Python

資料型態與浮點數

資料型態

整數、浮點數、複數與文字

在 Python 中，資料型態包含

數

- 整數 int
- 浮點數 float
- 複數 complex

字

Python支援複數的運算，
只是在數學當中的複數，我們是用「*i*」來表示虛數的部份，
而在Python當中則是用「j」或是「J」來表示。

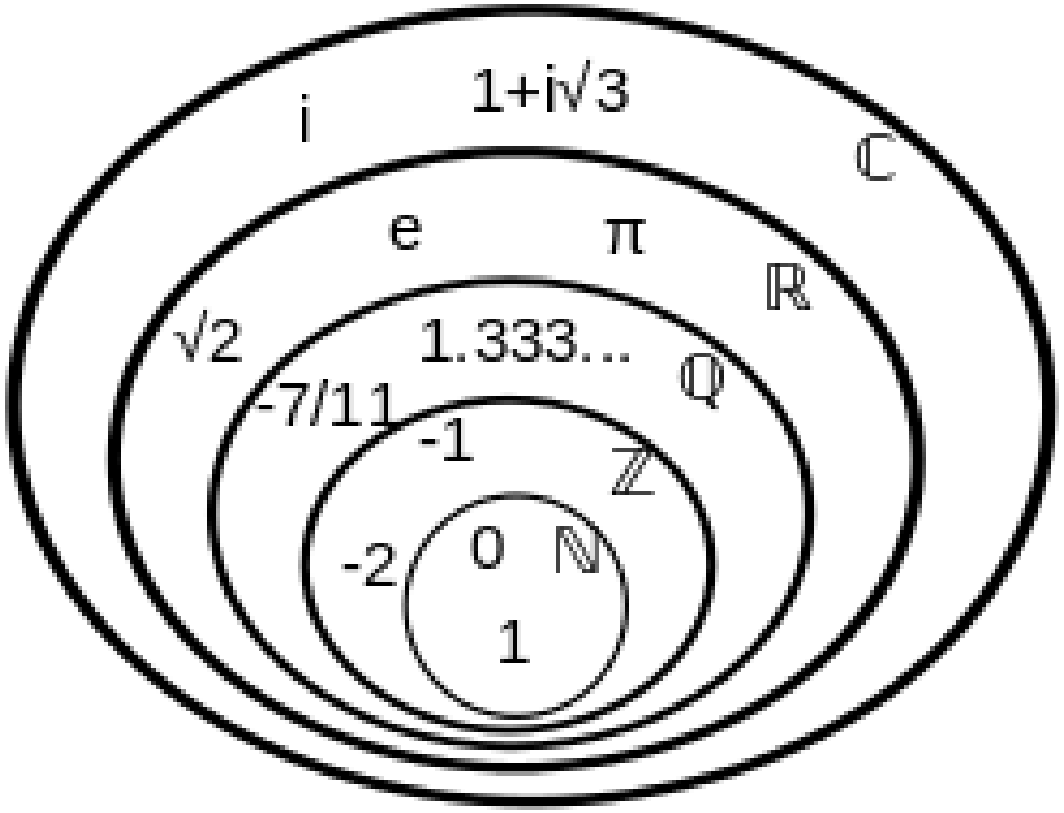
複數 complex

- 複數 = 實數 + 虛數
  - x + yi

- python 使用 j 代表虛數符號 i
  - x + yj => 實數 x + 虛數 y

在數學領域中，數系包含

- **自然數**(N):
  - 非負整數
- **整數**(Z):
  - 正整數、零、負整數
- **有理數**(Q):
  - 可以用兩個整數比表示的數
- **實數** (R):
  - 有理數、無理數(小數點右邊的數字個數無限)
- **複數** (C):
  - 由實數+虛數組合



浮點數的設定同整數

```
num1 = 10#指定整數變數
num2 = 10.01 #指定浮點數變數
num3 = 10 + 10j#指定複數變數
text= ‘DICE’ #文字加單引號
```

