Collections

Data Structure

최성철 교수
Director of TEAMLAB

Collections

Collections

- List, Tuple, Dict에 대한 Python Built-in 확장 자료 구조(모듈)
- 편의성, 실행 효율 등을 사용자에게 제공함
- 아래의 모듈이 존재함

```
from collections import deque
from collections import Counter
from collections import OrderedDict
from collections import defaultdict
from collections import namedtuple
```

deque

deque

- Stack과 Queue 를 지원하는 모듈
- List에 비해 효율적인 자료 저장 방식을 지원함

```
from collections import deque


deque_list = deque()
for i in range(5):
    deque_list.append(i)
print(deque_list)
```

```
deque_list.appendleft(10)
print(deque_list)
```

deque

- rotate, reverse등 Linked List의 특성을 지원함
- 기존 list 형태의 함수를 모두 지원함

```
deque_list.rotate(2)
print(deque_list)
deque_list.rotate(2)
print(deque_list)

print(deque_list)
print(deque(reversed(deque_list)))
```

```
deque_list.extend([5, 6, 7])
print(deque_list)


deque_list.extendleft([5, 6, 7])
print(deque_list)
```

deque

- deque 는 기존 list보다 효율적인 자료구조를 제공
- 효율적 메모리 구조로 처리 속도 향상

deque

```
from collections import deque
import time


start_time = time.clock()
deque_list = deque()
# Stack
for i in range(10000):
    for i in range(10000):
        deque_list.append(i)
        deque_list.pop()
print(time.clock() - start_time, "seconds")
```

general list

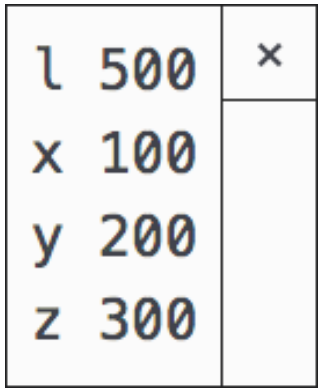
```
import time


start_time = time.clock()
just_list = []
for i in range(10000):
    for i in range(10000):
        just_list.append(i)
        just_list.pop()
print(time.clock() - start_time, "seconds")
```

OrderedDict

OrderedDict

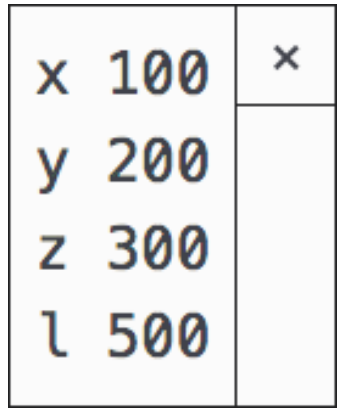
- Dict와 달리, 데이터를 입력한 순서대로 dict를 반환함

```
d = {}
d['x'] = 100
d['y'] = 200
d['z'] = 300
d['l'] = 500
```




```
for k, v in d.items():
    print(k, v)
```

```
from collections import OrderedDict

d = OrderedDict()
d['x'] = 100
d['y'] = 200
d['z'] = 300
d['l'] = 500
```




```
for k, v in d.items():
    print(k, v)
```

OrderedDict

- Dict type의 값을, value 또는 key 값으로 정렬할 때 사용 가능

```
for k, v in OrderedDict(sorted(d.items(), key=lambda t: t[0])).items():
    print(k, v)
```

```
l 500
x 100
y 200
z 300
```

```
for k, v in OrderedDict(sorted(d.items(), key=lambda t: t[1])).items():
    print(k, v)
```

```
x 100
y 200
z 300
l 500
```

defaultdict

defaultdict

- Dict type의 값에 기본 값을 지정, 신규값 생성시 사용하는 방법

```
d = dict()
print(d["first"])
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call l
<ipython-input-1-3b4ee8f5b4c3> in <module>()
      1 d = dict()
----> 2 print(d["first"])

KeyError: 'first'
```

defaultdict

- Dict type의 값에 기본 값을 지정, 신규값 생성시 사용하는 방법

```
from collections import defaultdict
d = defaultdict(object)     # Default dictionary를 생성
d = defaultdict(lambda: 0)  # Default 값을 0으로 설정합
print(d["first"])
```

