

## Тема 6. Списки

Списки - это упорядоченные изменяемые коллекции объектов произвольных типов. Более простыми словами это объединение нескольких ячеек данных в одну переменную.

Например список имен учащихся в школе, до сегодняшнего дня мы бы писали так:

```
Name1 = «Маша»  
Name2 = «Павел»  
.....  
NameN = «Дарья»
```

Это не удобно, для этого и ввели списки, давайте посмотрим как мы это можем сделать используя список.

```
lst_name = [«Маша», «Павел», «Дарья» ]
```

Для того, чтобы обратиться к какому-нибудь элементу списка мы должны написать имя списка и в [ ] написать индекс (порядковый номер элемента)

**Важно!!! Порядковые номера начинаются с 0, не с 1**

Поэтому написав `lst_name[1]` мы получим имя Павел, давайте проверим.

```
30 lst_name = ["Маша", "Павел", "Дарья"]  
31 print(lst_name[1])  
32
```

Run: Lesson7 ×

/Users/boss/Documents/programming/un  
Павел

Мы можем вывести весь список:

```
30 lst_name = ["Маша", "Павел", "Дарья"]  
31 print(lst_name)
```

Run: Lesson7 ×

/Users/boss/Documents/programming/u  
['Маша', 'Павел', 'Дарья']

Так же, мы можем создать и пустой список, в который потом научимся добавлять данные:

```
lst = []      # пустой список
```

Чтобы добавить данные в список необходимо вызвать у списка функцию `append(x)`, где `x` - данные которые хотим поместить в список. `append` помещает данные в конец списка

```
31 lst = []
32 lst.append(6)
33 lst.append("Python")
34 lst.append(True)
35 lst.append(3.14)
36 print(lst)
--
```

Run: Lesson7 x

/Users/boss/Documents/prog  
[6, 'Python', True, 3.14]

Если мы хотим поменять данные, то мы обращаемся к элементу списка по его индексу и изменяем его.

```
lst = [6, 'Python', True, 3.14]
print(lst)
lst[0] = 8
print(lst)
lst[3] += 100
print(lst)
```

Lesson7 x

/Users/boss/Documents/programming/u  
[6, 'Python', True, 3.14]  
[8, 'Python', True, 3.14]  
[8, 'Python', True, 103.14]

Если мы захотим вставить данные по конкретному индексу - существует функция `insert(i, x)`, где `i` - индекс, `x` - данные

```
lst = [6, 'Python', True, 3.14]
print(lst)
lst.insert(1, "Используем инсерт")
print(lst)
```

Lesson7 x

```
/Users/boss/Documents/programming/untitled/venv
[6, 'Python', True, 3.14]
5 [6, 'Используем инсерт', 'Python', True, 3.14]
```

`remove(x)` - удаляет первый элемент в списке, имеющий значение `x`

```
lst = [6, 'Python', True, 3.14]
print(lst)
lst.remove(True)
print(lst)
```

Lesson7 x

```
/Users/boss/Documents/programm
[6, 'Python', True, 3.14]
[6, 'Python', 3.14]
```

`copy()` - если нам необходимо данные из одного списка скопировать в другой

```
lst = [6, 'Python', True, 3.14]
lst2 = lst.copy()
lst2.append(2)
print(lst)
print(lst2)
```

Lesson7 x

```
/Users/boss/Documents/programming/unt
[6, 'Python', True, 3.14]
[6, 'Python', True, 3.14, 2]
```

**pop([i])** - удаляет i-ый элемент и возвращает его. Если индекс не указан, удаляется последний элемент

```
lst = [6, 'Python', True, 3.14]
print(lst.pop()) # удалит и вернет последний элемент
print(lst.pop(0)) # удалит и вернет первый элемент
print(lst) # выведет на экран список после удалений элементов
```

```
Lesson7 x
/Users/boss/Documents/programming/untitled/venv/bin/python /Users/bos
3.14
6
['Python', True]
```

Ну а теперь рассмотрим задачи.

1) Есть список `a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]`.

Выведите все элементы, которые меньше 5.

```
a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]

for l in a:
    if l < 5:
        print(l)
```

```
Lesson7 x
/Users/boss/Documents/programming/untitled/ve
1
1
2
3
```

2) Заполнить список 15 **нулями**, кроме первого и последнего элементов, которые должны быть равны **единице**;

```
a = []
i = 0
while i < 17:
    a.append(0)
    if i == 0 or i == 16:
        a[i] = 1
    i += 1
print(a)
```

Lesson7 x /Users/boss/Documents/programming/untitled/venv/bin/p  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]

3) Пользователь формирует список из пяти элементов. Необходимо найти наибольший элемент списка и вывести его на экран

```
lst = []
i = 0
while i < 5:
    lst.append(int(input("Введите число ")))
    i += 1

print("список - " + str(lst))
max_num = lst[0]

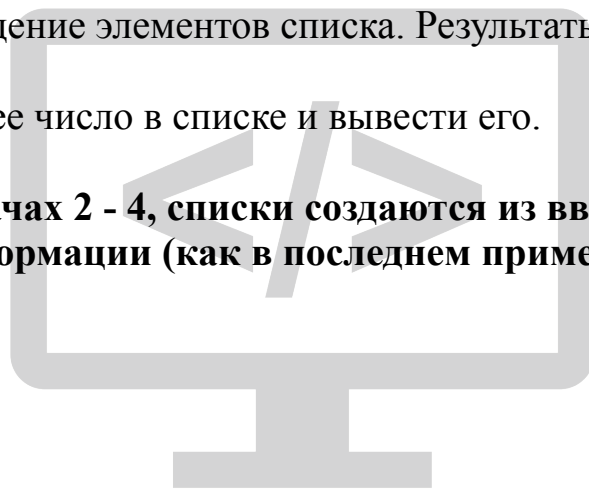
for l in lst:
    if l > max_num:
        max_num = l
print("максимальное число: {}".format(max_num))
```

Lesson7 x /Users/boss/Documents/programming/untitled/venv/bin/p  
Введите число 35  
Введите число 754  
Введите число 21  
Введите число 80  
Введите число 653  
список - [35, 754, 21, 80, 653]  
максимальное число: 754

### Практическое задание:

- 1) Сформировать возрастающий список из **чётных чисел** (количество элементов **12**)
- 2) Найдите сумму элементов списка. Результаты вывести на экран
- 3) Найдите произведение элементов списка. Результаты вывести на экран
- 4) Найти наименьшее число в списке и вывести его.

**Примечание:** в задачах 2 - 4, списки создаются из введенной пользователем информации (как в последнем примере)



OVERONE