

Вариант 1

1. Дан список `numbers = [10, 20, 30, 40, 50]`. Увеличьте первый элемент списка в 3 раза, а последний элемент — в 2 раза. Выведите измененный список.
2. Дан список содержаний слова: "яблоко", "банан", "апельсин", "виноград". Выведите на экран второй и четвертый элементы списка.
3. Напишите программу, которая генерирует список `squares`, содержащий квадраты целых чисел от 1 до 25. Выведите этот список.
4. На вход программе подается строка текста, разделенная пробелами. Напишите программу, которая создает из этой строки список, затем находит и выводит самый короткий элемент (строку) в этом списке. Если таких элементов несколько, выведите первый из них.

Пример ввода: ``привет мир python java``

Пример вывода: ``мир``

5. Дан список: `l = [Python,Java,C++,JavaScript,Ruby]`. Напишите программу, которая выводит строку вида: `Python@Java@C++@JavaScript@Ruby`.

Файл с решениями сюда: <https://forms.yandex.ru/u/68fa5db4068ff05f95e3ef80>

Вариант 2

1. Дан список `colors = ["красный", "зеленый", "синий"]`. Добавьте в конец списка элемент "желтый", а в начало — элемент "черный". Выведите полученный список и его длину.
2. Дан список `temperatures = [22.5, 18.0, 25.1, 20.8, 23.0]`. Найдите и выведите среднюю температуру (среднее арифметическое элементов списка).
3. Напишите программу, которая генерирует список `odd_numbers`, содержащий все нечетные числа от 11 до 121 включительно. Выведите этот список.
4. На вход программе подается строка текста, разделенная пробелами. Напишите программу, которая создает из этой строки список слов, а затем выводит список, состоящий только из тех слов, длина которых больше 5 символов.

Пример ввода: ``программирование это очень интересно и полезно``

Пример вывода: ``['программирование', 'интересно', 'полезно']``

5. Дана строка: ``sentence = "Я учусь программировать на языке Python"``. Разбейте строку на список слов. Сгенерируйте новый список, содержащий длины каждого слова из исходного списка (сколько букв в каждом слове). Выведите этот новый список. Например, для ``"Привет мир"`` результат был бы ``[6, 3]``.

Файл с решениями сюда: <https://forms.yandex.ru/u/68fa5db4068ff05f95e3ef80>

Вариант 3

1. Дан список `data = ["Python", 3.9, True, 2024]`. Измените второй элемент списка (по индексу 1) на `"Java"`, а третий — на `False`. Выведите измененный список.
2. Дан список `nums = [4, 8, 15, 16, 23, 42]`. Выведите на экран элементы списка с третьего по пятый (включительно).
3. Создайте список, содержащий 4 любых города (строки). Выведите список в обратном порядке.
4. На вход программе подается строка текста, содержащая целые числа, разделенные пробелами. Напишите программу, которая создает из этой строки список целых чисел, а затем выводит список, состоящий только из четных чисел исходного списка в том же порядке.

Пример ввода: ``4 7 12 5 8 3 10``

Пример вывода: ``[4, 12, 8, 10]``

5. Пользователь вводит числа, разделенные пробелами. Например: `"12 7 34 8 5"`. Преобразуйте эту строку в список целых чисел. Найдите сумму и произведение всех элементов этого списка. Выведите оба результата.

Файл с решениями сюда: <https://forms.yandex.ru/u/68fa5db4068ff05f95e3ef80>

Вариант 4

1. Дан список `letters = ['a', 'b', 'c', 'd']`. Поменяйте местами второй и третий элементы списка. Выведите результат.
2. Дан список `grades = [4, 5, 3, 4, 5, 5, 4]`. Посчитайте и выведите, сколько в списке отличных оценок (равных 5).
3. Напишите программу, которая генерирует список из 19 нулей. Затем замените центральный элемент (по индексу) на число 1. Выведите список.
4. На вход программе подается строка текста, разделенная пробелами. Напишите программу, которая создает из этой строки список, затем выводит список, в котором элементы идут в обратном порядке относительно исходного.

Пример ввода: ``сегодня хорошая погода``

Пример вывода: ``['погода', 'хорошая', 'сегодня']``

5. У вас есть список имен: `names = ["Анна", "Петр", "Мария", "Иван"]`. Создайте из этого списка строку:

`"Привет, Анна, Петр, Мария, Иван!"``.

Файл с решениями сюда: <https://forms.yandex.ru/u/68fa5db4068ff05f95e3ef80>