

### Практическая работа 3- Списки (list). Функции и их методы

Списки можно также назвать массивами, так как выполняют они схожую роль. В Python существует множество разновидностей таких "массивов". В уроке мы рассмотрели лишь первый тип, который называется списком и имеет следующие характеристики:

- его элементы можно переопределять;
- в него можно добавлять новые элементы;
- из него можно удалять элементы;
- в качестве ключей идут индексы (числа - 0, 1, 2...).

Это не все характеристики списка, тем не менее это ключевые моменты, которые отличают его от других "массивов" в языке Python.

Для списков существует большое количество функций, которые можно применять для добавления, удаления или же поиска элементов.

В списках можно хранить различную информацию. К примеру, вы можете хранить в нем сведения про всех студентов в группе.

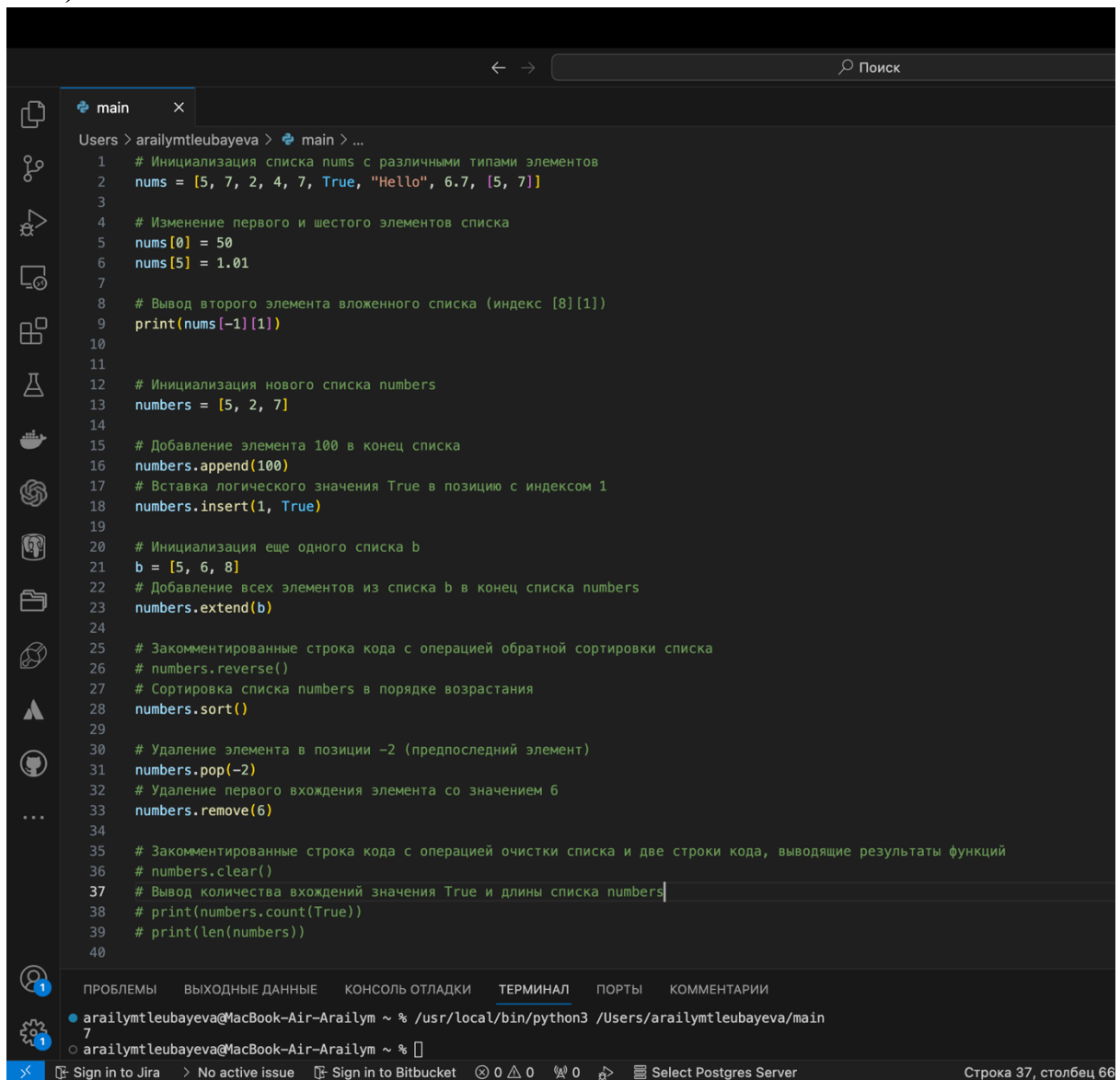
Пример:

```
lis = ["Андрей", "Иван", "Василий", "Петро", "Максим", "Дима"]
```

После такой записи в одном месте у вас будут собраны все имена студентов в группе. Согласитесь, это намного удобнее, нежели создавать 10 или даже 20 разных переменных. В дальнейшем мы изучим дополнительные вещи, которые позволят хранить не просто имена студентов, а еще и характеристики каждого из студентов.

