 10的2次方 = 100



```
In [33]: # Set eggs equal to 100 using exponentiation on line 3!

         eggs = 10 ** 2

         # Write your code above!

         print (eggs)

         100
```

簡單數學：次方

10 ** 2 ＝ 10的2次方 = 100



```
In [33]: # Set eggs equal to 100 using exponentiation on line 3!

         eggs = 10 ** 2

         # Write your code above!

         print (eggs)

         100
```

實務題庫：餐廳結帳

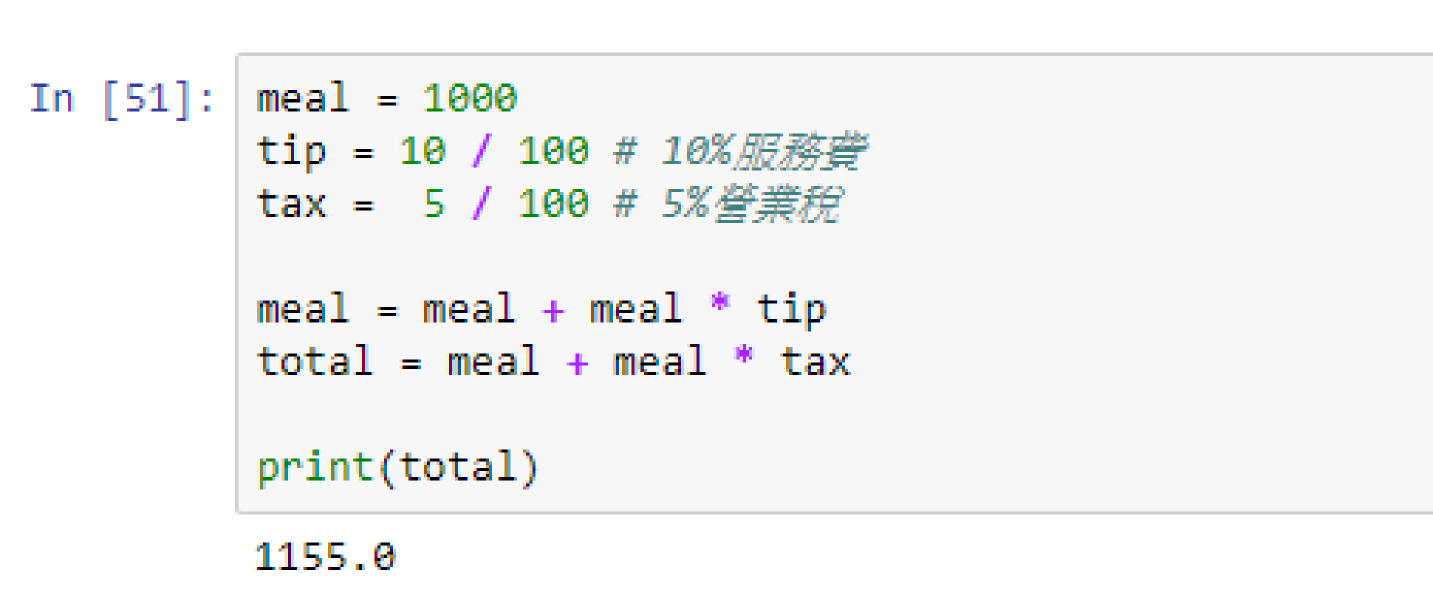
餐費單價=1000元

10%服務費 5%營業稅

實務題庫：餐廳結帳

餐費單價=1000元

10%服務費 5%營業稅

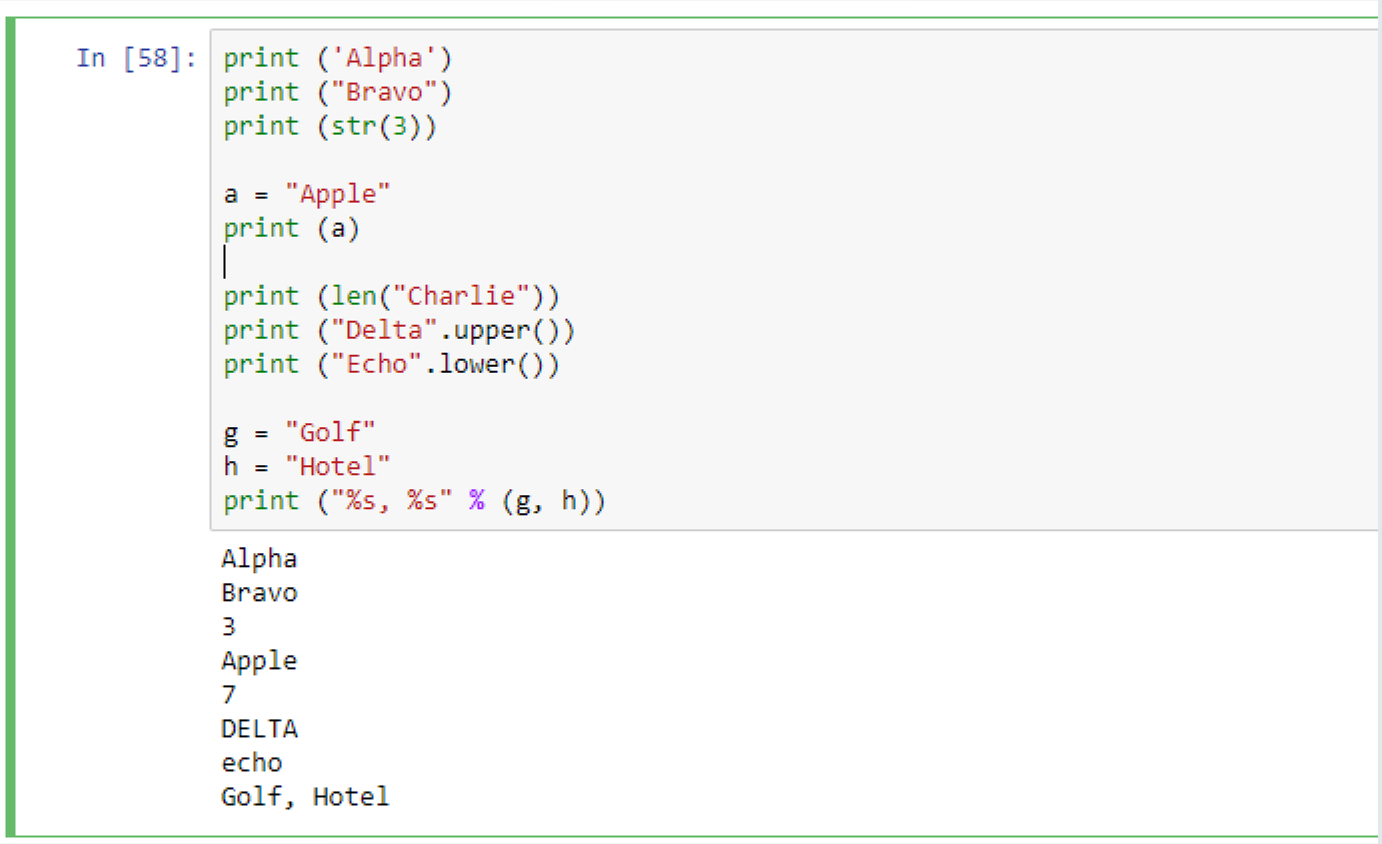


In [51]:

```
meal = 1000
tip = 10 / 100 # 10%服務費
tax =  5 / 100 # 5%營業稅

meal = meal + meal * tip
total = meal + meal * tax

print(total)
```

```
1155.0
```

String字串：Python的文字遊戲

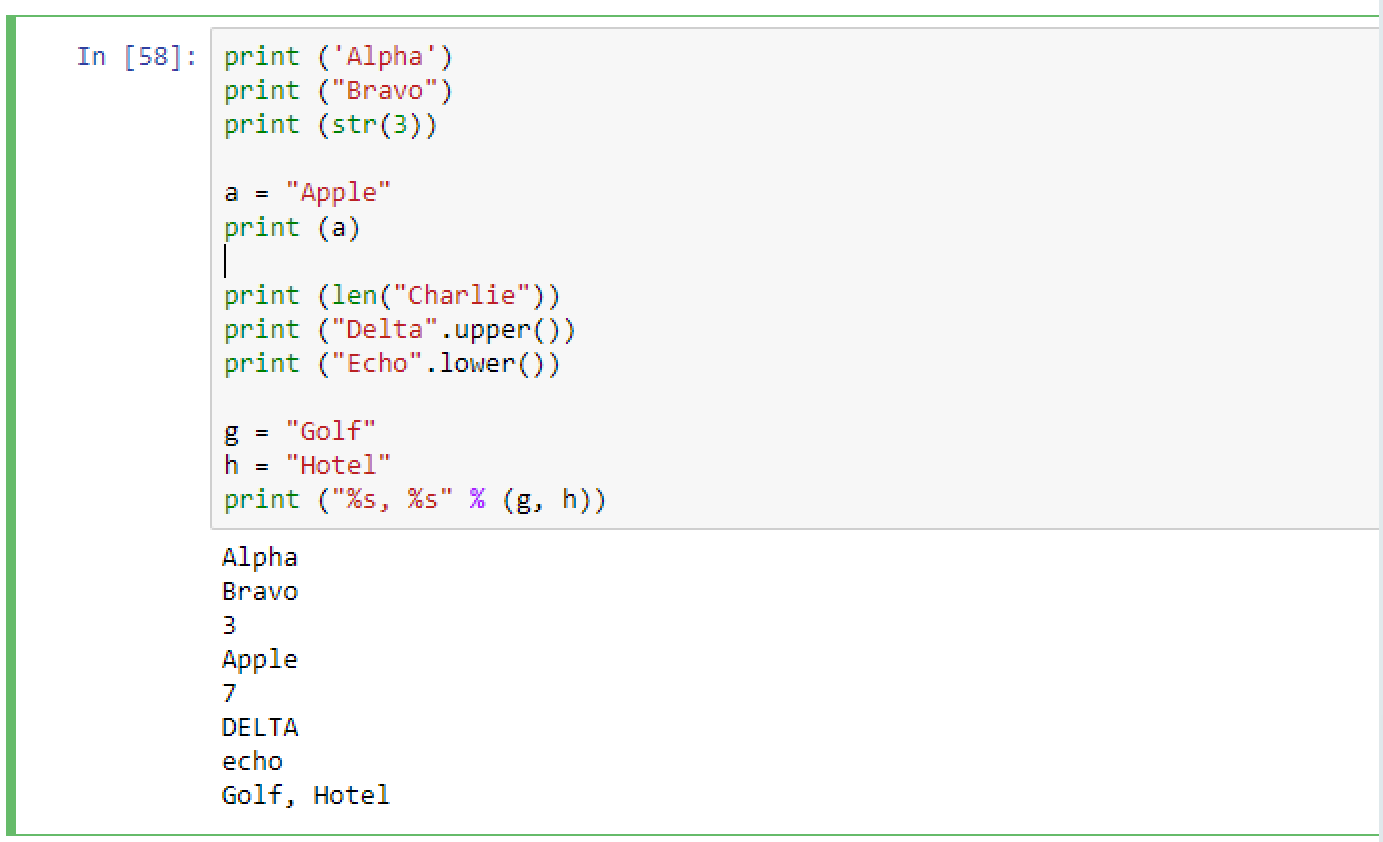


In [58]:

```python
print ('Alpha')
print ("Bravo")
print (str(3))

a = "Apple"
print (a)

print (len("Charlie"))
print ("Delta".upper())
print ("Echo".lower())

g = "Golf"
h = "Hotel"
print ("%s, %s" % (g, h))
```

```
Alpha
Bravo
3
Apple
7
DELTA
echo
Golf, Hotel
```

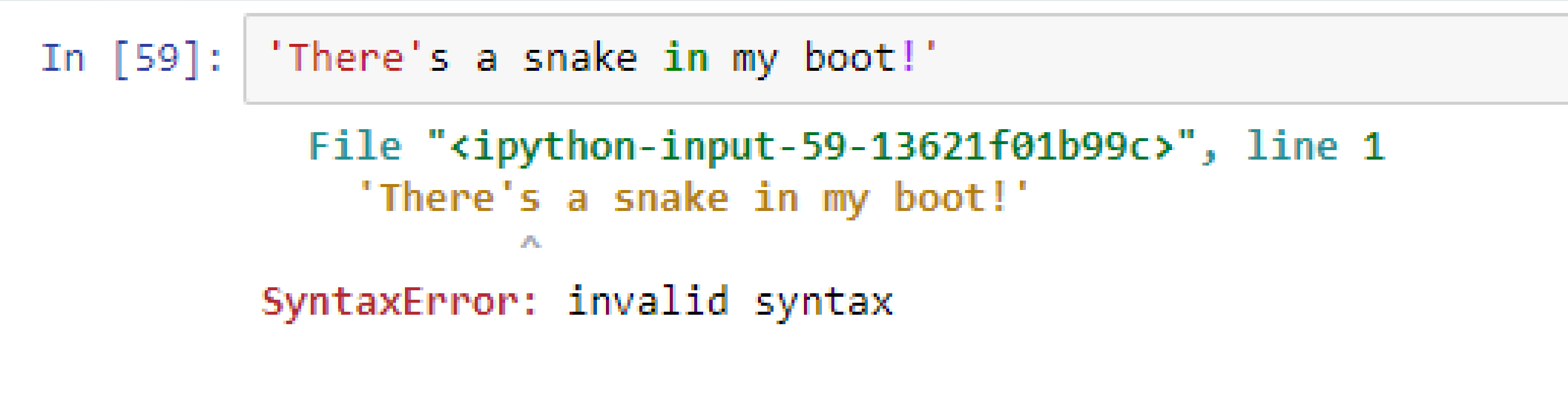
String字串：Python的文字遊戲



```
In [58]: print ('Alpha')
         print ("Bravo")
         print (str(3))

         a = "Apple"
         print (a)

         print (len("Charlie"))
         print ("Delta".upper())
         print ("Echo".lower())

         g = "Golf"
         h = "Hotel"
         print ("%s, %s" % (g, h))
```

```
Alpha
Bravo
3
Apple
7
DELTA
echo
Golf, Hotel
```

String字串：被斷句時

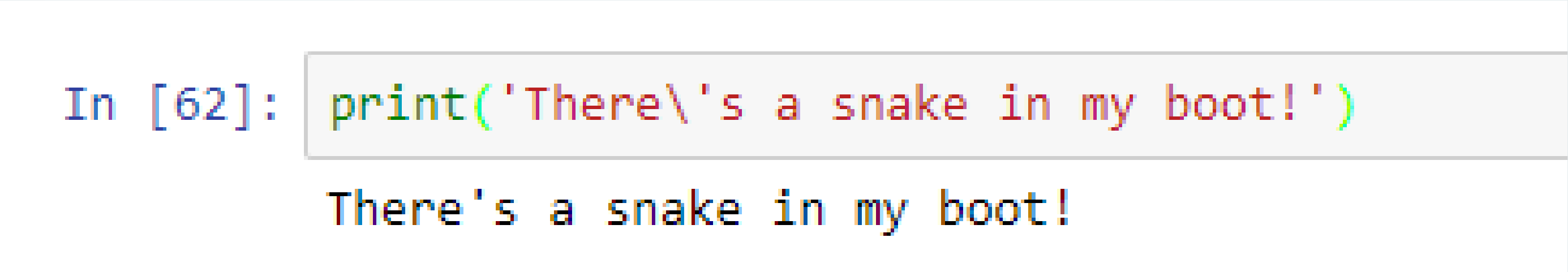


```
In [59]: 'There's a snake in my boot!'
```

```
  File "<ipython-input-59-13621f01b99c>", line 1
    'There's a snake in my boot!'
           ^
SyntaxError: invalid syntax
```

String字串：被斷句時

用 \ 反斜線前置處理



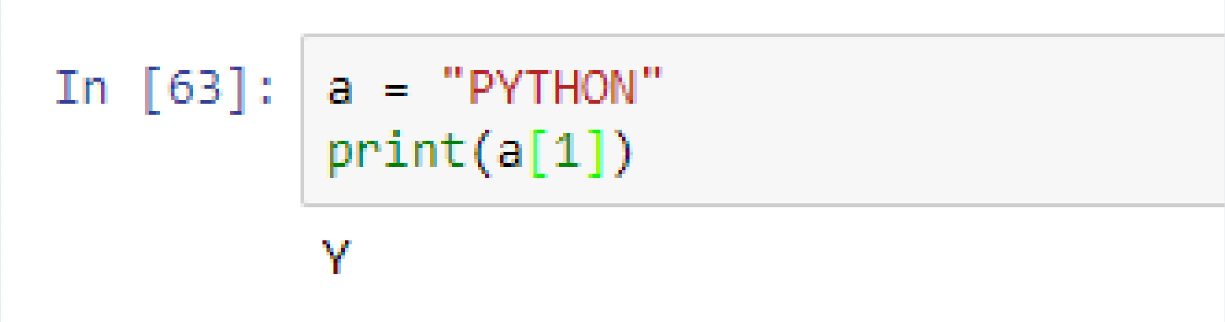
```
In [62]: print('There\'s a snake in my boot!')

         There's a snake in my boot!
```

String字串：第幾個字

當設定字串 a = “PYTHON”

P	Y	T	H	O	N
0	1	2	3	4	5

a[1] = 字串a的1號字母

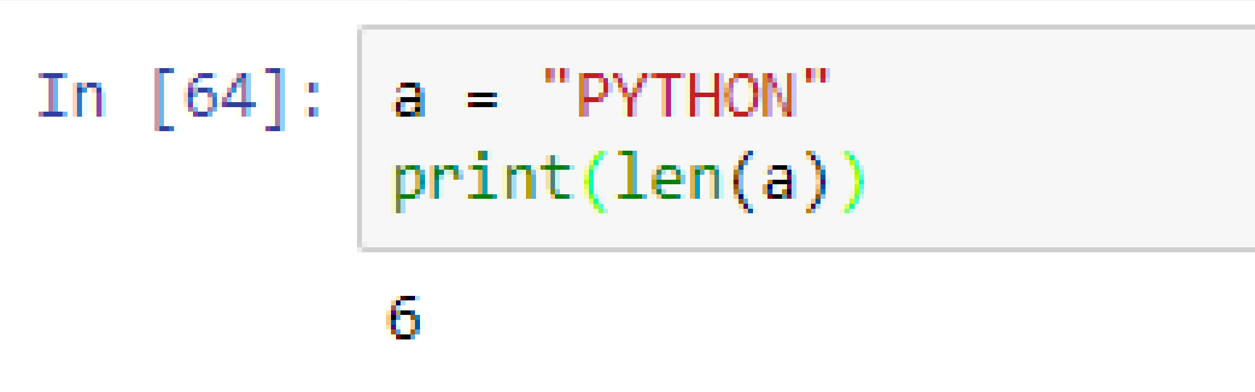


```
In [63]: a = "PYTHON"
         print(a[1])

         Y
```

String字串：字串長度

當設定字串 a = “PYTHON”

len(a) = 字串a的長度(字母個數)



In [64]:

