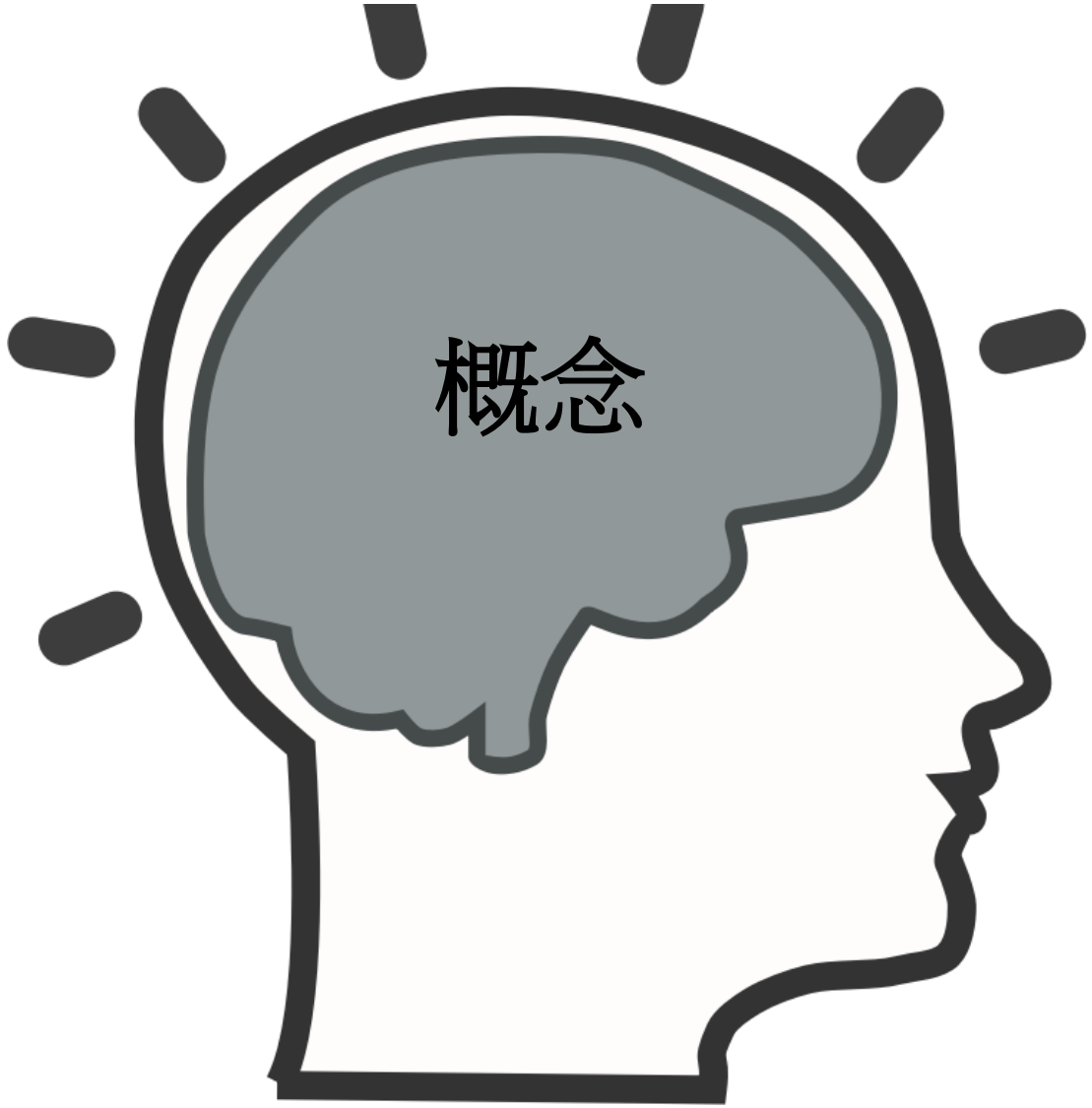
Python

資料也有家

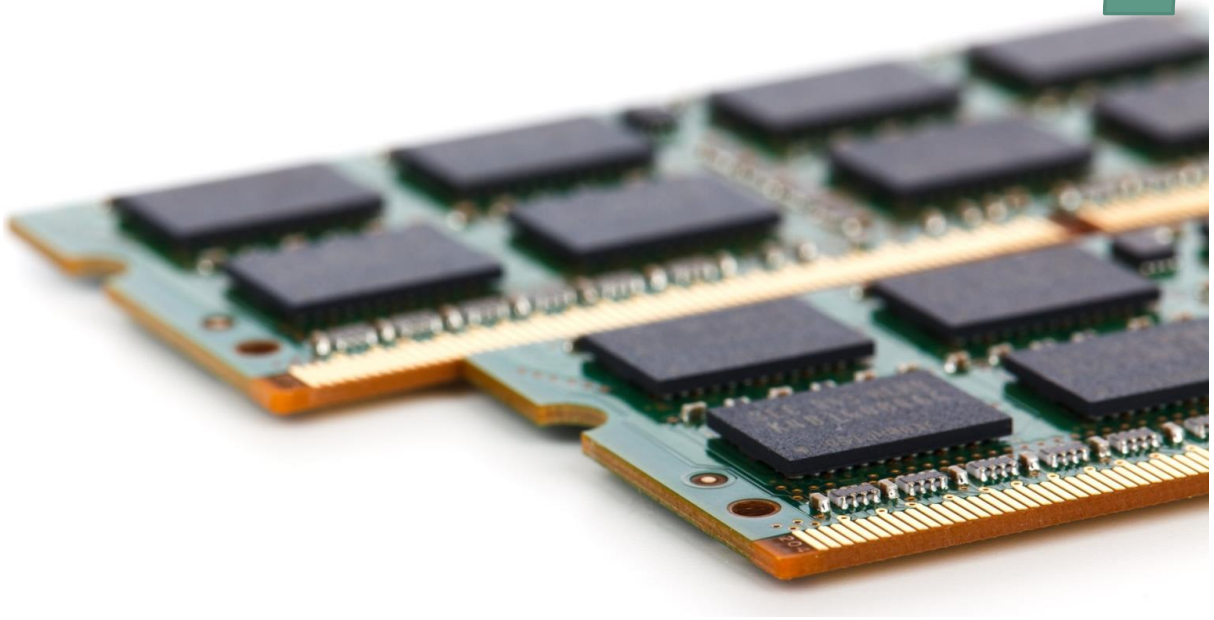
寫一個程式，

在記憶體置放一個整數3，

然後印出來。



資料放在電腦的記憶體，也必須先占地爲王



位置	存放的資料
第一個櫃子	3

程式碼

- a = 3 # 註解1 ,註解2：請看下頁說明
- print(a) # 註解3 ：請看下下頁說明

註解1：=的用法

輸入(寫法)

- a = 3

這裡的=不是等號，是指定運算子

解讀(功能)

將右邊的內容給左邊的內容，可以解讀成得到

執行(結果)

a得到3

註解2： a=3

- 代碼意義
  - 代表向電腦申請 一個櫃子 準備用來存放數據

- a是資料家的名字，

  也就是所謂的變數

- 3 是櫃子內存放的資料

櫃子	存放的資料
a	3

如何給資料的家取名

- 原則上隨便取
  - 可以是單獨的字母：a、b、c
  - 可以是多個字母的結合：aaa、bbb、ccc
  - 也可以是混數字的：number1、number2
- 例如：
  a = 3;
  num = 3.6
  num1 = 45.888
  num2 = 150