defaultdict

- 하나의 지문에 각 단어들이 몇 개나 있는지 세고 싶을경우?
- Text-mining 접근법 - Vector Space Model

```
text = """A press release is the quickest and easiest way to get free
publicity. If well written, a press release can result in multiple
published articles about your firm and its products. And that can mean
new prospects contacting you asking you to sell to them.
….""".lower().split()
```

```
print(text)
```

```
['a', 'press', 'release', 'is', 'the', 'quickest', 'and', 'easiest', 'way', 'to', 'get', 'f
```

defaultdict

```
from collections import OrderedDict
word_count = defaultdict(object)     # Default dictionary를 생성
word_count = defaultdict(lambda: 0)  # Default 값을 0으로 설정함
for word in text:
    word_count[word] += 1
for i, v in OrderedDict(sorted(
        word_count.items(), key=lambda t: t[1],
        reverse=True)).items():
    print(i, v)
```

```
a 12
to 10
and 9
the 9
press 8
release 8
that 7
of 5
your 4
```

Counter

Counter

- Sequence type의 data element들의 갯수를 dict 형태로 반환

```
from collections import Counter

c = Counter()                           # a new, empty counter
c = Counter('gallahad')                 # a new counter from an iterable
print(c) Counter({'a': 3, 'l': 2, 'g': 1, 'd': 1, 'h': 1})
```

Counter

- Dict type, keyword parameter 등도 모두 처리 가능

```
c = Counter({'red': 4, 'blue': 2})      # a new counter from a mapping
print(c)
print(list(c.elements()))
```

```
Counter({'red': 4, 'blue': 2})
['blue', 'blue', 'red', 'red', 'red', 'red']
```

```
c = Counter(cats=4, dogs=8)             # a new counter from keyword args
print(c)
print(list(c.elements()))
```

```
Counter({'dogs': 8, 'cats': 4})
['dogs', 'dogs', 'dogs', 'dogs', 'dogs', 'dogs', 'dogs', 'dogs', 'cats', 'cats', 'cats', 'cats']
```

Counter

- Set의 연산들을 지원함

```
c = Counter(a=4, b=2, c=0, d=-2)
d = Counter(a=1, b=2, c=3, d=4)
c.subtract(d)  # c- d
print(c) Counter({'a': 3, 'b': 0, 'c': -3, 'd': -6})
```

Counter

- Set의 연산들을 지원함

```
c = Counter(a=4, b=2, c=0, d=-2)
d = Counter(a=1, b=2, c=3, d=4)
print(c + d)
print(c & d)
print(c | d)
```

```
Counter({'a': 5, 'b': 4, 'c': 3, 'd': 2})
Counter({'b': 2, 'a': 1})
Counter({'a': 4, 'd': 4, 'c': 3, 'b': 2})
```

Counter

- word counter의 기능도 손쉽게 제공함

```
text = """A press release is the quickest and easiest way to get free
publicity. If well written, a press release can result in multiple
published articles about your firm and its products. And that can mean
new prospects contacting you asking you to sell to them.
….""".lower().split()

print(Counter(text))
print(Counter(text)["a"])
```

```
Counter({'a': 12, 'to': 10, 'the': 9, 'and': 9, 'release': 8, 'press': 8, 'that': 7, 'of': 5,
12
```

namedtuple

namedtuple

- Tuple 형태로 Data 구조체를 저장하는 방법
- 저장되는 data의 variable을 사전에 지정해서 저장함

```
from collections import namedtuple
Point = namedtuple('Point', ['x', 'y'])
p = Point(11, y=22)
print(p[0] + p[1])
```

```
x, y = p
print(x, y)
print(p.x + p.y)
print(Point(x=11, y=22))
```

```python
from collections import namedtuple
import csv
f = open("./code/7/collections/users.csv", "r")
next(f)
reader = csv.reader(f)
student_list = []
for row in reader:
    student_list.append(row)
    print(row)


coloumns = ["user_id", "integration_id", "login_id", "password", "first_name",
            "last_name", "full_name", "sortable_name", "short_name",
            "email", "status"]
Student = namedtuple('Student', " ".join(coloumns))
student_namedtupe_list = []
for row in student_list:
    student = Student(*row)
    student_namedtupe_list.append(student)
print(student_namedtupe_list)
print(student_namedtupe_list[0].full_name)
```

Human knowledge belongs to the world.