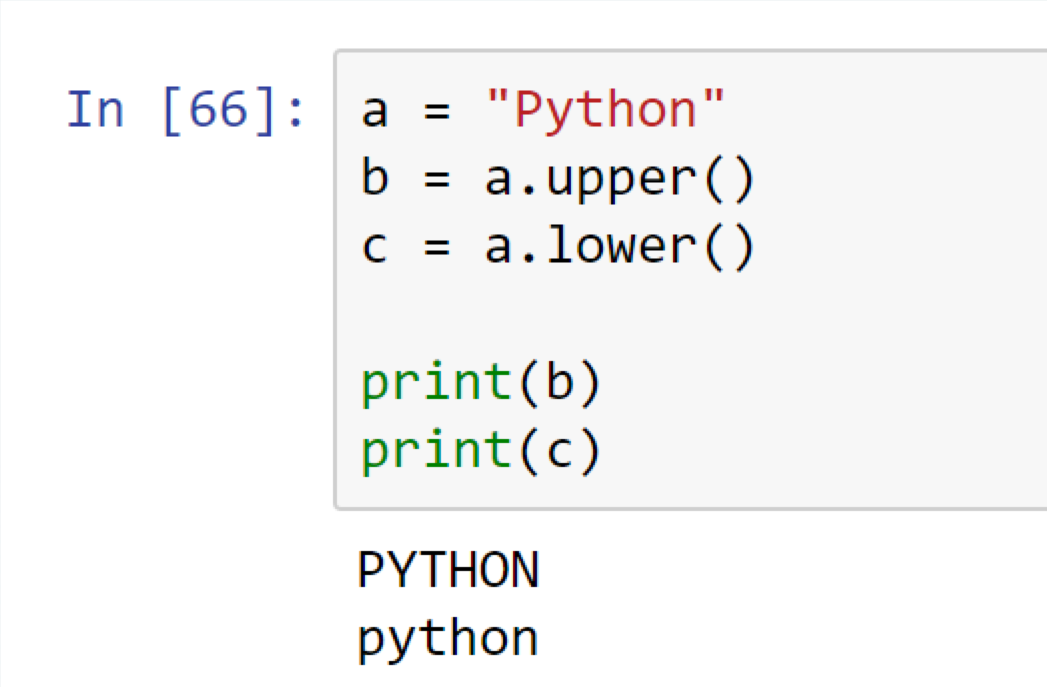
```
a = "PYTHON"
print(len(a))
```

```
6
```

String字串：大小寫切換

a.upper() = 字串a的大寫

a.lower() = 字串a的小寫



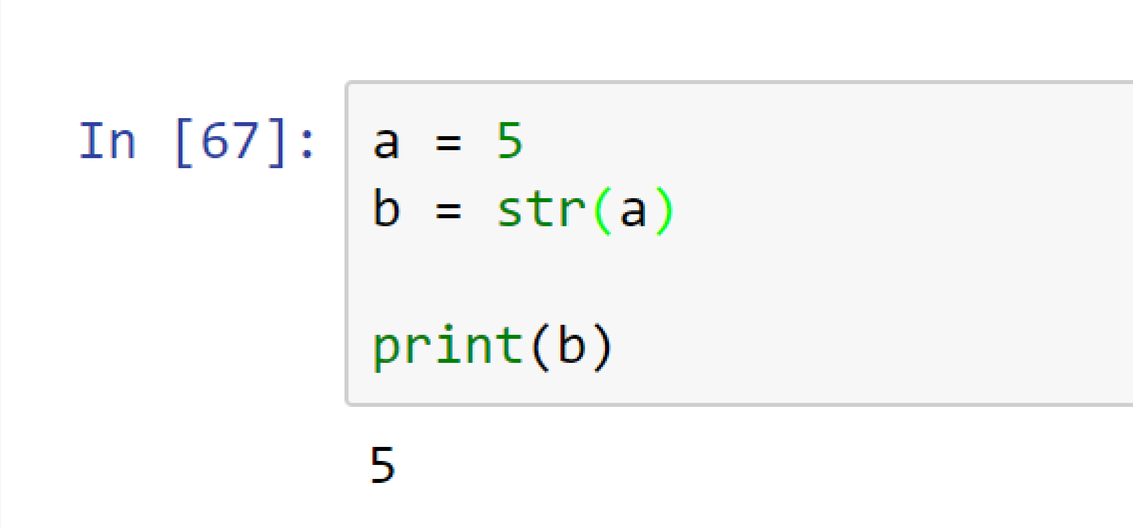
In [66]:

```
a = "Python"
b = a.upper()
c = a.lower()

print(b)
print(c)
```

```
PYTHON
python
```

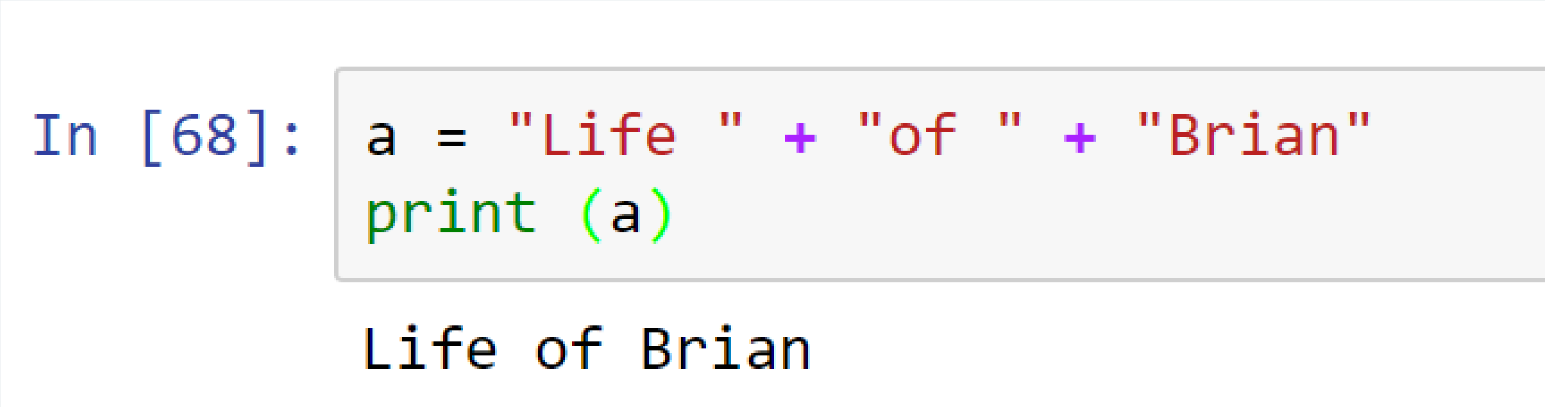
String字串：轉換成字串

str(a) = 將不是字串的a轉換成字串



```
In [67]: a = 5
         b = str(a)

         print(b)

         5
```

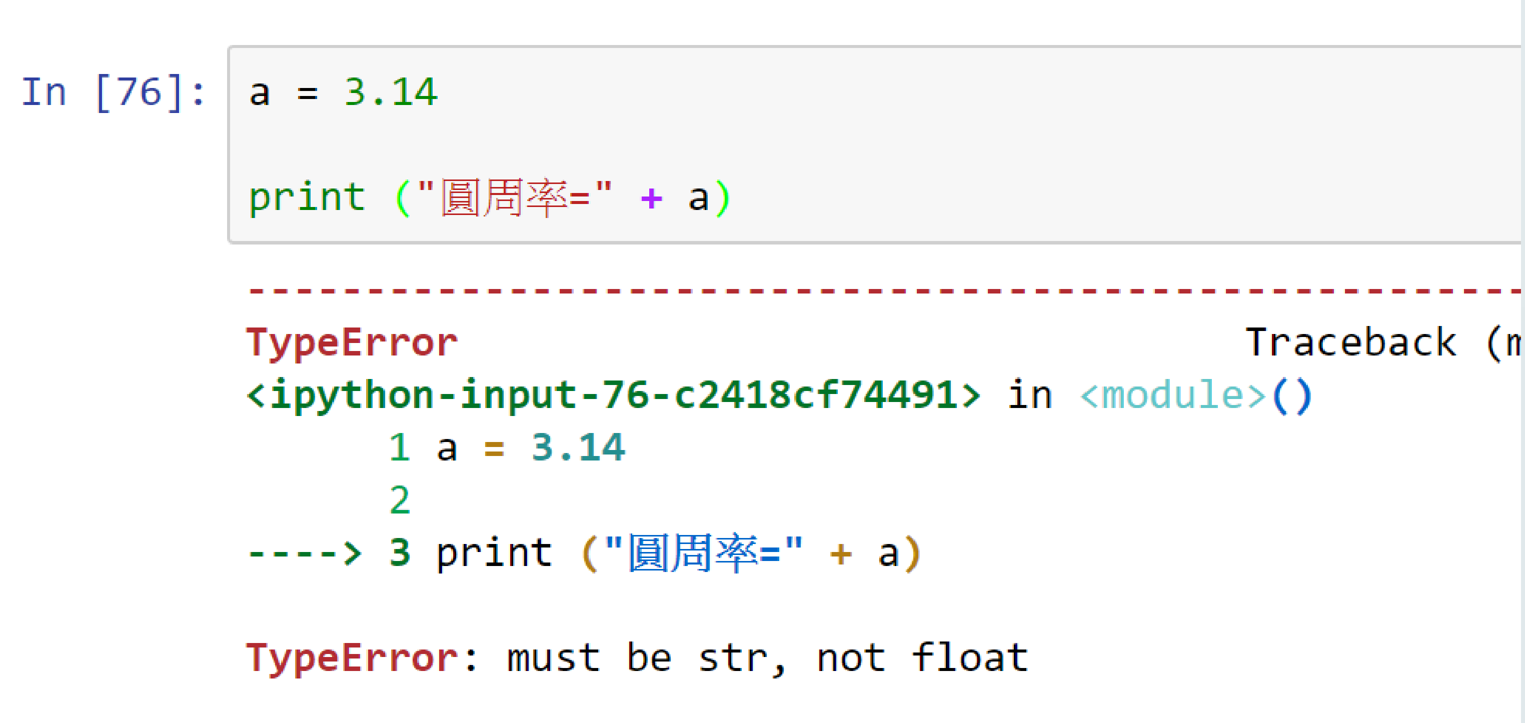
String字串：字串+字串



```python
a = "Life " + "of " + "Brian"
print (a)
```

```
Life of Brian
```

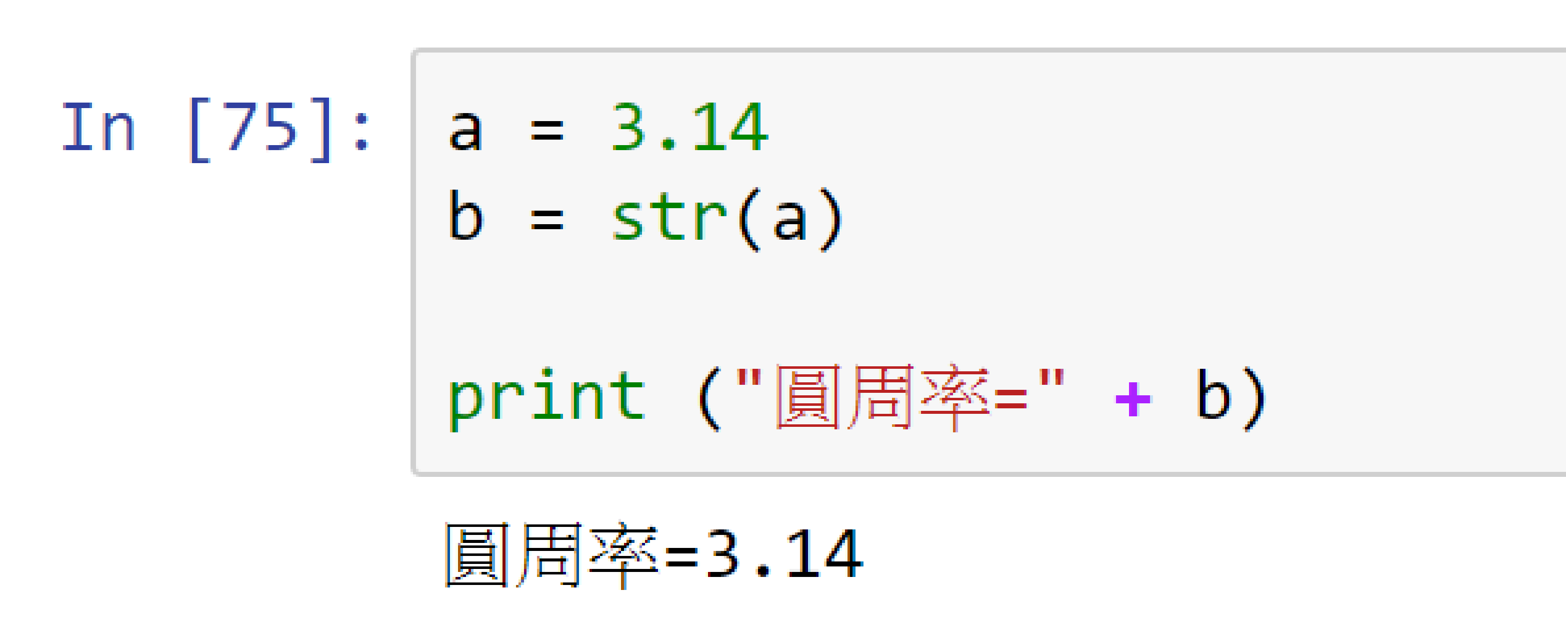
轉換成字串的必要性



```
In [76]: a = 3.14

         print ("圓周率=" + a)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (m
<ipython-input-76-c2418cf74491> in <module>()
      1 a = 3.14
      2 
----> 3 print ("圓周率=" + a)

TypeError: must be str, not float
```

轉換成字串的必要性



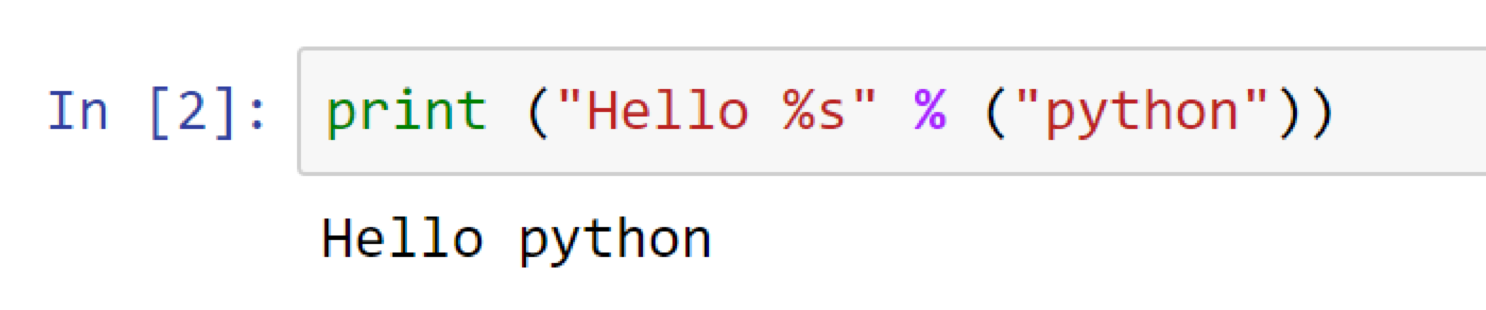
```
In [75]: a = 3.14
         b = str(a)

         print ("圓周率=" + b)
```

```
圓周率=3.14
```

字串格式化表示

字串當中加入%s ，將%後面的變數插入

“字串 %s” % (在%s要插入的內容)

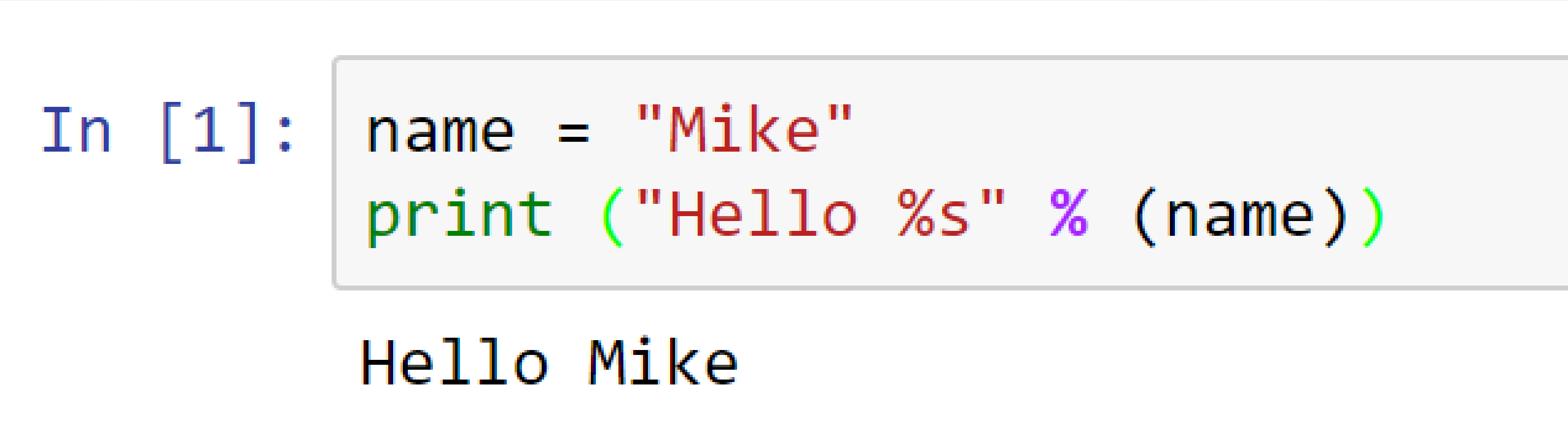


```
In [2]: print ("Hello %s" % ("python"))
```

```
Hello python
```

字串格式化表示

%s 可以在%後面插入任意string格式變數



```
In [1]: name = "Mike"
        print ("Hello %s" % (name))

        Hello Mike
```

字串格式化表示

常用的格式型態

%s	格式化字串
%d	格式化整數
%f	格式化浮點數

實務題庫：格式化字串

設以下變數：

name：你的英文名字

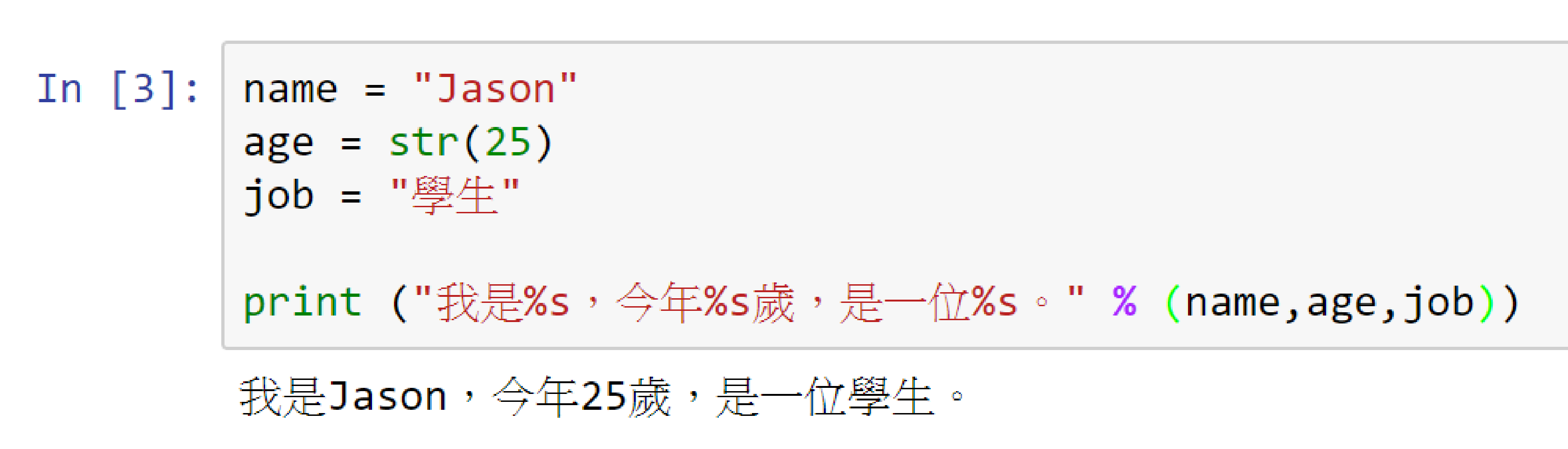
age：你的年齡

job： 你的職業(ex. 學生)

