

Python

2019年5月30日

下午 01:50

Default arguments

- print(1, 2, 3, sep='<')

⇒ 1 < 2 < 3 → 指插入的符號

sep=' ' → 空白

⇒ 1 _ 2 _ 3

List

(-) ordered sequence of value

ex: primes = [2, 3, 5, 7]

or primes = ['May', 'Bay', 'cat']

(⇒) list of list

ex: hands = [['J', 'Q', 'K'],
['2', '2', '2'],
['a', 'b', 'c']]

○ → 最後的, 縮小

(⇒) Index

ex primes[0] ⇒ May

primes[1] ⇒ Bay

primes[-1] ⇒ cat → 從後面取則負号

(IV) Slice

(四) Slice

→ 一次取多個

→ `planets[0:3]` → 取第 0 ~ ² 個 (共 3 個)

0 可不寫 → `[: 3]`

最後也可不寫 → `[2 :]` → 第 2 個 ~ 最後

→ `planets[1:-1]`

⇒ 頭尾不要

→ `planets[-3:]` ⇒ 最後 3 個

(五) change list

1. `planets[3] = 'Hi'` ⇒ 直接改

2. `planets[:3] = ['a', 'b', 'c']` ⇒ 一次改前 3 個

(六) list function

1. `len()`

→ list 的長度

→ `len(apple) ⇒ 3`

如果 `apple = ['a', 'b', 'c']`

1 2 3
→

2. `sorted()`

→ 排序

3. `sum()`

→ 將 list 中的數字加總

(t) list method

1. list.append()

→ 增 item 到 list

2. list.pop()

→ 移除且回傳最後一個元素

3. list.index()

→ 查順序

→ ex. apple.index('b')

⇒)

4. in

→ 查是否存在 list 中

→ ex. 'd' in apple ⇒ false

(八) Tuples

1. 和 list 很像，差在兩個地方

a. 是用 () 表示而非 []

i. t=1,2,3

ii. t

iii. (1,2,3)

b. tuple 不能修改

2. 常用在小數換分數

a. as_integer_ratio()

b. x=0.125

c. x.as_integer_ratio()

d. (1,8)

$$\rightarrow \frac{1}{8} = 0.125$$

3. 常用在 swap

i. ddcdsfdpython 方便的地方

a. a=1,b=0

b. a,b=b,a

c. Print(a,b)

d. 0 1

→ 自變量與常量的區別

Loop

(一) in

1. in 可以用在 list 也可在 tuple 只要是集合都行 字串也可以
2. sets=['a','b','c','d']
 For set in sets:
 Print(set,end="")
 → 變數
 Print(set,end="") 印在同一行
3. A='asdDfgR SAff'
 For char in A:
 If char.isupper():
 Print(char,end="")
 → DRSA

(二) range

1. 可回傳特定個數的數值
2. For i in range(5):
 Print(i)
 → 0 1 2 3 4

(三) while

```
i=0
While i<10:
    Print(i,end="") → 0 1 2 3 4 5 6 7 8 9
    i+=1
```

(四) list 的快速運算

1. Squares=[n**2 for n in range(10)]
 Squares *→ [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]*
 → 列表 - 行
2. 同樣算法
 Squares=[]
 For n in range(10):
 Squares.append(n**2)
 Squares
 ↑
 列表 (長度 < 6)
 list
3. If
 Short_planets = [planet for planet in planets if len(planet) < 6]
4. 簡化範例
 N_negative=0
 For num in nums:
 If num<0:
 N_negative=n_negative+1
 Return n_negative
 → return len([num for num in nums if num<0])

String

(一) string 表示

1. 雙引號 單引號皆可

2. 若要在字串中顯示" 需在前面輸入\
'pluto\'s a planet!'
3. \n 表示換行
4. Print()兩次也可當作換行
5. Print("hi",end='') *→ 才不會換*

(二) string 可看成字元的list

1. Index
2. Slice
3. Len

(三) string methods

1. Upper() 轉成大寫
2. Lower() 轉成小寫
3. index 可找子字串的第一個index

Claim='Pluto is a plante!'

Claim.index('plan') *⇒ ()*

4. Str.split() 切割

Words=claim.split()

Words

=>['Pluto','is','a','planet!']

Date='2019-5-30'

Year,month,day =Date.split('-')

5. Str.join() 合併

'/'.join([month,day,year])

=>05/30/2019

6. + 可結合成字串

但是只能變數是字串時才能直接用+

7. Str(變數) 轉成字串

8. Str.format()

"{}, you'll always be the {}th planet to me.".format(planet,position)

s="pluto's a {0}. NO, it's a {1}.{0}!{1}".format('planet','dwarf planet')

字串變數
數字
{0} *{1}*

Dictionary

(一) 是一種對應key和value的資料結構

1. Numbers={'one': 1, 'two': 2, 'three': 3}

key *value*

2. Numbers['one'] *⇒ 1*

3. 可以直接加或改

Numbers['eleven']=11

Numbers['one']='pluto'

Numbers

=>{'one': 'pluto', 'two': 2, 'three': 3, 'eleven': 11}

4. For/in/sort 和list很像

5. Dict.items() 可以輕易迭代key和value

```
d = { 'Adam': 95, 'Lisa': 85, 'Bart': 59, 'Paul': 74 }  
sum = 0  
for key, value in d.items():  
    sum = sum + value  
    print(key, ': ', value)  
print('平均:', sum / len(d))
```

=>Adam : 95 Lisa : 85 Bart : 59 Paul : 74 平均分为: 78.25

External Libraries

(一) math

(二) numpy