3.4. Dictionary



Fig. 3.4.1 Photo by Joshua Hoehne on Unsplash



Outline

- 1. Introduction
- 2. Examples
 - a. Ex1: == vs. is
 - b. Ex2: Operations
- 3. Summary: Container
 - a. Ex1: Tuple
 - b. Ex2: List
 - c. Ex3: Set
 - d. Ex4: Dictionary
 - e. Ex5: Comparison
- 4. Exercise



Note

Roadmap

1. This topic: Dictionary

myMaze, myBoard, myBlock

Container				
tuple	list	set	dict	del
Statement	Statement	Statement	Statement	Statement*

- 2. Course: Python 1
- 3. Subject: Programming
- 4. Field
- a. Software Engineering (SE)
- b. Computer Science and Information Engineering (CSIE)
- c. Electrical/Electronics Engineering (EE)

3.4.1. Introduction

- 1. dict為dictionary 的縮寫,具有Key-Value 對應的型態。
- 2. 字典是種配對型態,且具有多筆資料的物件, key 就是存取該筆value 的索引值。
- 3. 同樣要建立一個dict,有四種可以用的方法:

```
d1 = {1: "a", 2: "b"}
d2 = dict({1: "a", 2: "b"})
d3 = dict(zip((1, 2), ("a", "b")))
d4 = dict([[2, "b"], [1, "a"]])
```

4. 而dict的讀取、刪除、回傳與判斷等方式則如下面內容所示。

Compute	Description	
d[k]	從d中取得k所對應的值。	
d[k]=v	將d中k所對應的值指定為v。若d中沒有k這個key則新增一組。	
del d[k]	刪除d中k所屬的組合。	
k in d	判斷k是否在d的key值中。	
k not in d	判斷k是否不在d的key值中。	
iter(d)	回傳由d的key值所建立的迭代器。	
len(d)	回傳d的資料組數。	

3.4.2. Examples

3.4.2.1. Ex1: == vs. is

- 1. 這四種方法所建立出來的dict 是完全一樣的,用「==」檢驗會得到True,不過用is 就不會。
- 2. 還記得嗎?因為is 檢驗的項目是「是否指向同一物件」,如下所示。
- 3. Source

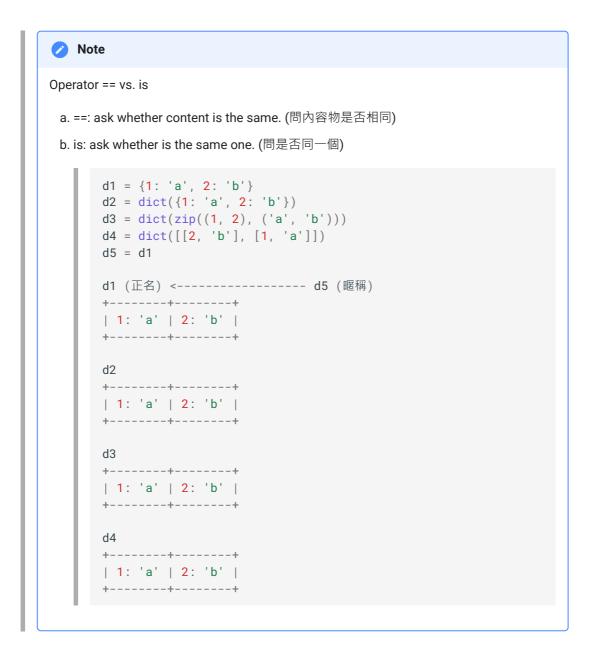
```
Listing 3.4.2.1.1 /src/Dict/p0710Dict.py

1 '''
2 Created on 2015年9月18日
```

```
3
 4
    @author: cph
 5
   d1 = {1: "a", 2: "b"}
 6
   d2 = dict({1: "a", 2: "b"})
d3 = dict(zip((1, 2), ( "a", "b")))
 7
8
9 d4 = dict([[2, "b"], [1, "a"]])
10 d5 = d1
11 print(d1 is d2)
12 print(d1 is d3)
13 print(d1 == d2)
14 print(d1 == d3)
15 print(d1 == d4)
16 print(d5 is d1)
```

```
False
False
True
True
True
True
```