並使用格式化字串輸出以下句子:

我是Jason，今年25歲，是一位學生。

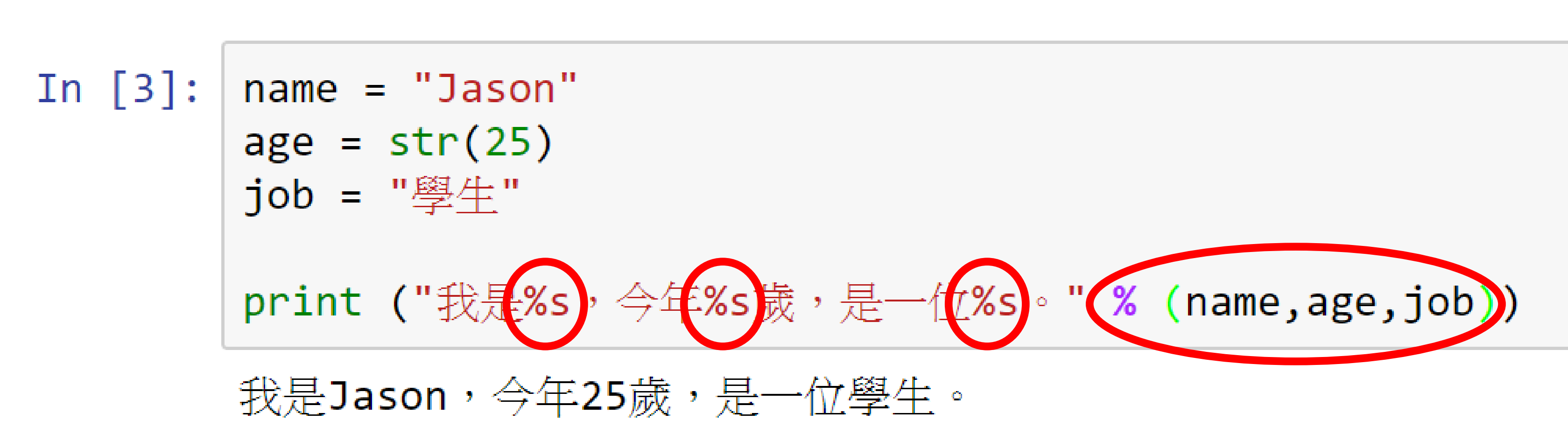
實務題庫：格式化字串



```
In [3]: name = "Jason"
        age = str(25)
        job = "學生"

        print ("我是%s，今年%s歲，是一位%s。" % (name,age,job))
```

```
我是Jason，今年25歲，是一位學生。
```

實務題庫：格式化字串



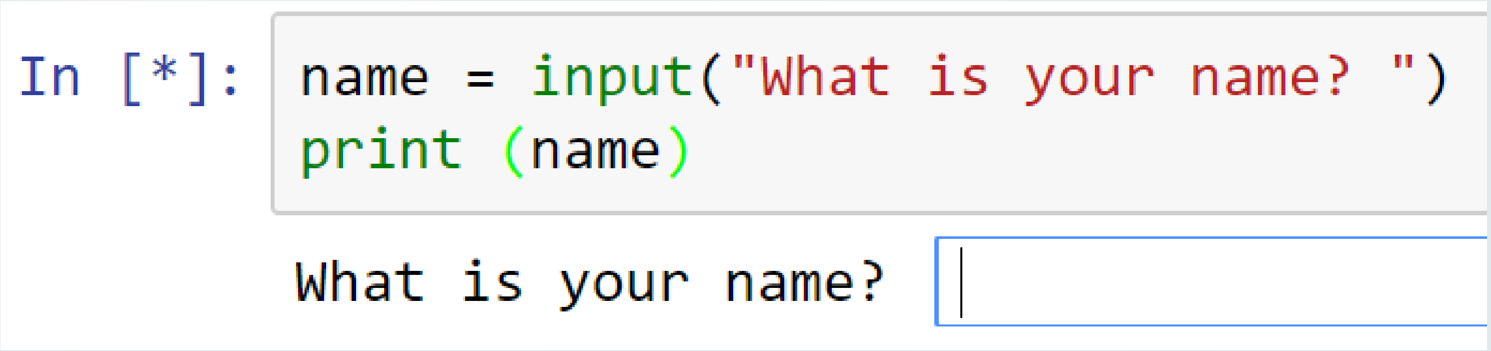
```
In [3]: name = "Jason"
        age = str(25)
        job = "學生"

        print ("我是%s，今年%s歲，是一位%s。" % (name,age,job))
```

```
我是Jason，今年25歲，是一位學生。
```

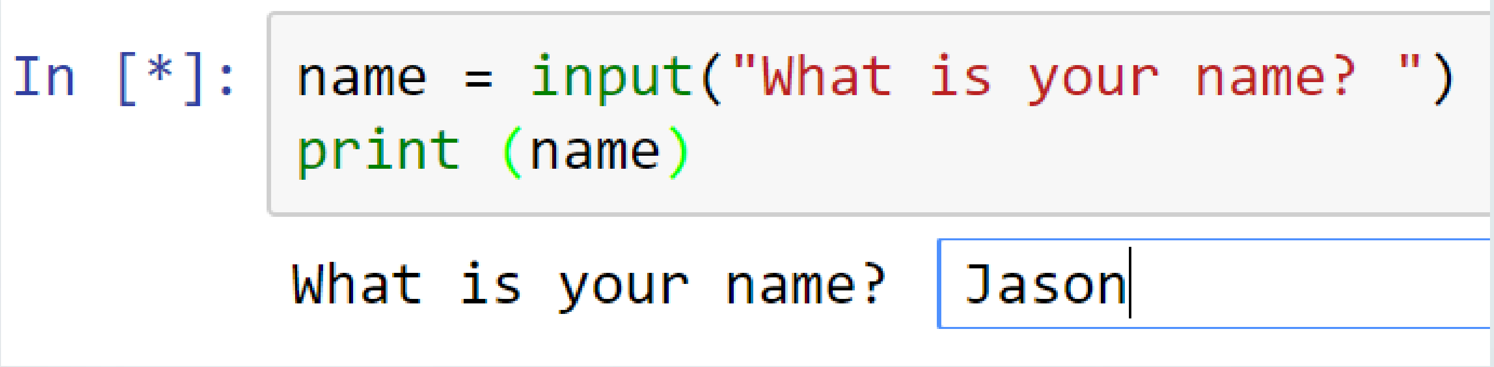
注意一個%s需要對應一個項目!!

多學一點：輸入項目

input (請使用者輸入內容)



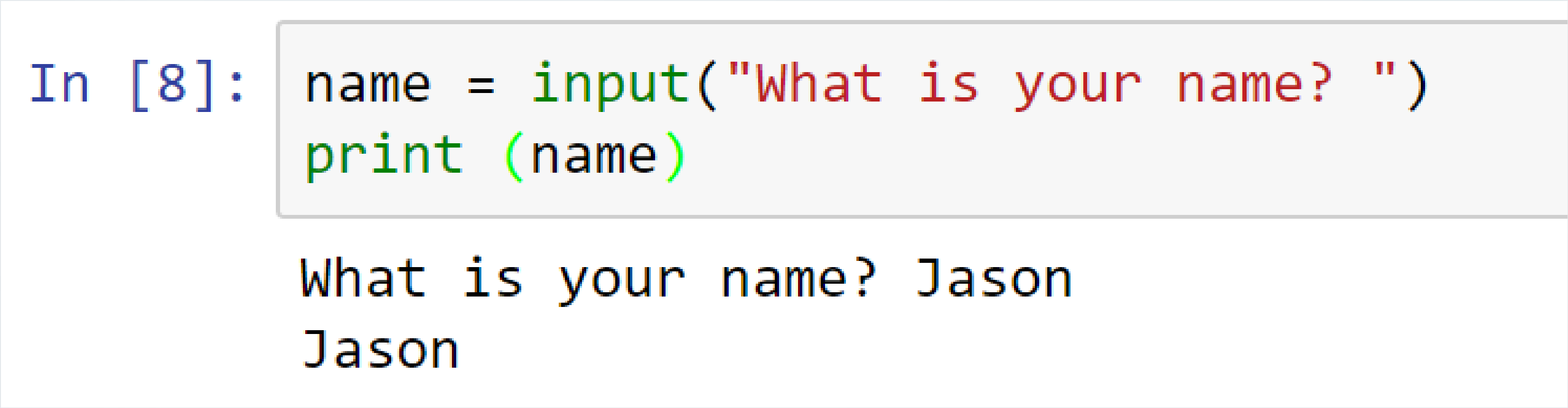
```
In [*]: name = input("What is your name? ")
        print (name)

        What is your name?
```



```
In [*]: name = input("What is your name? ")
        print (name)

        What is your name? Jason
```

多學一點：輸入項目

input (請使用者輸入內容)



```
In [8]: name = input("What is your name? ")
        print (name)

        What is your name? Jason
        Jason
```

實務題庫：輸入格式化字串

用input()設以下變數：

name : "What is your name? “

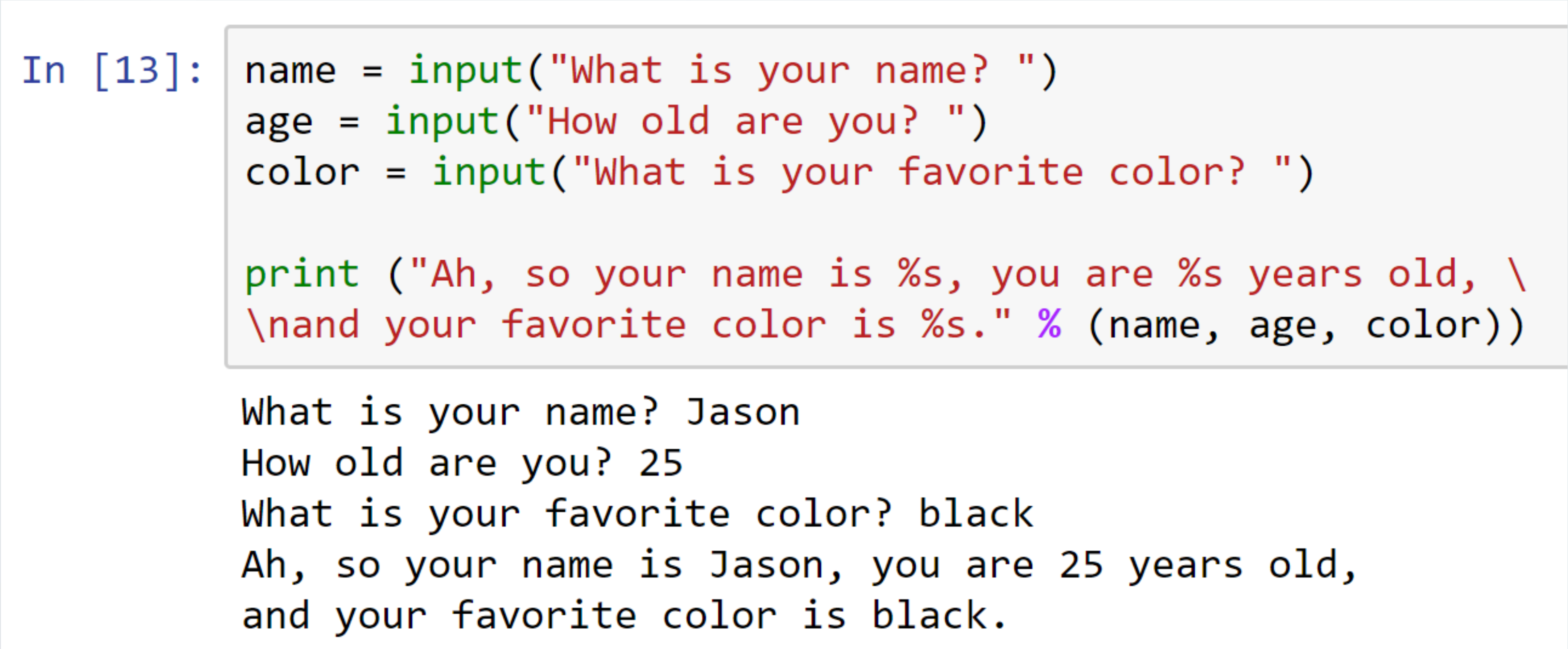
age : "How old are you? “

color : "What is your favorite color? “

並使用格式化字串輸出以下句子:

Ah, so your name is json, your are 25 years old, and your favorite color is black.

實務題庫：輸入格式化字串



In [13]:

```python
name = input("What is your name? ")
age = input("How old are you? ")
color = input("What is your favorite color? ")

print ("Ah, so your name is %s, you are %s years old, \
\nand your favorite color is %s." % (name, age, color))
```

```
What is your name? Jason
How old are you? 25
What is your favorite color? black
Ah, so your name is Jason, you are 25 years old,
and your favorite color is black.
```

list

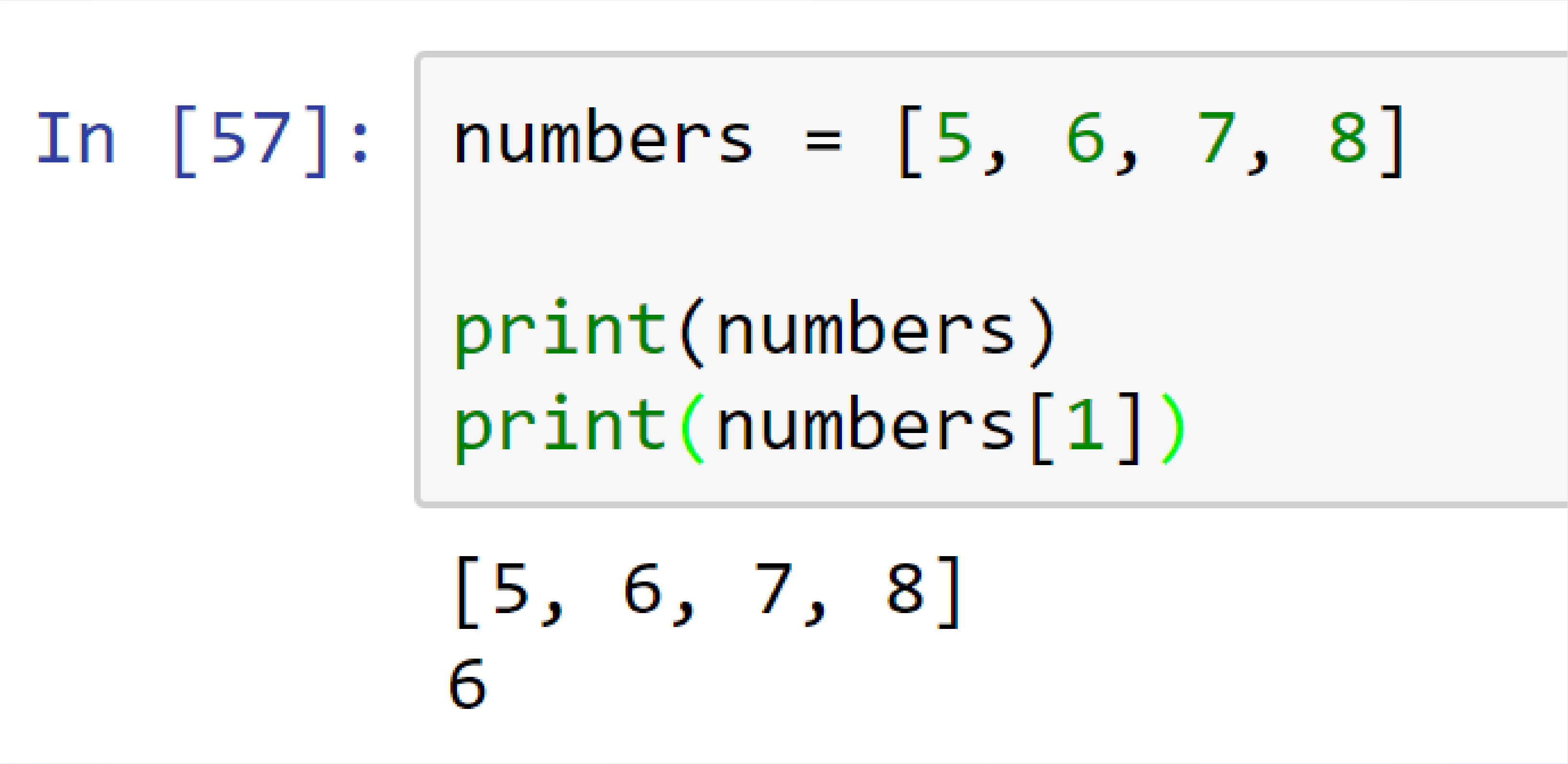
list = [物件1,物件2,……]

list 可為多個物件之物件組合

各物件以list[X]表示

X為物件順位，從零開始計算

list



```
numbers = [5, 6, 7, 8]

print(numbers)
print(numbers[1])
```

```
[5, 6, 7, 8]
6
```

dictionary

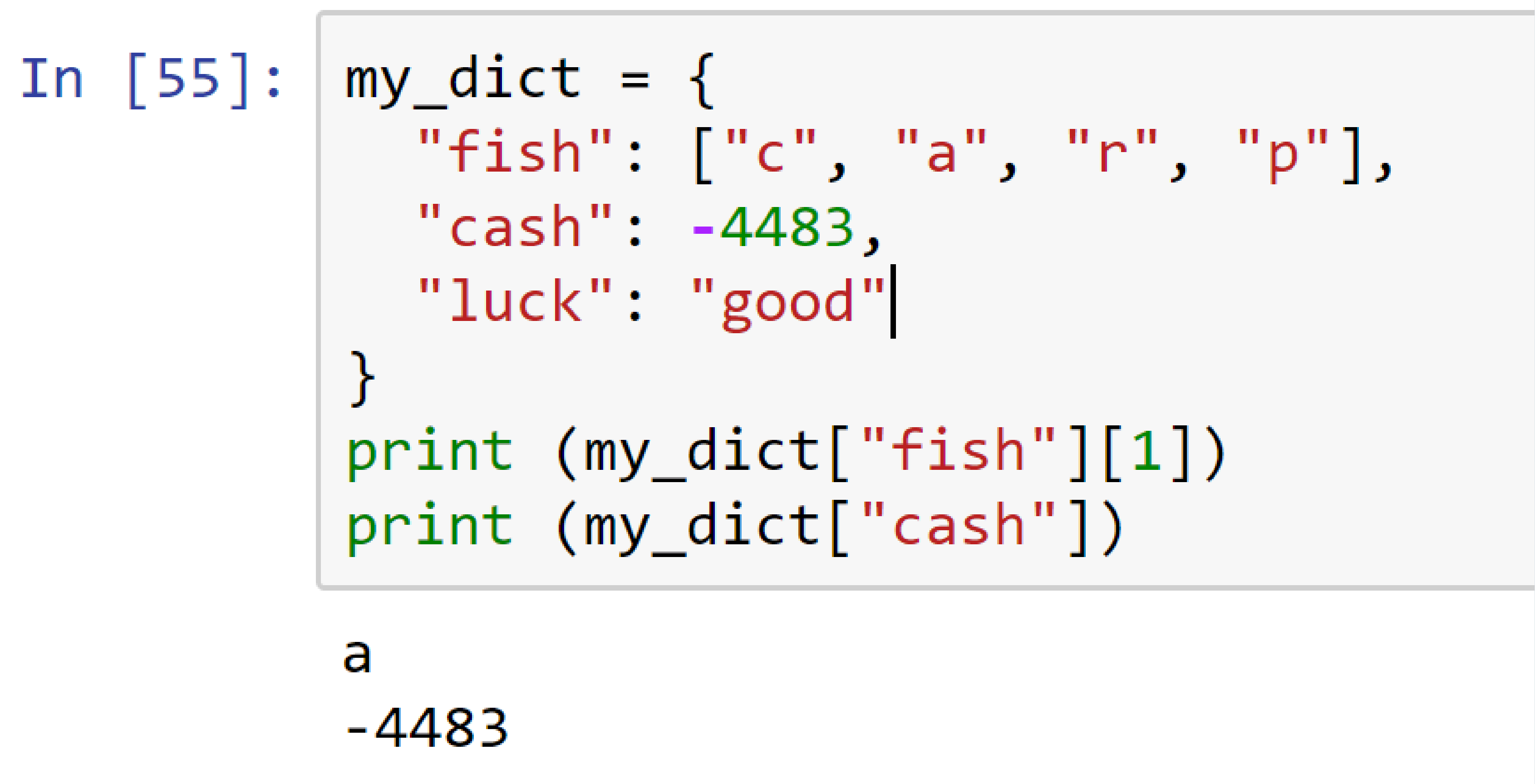
dictionary =
{key1:value1,key2:value2,……. }