使用變數名稱使用
指定運算子(=)
設定數值就可以

複數的符號表示相當的簡單，通常寫作：a+b*i*，這裡的a跟b是實數，
Python裡以j替代數學的i

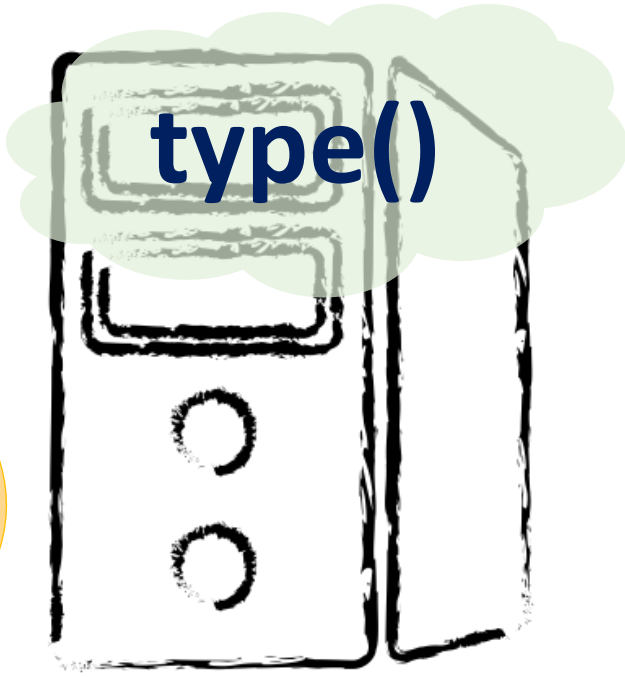
type

可以問出資料型態

type:判斷變數的資料型態

- 例如：num1的資料型態是甚麼？
  - 程式碼：
    - num1 = 10
      print(type(num1))
  - 輸出範例
    - <class 'int'>

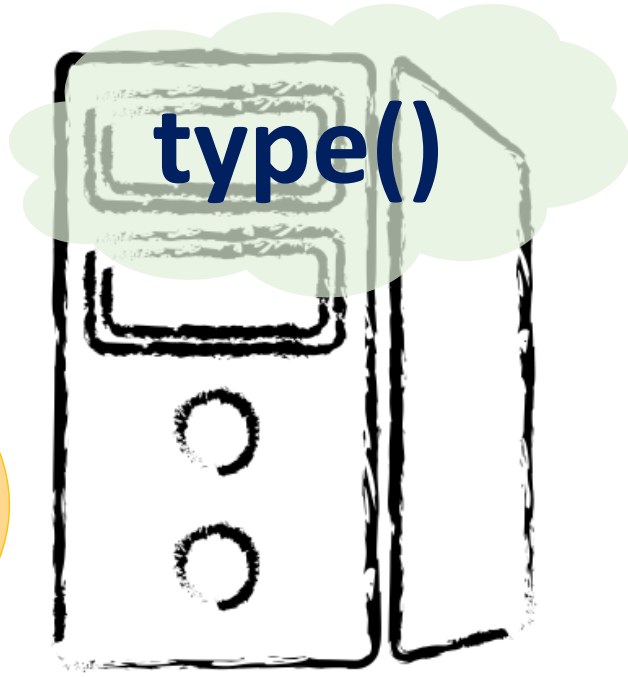
表示num1的資料型態是
int
int代表的是整數



再用type問資料型態

- 例如：請問num1的資料型態？
  - 程式碼：
    - num1 = 10.1
      print(type(num1))
  - 輸出範例
    - <class ‘float'>

表示num1的資料型態是
float
float代表的是浮點數



資料型態-
浮點數的輸出

狀況1：不控制輸出格式，寫法同整數

- int (整數):

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

11

- float (浮點數):

```
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？

浮點數 float與輸出

- 含小數點的數字就是浮點數
- 輸出浮點數資料時，可以使用，f是float代表浮點數
  - 不設定格式

    {變數位置:f}
  - 設定輸出格式

    {變數位置:[總寬度].[小數位數]f}

```
1 height = 160
2 weight = 55.22
3 print('My height is {0:10.2f} CM.'.format(height))
4 print('My weight is {0:10.2f} KG.'.format(weight))
```

```
[Python_4_4]$ python3 -d main.py | tee  main.py.err
My height is     160.00 CM.
My weight is      55.22 KG.
[Python_4_4]$
```

狀況2：f出馬，輸出預設小數點6位

- int (整數):

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

11

- float (浮點數):

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

11.110000

狀況3：印出小數點指定的位數，如2位

```
a=11.1111
print("{0:.2f}".format(a))
```

11.11

指定輸出格式控制:
{參數編號:輸出格式}

- **輸出格式**
  - f 表示輸出格式爲 float
  - [數字1].[數字2]f: 數字1表示整個數字要輸出的寬度位元數(含小數點下位數)，數字2表示小數要輸出的位數。若未指定，輸入值有幾位小數就會輸出相同位數。

狀況4：一個句子中，輸出3個浮點數

```
a=11.1111 #請以小數點後1位呈現輸出
b=22.2222 #請以小數點後2位呈現輸出
C=33.3333 #請以小數點後3位呈現輸出
print("{0:.1f} {1:.2f} {2:.3f} ".format(a,b,c))
```

執行結果