5. 由上可知· d1到d4的值都相同·但是所指向的物件都不相同;而d5因為被賦予的是d1的指標·因此d5 is d1結果會是True。



3.4.2.2. Ex2: Operations

1. Source

```
Listing 3.4.2.2.1 /src/Dict/p0711Dict2.py
1
2 Created on 20150918
3 @author: cph
   1.1.1
  d = dict(zip((1, 2, 3), ("a", "b", "c"))) # 建立dict
5
6 print(type(d)) # 印出d 的型別
7
  print(d[3])
                       # 印出d[3]
   d[4] = "d"
                     # 因為原本沒有4 這組key,因此此行新增一個
8
  key[4]·對應值為d
9
   print(d)
```

```
11 del d[1] # 删除d[1]
12 print(d)
13 print(3 in d) # 檢驗3 是否在d 的key 值中
14 print(2 not in d) # 檢驗2 是否不在d 的key 值中
15 for i in iter(d): # 建立for 迴圈。以iter(d) 為key 值所建立的迭
16 代器

print(i, end = " ") # 印出i
print(len(d)) # 印出d 的配對組數
```

```
<class "dict">
c
{1: "a", 2: "b", 3: "c", 4: "d"}
{2: "b", 3: "c", 4: "d"}
True
False
2 3 4 3
```

3.4.3. Summary: Container

3.4.3.1. Ex1: Tuple

- 1. 經過了菲絲恩的精神喊話, 普羅也對決賽產生了十足的信心。
- 2. 而比賽正準備開始, 前方放置了許多的箱子且排成了一列, 每個箱子內都寫有一個數字, 誰先把指定位子之箱子中的數字放到答案區就得分。
- 3. 普羅一聽到哨音就馬上衝了出去, 卻只見菲絲恩揮了揮手施展魔法, 就贏得了這個關卡。
- 4. Source

```
Listing 3.4.3.1.1 /src/Tuple/p07e1Tuple.py
1
2
  @since: 20150918
3 @author: cph
4 @note:
5
     Question:
         經過了菲絲恩的精神喊話,
6
7
         普羅也對決賽產生了十足的信心。
         而比賽正準備開始,
9
         前方放置了許多的箱子且排成了一列,
10
         每個箱子內都寫有一個數字,
         誰先把指定位子之箱子中的數字放到答案區就得分。
11
         普羅一聽到哨音就馬上衝了出去,
12
13
         卻只見菲絲恩揮了揮手施展魔法,
         就贏得了這個關卡。
14
15
16
   tiBoxes = (23, 45, 65, 23, 47) # 這是一個tuple
```

```
      17
      print(tiBoxes[2])
      # 取出第3個位置中的資料

      18
```

```
65
```

3.4.3.2. Ex2: List

- 1. 經過取數字的考驗後,所有通過的隊伍都被帶進一間書房。
- 2. 而下一個測驗是要參賽者利用魔法將自己面前書櫃裡及掉到地上的書按照次序排列整齊。
- 3. Source

```
Listing 3.4.3.2.1 /src/List/p07e2List.py
1
2
   @since: 20150918
   @author: cph
3
4
   @note:
5
       Question:
           經過取數字的考驗後,所有通過的隊伍都被帶進一間書房。
6
           而下一個測驗是要參賽者利用魔法將自己面前書櫃裡及掉到地上的書按照次序
   排列整齊。
9
   @see:
10
       PyBook1
11
    lsShelf = ["Book1", "Book2", "Book6", "Book4", "Book5"]
12
    lsShelf.append(lsShelf[2]) # lsShelf中第3個元素加到並新增到串列最後
lsShelf[2] = "Book3" # lsShelf中第3個元素更改為Book3
13
# 輸出結果
    print(lsShelf)
```