dictionary 可為多對鍵值之組合容器

各”值”以dictionary[鍵]回傳

.items()	將各組鍵值以list回傳

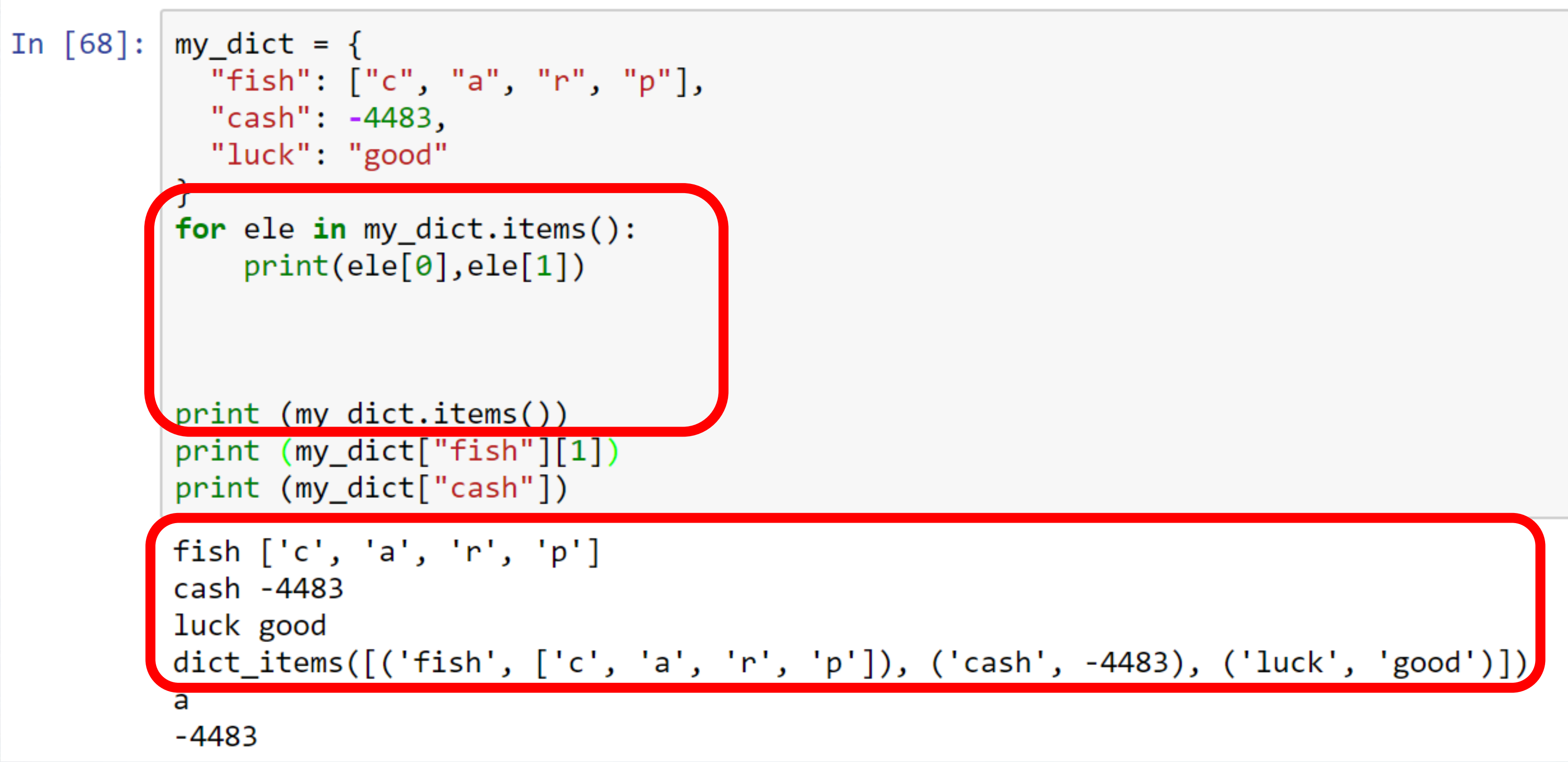
dictionary



```
my_dict = {
  "fish": ["c", "a", "r", "p"],
  "cash": -4483,
  "luck": "good"
}
print (my_dict["fish"][1])
print (my_dict["cash"])
```

```
a
-4483
```

Dic.items()



In [68]:

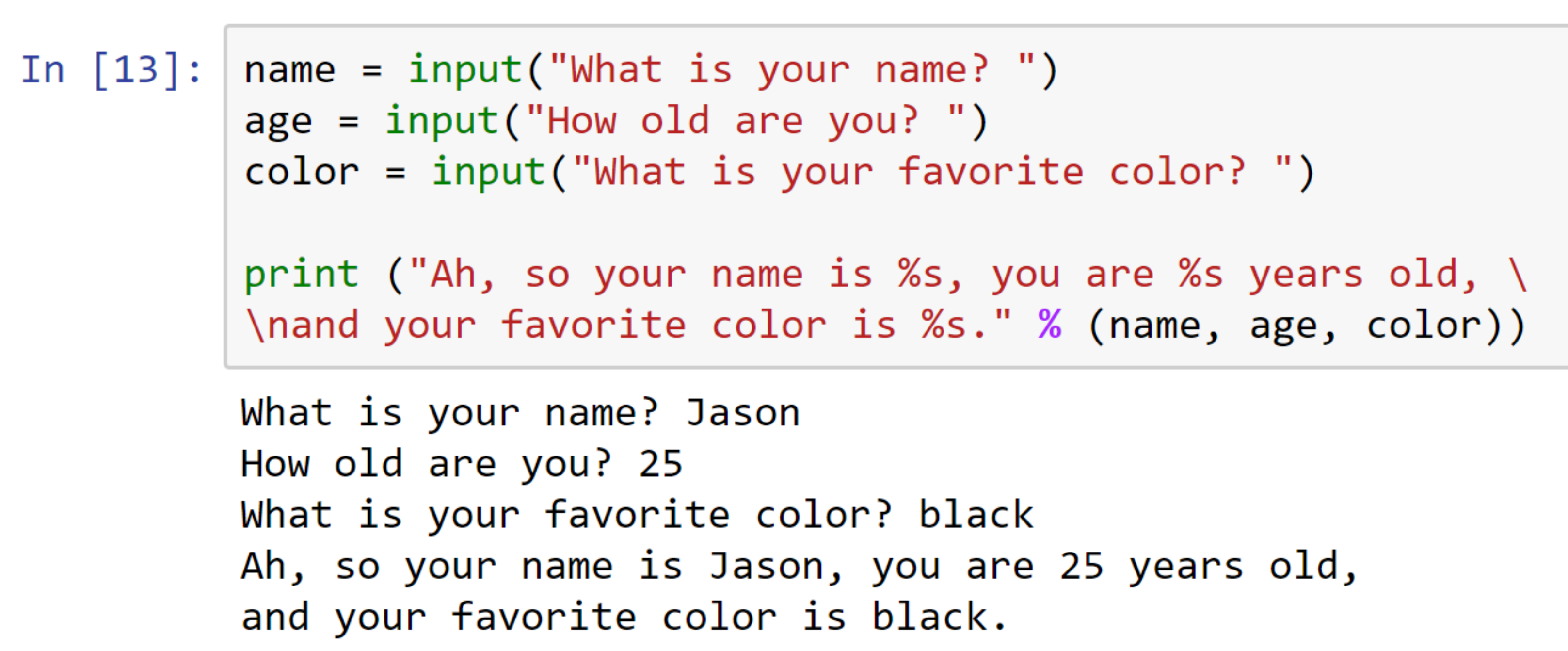
```
my_dict = {
  "fish": ["c", "a", "r", "p"],
  "cash": -4483,
  "luck": "good"
}
for ele in my_dict.items():
    print(ele[0],ele[1])



print (my_dict.items())
print (my_dict["fish"][1])
print (my_dict["cash"])
```

```
fish ['c', 'a', 'r', 'p']
cash -4483
luck good
dict_items([('fish', ['c', 'a', 'r', 'p']), ('cash', -4483), ('luck', 'good')])
a
-4483
```

小技巧：程式換行&輸出換行

\ = 程式內換行，不影響程式運行

\n = 輸出結果換行，影響美觀

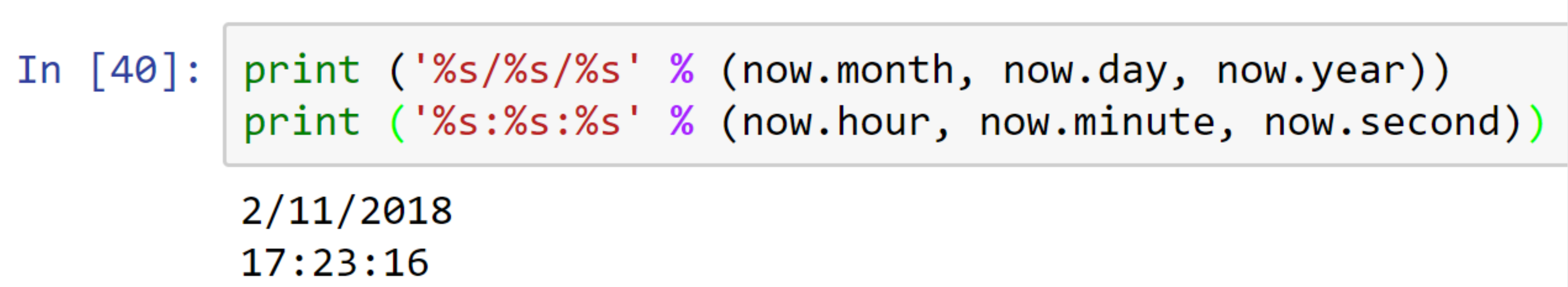


In [13]:

```
name = input("What is your name? ")
age = input("How old are you? ")
color = input("What is your favorite color? ")

print ("Ah, so your name is %s, you are %s years old, \
\nand your favorite color is %s." % (name, age, color))
```

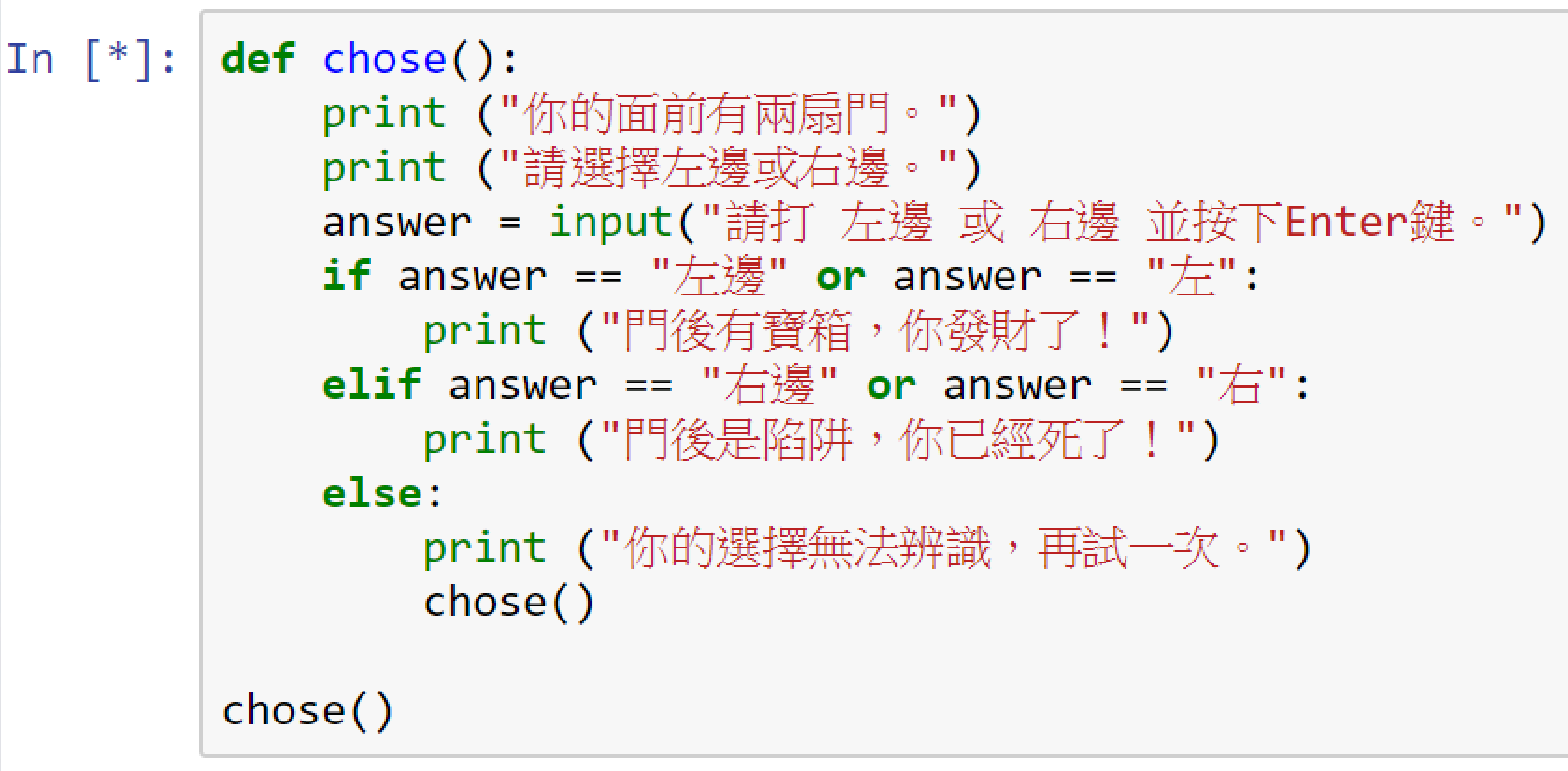
```
What is your name? Jason
How old are you? 25
What is your favorite color? black
Ah, so your name is Jason, you are 25 years old,
and your favorite color is black.
```

指令小補帖：時間



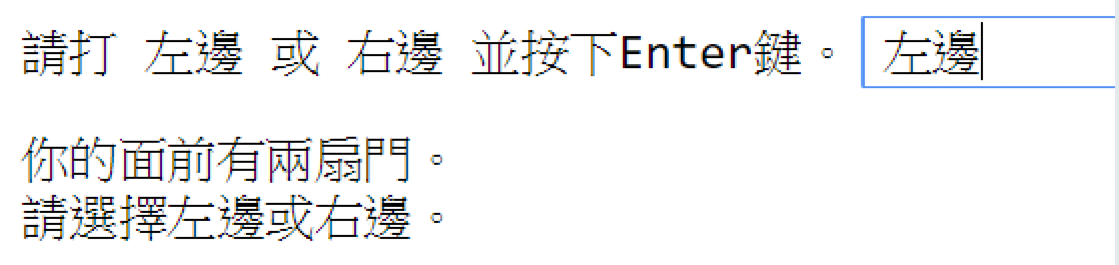
```
In [40]: print ('%s/%s/%s' % (now.month, now.day, now.year))
         print ('%s:%s:%s' % (now.hour, now.minute, now.second))

         2/11/2018
         17:23:16
```

def、if語法：選擇題範例

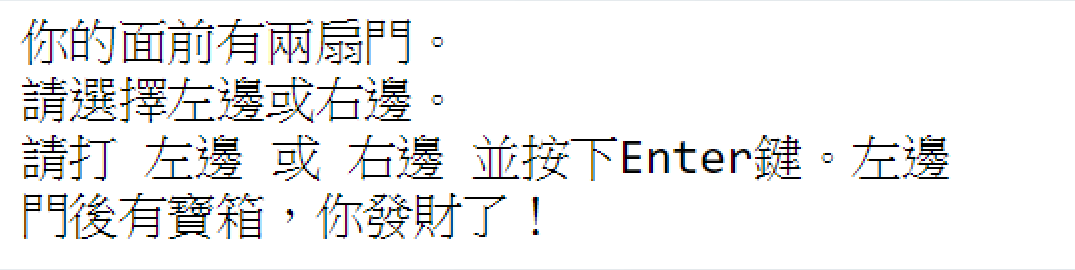


```python
def chose():
    print ("你的面前有兩扇門。")
    print ("請選擇左邊或右邊。")
    answer = input("請打 左邊 或 右邊 並按下Enter鍵。")
    if answer == "左邊" or answer == "左":
        print ("門後有寶箱，你發財了！")
    elif answer == "右邊" or answer == "右":
        print ("門後是陷阱，你已經死了！")
    else:
        print ("你的選擇無法辨識，再試一次。")
        chose()

chose()
```

def、if語法：選擇題範例

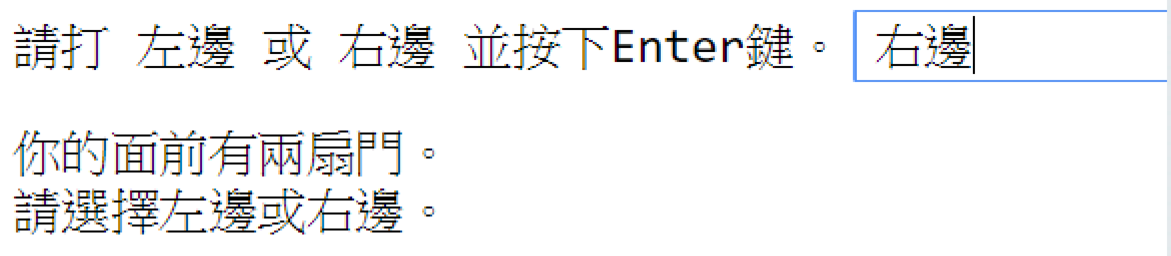


```
請打 左邊 或 右邊 並按下Enter鍵。 左邊

你的面前有兩扇門。
請選擇左邊或右邊。
```