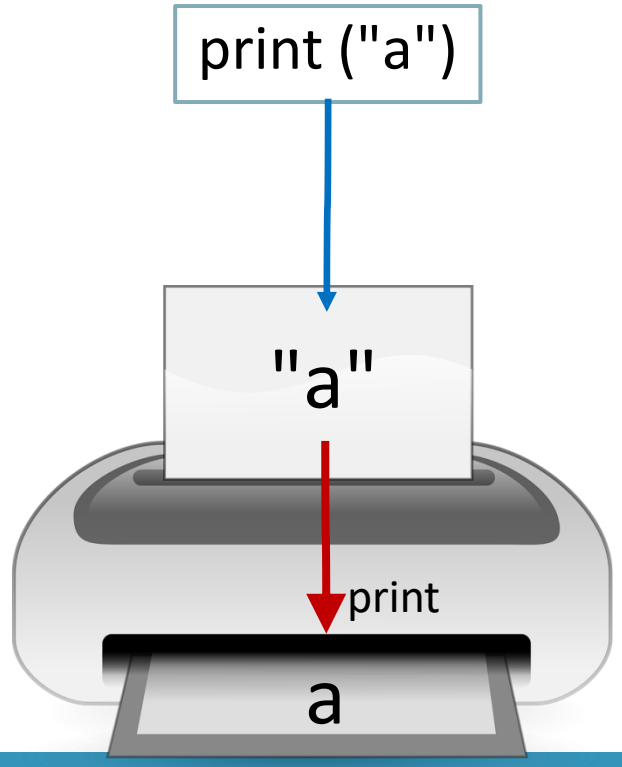
資料內容的類型

- a=3
- 3 是資料內容
- 資料可以分成不同類型(**資料型態**)，

有整數、小數(浮點數)與文字等

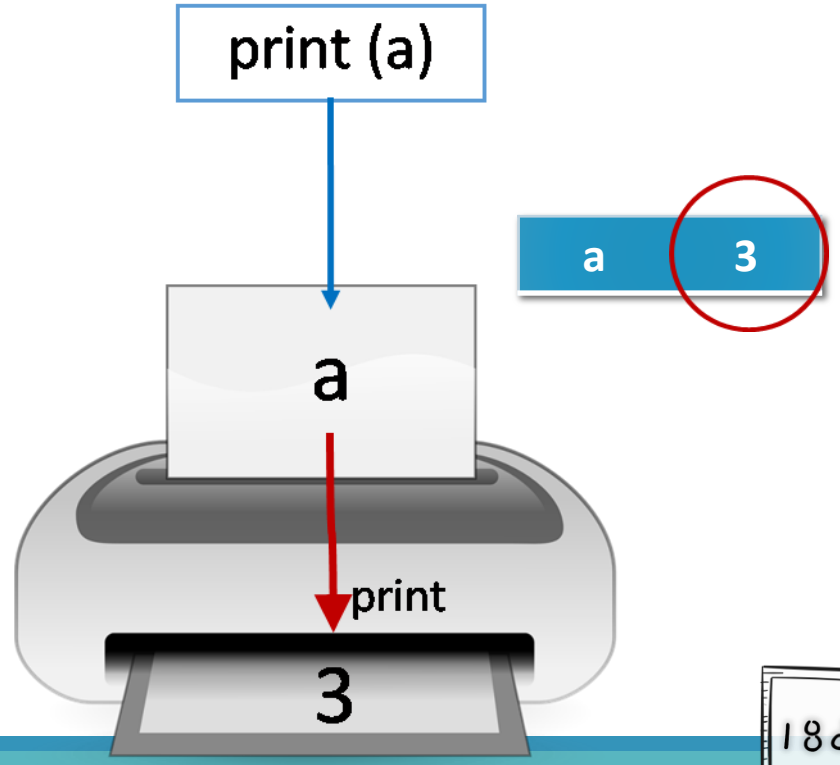
