

# **Списки. Срезы СПИСКОВ**

## СПИСКИ

```
students = ['Ivan', 'Masha', 'Sasha']
```

```
for student in students:
```

```
    print("Hello, " + student + "!")
```

Hello, Ivan!

students[0] 'Ivan'

Hello, Masha!

students[1] 'Masha'

Hello, Sasha!

students[2] 'Sasha'

# Изменение списков

В отличие от изученных типов данных (**int**, **float**, **str**) списки (**list**) являются изменяемыми.

Можно изменить конкретный элемент списка:

```
students = ['Ivan', 'Masha', 'Sasha']
```

```
students[1] = 'Oleg'
```

```
print(students)
```

```
['Ivan', 'Oleg', 'Sasha']
```

## Доступ к элементам списка

```
students = ['Ivan', 'Masha', 'Sasha']
```

Длина списка: `len(students)`

Результат: 3

<code>students[0]</code>	<code>'Ivan'</code>	<code>students[-1]</code>	<code>'Sasha'</code>
<code>students[1]</code>	<code>'Masha'</code>	<code>students[-2]</code>	<code>'Masha'</code>
<code>students[2]</code>	<code>'Sasha'</code>	<code>students[-3]</code>	<code>'Ivan'</code>

# Операции со списками

+  
students = ['Ivan', 'Masha', 'Sasha']  
teachers = ['Oleg', 'Alex']  
students + teachers  
Результат: ['Ivan', 'Masha', 'Sasha', 'Oleg', 'Alex']

\*  
[0, 1] \* 4  
Результат: [0, 1, 0, 1, 0, 1, 0, 1]

# Операции со списками

---

```
A = [1, 3, 4, 23, 5]
```

```
A = [1, 3] + [4, 23] + [5]
```

```
[1, 3, 4, 23, 5]
```

```
A = [0] * 10
```

```
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

```
A = list ( range (10) )
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

## МЕТОДЫ СПИСКОВ

Метод	Что делает
<b>list.append(x)</b>	Добавляет элемент в конец списка
<b>list.extend(L)</b>	Расширяет список list, добавляя в конец все элементы списка L
<b>list.insert(i, x)</b>	Вставляет на i-ый элемент значение x
<b>list.remove(x)</b>	Удаляет первый элемент в списке, имеющий значение x. ValueError, если такого элемента не существует
<b>list.pop( i )</b>	Удаляет i-ый элемент и возвращает его. Если индекс не указан, удаляется последний элемент
<b>list.index(x, [start [, end]])</b>	Возвращает положение первого элемента со значением x (при этом поиск ведется от start до end)
<b>list.count(x)</b>	Возвращает количество элементов со значением x
<b>list.sort()</b>	Сортирует список
<b>list.reverse()</b>	Разворачивает список
<b>list.clear()</b>	Очищает список

# Добавление элементов в список

```
students = ['Ivan', 'Masha', 'Sasha']
```

```
students.append('Olga')
```

Результат: ['Ivan', 'Masha', 'Sasha', 'Olga']

```
students += ['Olga']
```

Результат: ['Ivan', 'Masha', 'Sasha', 'Olga', 'Olga']

```
students += ['Boris', 'Sergey']
```

Результат: ['Ivan', 'Masha', 'Sasha', 'Olga', 'Olga', 'Boris', 'Sergey']

Пустой список: students = []

Вставка  
элемента:

```
students = ['Ivan', 'Masha', 'Sasha']
```

```
students.insert(1, 'Olga')
```

Результат: ['Ivan', 'Olga', 'Masha', 'Sasha']



## Удаление элемента из списка

```
students = ['Ivan', 'Masha', 'Sasha']
```

```
students.remove('Sasha')
```

Результат: ['Ivan', 'Masha']

```
del students[0]
```

Результат: 'Masha'

## Список в обратном порядке

```
students = ['Sasha', 'Ivan', 'Masha']
```

```
students.reverse()
```

Результат: ['Masha', 'Ivan', 'Sasha']

# Генераторы списков

```
A = [ i for i in range(10) ]
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
A = [ i*i for i in range(10) ]
```

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

```
A = [ i*i for i in range(10) if  
i%2==0] [0, 4, 16, 36, 64]
```

условие  
отбора

```
A = list ( range(10) )
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```