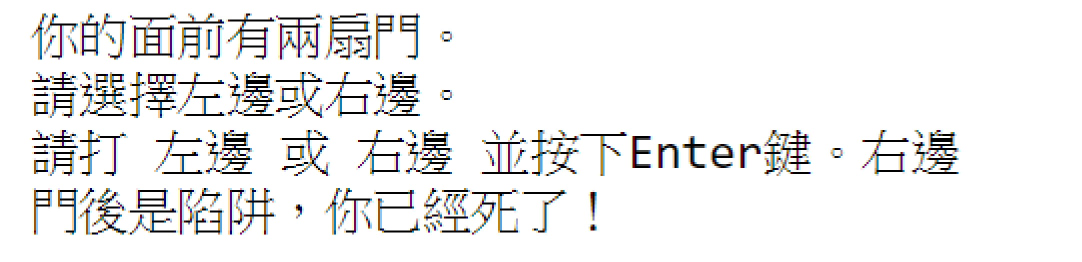


```
你的面前有兩扇門。
請選擇左邊或右邊。
請打 左邊 或 右邊 並按下Enter鍵。左邊
門後有寶箱，你發財了！
```

def、if語法：選擇題範例

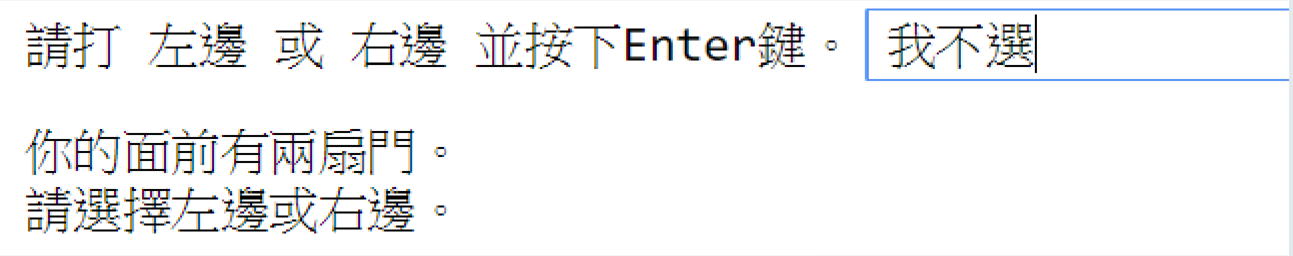


```
請打 左邊 或 右邊 並按下Enter鍵。 右邊

你的面前有兩扇門。
請選擇左邊或右邊。
```

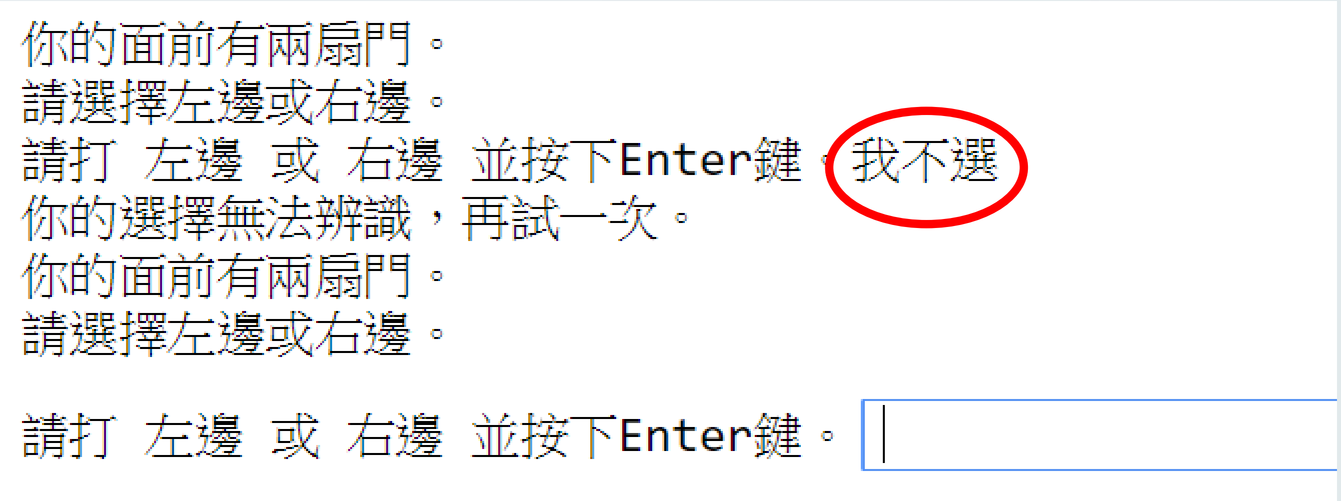


```
你的面前有兩扇門。
請選擇左邊或右邊。
請打 左邊 或 右邊 並按下Enter鍵。右邊
門後是陷阱，你已經死了！
```

def、if語法：選擇題範例



```
請打 左邊 或 右邊 並按下Enter鍵。 我不選

你的面前有兩扇門。
請選擇左邊或右邊。
```



```
你的面前有兩扇門。
請選擇左邊或右邊。
請打 左邊 或 右邊 並按下Enter鍵。我不選
你的選擇無法辨識，再試一次。
你的面前有兩扇門。
請選擇左邊或右邊。

請打 左邊 或 右邊 並按下Enter鍵。
```

if語法：選擇題

```
If  判斷式1 ：
      指令1
elif  判斷式2 ：
      指令 2
Else ：
      指令 3
```

if語法：選擇題

```
If  判斷式1 ：
        指令1
elif  判斷式2 ：
        指令 2
Else ：
        指令 3
```

注意if條件要求下的指令，
指令前要空格，以判斷段落

if語法：選擇題

```
If  判斷式1 ：
        指令1
elif  判斷式2 ：
        指令 2
Else ：
        指令 3
```

此段elif可以重複多次

判斷式符號

==	等於
!=	不等於
>	大於
<	小於
>=	大於等於
<=	小於等於

連接詞

and	且
or	或
not	非

連接詞

and	且
or	或
not	非

實務題庫：左邊還是右邊？

如果使用者輸入”左邊”或”左”

則顯示”門後有寶箱，你發財了！”

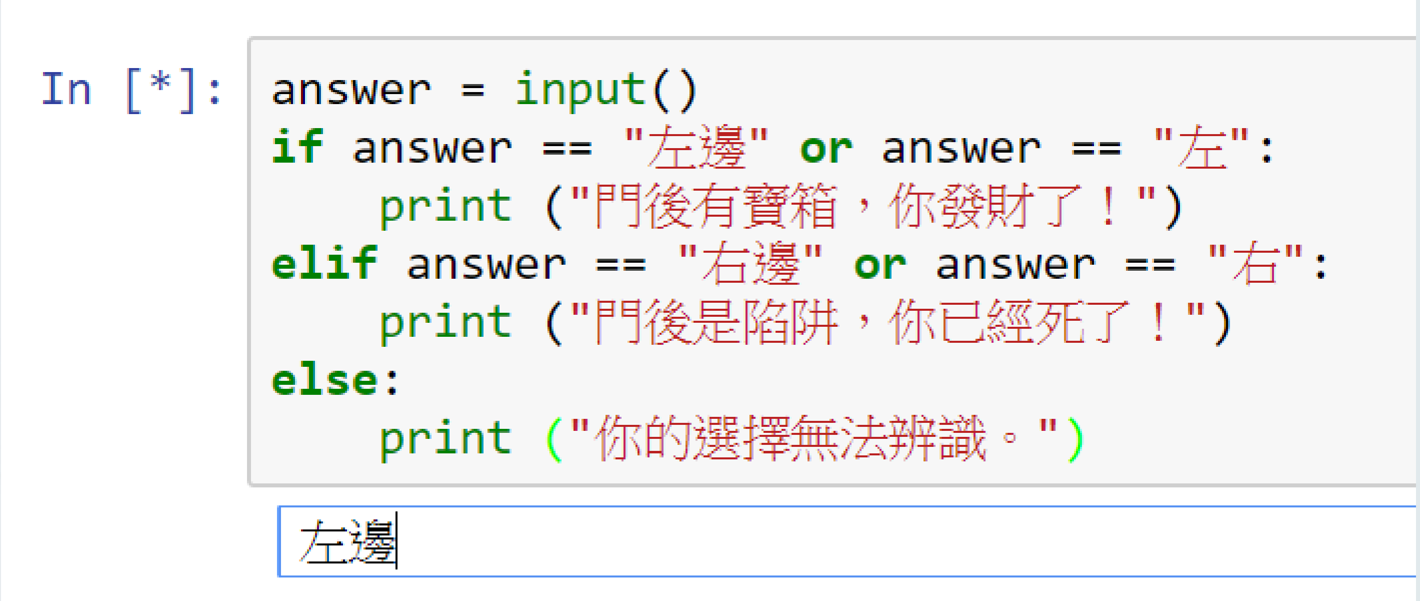
如果使用者輸入”右邊”或”右”

則顯示”門後是陷阱，你已經死了！”

如果輸入其他內容

則”你的選擇無法辨識。”

實務題庫：左邊還是右邊？



```
In [*]: answer = input()
        if answer == "左邊" or answer == "左":
            print ("門後有寶箱，你發財了！")
        elif answer == "右邊" or answer == "右":
            print ("門後是陷阱，你已經死了！")
        else:
            print ("你的選擇無法辨識。")
```

```
左邊
```

```
左邊
門後有寶箱，你發財了！
```

def語法：定義功能

Def 功能名稱(功能中要使用的變數) ：

功能指令

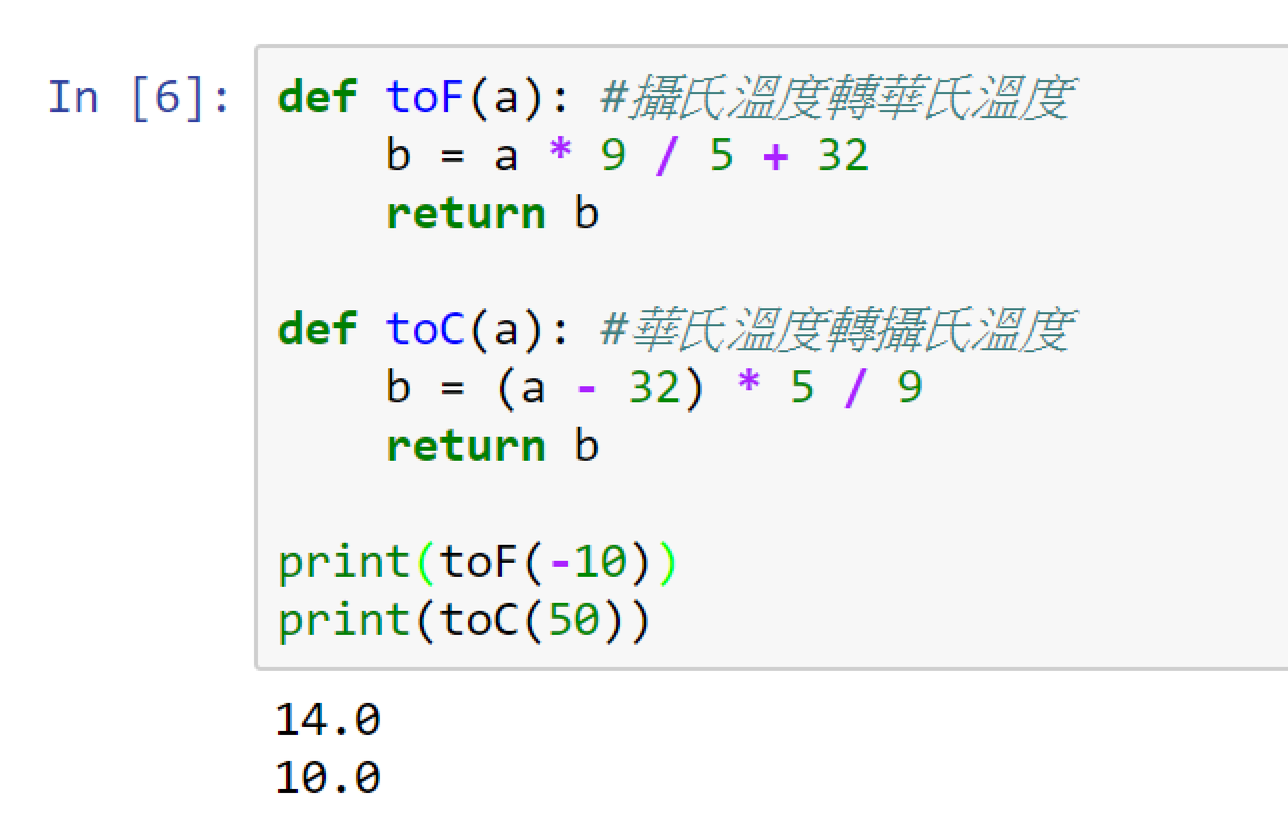
使用def語法可以將複雜的多道指令

定義為一個”功能(Function)”

可以在之後重複使用

return	**通常用於def結尾，可以直接將值回傳成功能結果**

def語法：範例—溫度轉換



```
In [6]: def toF(a): #攝氏溫度轉華氏溫度
            b = a * 9 / 5 + 32
            return b

        def toC(a): #華氏溫度轉攝氏溫度
            b = (a - 32) * 5 / 9
            return b

        print(toF(-10))
        print(toC(50))

        14.0
        10.0
```

實務題庫：選擇題防呆機制

延伸前面if語法的左右選擇題

我們希望可以設計防呆機制

讓使用者重複直到選”左”或”右”為止

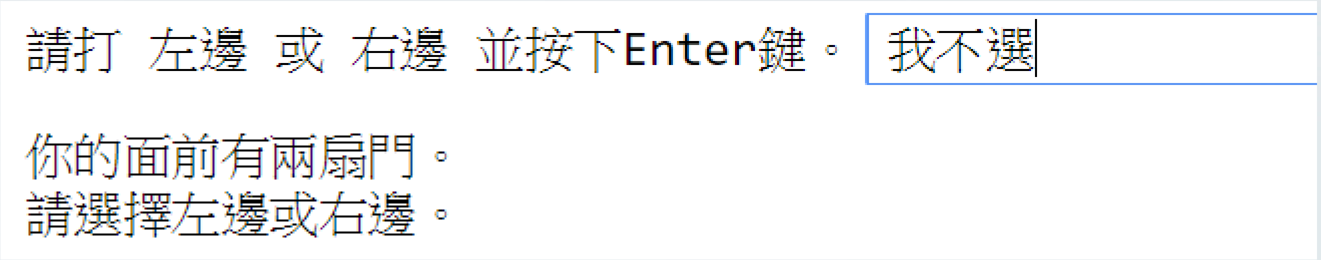
實務題庫：選擇題防呆機制

延伸前面if語法的左右選擇題

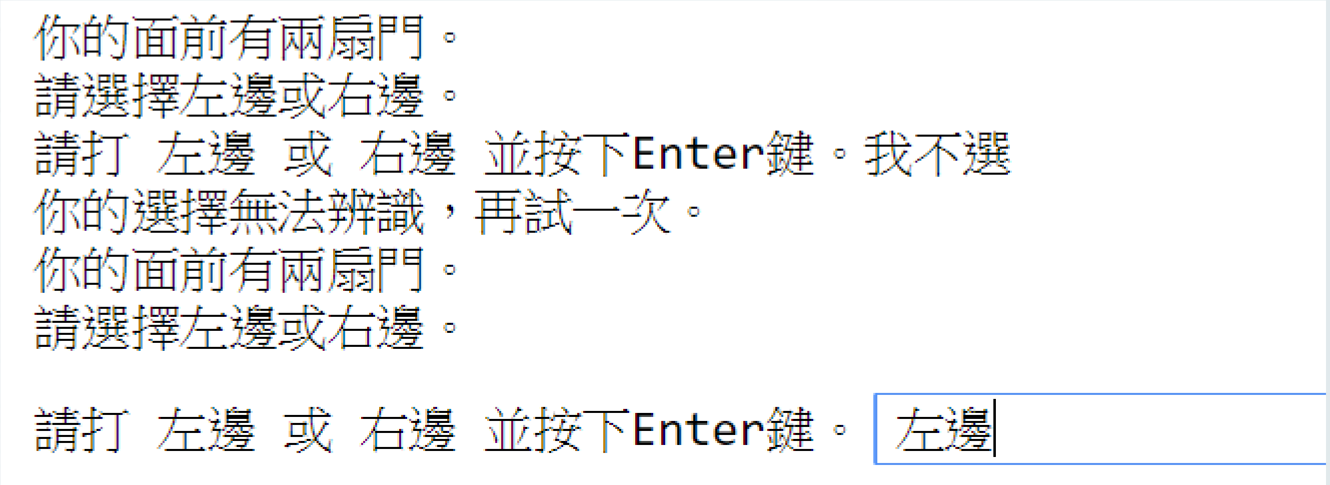
我們希望可以設計防呆機制

讓使用者重複直到選”左”或”右”為止

實務題庫：選擇題防呆機制

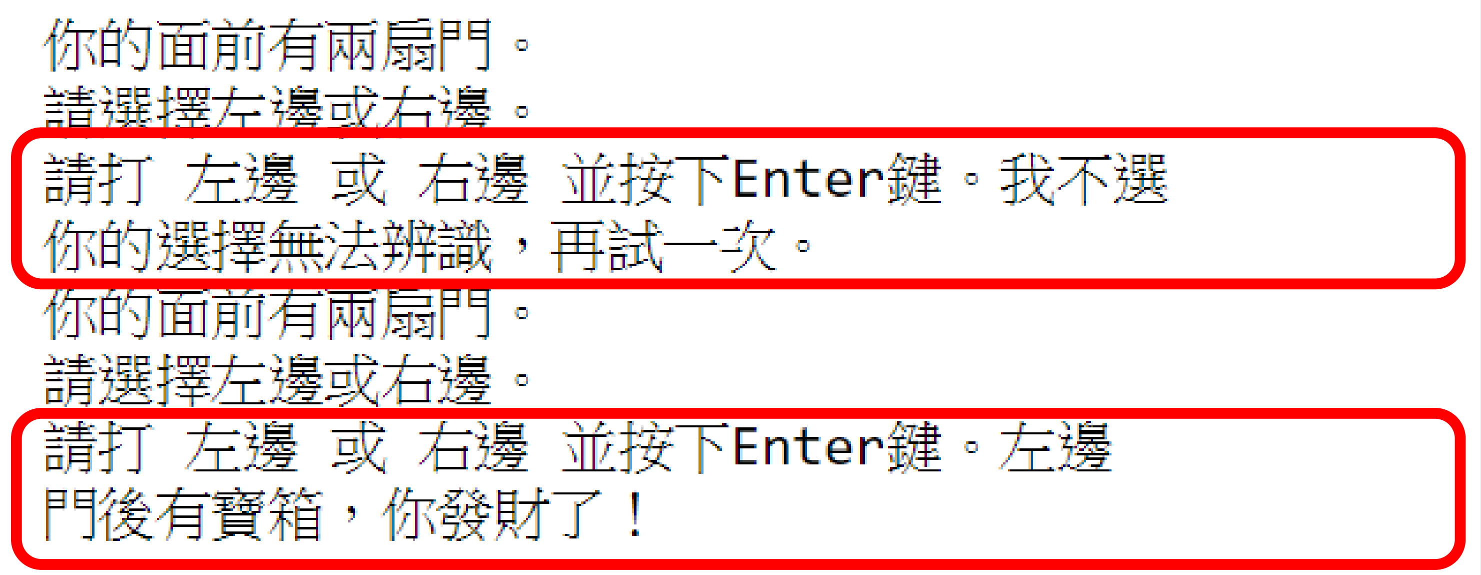


```
請打 左邊 或 右邊 並按下Enter鍵。 我不選

你的面前有兩扇門。
請選擇左邊或右邊。
```