## Отработка примеров:

### 1) Списки



The screenshot shows a code editor with a file named 'main'. The code is in Russian and demonstrates various list operations in Python. The terminal at the bottom shows the command to run the script using Python 3.

```
1 # Инициализация списка nums с различными типами элементов
2 nums = [5, 7, 2, 4, 7, True, "Hello", 6.7, [5, 7]]
3
4 # Изменение первого и шестого элементов списка
5 nums[0] = 50
6 nums[5] = 1.01
7
8 # Вывод второго элемента вложенного списка (индекс [8][1])
9 print(nums[-1][1])
10
11
12 # Инициализация нового списка numbers
13 numbers = [5, 2, 7]
14
15 # Добавление элемента 100 в конец списка
16 numbers.append(100)
17 # Вставка логического значения True в позицию с индексом 1
18 numbers.insert(1, True)
19
20 # Инициализация еще одного списка b
21 b = [5, 6, 8]
22 # Добавление всех элементов из списка b в конец списка numbers
23 numbers.extend(b)
24
25 # Закомментированная строка кода с операцией обратной сортировки списка
26 # numbers.reverse()
27 # Сортировка списка numbers в порядке возрастания
28 numbers.sort()
29
30 # Удаление элемента в позиции -2 (предпоследний элемент)
31 numbers.pop(-2)
32 # Удаление первого вхождения элемента со значением 6
33 numbers.remove(6)
34
35 # Закомментированная строка кода с операцией очистки списка и две строки кода, выводящие результаты функций
36 # numbers.clear()
37 # Вывод количества вхождений значения True и длины списка numbers
38 # print(numbers.count(True))
39 # print(len(numbers))
40
```

Terminal output:

```
arailymtleubayeva@MacBook-Air-Arilym ~ % /usr/local/bin/python3 /Users/arailymtleubayeva/main
7
arailymtleubayeva@MacBook-Air-Arilym ~ %
```

### 2) Работа со списка

← →

Поиск

main

×

Users > arailymtleubayeva > main > ...

```
1 # Создание списка nums с различными типами данных
2 nums = [5, 2, 7, "50", False]
3
4 # Цикл for, проходящий по каждому элементу списка nums
5 # и умножающий каждый элемент на 2, затем выводящий результат.
6 # Обратите внимание, что это не изменит оригинальный список, так как
7 # переменная el в данном случае создаёт копию элемента, а не ссылается на него.
8 for el in nums:
9     el *= 2
10    print(el)
11
12
13 # Ввод значения длины списка от пользователя и приведение его к типу int
14 n = int(input("Enter length: "))
15
16 # Инициализация пустого списка user_list
17 user_list = []
18
19 # Инициализация счётчика i значением 0
20 i = 0
21 # Цикл while, который выполняется, пока i меньше n.
22 # На каждой итерации цикла от пользователя запрашивается ввод нового элемента,
23 # который добавляется в список user_list.
24 while i < n:
25     string = "Enter element #" + str(i + 1) + ": "
26     user_list.append(input(string))
27     i += 1 # Инкрементация счётчика i на 1 на каждой итерации
28
29 # Вывод полного списка user_list после завершения цикла
30 print(user_list)
31
```

ПРОБЛЕМЫ

ВЫХОДНЫЕ ДАННЫЕ

КОНСОЛЬ ОТЛАДКИ

ТЕРМИНАЛ

ПОРТЫ

КОММЕНТАРИИ

arailymtleubayeva@MacBook-Air-Arilym ~ % /usr/local/bin/python3 /Users/arailymtleubayeva/main

10

4

14

5050

0

Enter length:

Sign in to Jira

> No active issue

Sign in to Bitbucket

0 0 0

0

Select Postgres Server

Строка 24, столбе

## Самостоятельная работа

### 1) Работа со списками

Создайте список при помощи цикла for, который будет состоять из 5 элементов.

Создайте пустой список и выполните над ним операции:

добавьте в него число 5 и -6

добавьте в него целиком весь первый список

выполните сортировку списка

Выведите на экран оба списка без использования циклов.

### 2) Создание списка пользователем

В программе получите строку введенную пользователем с клавиатуры.

При помощи цикла разбейте строку на символы (отдельные буквы) и каждую отдельную букву добавьте как новый элемент списка.

### 3) Вывод по отдельности

У вас есть список:

```
lis = ["Андрей", "Иван", "Василий", "Петро", "Максим", "Дима"]
```

Вам необходимо вывести каждое слово из списка, после чего вывести каждую букву из каждого слова, например:

Андрей # Сначала слово целиком

А # Потом

н # каждая

д # буква

р # из этого

е # слова

й # отдельно

### 4) Наименьший элемент

Есть список чисел:

```
lis = [1, 34, 8, 0, -5, 7, 32, 74, 59, 92, 41, 10, -2]
```

Найдите самый маленький элемент списка и переместите его в зависимости от условия:

если найденный элемент меньше за ноль, то поместите его в конец списка

если найденный элемент больше или равен нулю, то поместите его в начало списка.

### 5) Добавление списка пользователем

Попросите пользователя ввести число в переменную N.

Добавьте в список some N количество новых элементов. Каждый элемент также должен вводить пользователь с клавиатуры.

Список:

```
some = [9, "Hi", 23.5, "A"]
```

#### **6) Задача**

Попросите пользователя ввести три числа с клавиатуры. На основе этих чисел создайте программу, которая будет объединять три значения в список и отдельно в кортеж. Данные должны объединены через запятую.

- 1) Отработать все примеры , обязательно скриншот кода и результата с решением задач в комментариях!**
- 2) Подготовьте отчет(Word файл) по самостоятельной работе с скриншотами исходного кода и результата компиляции.**
- 3) На скриншоте пусть будет видно что это ваша работа (Создайте файл ru с вашим именем).**
- 4) Исходные файлы с кодом загрузите в репозиторий GitHub и оставьте ссылку в отчете.**
- 5) Дедлайн до 19 октября**