

Задание 1.

Исходные данные:

Создать список и заполнить его элементами различных типов данных. Реализовать скрипт проверки типа данных каждого элемента. Использовать функцию **type()** для проверки типа. Элементы списка можно не запрашивать у пользователя, а указать явно, в программе.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> number1 = 25

>>> object1 = number1

>>> number_1 = 75.5

>>> object2 = number_1

>>> a = 'lego'

>>> object3 = a

>>> b = [15, 77]

>>> object4 = b

>>> c = (15, 55)

>>> object5 = c

>>> d = {55, 77}

>>> object6 = d

>>> my_list = [object1, object2, object3, object4, object5, object6]

>>> type(my_list)
<class 'list'>

>>> type(object1)
<class 'int'>

>>> type(object2)
<class 'float'>

>>> type(object3)
<class 'str'>

>>> type(object4)
<class 'list'>

>>> type(object5)
<class 'tuple'>

>>> type(object6)
```

```
<class 'set'>
```

```
>>>
```

Задание 2.

Исходные данные:

Для списка реализовать обмен значений соседних элементов. Значениями обмениваются элементы с индексами 0 и 1, 2 и 3 и т. д. При нечётном количестве элементов последний сохранить на своём месте. Для заполнения списка элементов нужно использовать функцию **input()**.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> a = int(input("Введите целое число: "))
```

Введите целое число: 25

```
>>> n1 = 25
```

```
>>> # число получено от пользователя.
```

```
>>> b = int(input("Введите целое число: "))
```

Введите целое число: 55

```
>>> n2 = 55
```

```
>>> # число получено от пользователя.
```

```
>>> c = float(input("Введите десятичное число: "))
```

Введите десятичное число: 75.77

```
>>> n3 = 75.77
```

```
>>> # число получено от пользователя.
```

```
>>> d = float(input("Введите десятичное число: "))
```

Введите десятичное число: 99.55

```
>>> n4 = 99.55
```

```
>>> # число получено от пользователя.
```

```
>>> e = input("Введите текст: ")
```

Введите текст: 'car'

```
>>> n5 = 'car'
```

```
>>> # значение получено от пользователя.
```

```
>>> f = input("Введите текст: ")
```

Введите текст: 'train'

```

>>> n6 = 'train'
>>> my_list = [n1, n2, n3, n4, n5, n6]
>>> my_list
[25, 55, 75.77, 99.55, 'car', 'train']
>>> n1, n2 = 25, 55
>>> n1, n2 = n2, n1
>>> n2, n1 = 55, 25
>>> my_list[0] = n2
>>> my_list[1] = n1
>>> my_list
[55, 25, 75.77, 99.55, 'car', 'train']
>>> n3, n4 = 75.77, 99.55
>>> n3, n4 = n4, n3
>>> n4, n3 = 99.55, 75.77
>>> my_list[2] = n4
>>> my_list[3] = n3
>>> my_list
[55, 25, 99.55, 75.77, 'car', 'train']
>>> n5, n6 = 'car', 'train'
>>> n5, n6 = n6, n5
>>> n6, n5 = 'train', 'car'
>>> my_list[4] = 'train'
>>> my_list[5] = 'car'
>>> print(my_list)
[55, 25, 99.55, 75.77, 'train', 'car']
>>>

```

Задание 3.

Исходные данные:

Пользователь вводит месяц в виде целого числа от 1 до 12. Сообщить, к какому времени года относится месяц (зима, весна, лето, осень). Напишите решения через list и dict.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```

>>> a = int(input("Введите целое число: "))
Введите целое число: 6
>>> n = 6
>>> # число получено от пользователя.
>>> d = dict(long='seasons', time='munth')
>>> d = ((1, 1), (1, 2), (2, 3), (2, 4), (2, 5), (3, 6), (3, 7), (3, 8), (4, 9), (4, 10), (4, 11), (1, 12))]
>>> d
[(1, 1), (1, 2), (2, 3), (2, 4), (2, 5), (3, 6), (3, 7), (3, 8), (4, 9), (4, 10), (4, 11), (1, 12)]
>>> d[5]
(3, 6)
>>> print("summer")
summer
>>>

```

Задание 4.

Исходные данные:

Пользователь вводит строку из нескольких слов, разделённых пробелами. Вывести каждое слово с новой строки. Строки нужно пронумеровать. Если слово длинное, выводить только первые 10 букв в слове.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```

>>> 'one' 'car' 'dog' 'show'
'onecardogshow'
>>> for ind, el in enumerate(['one', 'car', 'dog', 'show']):
...     print(ind, el)
...
0 one
1 car
2 dog
3 show
>>>

```

Задание 5.