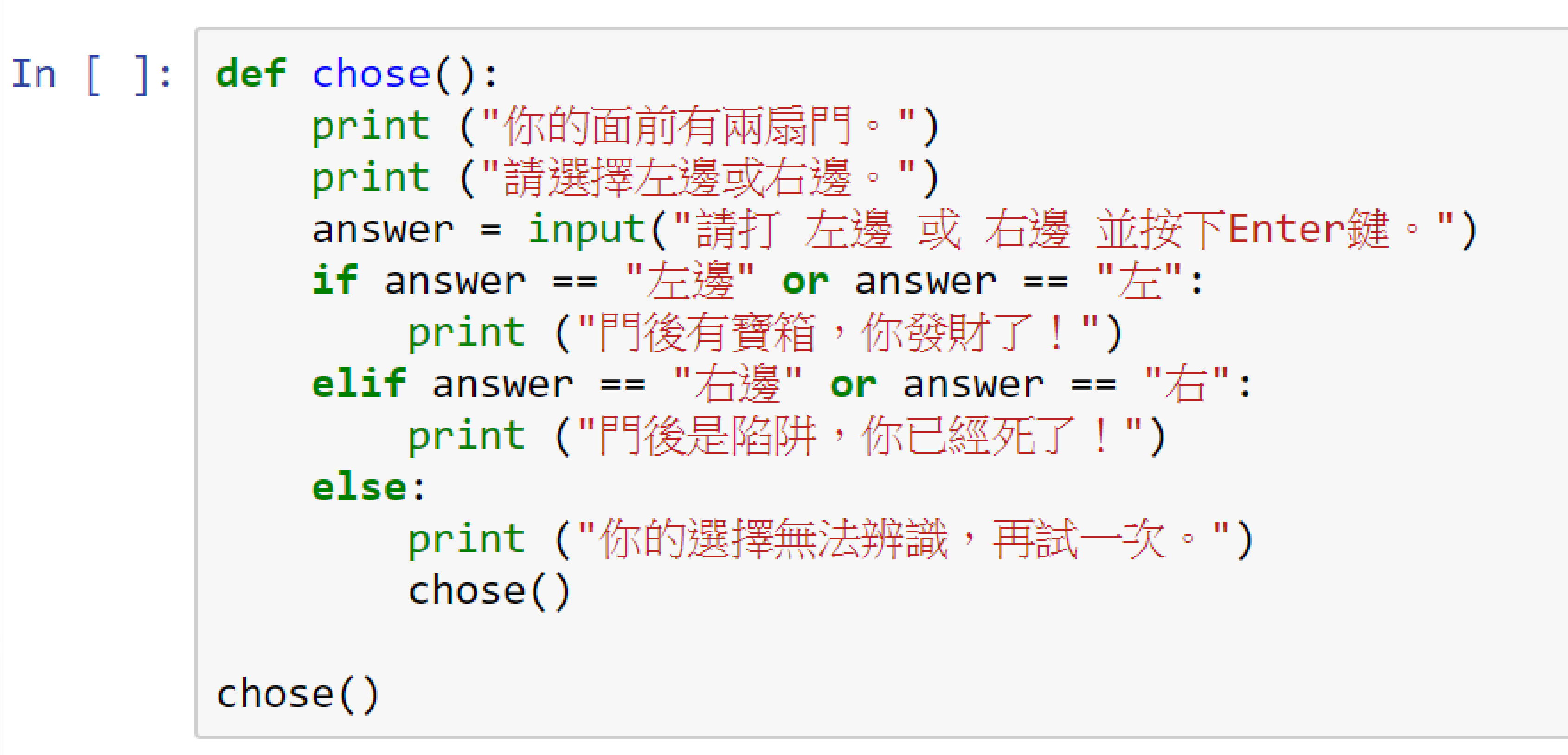
```
你的面前有兩扇門。
請選擇左邊或右邊。
請打 左邊 或 右邊 並按下Enter鍵。我不選
你的選擇無法辨識，再試一次。
你的面前有兩扇門。
請選擇左邊或右邊。

請打 左邊 或 右邊 並按下Enter鍵。 左邊
```

實務題庫：選擇題防呆機制



```
你的面前有兩扇門。
請選擇左邊或右邊。
請打 左邊 或 右邊 並按下Enter鍵。我不選
你的選擇無法辨識，再試一次。
你的面前有兩扇門。
請選擇左邊或右邊。
請打 左邊 或 右邊 並按下Enter鍵。左邊
門後有寶箱，你發財了！
```

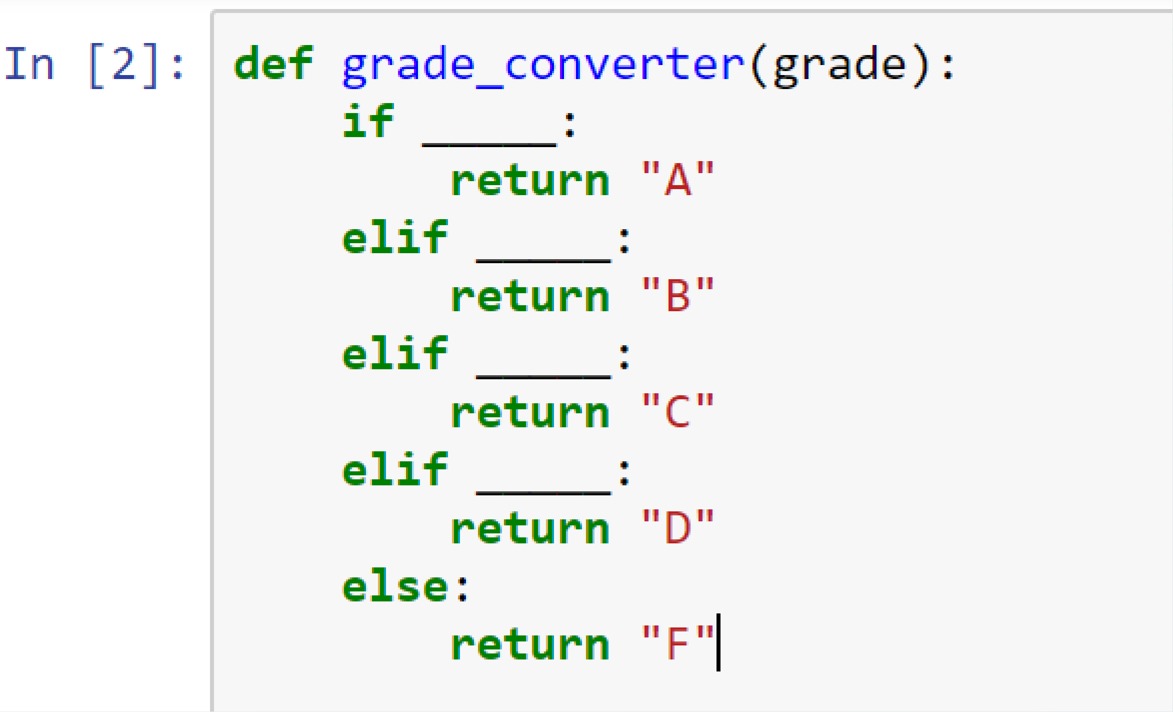
實務題庫：選擇題防呆機制



In [ ]:

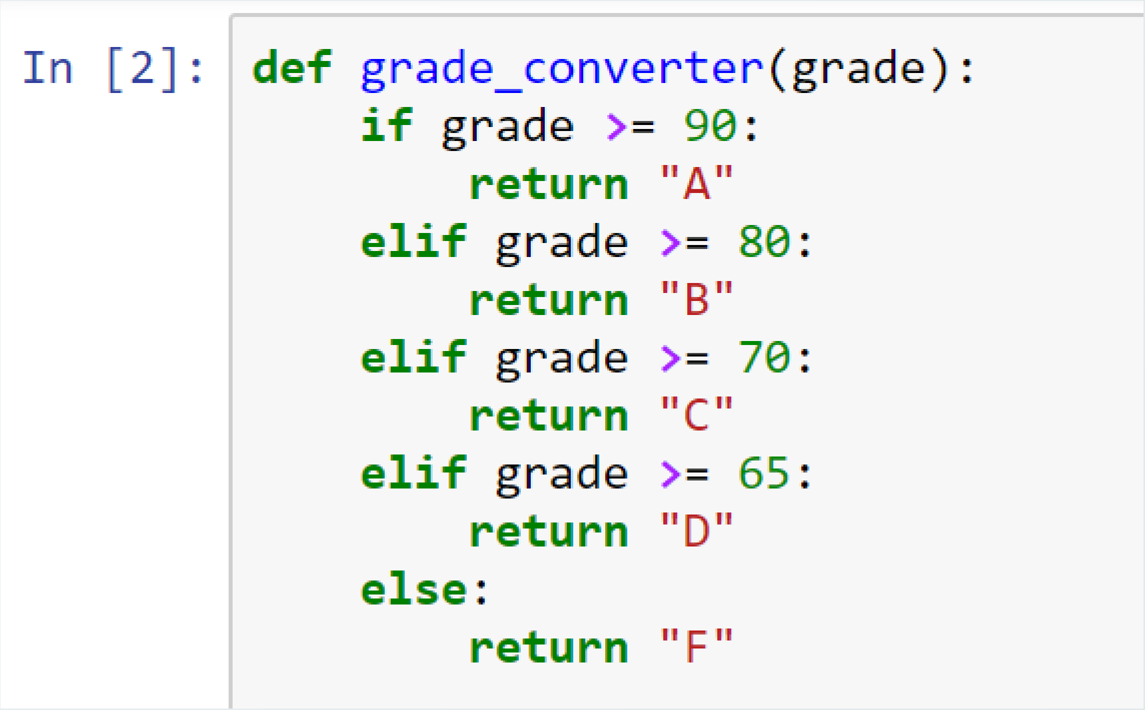
```
def chose():
    print ("你的面前有兩扇門。")
    print ("請選擇左邊或右邊。")
    answer = input("請打 左邊 或 右邊 並按下Enter鍵。")
    if answer == "左邊" or answer == "左":
        print ("門後有寶箱，你發財了！")
    elif answer == "右邊" or answer == "右":
        print ("門後是陷阱，你已經死了！")
    else:
        print ("你的選擇無法辨識，再試一次。")
        chose()

chose()
```

實務題庫：成績轉換工具



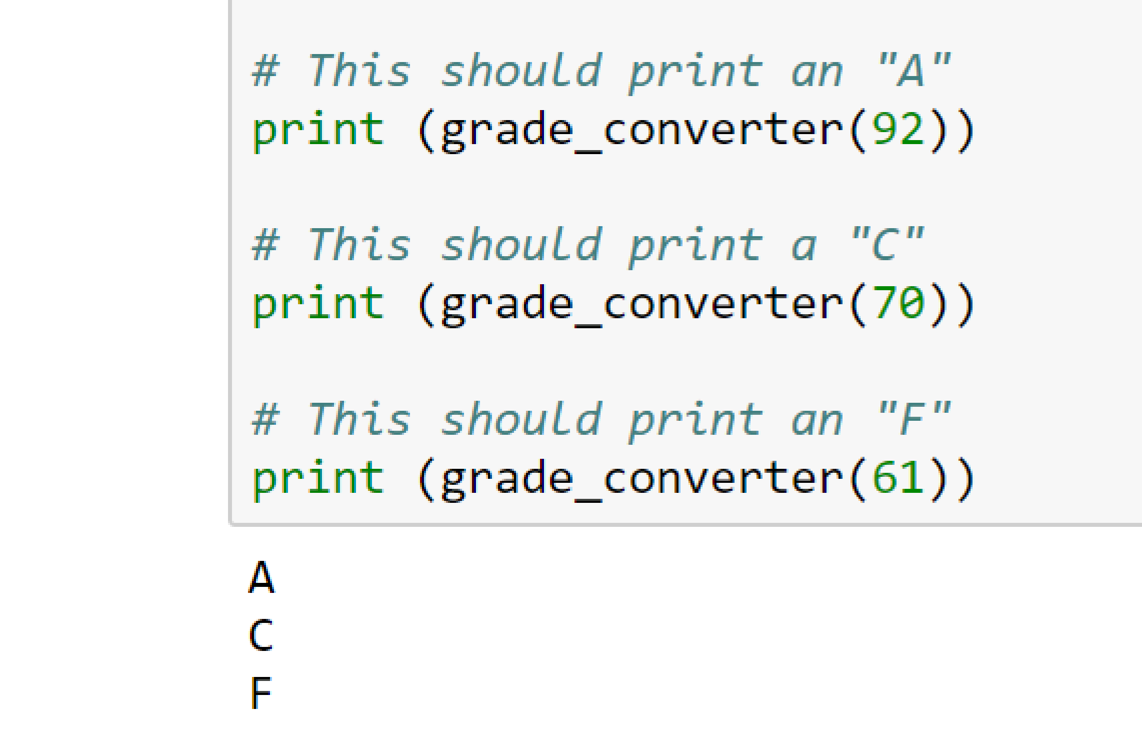
```
In [2]:  def grade_converter(grade):
             if _____:
                 return "A"
             elif _____:
                 return "B"
             elif _____:
                 return "C"
             elif _____:
                 return "D"
             else:
                 return "F"
```

實務題庫：成績轉換工具



```
In [2]: def grade_converter(grade):
            if grade >= 90:
                return "A"
            elif grade >= 80:
                return "B"
            elif grade >= 70:
                return "C"
            elif grade >= 65:
                return "D"
            else:
                return "F"
```

實務題庫：成績轉換工具



```
# This should print an "A"
print (grade_converter(92))

# This should print a "C"
print (grade_converter(70))

# This should print an "F"
print (grade_converter(61))
```

```
A
C
F
```

For迴圈：針對個別物件執行

for 自訂變數 in 物件組合：

功能指令

使用for迴圈可以將重複的動作

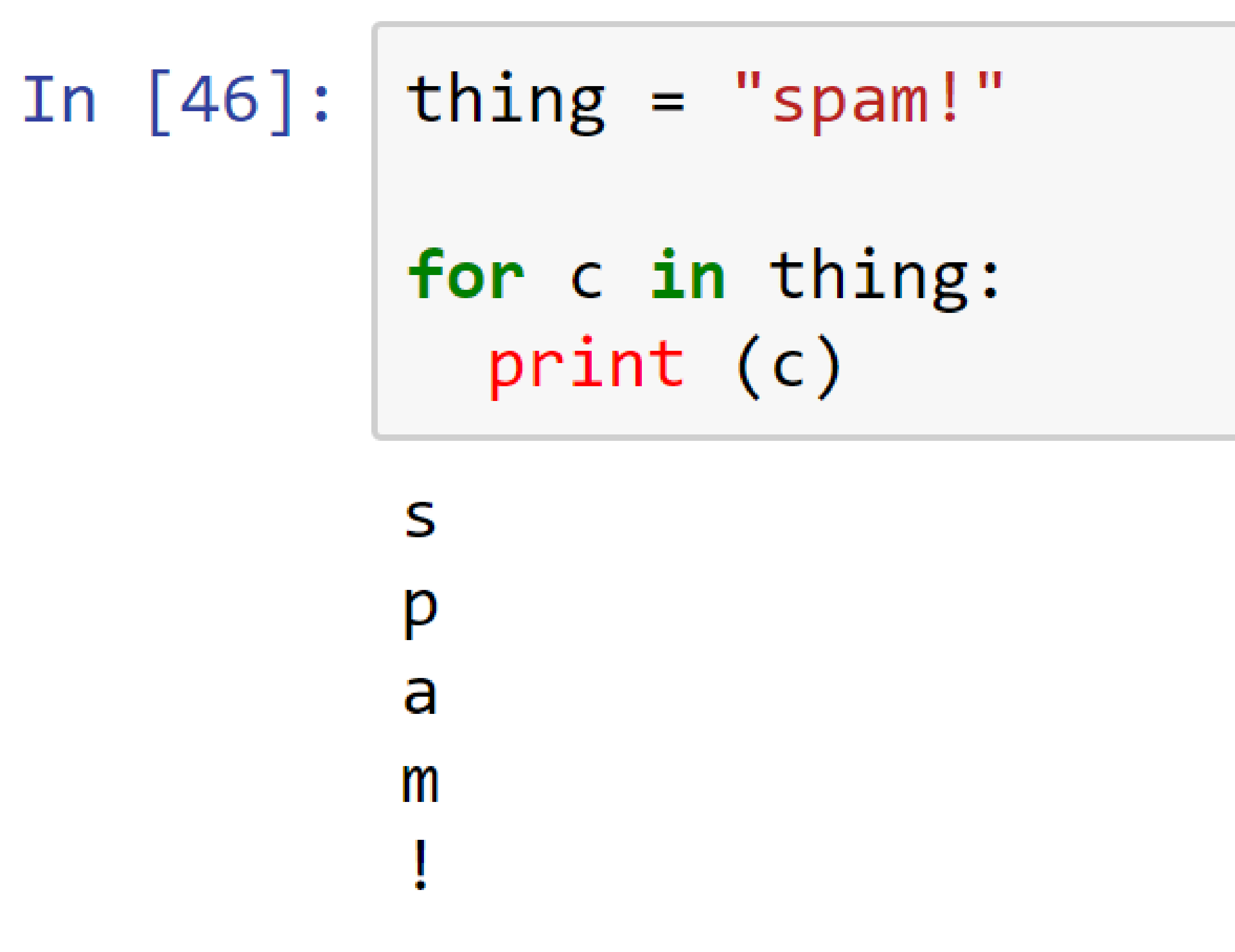
針對物件集合的個別物件自動重複執行

break	用於特定條件直接打破迴圈

For 迴圈

物件組合	個別物件
字串	字
list	物件
dictionary	key

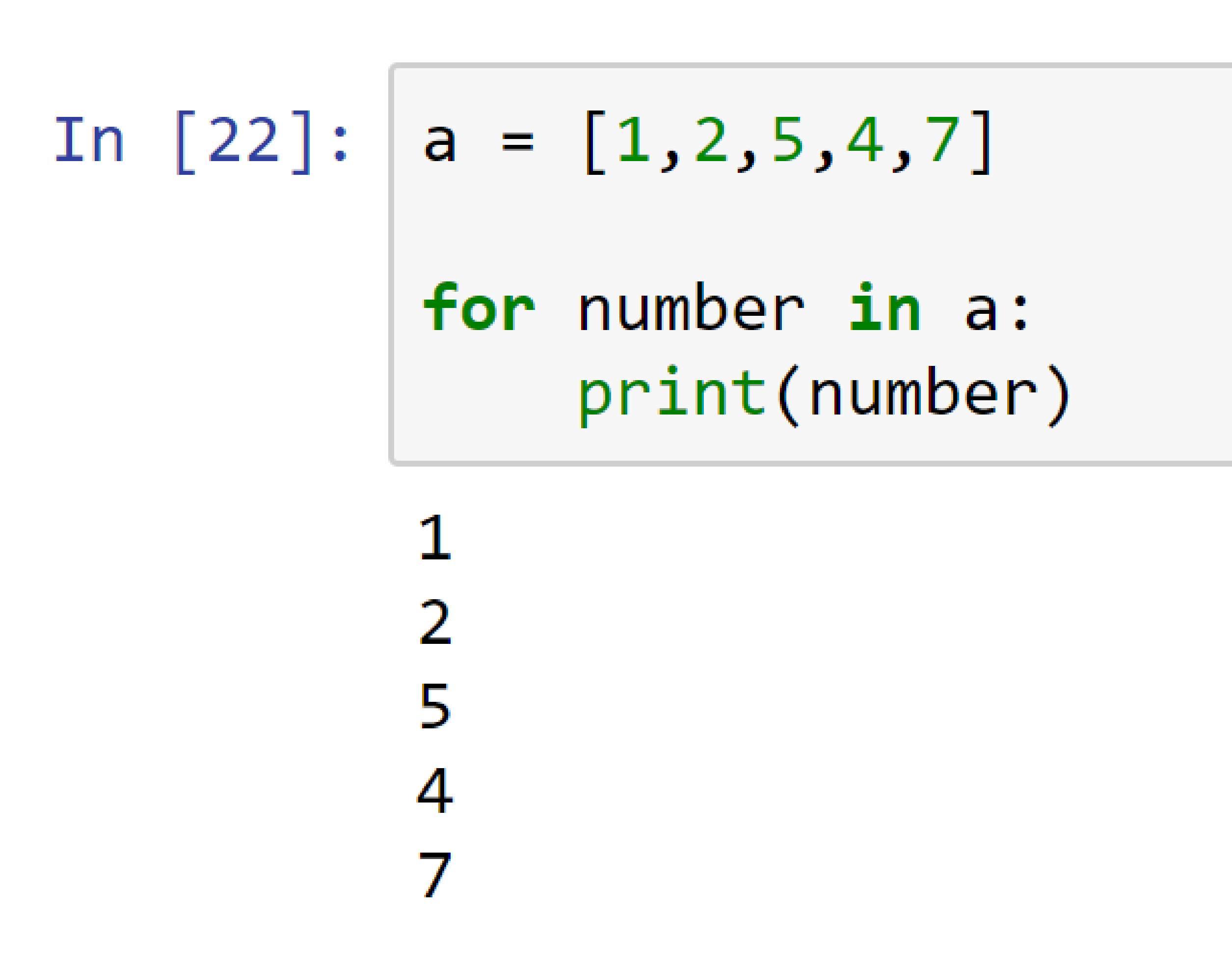
For 迴圈



```
In [46]: thing = "spam!"

         for c in thing:
           print (c)
```

```
s
p
a
m
!
```

