02.10-dictionaries

February 21, 2020

1

dictionary hash map

1.1

1.1.1

Python {} dict()

```
In [1]: a = {}
        type(a)
```

```
Out[1]: dict
```

```
In [2]: a = dict()
        type(a)
```

```
Out[2]: dict
```

dict

1.1.2

```
In [3]: a["one"] = "this is number 1"
        a["two"] = "this is number 2"
        a
```

```
Out[3]: {'one': 'this is number 1', 'two': 'this is number 2'}
```

1.1.3

```
In [4]: a['one']
```

```
Out[4]: 'this is number 1'
```

1.1.4

```
In [5]: a["one"] = "this is number 1, too"
        a

Out[5]: {'one': 'this is number 1, too', 'two': 'this is number 2'}
```

1.1.5

Pythonkey: value

```
In [6]: b = {'one': 'this is number 1', 'two': 'this is number 2'}
        b['one']

Out[6]: 'this is number 1'
```

1.1.6

`print` **Python**,

```
In [7]: print (a)

{'one': 'this is number 1, too', 'two': 'this is number 2'}


In [8]: print (b)

{'one': 'this is number 1', 'two': 'this is number 2'}
```

Python

```
In [9]: #
        a[0]


        ---------------------------------------------------------------------------

        KeyError                                  Traceback (most recent call last)

        <ipython-input-9-f76a772240d5> in <module>
          1 #
    ----> 2 a[0]


        KeyError: 0
```

1.1.7

hashPythonPython

```
In [10]: synonyms = {}
         synonyms['mutable'] = ['changeable', 'variable', 'varying', 'fluctuating',
                                'shifting', 'inconsistent', 'unpredictable', 'inconstant',
                                'fickle', 'uneven', 'unstable', 'protean']
         synonyms['immutable'] = ['fixed', 'set', 'rigid', 'inflexible',
                                  'permanent', 'established', 'carved in stone']
         synonyms
```

```
Out[10]: {'mutable': ['changeable',
           'variable',
           'varying',
           'fluctuating',
           'shifting',
           'inconsistent',
           'unpredictable',
           'inconstant',
           'fickle',
           'uneven',
           'unstable',
           'protean'],
          'immutable': ['fixed',
           'set',
           'rigid',
           'inflexible',
           'permanent',
           'established',
           'carved in stone']}
```

```
In [12]: #
         e1 = {'mag': 0.05, 'width': 20}
         e2 = {'mag': 0.04, 'width': 25}
         e3 = {'mag': 0.05, 'width': 80}
         e4 = {'mag': 0.03, 'width': 30}
         #
         events = {500: e1, 760: e2, 3001: e3, 4180: e4}
         events
```

```
Out[12]: {500: {'mag': 0.05, 'width': 20},
          760: {'mag': 0.04, 'width': 25},
          3001: {'mag': 0.05, 'width': 80},
          4180: {'mag': 0.03, 'width': 30}}
```

```
In [13]: people = [
             {'first': 'Sam', 'last': 'Malone', 'name': 35},
             {'first': 'Woody', 'last': 'Boyd', 'name': 21},
             {'first': 'Norm', 'last': 'Peterson', 'name': 34},
             {'first': 'Diane', 'last': 'Chambers', 'name': 33}
         ]
         people

Out[13]: [{'first': 'Sam', 'last': 'Malone', 'name': 35},
          {'first': 'Woody', 'last': 'Boyd', 'name': 21},
          {'first': 'Norm', 'last': 'Peterson', 'name': 34},
          {'first': 'Diane', 'last': 'Chambers', 'name': 33}]
```

1.1.8 dict

```
dict()
```

```
In [14]: inventory = dict(
             [('foozelator', 123),
              ('frombicator', 18),
              ('spatzleblock', 34),
              ('snitzelhogen', 23)
             ])
         inventory

Out[14]: {'foozelator': 123, 'frombicator': 18, 'spatzleblock': 34, 'snitzelhogen': 23}
```

```
In [15]: inventory['frombicator'] += 1
         inventory

Out[15]: {'foozelator': 123, 'frombicator': 19, 'spatzleblock': 34, 'snitzelhogen': 23}
```

1.2

```
In [16]: data = {}
         data[1.1 + 2.2] = 6.6
         #
         data[3.3]


         ---------------------------------------------------------------------------

         KeyError                                  Traceback (most recent call last)

         <ipython-input-16-9d18186b97d9> in <module>
```

```
      2 data[1.1 + 2.2] = 6.6
      3 #
----> 4 data[3.3]


KeyError: 3.3
```

data

```
In [17]: data

Out[17]: {3.3000000000000003: 6.6}
```

```
In [18]: connections = {}
         connections[('New York', 'Seattle')] = 100
         connections[('Austin', 'New York')] = 200
         connections[('New York', 'Austin')] = 400
```

('New York', 'Austin') ('Austin', 'New York')

```
In [19]: print (connections[('Austin', 'New York')])
         print (connections[('New York', 'Austin')])

200
400
```

1.3

1.3.1 get

Python get

`d.get(key, default = None)`

key default None

```
In [20]: a = {}
         a["one"] = "this is number 1"
         a["two"] = "this is number 2"
```

```
In [21]: a["three"]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)

<ipython-input-21-ba528808dd1a> in <module>
----> 1 a["three"]


KeyError: 'three'
```

get

```
In [22]: print (a.get("three"))
None
```

```
In [23]: a.get("three", "undefined")
Out[23]: 'undefined'
```

1.3.2 pop

pop

`d.pop(key, default = None)`

key default None

```
In [24]: a
Out[24]: {'one': 'this is number 1', 'two': 'this is number 2'}
```

```
In [25]: a.pop("two")
Out[25]: 'this is number 2'
In [26]: a
Out[26]: {'one': 'this is number 1'}
```

```
In [27]: a.pop("two", 'not exist')
Out[27]: 'not exist'
```

del

```
In [28]: del a["one"]
         a
Out[28]: {}
```

1.3.3 update

update

`d.update(newd)`

newdd

```
In [29]: person = {}
         person['first'] = "Jmes"
         person['last'] = "Maxwell"
         person['born'] = 1831
         print (person)
```

```
{'first': 'Jmes', 'last': 'Maxwell', 'born': 1831}
```

'first''James''middle''Clerk'

```
In [30]: person_modifications = {'first': 'James', 'middle': 'Clerk'}
         person.update(person_modifications)
         print (person)
```

```
{'first': 'James', 'last': 'Maxwell', 'born': 1831, 'middle': 'Clerk'}
```

1.3.4 in

```
In [31]: barn = {'cows': 1, 'dogs': 5, 'cats': 3}
```

in

```
In [32]: 'chickens' in barn
```

```
Out[32]: False
```

```
In [33]: 'cows' in barn
```

```
Out[33]: True
```

1.3.5 keys values items

`d.keys()`

`d.values()`

`d.items()`

```
In [34]: barn.keys()

Out[34]: dict_keys(['cows', 'dogs', 'cats'])

In [35]: barn.values()

Out[35]: dict_values([1, 5, 3])

In [36]: barn.items()

Out[36]: dict_items([('cows', 1), ('dogs', 5), ('cats', 3)])
```