註解3：關於資料的輸出

- 雙引號內的會視為一般文字印出



- 不是放在雙引號內的會視為變數，程式會先找變數存放的值，再印出找到的那個存放值

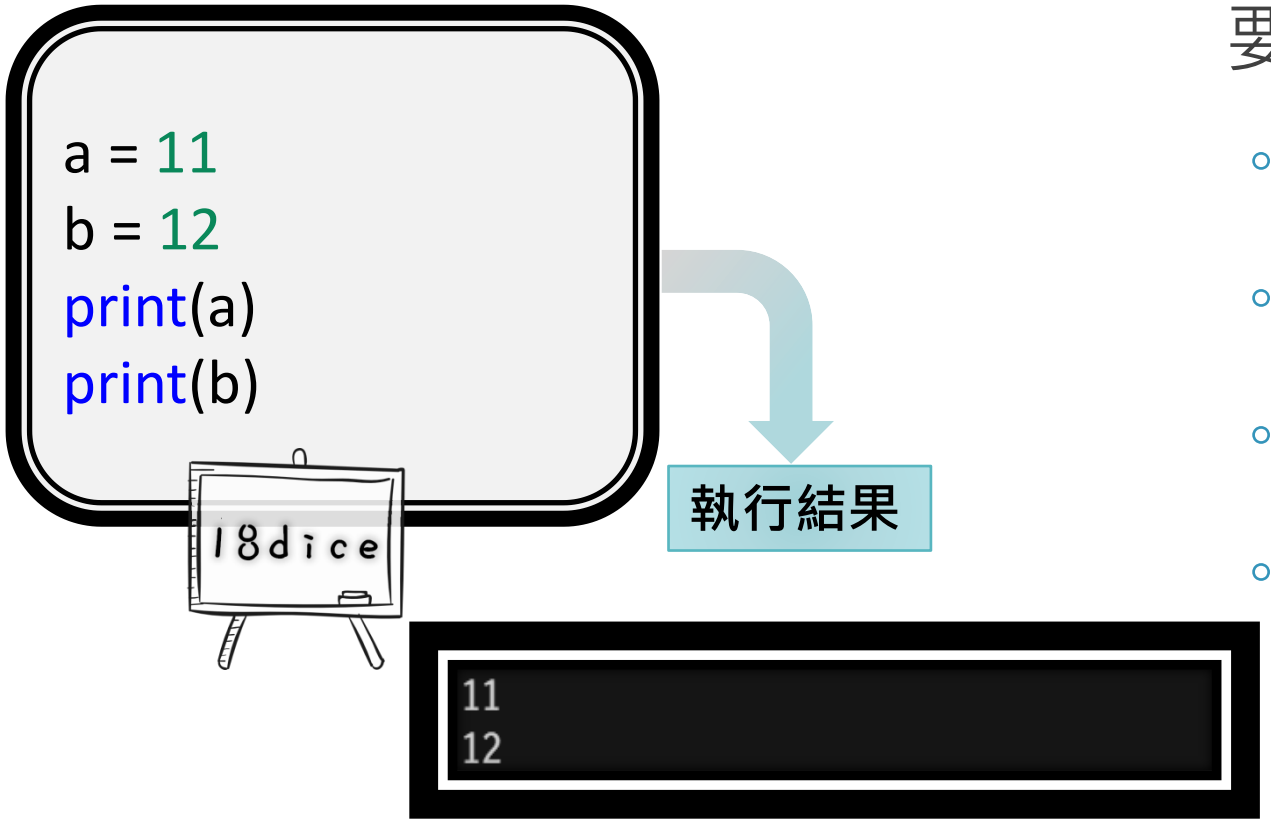
a=3
print(a)