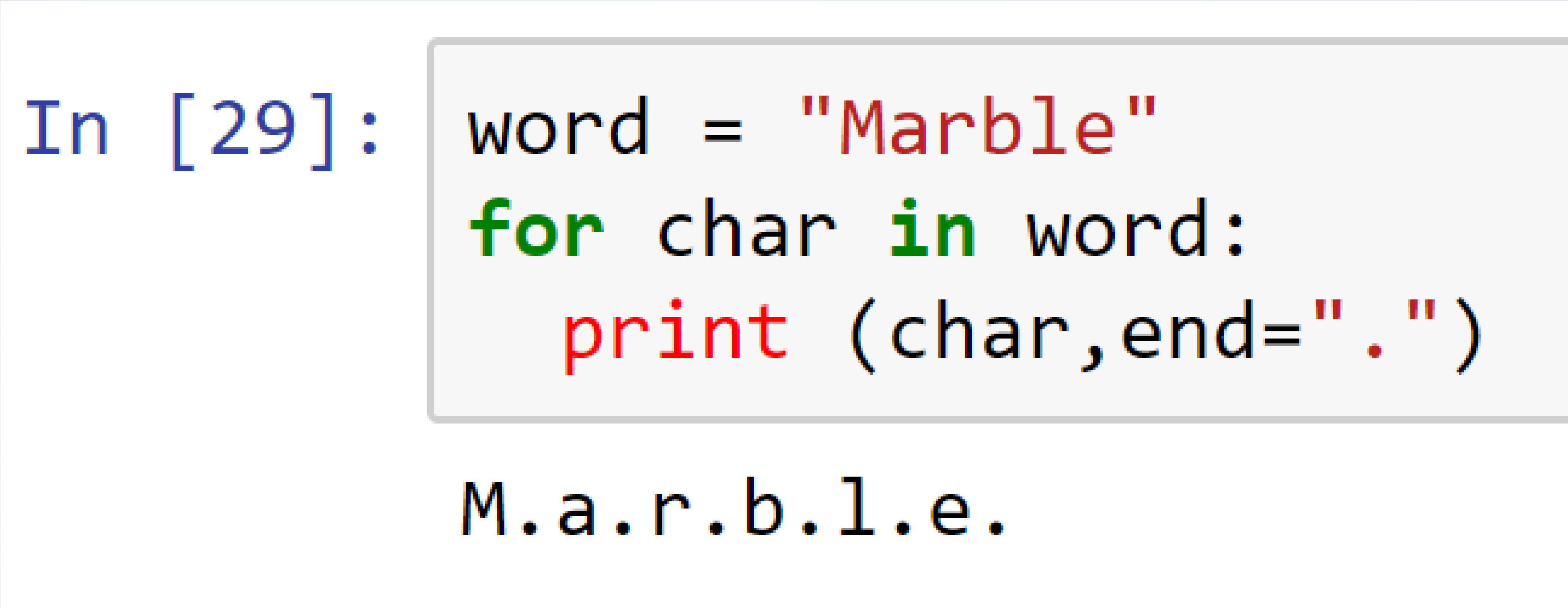
For 迴圈



```python
a = [1,2,5,4,7]

for number in a:
    print(number)
```

```
1
2
5
4
7
```

For 迴圈



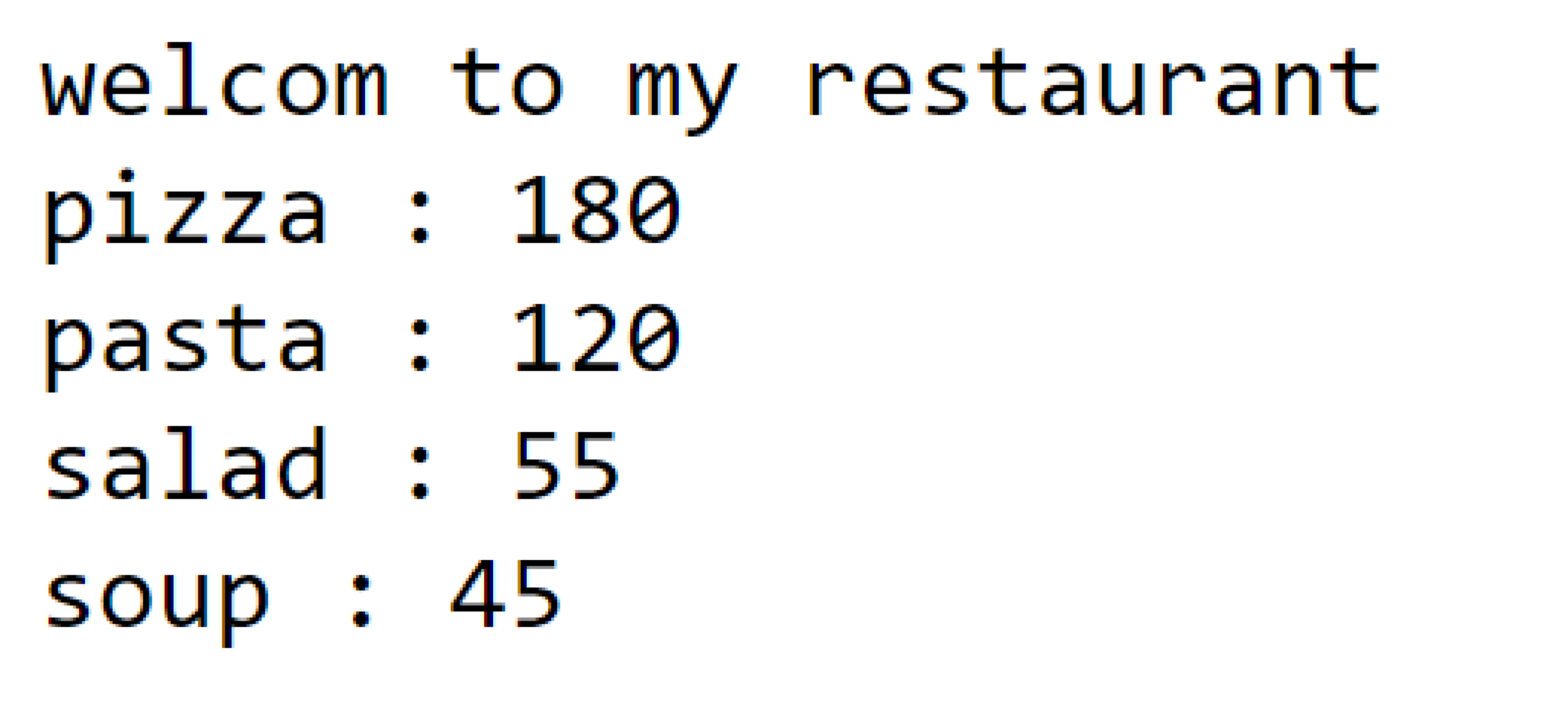
In [29]:

```
word = "Marble"
for char in word:
  print (char,end=".")
```

```
M.a.r.b.l.e.
```

Print(物件,end=“取代跳行符號”)

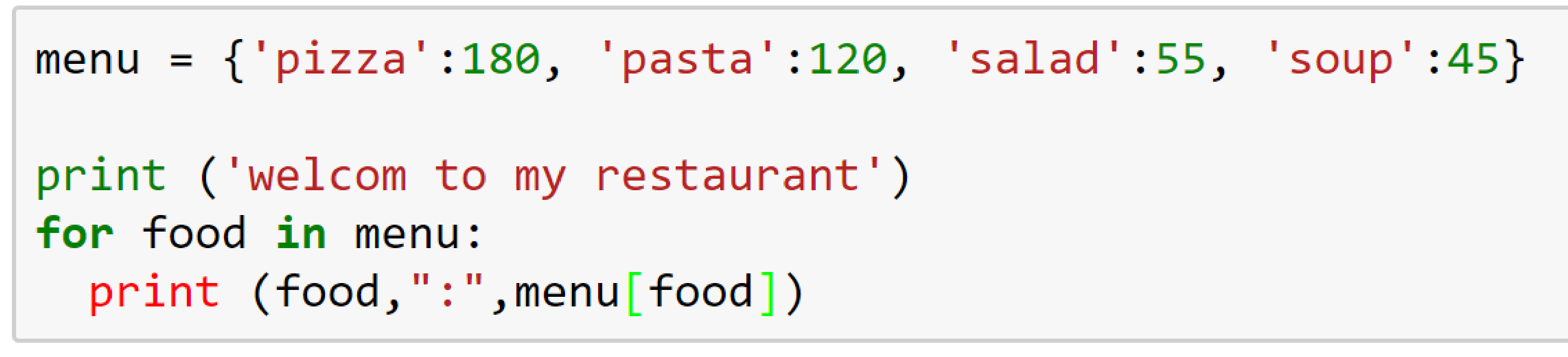
For迴圈 實務題庫



```
welcom to my restaurant
pizza : 180
pasta : 120
salad : 55
soup : 45
```

With a dictionary!!

For迴圈　實務題庫



```
menu = {'pizza':180, 'pasta':120, 'salad':55, 'soup':45}

print ('welcom to my restaurant')
for food in menu:
  print (food,":",menu[food])
```

While迴圈：符合條件就重複

while 判斷式 ：

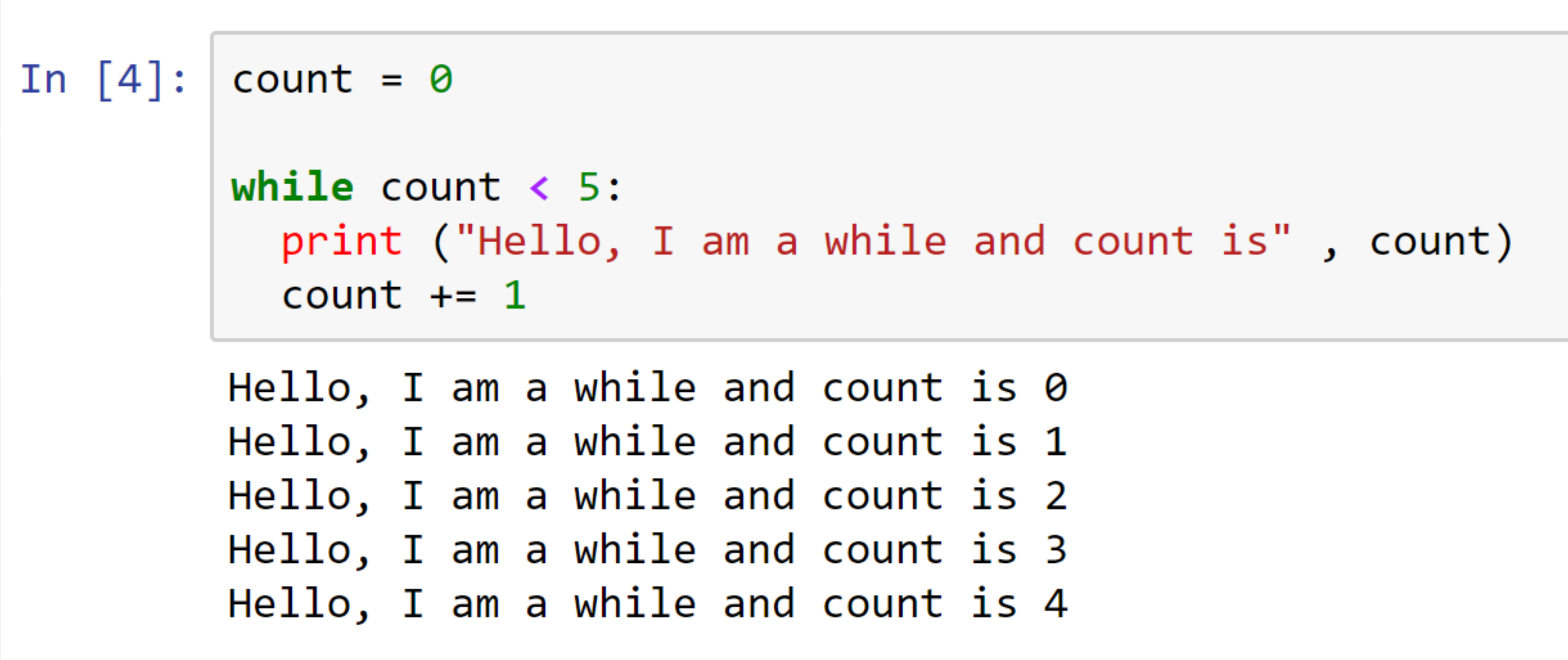
功能指令

使用while迴圈在特定條件下

不停重複此動作

break	用於特定條件直接打破迴圈

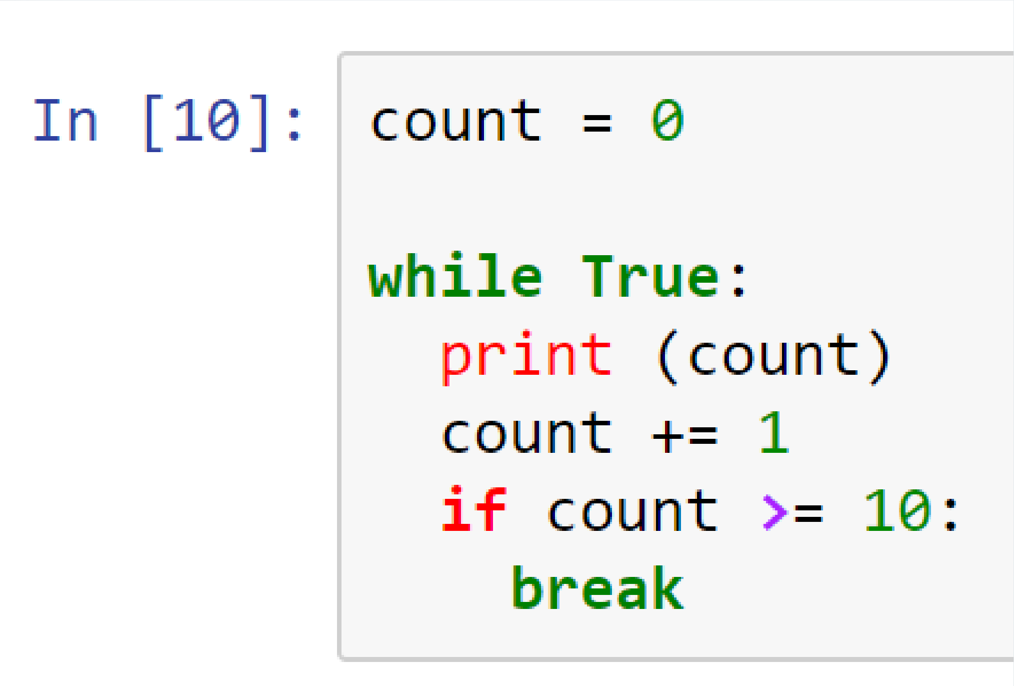
while 迴圈



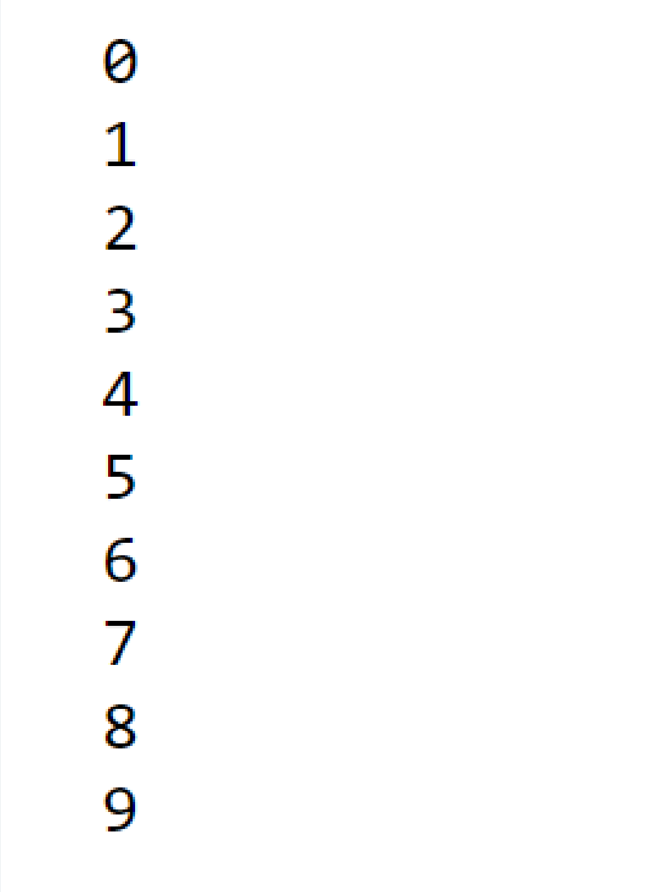
```python
count = 0

while count < 5:
  print ("Hello, I am a while and count is" , count)
  count += 1
```

```
Hello, I am a while and count is 0
Hello, I am a while and count is 1
Hello, I am a while and count is 2
Hello, I am a while and count is 3
Hello, I am a while and count is 4
```

while 迴圈



```
In [10]: count = 0

         while True:
           print (count)
           count += 1
           if count >= 10:
             break
```



```
0
1
2
3
4
5
6
7
8
9
```

While迴圈　實務題庫

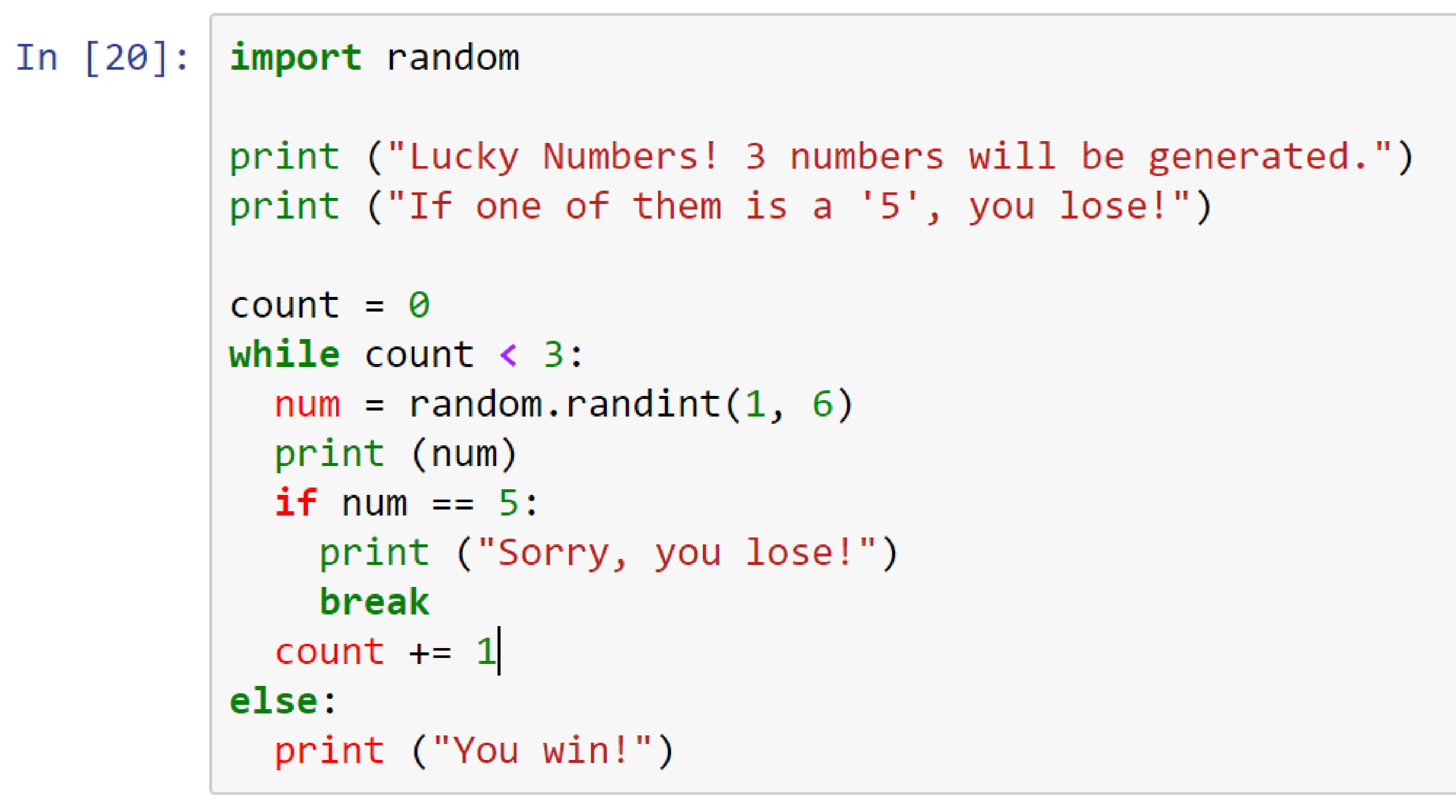
Lucky Numbers! 3 numbers will be generated.

