

Dict

hyundai 8.15

昨天提到了 List

昨天提到了 List

- 可以動態增長長度
- 可以塞任何東西
- 透過 index 來取得對應的資料

如果我的 index 不想是數字呢？

再一次運用想像力

- 我們有另外一種櫃子，上面會貼有一種**標籤**
- 比如放在「**水果**」標籤中的東西
- 又如放在「**OO**」標籤中的東西
- 這個**標籤**表示東西放在哪邊，叫做 **key (鍵值)**
- 標籤可以是**任何東西**，裡面**什麼都能放**

這世界其實沒這麼美好

再一次走出櫃子，看看 Dict

- Dict = Dictionary
- 我們可以有放在 '**fruits**' 裡的變數
- 也能有放在 **123** 裡面的變數
- **Key (鍵值)** 只能是**不可修改**的型態：str, int, float
- 變數則和 List 一樣，什麼都可以放

Dict 語法

- `d = {'cat': ['Pusheen', 'Kitty'], 'dog': 'snoopy'}`
- dict 用**大括弧** `{}` 包住，元素之間也是逗號隔開
- 元素會以 **key : value** 的 K-V Pair 組成
- Key 後面要接冒號，冒號後面就是變數
- 使用 **`d[key]`** 取出變數，比如 `d['dog']` 會得到 `'snoopy'`

不保證 Key 的順序

```
d = {'one':1, 'two':2, 'three':3}  
print(d)
```

```
{'two': 2, 'three': 3, 'one': 1}
```

猜猜看時間

這個 Dict 有大？

```
a = {'name': 'hydai', 'id': 1234}
```

用 len 就能拿到大小了

```
a = { 'name': 'hydai', 'id': 1234 }
```

```
print(len(a))
```

```
2
```

那有 List 的這個呢？

```
b = {1: ['one', True, 3.14, 689], 'c8763': 'GG'}
```

一樣，我們只看 Key 有多少

```
b = {1: ['one', True, 3.14, 689], 'c8763': 'GG'}  
print(len(a))
```

2

用 d[key] 取值

```
a = { 'name': 'hydai', 'id': 1234 }  
  
print(a[ 'name' ])  
print(a[ 'id' ])
```

```
hydai  
1234
```

繼續來作死

```
print(a['not found'])
```

```
hydai  
1234
```

```
-----  
--  
KeyError                                Traceback (most recent call last)  
t)  
<ipython-input-5-98ff1ef63bfd> in <module>()  
      3 print(a['name'])  
      4 print(a['id'])  
----> 5 print(a['not found'])  
  
KeyError: 'not found'
```


練習題

- 輸入五次從 0~9 的整數
- 把該數字對應的英文印出來

```
0
zero
3
three
2
two
1
one
5
five
```

Dict 的操作

創建與刪除

- `d = {}`
 - 建立空的 dict
- `del d[key]`
 - 刪除特定的 key-value pair

增加與更新

- `d[key] = value`
 - 如果 `key` 不存在，會增加這組 K-V
 - 如果 `key` 存在，會更新這組 K-V

遍歷 dict

```
d = {0: 'zero', 1: 'one', 2: 'two', 3: 'three', 4: 'four', 5: 'five'}  
for key in d:  
    print('key =', key, ', value =', d[key])
```

```
key = 0 , value = zero  
key = 1 , value = one  
key = 2 , value = two  
key = 3 , value = three  
key = 4 , value = four  
key = 5 , value = five
```

語法 for...in

- for [變數名稱] in dict:
- [縮排] blah blah
- 每一回合會拿到的一把在 dict 裡面的 **Key**
- 就可以用 **dict[key]** 拿到對應的變數

練習題

- 讓使用者輸入 5 組 Key-Value Pair
- key 是字串，value 是數字
- 然後把最後的 dict 印出來

反查 dict

```
d = {0: 'zero', 1: 'one', 2: 'two', 3: 'three', 4: 'four', 5: 'five'}
target = 'two'
for key in d:
    if d[key] == target:
        print('Found! key =', key)
        break
    else:
        print('Keep going...')
```

Keep going...

Keep going...

Found! key = 2

檢查 dict 有沒有包含特定 key

```
d = {0: 'zero', 1: 'one', 2: 'two', 3: 'three', 4: 'four', 5: 'five'}  
print(0 in d)  
print('1' in d)
```

True

False

拿到 keys & values

- `dict.keys()`
 - 可以拿到一個由所有 key 組成的 list
- `dict.values()`
 - 可以拿到一個由所有 value 組成的 list

檢查是否 K, V 在 dict 中

```
d = {0: 'zero', 1: 'one', 2: 'two'}  
print(0 in d.keys())  
print('zero' in d.values())  
print(123 in d.keys())  
print('123' in d.values())
```

True

True

False

False

取值

- `d[key]`
 - 不安全，key 如果不存在會出現 error
- `d.get(key, default_value)`
 - 安全，如果 key 不存在會回傳 default_value

試試看

```
d = {0: 'zero', 1: 'one', 2: 'two'}  
print(d[0])  
print(d.get(0, 'Not found'))  
print(d.get(12764871, 'Not found'))
```

zero

zero

Not found

練習題

- 輸入五個人的名字跟成績
- 接下來會有五次查詢，每次查詢會輸入一個名字
- 如果不在名單中，請輸出 '404 Not Found'
- 如果在名單中，請輸出他的名字以及成績

練習題

- 輸入 10 個人名
- 請你算出每個人名各出現幾次