

Основы программирования на языке Python

Тема 12: Словари.

Создание словаря

```
empty_dict = dict()
empty_dict = {}
my_dict = {'норм': 'нормально', 'спс': 'спасибо', 'нзч': 'не за что'}
```

```
my_dict = {1: 'один', 2: 'два', 3: 'три', 1: 'раз'}
print(my_dict)
```

```
{1: 'раз', 2: 'два', 3: 'три'}
```

```
|
```

```
Process finished with exit code 0
```

```
strange_dict = {'строка': 42,
                17: ['нечетное', 'простое'],
                (55.75, 37.52): 'Москва',
                0: 'число',
                False: 'ложь'}
```

```
print(strange_dict)
```

```
{'строка': 42, 17: ['нечетное', 'простое'], (55.75, 37.52): 'Москва', 0: 'ложь'}
```

```
Process finished with exit code 0
```

```
|
```

Базовые операции над словарями

```
name2full = {'Дима': 'Дмитрий',  
             'Петя': 'Петр',  
             'Настя': 'Анастасия',  
             'Вика': 'Виктория'}
```

```
print(name2full['Настя'], name2full['Петя'], sep='\n')
```

```
Анастасия  
Петр|
```

```
Process finished with exit code 0
```

```
name2full['Катя'] = 'Екатерина'
```

Базовые операции над словарями

```
teen2adult = { 'спс': 'спасибо', 'нзч': 'не за что', 'норм': 'нормально' }
```

```
print('спс' in teen2adult)    # есть ли ключ 'спс' в словаре teen2adult
print(len(teen2adult))       # каков размер (кол-во пар) словаря?
```

```
del teen2adult['спс']        # удалить пару с ключем 'спс'
print(teen2adult)
```

```
True
```

```
3
```

```
{'нзч': 'не за что', 'норм': 'нормально'}
```

```
Process finished with exit code 0
```

```
|
```

Методы словарей

Метод	Пример	Описание
<code>clear</code>	<code>d.clear()</code>	Удалить все элементы из словаря.
<code>copy</code>	<code>d.copy()</code>	Создать новый словарь, содержащий то же самое, что и исходный.
<code>get</code>	<code>d.get(key)</code>	Получить значение по заданному ключу.
<code>setdefault</code>	<code>d.setdefault(key, value)</code>	Добавить пару <code>key->value</code> , если ключа <code>key</code> не было в словаре, иначе ничего не делать.
<code>keys</code>	<code>d.keys()</code>	Получить ключи словаря.
<code>values</code>	<code>d.values()</code>	Получить значения словаря.
<code>items</code>	<code>d.items()</code>	Получить пары словаря в виде кортежей, содержащий ключ и соответствующее значение.
<code>pop</code>	<code>d.pop(key)</code>	Удалить пару с ключем <code>key</code> из словаря и вернуть значение, соответствующее ключу <code>key</code> .
<code>popitem</code>	<code>d.popitem()</code>	Удалить случайную пару из словаря и вернуть ее в виде кортежа.

Пример

```
d = dict()

d['sin'] = 'синус'
print(d.get('sin'))  # то же, что и print(d['sin'])
print()

# разница между d.get('sin') и d['синус'] в том, что у get можно задать второй аргумент
# Если переданного в get ключа нет в словаре, то результатом будет переданный второй аргумент
print(d.get('cos', 'Нет такого ключа'))
print()

d.setdefault('cos', 'косинус')  # создаст пару 'cos'-'косинус', так как раньше ключа 'cos' не было в словаре
d.setdefault('cos', 'не знаю')  # этот вызов уже ничего не сделает, так как ключ 'cos' уже был в словаре
print(d)
print()

print(list(d.keys()))          # Нет такого ключа
print(list(d.values()))
print(list(d.items()))
print()

d1 = d.copy()
print(d.pop('cos'))
print(d.popitem())
print(d)
print(d1)
print()

d1.clear()
print(d1)

{'sin': 'синус', 'cos': 'косинус'}

['sin', 'cos']
['синус', 'косинус']
[('sin', 'синус'), ('cos', 'косинус')]

косинус
('sin', 'синус')
{}
{'sin': 'синус', 'cos': 'косинус'}

{}

Process finished with exit code 0
```

Базовые операции над словарями

```
d = { 'Россия': 'Москва',  
      'Польша': 'Варшава',  
      'Украина': 'Киев',  
      'Шри-Ланка': 'Шри-Джаявардинепура-Котте' }
```

Способ №1. Перебор ключей

```
for key in d: # то же, что и for key in d.keys()  
    print(key, '->', d[key])  
print()
```

```
Россия -> Москва  
Польша -> Варшава  
Украина -> Киев  
Шри-Ланка -> Шри-Джаявардинепура-Котте
```

Способ №2. Перебор пар

```
for key, value in d.items():  
    print(key, '->', value)  
print()
```

```
Россия -> Москва  
Польша -> Варшава  
Украина -> Киев  
Шри-Ланка -> Шри-Джаявардинепура-Котте
```

Способ №3. Перебор значений

```
for value in d.values():  
    print(value)  
print()
```

```
Москва  
Варшава  
Киев  
Шри-Джаявардинепура-Котте
```

```
Process finished with exit code 0
```

Основы программирования на языке Python

Тема 12: Словари.