Python

延伸的概念

概念1：兩個變數的輸入與輸出

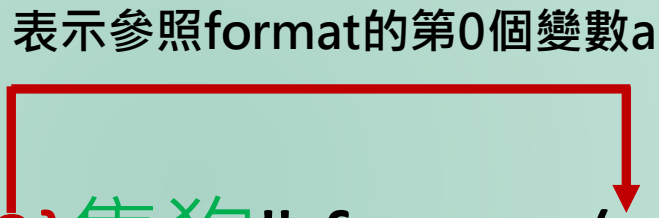


要印出一個變數就寫一個 print

- 1 個變數 1行
- 2 個變數 2 行
- 5 個變數 5 行
- 10 個變數 10 行

概念2：除了輸出變數內容外，加了其他說明

```
a=11
b=12
print(“我家有{0}隻狗".format(a))
print(“我家有{0}隻貓".format(b))
```



概念3：多個變數在同一行輸出

```
a = 11
b = 12
print("{0}\n{1}".format(a, b))
```

表示輸出時會換行

執行結果

```
11
12
```

在大括號內加上編號，

代表是對應到 format() 內第幾個變數

注意: 編號是由 **0** 開始喔！

- 0對應到format的第一個變數
- 1對應到format的第二個變數

概念 4：
在同一行輸出兩個變數，並且加上其他說明

```
a=11
b=12
print("我家有{0}隻狗{1}隻貓".format(a,b))
```

執行結果

我家有11隻狗12隻貓

5：小數(浮點數)如何處理？

整數 vs 小數

- int (整數):
  - int 是英文單字integer的縮寫
  - 不含小數點的數位爲整數

- float (小數):
  - 含小數點的數字爲浮點數

- 3.1415
- 21323.2323
- 233.54
- 87.45
- 999.444
- 99.00

程式碼

代碼都一樣？？？

- Int(整數):

```
a=11
print("{0}".format(a))
```

11

- Float(小數):

```
a=11.11
print("{0}".format(a))
```

11.11

Python format會自動根據變量值，

調整輸出格式

But

若要控制小數輸出格式呢？

小數點位數的多樣呈現

- a=10.100000
- print(“{0}”.format(a))　#去掉a的小數點並印出
- print(“{0:f}” .format(a))　#預設小數點後6位
- print(“{0:.2f}” .format(a))　#想要小數點後有幾位，:後面就點幾位

```
a=10.100000
b=10.22
print("num1={0}".format(a))
print("num2={0:f}".format(a))
print("num3={0:.2f}".format(a))
print("num1={0}".format(b))
print("num2={0:f}".format(b))
print("num3={0:.2f}".format(b))
```

```
[Python_3_6]$ python3 -d main.py | tee  main.py.err
num1=10.1
num2=10.100000
num3=10.10
num1=10.22
num2=10.220000
num3=10.22
[Python_3_6]$
```