Python Collections: Lists and Dictionaries

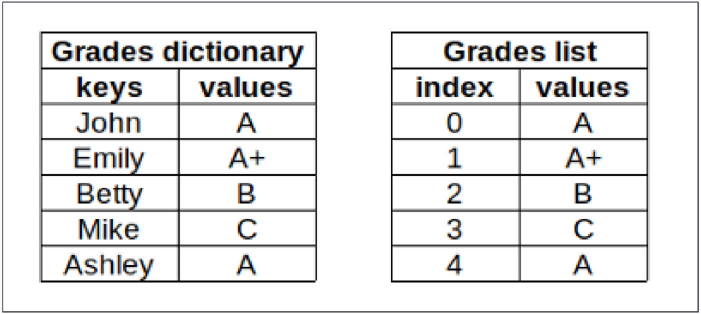
Kristoffer Nielbo

Center for Humanities Computing AArhus|chcaa.io
aarhus university, denmark

SOLIDUM PETIT IN PROFUNDIS · UNIVERSITAS ARHUSIENSIS

CENTER FOR HUMANITIES
COMPUTING AARHUS

Collections



Grades dictionary	
keys	values
John	A
Emily	A+
Betty	B
Mike	C
Ashley	A

Grades list	
index	values
0	A
1	A+
2	B
3	C
4	A

lists, `list` and dictionaries, `dict`, are used to store collections of data.
both can be used to store multi-dimensional data.

Lists

Remember the Pannang Curry 1-2-3!

1. Open your right refrigerator door and remove ingredients from the following locations: Door shelf 2. Spot 1; Crisper drawer 1, Spot 3; Crisper drawer 1, Spot 5.

```
crisper_drawer = [’pepper’, ’zucchini’, ’onion’]

fridge = [[’pepper’, ’zucchini’, ’onion’],
          [’cabbage’, ’lettuce’, ’garlic’],
          [’apple’, ’pear’, ’banana’]]
```

The `list` is an ordered collection that allow you to store multiple items in a specific sequence

Dictionaries

With the container data structures, we can begin to model complex relational real-world objects!

```
crisper_drawer = {’pepper’: 3, ’zucchini’: 1, ’onion’: 5}

fridge = { ’crisper_1’: {’chili’: 2, ’garlic’: 3, ’onion’: 5},
         ’crisper_2’: {’cabbage’: 2, ’lettuce’: 1, ’peper’: 3},
         ’door’: {
             ’shelf_1’: {’yeast’: 1, ’eggs’: 5},
             ’shelf_2’: {’milk’: 2, ’yogurt’: 1},
             ’shelf_3’: {’water’: 3}
             }
         }
```

The `dictionary` is an unordered* collection that allow you to store items as `key:value` pairs.

Hash Table

Dictionaries is Python's builtin implementation of hash tables, that is, an unordered collection of key-value pairs, where each key is unique.

- used to implement map and set data structures in most common programming languages.
- offer a combination of efficient lookup, insert and delete operations.
- neither arrays nor linked lists can achieve this

```
if questions:
   try:
      answer()
   except RunTimeError:
      pass
   else:
      print('break')
```