If one of them is a '5', you lose!

```
import random
random.randint(1,6)
```

製造一個1到6之間的隨機整數

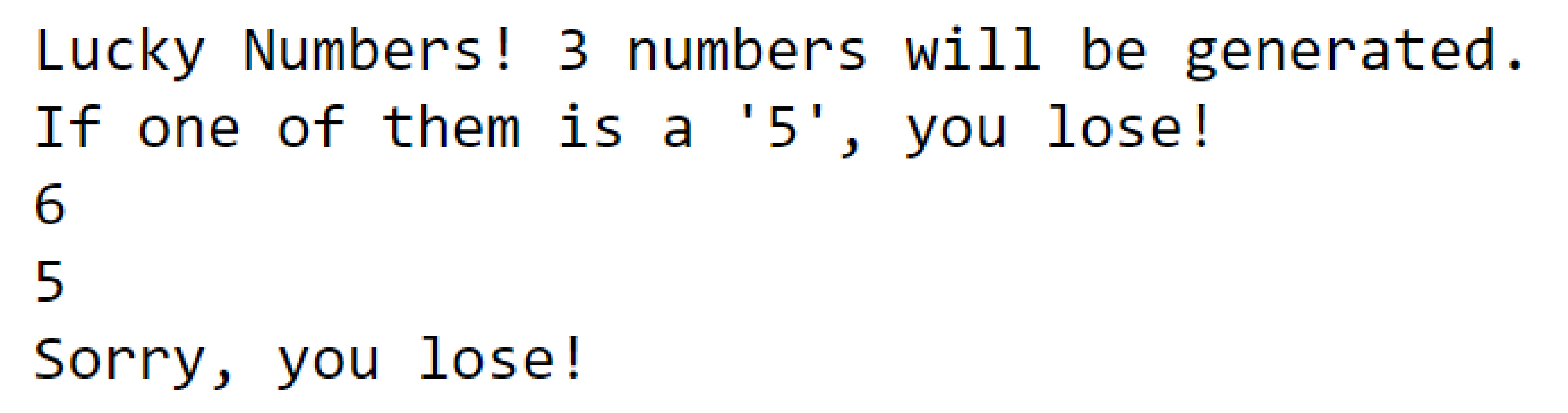
While迴圈　實務題庫



In [20]:

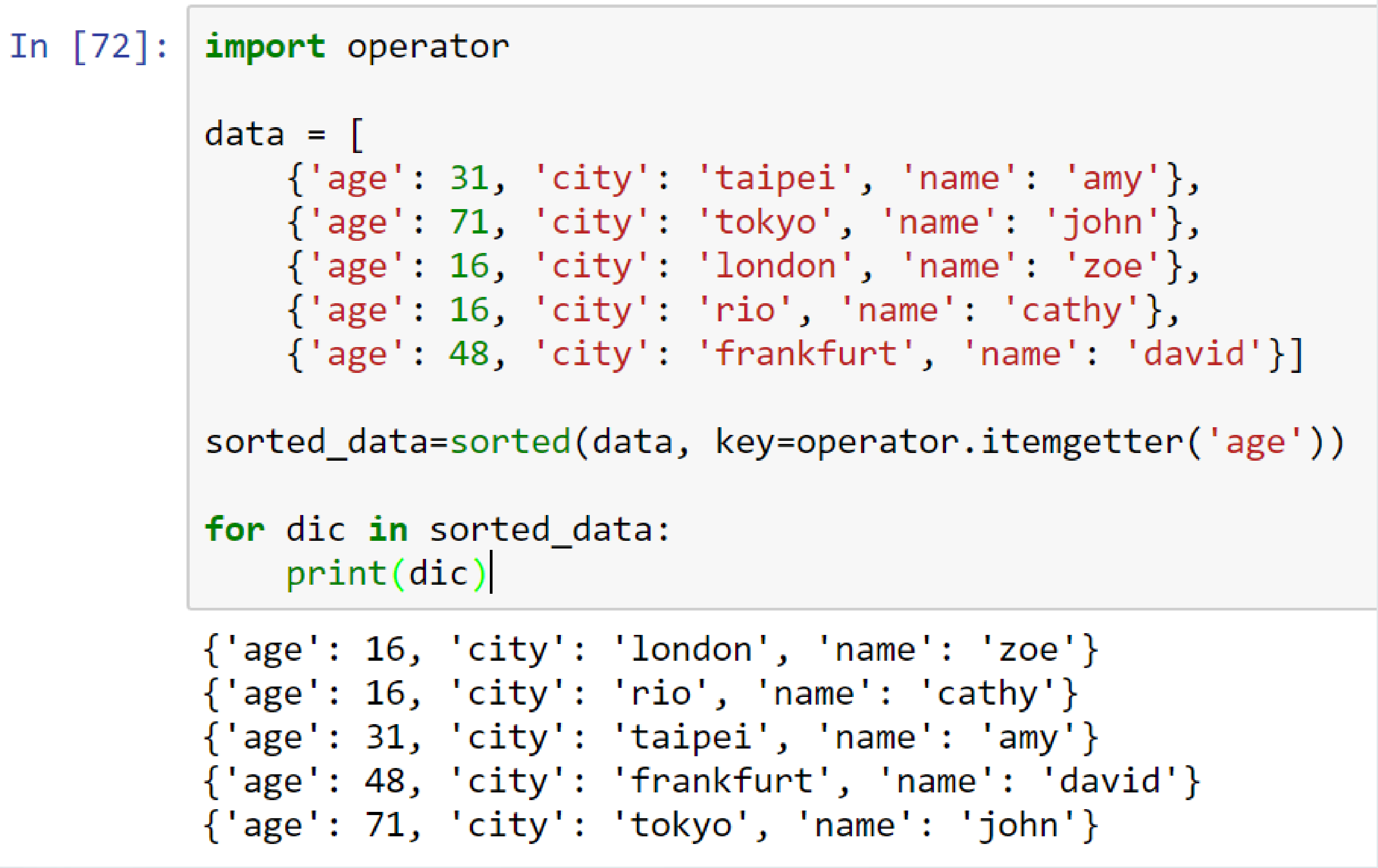
```python
import random

print ("Lucky Numbers! 3 numbers will be generated.")
print ("If one of them is a '5', you lose!")

count = 0
while count < 3:
  num = random.randint(1, 6)
  print (num)
  if num == 5:
    print ("Sorry, you lose!")
    break
  count += 1
else:
  print ("You win!")
```

While迴圈　實務題庫



```
Lucky Numbers! 3 numbers will be generated.
If one of them is a '5', you lose!
6
5
Sorry, you lose!
```

Operator + sorted



```
import operator

data = [
    {'age': 31, 'city': 'taipei', 'name': 'amy'},
    {'age': 71, 'city': 'tokyo', 'name': 'john'},
    {'age': 16, 'city': 'london', 'name': 'zoe'},
    {'age': 16, 'city': 'rio', 'name': 'cathy'},
    {'age': 48, 'city': 'frankfurt', 'name': 'david'}]

sorted_data=sorted(data, key=operator.itemgetter('age'))

for dic in sorted_data:
    print(dic)
```

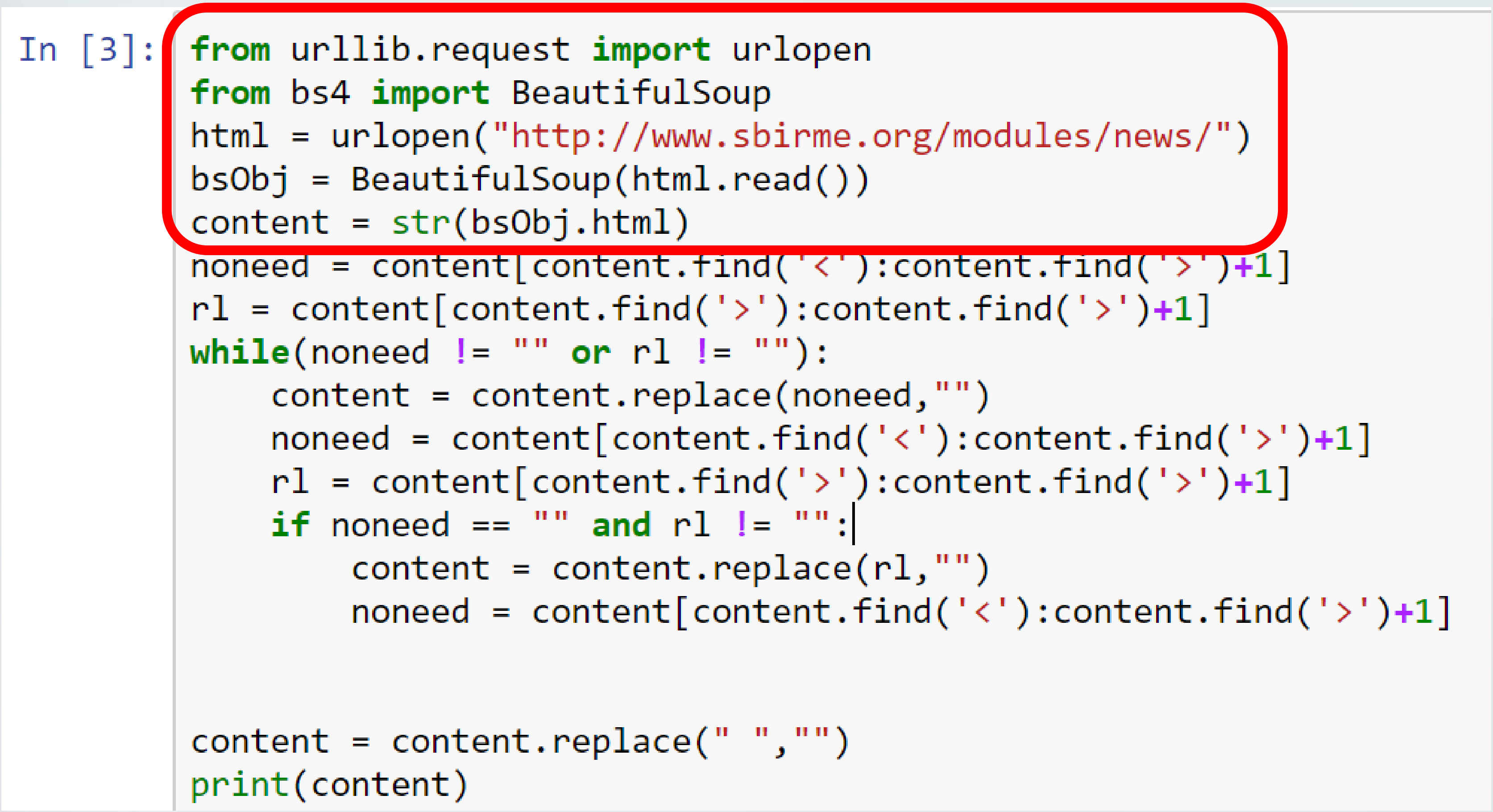
```
{'age': 16, 'city': 'london', 'name': 'zoe'}
{'age': 16, 'city': 'rio', 'name': 'cathy'}
{'age': 31, 'city': 'taipei', 'name': 'amy'}
{'age': 48, 'city': 'frankfurt', 'name': 'david'}
{'age': 71, 'city': 'tokyo', 'name': 'john'}
```

網路爬蟲 常用抓資料指令

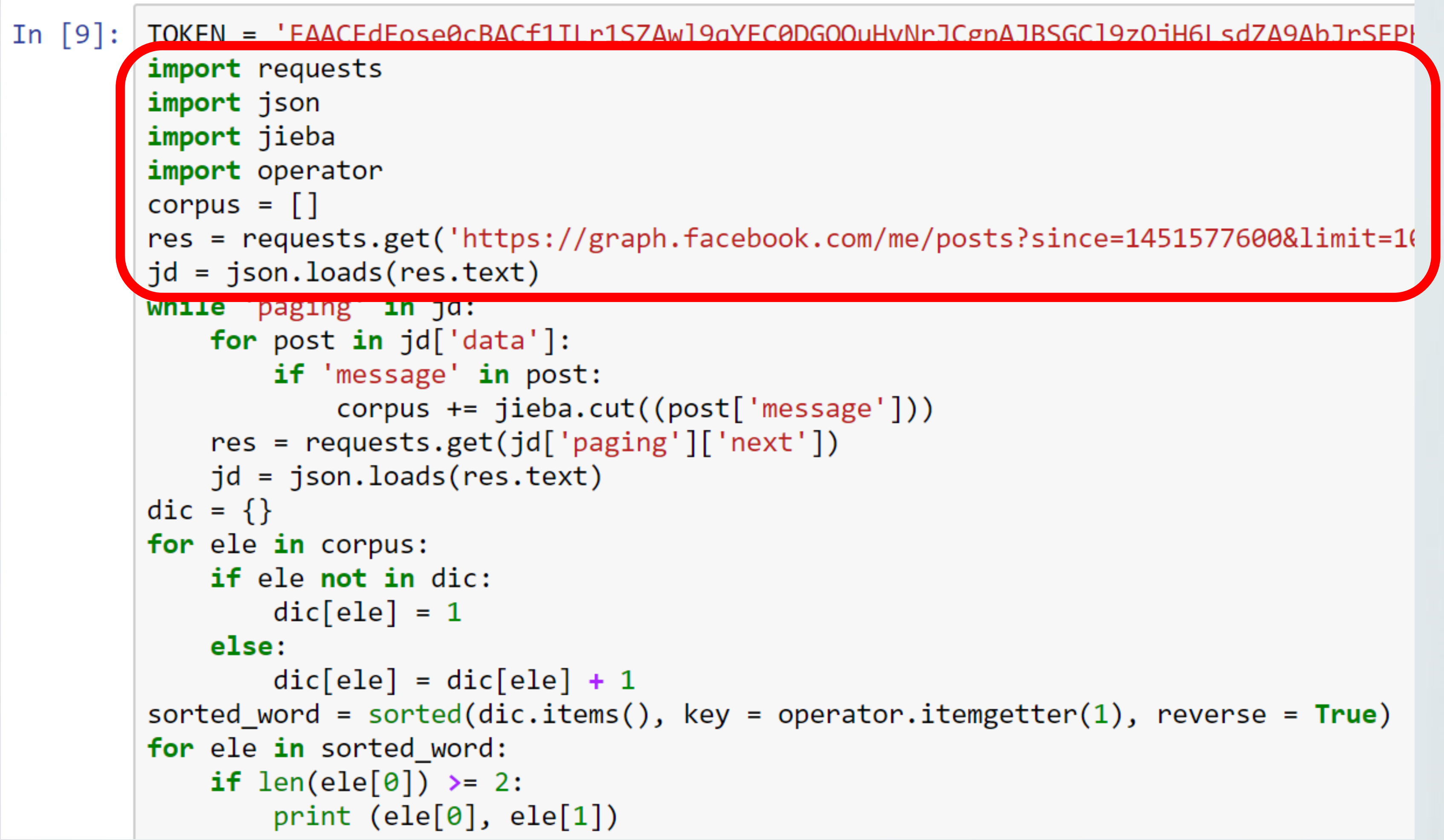
Requests.get

urlopen

Beautifulsoup



```
In [3]: from urllib.request import urlopen
        from bs4 import BeautifulSoup
        html = urlopen("http://www.sbirme.org/modules/news/")
        bsObj = BeautifulSoup(html.read())
        content = str(bsObj.html)
        noneed = content[content.find('<'):content.find('>')+1]
        rl = content[content.find('>'):content.find('>')+1]
        while(noneed != "" or rl != ""):
            content = content.replace(noneed,"")
            noneed = content[content.find('<'):content.find('>')+1]
            rl = content[content.find('>'):content.find('>')+1]
            if noneed == "" and rl != "":
                content = content.replace(rl,"")
                noneed = content[content.find('<'):content.find('>')+1]


        content = content.replace(" ","")
        print(content)
```



In [9]:

```
TOKEN = 'EAACEdEose0cBACf1ILr1SZAwl9gYEC0DGOQuHyNrJCgpAJBSGCl9zOjH6LsdZA9AbJrSEP
import requests
import json
import jieba
import operator
corpus = []
res = requests.get('https://graph.facebook.com/me/posts?since=1451577600&limit=1
jd = json.loads(res.text)
while 'paging' in jd:
    for post in jd['data']:
        if 'message' in post:
            corpus += jieba.cut((post['message']))
    res = requests.get(jd['paging']['next'])
    jd = json.loads(res.text)
dic = {}
for ele in corpus:
    if ele not in dic:
        dic[ele] = 1
    else:
        dic[ele] = dic[ele] + 1
sorted_word = sorted(dic.items(), key = operator.itemgetter(1), reverse = True)
for ele in sorted_word:
    if len(ele[0]) >= 2:
        print (ele[0], ele[1])
```