

## Лабораторная работа №3

**Тема:** Работа с итераторами, генераторами и генераторными выражениями.

**Цель работы:** Изучить понятия итератора и генератора в Python, а также их преимущества; ознакомиться с примерами их использования.

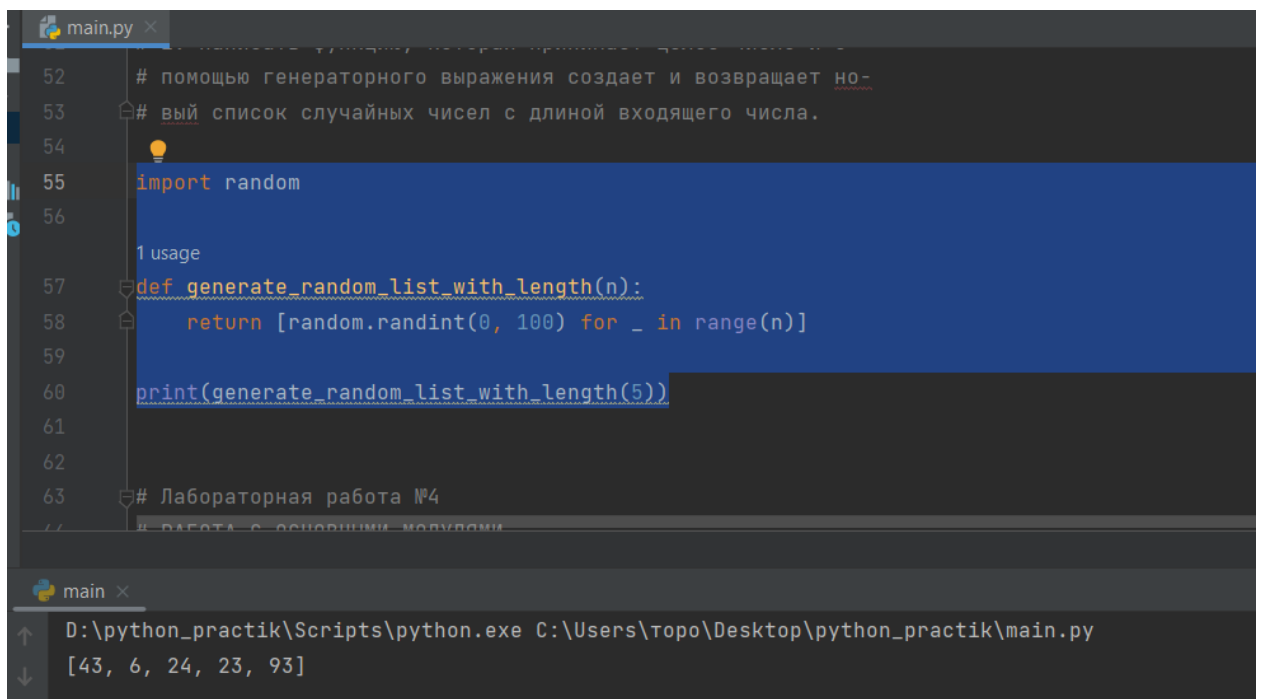
**Ход выполнения работы:**

**Задание 1:** Написать функцию, которая принимает целое число и с помощью генераторного выражения создает и возвращает новый список случайных чисел с длиной, равной входящему числу.

**Решение:**

Для генерации случайных чисел будем использовать библиотеку random, а с помощью генераторного выражения создадим список нужной длины.

**Код функции:**

A screenshot of a Python IDE with a dark theme. The editor window shows a file named 'main.py' with the following code: Line 52: A comment in Russian. Line 53: A comment in Russian. Line 54: A yellow lightbulb icon. Line 55: 'import random'. Line 56: A blank line. Line 57: '1 usage'. Line 58: 'def generate\_random\_list\_with\_length(n):'. Line 59: ' return [random.randint(0, 100) for \_ in range(n)]'. Line 60: 'print(generate\_random\_list\_with\_length(5))'. Line 61: A blank line. Line 62: A blank line. Line 63: A comment in Russian. Line 64: A comment in Russian. The console window at the bottom shows the command 'D:\python\_praktik\Scripts\python.exe C:\Users\topo\Desktop\python\_praktik\main.py' and the output '[43, 6, 24, 23, 93]'.

```
52 # помощью генераторного выражения создает и возвращает но-
53 # вый список случайных чисел с длиной входящего числа.
54
55 import random
56
57 1 usage
58 def generate_random_list_with_length(n):
59     return [random.randint(0, 100) for _ in range(n)]
60 print(generate_random_list_with_length(5))
61
62
63 # Лабораторная работа №4
64 # РАБОТА С ОСНОВНЫМИ МОДУЛЯМИ
```

main x

D:\python\_praktik\Scripts\python.exe C:\Users\topo\Desktop\python\_praktik\main.py

[43, 6, 24, 23, 93]

**Вывод:**

1. Функция корректно генерирует список случайных чисел с длиной, равной входному числу.
2. Если передать 0 в качестве аргумента, возвращается пустой список.

3. Генераторное выражение эффективно создает список с минимальными затратами памяти.

Выполнил студент ФИТУ, 3 курс, 090302-ИСТа-о22:

- Аксёнов Роман Владимирович