

## Тема 1.4. Словари. Работа со словарями

# словари - наборы объектов, доступ к которым происходит не по индексу, а по ключу

# ключ - неизменяемый объект - число, строка или кортеж

# элементы словаря могут содержать объекты произвольного типа данных и иметь неограниченную степень вложенности

# элементы в словаре располагаются в произвольном порядке

# для получения элемента надо указать ключ, который использовался при сохранении значения.

# словари относятся к отображениям, а не к последовательностям, поэтому

# к словарям не применимы функции для работы с последовательностями,

# а также операции извлечения среза, конкатенации, повторения и др.

# словари, как и списки, относятся к изменяемым типам данных

# создать словарь можно несколькими способами:

# функция dict() Формат:

# dict(<Ключ1>=<Значение1>[,...,<КлючN>=<ЗначениеN>])

# dict(<Словарь>)

# dict(<Список кортежей с двумя элементами (Ключ, Значение)>)

# dict(<Список списков с двумя элементами (Ключ, Значение)>)

# Если параметры не указаны, то создается пустой словарь

```
d = dict()      # создается пустой словарь d= {}
```

```
print('d=', d)
```

```
d = dict(a = 1, b = 2)
```

```
print('d=', d)   # d = {'b': 2, 'a': 1}
```

```
d = dict({'a': 1, 'b': 2, 'c': 3, 'd': 4}) # создать словарь
```

```
print('d=', d)          # d = {'b': 2, 'd': 4, 'c': 3, 'a': 1}
```

```
d = dict([('a', 1), ('b', 2), ('c', 5)])    # список кортежей
```

```
print('d=', d)          # d = {'b': 2, 'a': 1, 'c': 5}
```

```
d = dict([['a', 1], ['b', 8], ['c', 12]])  # список списков
```

```
print('d=', d)          # d = {'a': 1, 'b': 8, 'c': 12}
```

```
# объединить два списка в список кортежей и создать словарь
```

```
k = ['z1', 'z2', 'z3', 'z4']              # список ключей
```

```
v = [1, 2, 3, 4]                          # список значений
```

```
zip(k, v)                                  # создать список кортежей
```

```
d = dict(zip(k, v))                       # создать словарь
```

```
print('d=', d)                          # d= {'z4': 4, 'z2': 2, 'z3': 3, 'z1': 1}
```

```
k = ['z1', 'z2', 'z3', 'z4']              # список ключей
```

```
v = [5, 7]                                # список значений - меньше, чем список ключей
```

```
zip(k, v)                                  # создать список кортежей
```

```
d = dict(zip(k, v))                       # создать словарь
```

```
print('d=', d)                          # d = {'z2': 7, 'z1': 5}
```

```
# создать словарь:
```

```
d = {} # создать пустой словарь
```

```
d = {'a': 1, 'b': 2, 'c': 3} # создать словарь с элементами
```

```
print(d)                                # {'a': 1, 'b': 2, 'c': 3}
```

```
# заполнить словарь поэлементно
```

```
d = {}
```

```
d['a'] = 1
```

```
d['b'] = 2
```

```
d['c'] = 7
```

```
print(d) # {'b': 2, 'a': 1, 'c': 7}
```

```
a = 's'
```

```
b = 'z'
```

```
c = 'f'
```

```
d = {}
```

```
d[a] = 5
```

```
d[b] = 6
```

```
d[c] = 7
```

```
print(d) # {'z': 6, 's': 5, 'f': 7}
```

```
# метод dict.fromkeys(<Последовательность>[, <Значение>])
```

```
d = dict.fromkeys(['a', 'b', 'c'])
```

```
print(d) # {'a': None, 'b': None, 'c': None}
```

```
d = dict.fromkeys(['a', 'b', 'c'], 0) # указан список
```

```
print(d) # {'a': 0, 'b': 0, 'c': 0}
```

```
d = dict.fromkeys(['a', 'b', 'c'], 7)
```

```
print(d) # {'b': 7, 'a': 7, 'c': 7}
```

```
d = dict.fromkeys(('a', 'b', 'c'), 4) # указан кортеж
```

```
print(d) # {'b': 4, 'c': 4, 'a': 4}
```

```
# для списков и словарей НЕЛЬЗЯ делать групповое присваивание
```

```
d1 = d2 = {'a':1, 'b':2} # обе переменные ссылаются на один и тот же объект
# если изменится объект по ссылке d1, то изменится и объект по ссылке d2
d2['b'] = 10
print('d2["b"] = ', d2['b'])    # d2[b] = 10
print(d1,d2)    # {'b': 10, 'a': 1} {'b': 10, 'a': 1}
```

# создать поверхностную копию словаря с помощью функции dict()

```
d1 = {'a':1, 'b':2} # создаем словарь
d2 = dict(d1)    # создаем поверхностную копию
print('d1=', d1) # d1= {'a': 1, 'b': 2}
print('d2=', d2) # d2= {'a': 1, 'b': 2}
print(d2 is d1)  # False - d2 и d1 это разные объекты
d1['a'] = 15
print('d1=', d1) # d1= {'a': 15, 'b': 2}
print('d2=', d2) # d2= {'a': 1, 'b': 2}
```

# создать поверхностную копию словаря с помощью метода copy()

```
d1 = {'a':1, 'b':2} # создаем словарь
d2 = d1.copy()    # создаем поверхностную копию
print('d1=', d1) # d1 = {'a': 1, 'b': 2}
print('d2=', d2) # d2 = {'a': 1, 'b': 2}
print(d2 is d1)   # False - d2 и d1 это разные объекты
```

```
d1['a'] = 23
print('d1=', d1) # d1= {'a': 23, 'b': 2}
print('d2=', d2) # d2= {'a': 1, 'b': 2}
```

# создание полной копии словаря функцией deepcopy() из модуля copy

```
d1 = {'a': 1, 'b': [20, 30, 40]}
d2 = dict(d1) # создаем поверхностную копию
```

```
print('d1=', d1) # d1= {'b': [20, 30, 40], 'a': 1}
print('d2=', d2) # d2= {'b': [20, 30, 40], 'a': 1}
print(d2 is d1) # False - d2 и d1 это разные объекты
d2['b'][0] = 'test'
print('d1=', d1) # d1= {'b': ['test', 30, 40], 'a': 1}
print('d2=', d2) # d2 = {'b': ['test', 30, 40], 'a': 1}
```

```
import copy
d3 = copy.deepcopy(d1) # создаем полную копию
d3['b'][1] = 800
print('d3=', d3) # d3 = {'b': ['test', 800, 40], 'a': 1}
print('d1=', d1) # d1= {'b': ['test', 30, 40], 'a': 1}
```