

Практическое занятие № 13

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.

Постановка задачи:

Задание 1. В последовательности на n целых чисел найти и вывести:

1. минимальный среди положительных
2. элементы кратные пяти
3. их среднее арифметическое

Задание 2. Из заданной строки отобразить только символы пунктуации.

Использовать библиотеку string.

Строка: --msg-template="\$FileDir\$\{path}:{line}:{column}:{C}:{symbol}){msg}"

Текст программы:

Задание № 1.

```
from functools import reduce
import math
```

```
# В последовательности на n целых чисел найти и вывести:
```

```
# 1.минимальный среди положительных
```

```
# 2.элементы кратные пяти
```

```
# 3.их среднее арифметическое
```

```
# Находим минимальный среди положительных
```

```
numbers = [12, -3, 2, -6, 8, -1, 57, 3, 20]
```

```
positive_numbers = list(filter(lambda x: x > 0, numbers))
```

```
positive_numbers_2 = str(positive_numbers)
```

```
print('Список положительных чисел:', positive_numbers_2.strip('[]'))
```

```
print("Минимальное положительное число:", min(positive_numbers), '\n')
```

```
# Находим элементы кратные пяти
```

```
f_list = str(numbers)
```

```
print('Исходный список:', f_list.strip('[]'))
```

```
print("Числа кратные 5:", *list(filter(lambda x: x % 5 == 0, numbers)), '\n')
```

```
# Находим среднее арифметическое списка
middle_number = reduce(lambda x, y: x + y, numbers)
number = middle_number / len(numbers)
print("Среднее арифметическое число списка =", round(number, 3))
```

Задание № 2.

```
import string
from string import ascii_letters

# Из заданной строки отобразить только символы пунктуации. Использовать
# библиотеку string.
# Строка: --msg-template="$FileDir$\{path}:{line}:{column}:{C}:(\{symbol\}){msg}"

text = '--msg-template="$FileDir$\{path}:{line}:{column}:{C}:(\{symbol\}){msg}"

print(f'Исходная строка: {text}')
print("Знаки пунктуации строки:", *set(list(filter(lambda x: x in string.punctuation, text))))
```

Протокол работы программы:

1-ая программа:

Список положительных чисел: 12, 2, 8, 57, 3, 20

Минимальное положительное число: 2

Исходный список: 12, -3, 2, -6, 8, -1, 57, 3, 20

Числа кратные 5: 20

Среднее арифметическое число списка = 10.222

Process finished with exit code 0

2-ая программа:

Исходная строка: --msg-template="\$FileDir\$\{path}:{line}:{column}:{C}:(\{symbol\}){msg}"

Знаки пунктуации строки: = } (: - " \$ \ { }

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработал(а) навыки составление программ в функциональном стиле в IDE PyCharm Community.

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

Готовые программные коды выложены на [GitHub](#).