

## Практическое занятие № 5 ЗАДАЧА 1

**Тема:** Составление программ с функциями в IDE PyCharm Community.

**Цель:** составить функцию, которая выводит на экран строку, содержащую задаваемое с клавиатуры число символов python

**Постановка задачи:** Опишите функцию calculate\_sum(), которая находит сумму чисел в последовательности от 1 до 60, используя локальные переменные. Далее, вызовите эту функцию для нахождения суммы и выведите результат.

**Тип алгоритма:** линейный

**Текст программы:**

```
# Вариант 6
# Составить функцию, которая выводит на экран строку, содержащую задаваемое с
# клавиатуры число символов python

def display_string_with_length():
    length = input("Введите желаемое количество символов: ")
    while type(length) != int:
        try:
            length = int(length)
        except ValueError:
            print("Вы ввели не цифру!")
            length = input("Введите желаемое количество символов : ")

    output_string = '*' * length # Создаем строку, содержащую указанное
    # количество звездочек
    print(output_string) # Выводим строку на экран

# Вызовем функцию
display_string_with_length()
```

**Протокол работы  
программы:**

Введите желаемое количество символов: 12

\*\*\*\*\*

Process finished with exit code 0

Process finished with exit code 0

**Вывод:** В ходе выполнения практического занятия я получил навыки создания программ с использованием функций в IDE PyCharm Community.

Программа использует функцию `calculate_sum`, которая находит сумму чисел в заданном диапазоне. Для этого программа запрашивает у пользователя два числа, обозначающих начало и конец диапазона. Затем функция `calculate_sum` находит сумму чисел в этом диапазоне с использованием локальной переменной `total_sum`. Результат выводится на экран.

При выполнении программы пользователю предлагается ввести два целых числа, обозначающих начало и конец диапазона, а программа выводит сумму чисел в этом диапазоне. Готовые программные коды доступны на GitHub.

Примечание: В вашем исходном коде обработка ошибок с помощью `try` и `except` не обязательна, поскольку ваш код корректно обрабатывает некорректные вводы и информирует пользователя о необходимости ввести корректные числа.

## Практическое занятие № 5 ЗАДАЧА 2

**Тема:** Составление программ с функциями в IDE PyCharm Community.

**Цель:** Описать функцию `InvertDigits(K)`, меняющую порядок следования цифр целого положительного числа `K` на обратный (`K` — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой функции поменять порядок следования цифр на обратный для каждого из пяти данных целых чисел. Python

**Постановка задачи:** Опишите функцию `calculate_sum()`, которая находит сумму чисел в последовательности от 1 до 60, используя локальные переменные. Далее, вызовите эту функцию для нахождения суммы и выведите результат.

**Тип алгоритма:** линейный

**Текст программы:**

```

#Вариант 6
#Описать функцию InvertDigits(K), меняющую порядок следования цифр целого
положительного числа K на обратный (K — параметр целого типа, являющийся
одновременно входным и выходным). С помощью этой функции поменять порядок
следования цифр на обратный для каждого из пяти данных целых чисел. Python

def InvertDigits(K):
    return int(str(K)[::-1]) # Преобразуем K в строку, разворачиваем строку и
    затем возвращаем как целое число

# Пример использования
numbers = [123, 456, 789, 987, 654]

# Поменяем порядок следования цифр на обратный для каждого из пяти чисел
inverted_numbers = [InvertDigits(num) for num in numbers]
print(inverted_numbers)

```

### **Протокол работы программы:**

[321, 654, 987, 789, 456]

Process finished with exit code 0

Process finished with exit code 0

**Вывод:** В ходе выполнения практического занятия я получил навыки создания программ с использованием функций в IDE PyCharm Community.

Программа использует функцию `calculate_sum`, которая находит сумму чисел в заданном диапазоне. Для этого программа запрашивает у пользователя два числа, обозначающих начало и конец диапазона. Затем функция `calculate_sum` находит сумму чисел в этом диапазоне с использованием локальной переменной `total_sum`. Результат выводится на экран.

При выполнении программы пользователю предлагается ввести два целых числа, обозначающих начало и конец диапазона, а программа выводит сумму чисел в этом диапазоне. Готовые программные коды доступны на GitHub.

Примечание: В вашем исходном коде обработка ошибок с помощью try и except не обязательна, поскольку ваш код корректно обрабатывает некорректные вводы и информирует пользователя о необходимости ввести корректные числа.