4. Output

```
["Book1", "Book2", "Book3", "Book4", "Book5", "Book6"]
```

3.4.3.3. Ex3: Set

1. Source

```
Listing 3.4.3.3.1 /src/Set/p07e3Set.py

1 '''
2 @since: 20150918
3 @author: cph
4 @note:
5
```

```
6 @see:
7 '''
8 ssItemsA = {"蘋果", "香蕉", "鳳梨", "芭樂"} # 這是第一個集合
9 ssItemsB = {"鳳梨", "蘋果", "水梨", "蓮霧"} # 這是第二個集合
print(ssItemsA ^ ssItemsB) # 利用XOR把沒有重複的項目挑出來
```

```
{"芭樂", "蓮霧", "水梨", "香蕉"}

Note

a. XOR = Exclusive OR = 互斥 = 有你就沒有我
```

3.4.3.4. Ex4: Dictionary

- 1. 今天的賽程即將告一段落, 菲絲恩與普羅終於要前往今天的最後一道關卡。
- 2. 只見桌上放了一本密碼簿及一串文字(5-4-2.2-'1'-"Six")·不管怎麼想就是解不出來·最後菲 絲恩把這串字當成了咒語施展·卻意外地解開了這道密碼。
- 3. Source

```
Listing 3.4.3.4.1 /src/Dict/p07e4Dict.py
2 @since: 20150918
   @author: cph
4 @note:
5
       Question:
           今天的賽程即將告一段落,菲絲恩與普羅終於要前往今天的最後一道關卡。
6
7
           只見桌上放了一本密碼簿及一串文字(5-4-2.2-'1'-"Six")·
           不管怎麼想就是解不出來,最後菲絲恩把這串字當成了咒語施展,
8
           卻意外地解開了這道密碼。
10
   @see:
       PyBook1
11
12
13 # 這是一個字典(密碼簿)
   dPassBook = {'1': 'Wor',
14
               2.2: 'gic '
15
               'Three': 'rd',
16
17
             4: 'Ma',
               'Six': 'ld.',
18
               5: 'A'}
19
20 print(dPassBook[5], end = '') # print 出內容
21 print(dPassBook[4], end = '')
22 print(dPassBook[2.2], end = '')
    print(dPassBook['1'], end = '')
```

```
print(dPassBook['Six'], end = '')
24
```

```
A Magic World.
```

3.4.3.5. Ex5: Comparison

1. Source

```
Listing 3.4.3.5.1 /src/Dict/Comparison.py
    1.1.1
1
 2
   author: cph
   since: 20230725
3
5
    11 = [1, 2, 3, 4]
 6
    t1 = (1, 2, 3, 4)
7
    s1 = \{1, 2, 3, 4\}
8 d1 = \{1: 'a', 2: 'b', 3: 'c', 4: 'd'\}
    print(f'Len: List={len(l1)}; Tuple={len(t1)}; Set={len(s1)};
9
10
    Dict={len(d1)}')
    print(f'Max: List={max(l1)}; Tuple={max(t1)}; Set={max(s1)};
11
12
   Dict={max(d1)}'
13
    print(f'Min: List={min(l1)}; Tuple={min(t1)}; Set={min(s1)};
14
    Dict={min(d1)}')
15
   print(f'Sum: List={sum(l1)}; Tuple={sum(t1)}; Set={sum(s1)};
16
   Dict={sum(d1)}')
17
    12 = [1, 2, 5, 6]
18
   t2 = (1, 2, 5, 6)
19
20
   s2 = \{1, 2, 5, 6\}
21 d2 = {1: 'a', 2: 'b', 3: 'C', 4: 'D'}
    print(f'List MERGE: {11 + 12}')
22
23
   print(f'Tuple MERGE: {t1 + t2}')
24
   print(f'Set AND : {s1 & s2}')
25
   print(f'Set OR : {s1 | s2}')
26
    print(f'Set DIF : {s1 - s2}')
    print(f'Set DIF : {s2 - s1}')
    print(f'Set XOR : {s1 ^ s2}')
    print(f'Dict MERGE: {d1 | d2}')
    print(f'Dict MERGE: {d2 | d1}')
```