```
11.1 22.22 33.333
```

資料型態-
浮點數輸出固定寬度設定

浮點數固定寬度輸出範例

寫一個程式，
指定一個整數為160，一個浮點數為 55.22，分別代表身高與體重，
並皆以小數型式印出其值，
小數位數請印出兩位數，總數字寬度印出 10 位。

{0:10.2f}

```
1 height = 160
2 weight = 55.22
3 print('My height is {0:10.2f} CM.'.format(height))
4 print('My weight is {0:10.2f} KG.'.format(weight))
```

```
[Python_4_4]$ python3 -d main.py | tee  main.py.err
My height is     160.00 CM.
My weight is      55.22 KG.
[Python_4_4]$
```

浮點數 float輸出格式加強版

```
num = 33.333333333333333333333
print('num=<{num}>'.format(num=num))
print('num=<{num:f}>'.format(num=num))
print('num=<{num:.0f}>'.format(num=num)) # 不輸出小數值
print('num=<{num:.1f}>'.format(num=num)) # 輸出小數位數 1 位

print('num=<{num:1.1f}>'.format(num=num))
# 格式需求: 小數位數(1), 總寬度(1) => 整數寬度(1-1=0)
# 設定的整數寬度(1-1=0) < 整數最小寬度(2) => 無效！

print('num=<{num:5.4f}>'.format(num=num))
# 格式需求: 小數位數(4), 總寬度(5) => 整數寬度(5-4=1)
# 設定的整數寬度(5-4=1) < 整數最小寬度(2) => 無效！

print('num=<{num:15.5f}>'.format(num=num)) # 長度補滿 5
# 格式需求: 小數位數(5), 總寬度(15) => 整數寬度(15-5=10)
# 設定的整數寬度(15-5=10) > 整數最小寬度(2) => 合理的設定！
```

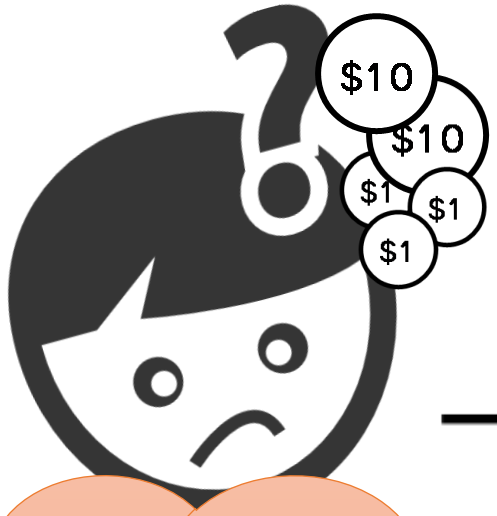
執行結果

```
num=<33.333333333333336>
num=<33.333333>
num=<33>
num=<33.3>
num=<33.3>
num=<33.3333>
num=<        33.33333>
```

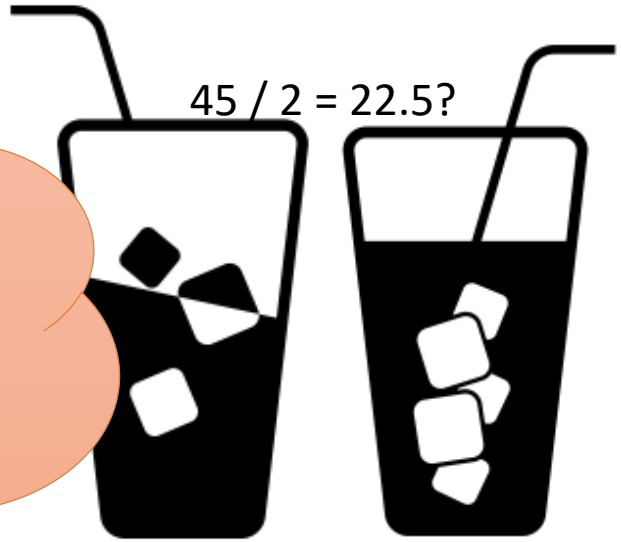
資料型態-
強迫轉換資料型態

浮點數轉整數的案例

兩個人買了兩杯飲料
總共 45 元，
那一個人要付多少錢？



45 / 2 = 22.5?



$ 45 元

45 / 2 = 22.5
一個人付 int(22.5) = 22元
另一個人付 45 – 22 = 23元

浮點數 -> 整數 (強迫轉型)

```
print(int(1))
print(int(2.8))
print(int("3"))
print(float(1))
print(float(2.8))
print(float("3"))
print(float("4.2"))
print(complex(2))
print(complex(2, 3))
print(complex(5, 0))
```

int(): 強迫輸出整數值
float() : 強迫輸出浮點數值
complex(): 強迫產生複數值

執行結果

```
1
2
3
1.0
2.8
3.0
4.2
(2+0j)
(2+3j)
(5+0j)
```