

Практическое занятие №12

Тема: Составление программ в функциональном стиле в IDE PyCharm Community.

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с использованием списковых включений, итераторов, генераторов в IDE PyCharm Community.

Задача №1

Постановка задачи.

Организовать и вывести последовательность из N случайных целых чисел. Из исходной последовательности организовать первую последовательность, содержащую числа кратные трем, и вторую – для всех остальных. Найти количество элементов в полученных последовательностях.

Текст программы:

```
import random

# Ввод числа N
N = int(input("Введите количество чисел в последовательности: "))

# Генерация случайной последовательности из N чисел
random_sequence = [random.randint(1, 100) for _ in range(N)]

# Разделение последовательности на числа кратные трем и не кратные трем
divisible_by_three = [num for num in random_sequence if num % 3 == 0]
not_divisible_by_three = [num for num in random_sequence if num % 3 != 0]

# Подсчет количества элементов в каждой последовательности
count_divisible_by_three = sum(1 for _ in divisible_by_three)
count_not_divisible_by_three = sum(1 for _ in not_divisible_by_three)

# Вывод результатов
print("Случайная последовательность:", random_sequence)
```

```
print("Числа кратные трем:", divisible_by_three)
print("Числа не кратные трем:", not_divisible_by_three)
print("Количество элементов в последовательности кратных трем:",
count_divisible_by_three)
print("Количество элементов в последовательности не кратных трем:",
count_not_divisible_by_three)
```

Протокол работы программы:

Введите количество чисел в последовательности: 10

Случайная последовательность: [26, 17, 95, 15, 25, 51, 19, 48, 28, 59]

Числа кратные трем: [15, 51, 48]

Числа не кратные трем: [26, 17, 95, 25, 19, 28, 59]

Количество элементов в последовательности кратных трем: 3

Количество элементов в последовательности не кратных трем: 7

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработал навыки составления программ с применением списковых включений, итераторов, генераторов в IDE PyCharm Community. Были использованы языковые структуры **for, in, if, import**.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.

Готовые программные коды выложены на GitHub.

Задача №2

Постановка задачи.

Составить генератор (yield), который выводит из строки только цифры

Текст программы:

```
string = input("Введите строку: ")
digits_generator = (int(char) for char in string if char.isdigit())
print(list(digits_generator))
```

Протокол работы программы:

Введите строку: avmd224hsj4901jsk4
[2, 2, 4, 4, 9, 0, 1, 4]

Вывод: в процессе выполнения практического занятия выработал навыки составления программ с применением списковых включений, итераторов, генераторов в IDE PyCharm Community. Были использованы языковые структуры **for, in, if**.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.

Готовые программные коды выложены на GitHub.