2. Output

```
1 Len: List=4; Tuple=4; Set=4; Dict=4
2 Max: List=4; Tuple=4; Set=4; Dict=4
3 Min: List=1; Tuple=1; Set=1; Dict=1
4 Sum: List=10; Tuple=10; Set=10; Dict=10
```

```
5 List MERGE: [1, 2, 3, 4, 1, 2, 5, 6]
6 Tuple MERGE: (1, 2, 3, 4, 1, 2, 5, 6)
7 Set AND : {1, 2}
8 Set OR : {1, 2, 3, 4, 5, 6}
9 Set DIF : {3, 4}
10 Set DIF : {5, 6}
11 Set XOR : {3, 4, 5, 6}
12 Dict MERGE: {1: 'a', 2: 'b', 3: 'C', 4: 'D'}
13 Dict MERGE: {1: 'a', 2: 'b', 3: 'c', 4: 'd'}
```

3.4.4. Exercise

3.4.4.1. Ex. 1

1. Question

```
當使用者分別輸入「P」、「R」或「J」時·
請利用字典(dict)的方法分別輸出「Python」、「Ruby」或「Java」。
```

2. Code

```
dic = {"P": "Python", "R": "Ruby", "J": "Java"}
Str = input("請輸入P/R/J:")
print(dic[Str])
```

Note

- 2. System Environment:

```
1. Start: 20170719
    Listing 3.4.4.1.1 requirements.txt
                                       # Sphinx
        sphinx==7.1.2
        graphviz > = 0.20.1
                                      # Graphviz
     2
        sphinxbootstrap4theme>=0.6.0
                                      # Theme: Bootstrap
     3
                                      # Theme: Material
     4
        sphinx-material>=<mark>0.0.35</mark>
                                      # PlantUML
        sphinxcontrib-plantuml>=0.25
     5
        sphinxcontrib.bibtex>=2.5.0
                                      # Bibliography
     6
     7
        sphinx-autorun>=1.1.1
                                      # ExecCode: pycon
        sphinx-execute-code-python3>=0.3 # ExecCode
     8
        btd.sphinx.inheritance-diagram>=2.3.1 # Diagram
     9
        sphinx-copybutton>=<mark>0.5.1</mark>
                                      # Copy button
    10
    11
        sphinx_code_tabs>=0.5.3
                                      # Tabs
    12
        sphinx-immaterial>=0.11.3
                                      # Tabs
    13
    14
        #-- Library Upgrade Error by Library Itself
    15
    16
        # >> It needs to fix by library owner
        # >> After fixed, we need to try it later
    17
    18
    19
        pydantic==1.10.10
                                       # 2.0: sphinx compiler
    20
        error, 20230701
    21
    22
        #-- Minor Extension
    23
    24
        sphinxcontrib.httpdomain>=1.8.1 # HTTP API
    25
    26
    27
        #sphinxcontrib-actdiag>=3.0.0
                                      # Diagram: activity
    28
        #sphinxcontrib-nwdiag>=2.0.0
                                      # Diagram: network
    29
        #sphinxcontrib-seqdiag>=3.0.0
                                      # Diagram: sequence
    30
    31
    32
    33
        #-- Still Wait For Upgrading Version
        #-----
    34
    35
    36
    37
        #-- Still Under Testing
        #-----
    38
    39
        #numpy>=1.24.2
                                      # Figure: numpy
    40
    41
        #-----
    42
    43
        #-----
    44
        #sphinxcontrib.jsdemo==0.1.4 # ExecCode: Need replace
    45
        add_js_file()
```

```
#sphinxcontrib.slide==1.0.0  # Slide: Slideshare
#hieroglyph==2.1.0  # Slide: make slides
#matplotlib>=3.7.1  # Plot: Need Python >= v3.8
#manim==0.17.2  # Diagram: scipy, numpy need
gcc
#sphinx_diagrams==0.4.0  # Diagram: Need GKE access
#sphinx-tabs>=3.4.1  # Tabs: Conflict w/
sphinx-material
```