

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Кафедра инфокоммуникаций

**Основы кроссплатформенного программирования
Отчет по лабораторной работе №2.12**

Декораторы функций в языке Python

Выполнила студентка группы
ИТС-б-о-20-1 (2)

Швецова К.С. « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил к.т.н., доцент

Кафедры инфокоммуникаций

Воронкин Р.А.

(подпись)

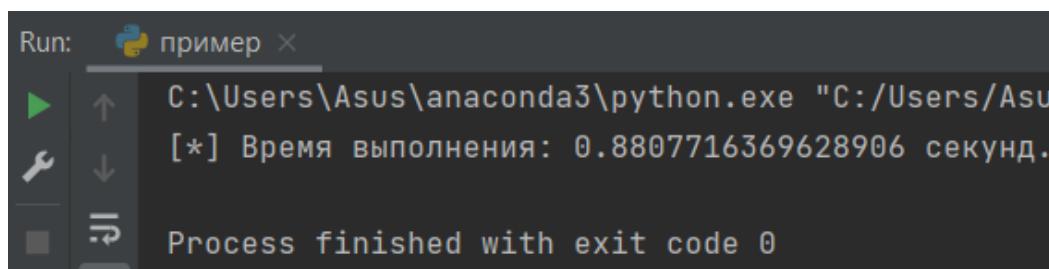
Ставрополь 2021

Цель работы: приобретение навыков по работе с декораторами функций при написании программ с помощью языка программирования Python версии 3.x.

Ссылка на репозиторий - <https://github.com/ShveczovaKS/6lab2k>

Ход работы:

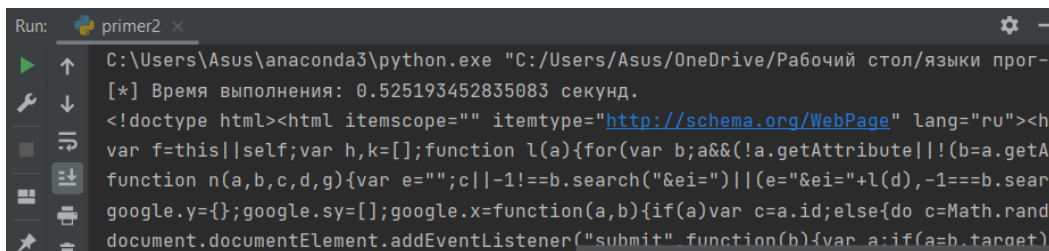
Пример 1. Создаём декоратор, измеряющий время выполнения функции.



```
Run: пример x
C:\Users\Asus\anaconda3\python.exe "C:/Users/Asu
[*] Время выполнения: 0.8807716369628906 секунд.
Process finished with exit code 0
```

Рисунок 1. Результат выполнения программы

Пример 2. Модифицируем декоратор для измерения времени выполнения.

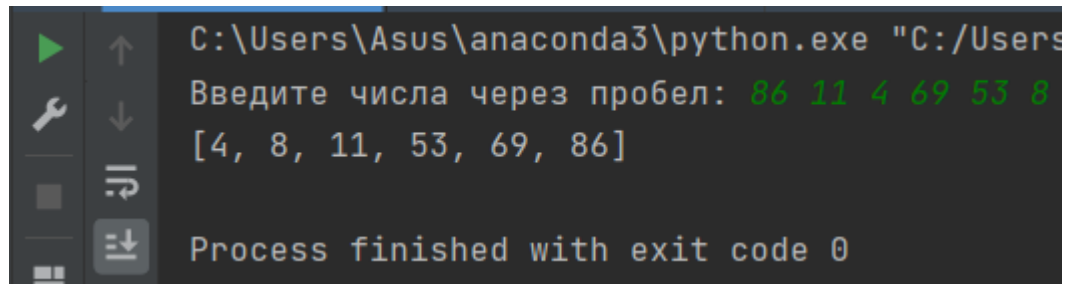


```
Run: primer2 x
C:\Users\Asus\anaconda3\python.exe "C:/Users/Asus/OneDrive/Рабочий стол/языки прог-и
[*] Время выполнения: 0.525193452835083 секунд.
<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang="ru"><he
var f=this||self;var h,k=[];function l(a){for(var b;a&&(!a.getAttribute)||!(b=a.getA
function n(a,b,c,d,g){var e="";c||-1!==b.search("&ei=")||e="&ei="+l(d),-1===b.searc
google.y={};google.sy=[];google.x=function(a,b){if(a)var c=a.id;else{do c=Math.randc
document.documentElement.addEventListener("submit",function(h){var a;if(a=b.target)-
```

Рисунок 2. Результат выполнения программы

Индивидуальное задание. 2 вариант. На вход программы поступает строка из целых чисел, записанных через пробел. Напишите функцию `get_list`, которая преобразовывает эту строку в список из целых чисел и возвращает его. Определите декоратор для этой функции, который сортирует список чисел, полученный из вызываемой в нем функции. Результат сортировки должен возвращаться при вызове декоратора. Вызовите

декорированную функцию `get_list` и отобразите полученный отсортированный список на экране.



```
C:\Users\Asus\anaconda3\python.exe "C:/Users
Введите числа через пробел: 86 11 4 69 53 8
[4, 8, 11, 53, 69, 86]
Process finished with exit code 0
```

Рисунок 3. Результат выполнения программы

Контрольные вопросы:

1. Что такое декоратор?

Декоратор — это функция, которая позволяет обернуть другую функцию для расширения её функциональности без непосредственного изменения её кода.

2. Почему функции являются объектами первого класса?

Объектами первого класса в контексте конкретного языка программирования называются элементы, с которыми можно делать всё то же, что и с любым другим объектом: передавать как параметр, возвращать из функции и присваивать переменной.

В Python всё является объектом, а не только объекты, которые вы создаёте из классов. Это значит, что в Python всё это — объекты:

- числа;
- строки;
- классы;
- функции.

3. Каково назначение функций высших порядков?

Функции высших порядков — это такие функции, которые могут принимать в качестве аргументов и возвращать другие функции.

4. Как работают декораторы?

Декоратор — это функция, которая позволяет обернуть другую функцию с помощью символа «@»

5. Какова структура декоратора функций?

Сначала записывается функция — декоратор. Потом идет его вызов с помощью @, а затем основная функция, которую оборачивает декоратор.

6. Самостоятельно изучить, как можно передать параметры декоратору, а не декорируемой функции?

Используя замыкание функций.

Вывод: в ходе лабораторной работы были приобретены навыки по работе с функциями с переменным числом параметров при написании программ с помощью языка программирования Python версии 3.x.