Исходные данные:

Реализовать структуру «Рейтинг», представляющую собой набор натуральных чисел, который не возрастает. У пользователя нужно запрашивать новый элемент рейтинга. Если в рейтинге существуют элементы с одинаковыми значениями, то новый элемент с тем же значением должен разместиться после них.

Подсказка. Например, набор натуральных чисел: 7, 5, 3, 3, 2.

Пользователь ввёл число 3. Результат: 7, 5, 3, 3, 3, 2.

Пользователь ввёл число 8. Результат: 8, 7, 5, 3, 3, 2.

Пользователь ввёл число 1. Результат: 7, 5, 3, 3, 2, 1.

Набор натуральных чисел можно задать сразу в коде, например, `my_list = [7, 5, 3, 3, 2]`.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> a = int(input("Введите натуральное число: "))
```

Введите натуральное число: 25

```
>>> n = 25
```

```
>>> # число получено от пользователя.
```

```
>>> my_list = [5, 7, 9, 15, 18]
```

```
>>> print(my_list)
```

```
[5, 7, 9, 15, 18]
```

```
>>> my_list.append(25)
```

```
>>> print(my_list)
```

```
[5, 7, 9, 15, 18, 25]
```

```
>>> b = int(input("Введите натуральное число: "))
```

Введите натуральное число: 3

```
>>> n = 3
```

```
>>> # число получено от пользователя.
```

```
>>> print(my_list)
```

```
[5, 7, 9, 15, 18, 25]
```

```
>>> my_list.insert(0, 3)
```

```
>>> print(my_list)
```

```
[3, 5, 7, 9, 15, 18, 25]
```

```
>>> c = int(input("Введите натуральное число: "))
```

Введите натуральное число: 12

```
>>> n = 12

>>> # число полученно от пользователя.

>>> print(my_list)
[3, 5, 7, 9, 15, 18, 25]

>>> my_list.insert(4, 12)

>>> print(my_list)
[3, 5, 7, 9, 12, 15, 18, 25]

>>> d = int(input("Введите натуральное число: "))
Введите натуральное число: 18

>>> n = 18

>>> # число полученно от пользователя.

>>> print(my_list)
[3, 5, 7, 9, 12, 15, 18, 25]

>>> my_list.append(18)

>>> print(my_list)
[3, 5, 7, 9, 12, 15, 18, 25, 18]

>>> f = int(input("Введите натуральное число: "))
Введите натуральное число: 15

>>> n = 15

>>> # число полученно от пользователя.

>>> print(my_list)
[3, 5, 7, 9, 12, 15, 18, 25, 18]

>>> my_list.append(15)

>>> print(my_list)
[3, 5, 7, 9, 12, 15, 18, 25, 18, 15]

>>> g = int(input("Введите натуральное число: "))
Введите натуральное число: 12

>>> n = 12

>>> # число полученно от пользователя.

>>> print(my_list)
[3, 5, 7, 9, 12, 15, 18, 25, 18, 15]

>>> my_list.append(12)

>>> print(my_list)
[3, 5, 7, 9, 12, 15, 18, 25, 18, 15, 12]

>>>
```

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> a = int(input("Введите натуральное число: "))
```

Введите натуральное число: 9

```
>>> n = 9
```

```
>>> # число получено от пользователя.
```

```
>>> my_list = [12, 0]
```

```
>>> my_list.insert(1, 9)
```

```
>>> print(my_list)
```

```
[12, 9, 0]
```

```
>>> b = int(input("Введите натуральное число: "))
```

Введите натуральное число: 6

```
>>> n = 6
```

```
>>> # число получено от пользователя.
```

```
>>> print(my_list)
```

```
[12, 9, 0]
```

```
>>> my_list.insert(2, 6)
```

```
>>> print(my_list)
```

```
[12, 9, 6, 0]
```

```
>>> c = int(input("Введите натуральное число: "))
```

Введите натуральное число: 3

```
>>> n = 3
```

```
>>> # число получено от пользователя.
```

```
>>> print(my_list)
```

```
[12, 9, 6, 0]
```

```
>>> my_list.insert(3, 3)
```

```
>>> print(my_list)
```

```
[12, 9, 6, 3, 0]
```

```
>>> [20~
```