

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

Отчет по лабораторной работе № 2.12

Декораторы функций в языке Python

Выполнил студент группы ИВТ-б-о-20-1

Ищенко М.А.

Работа защищена « » _____ 20__ г.

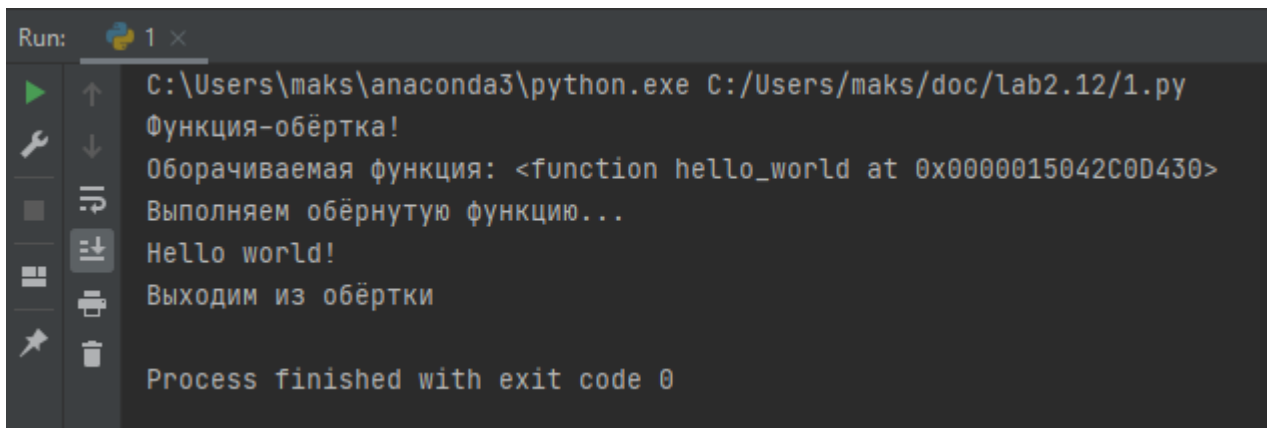
Проверил(а) _____

Ставрополь 2021

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

Создан общедоступный репозиторий на GitHub. Дополнен файл .gitignore необходимыми правилами для работы с IDE PyCharm.

Проработан пример из лабораторной работы, рис. 1

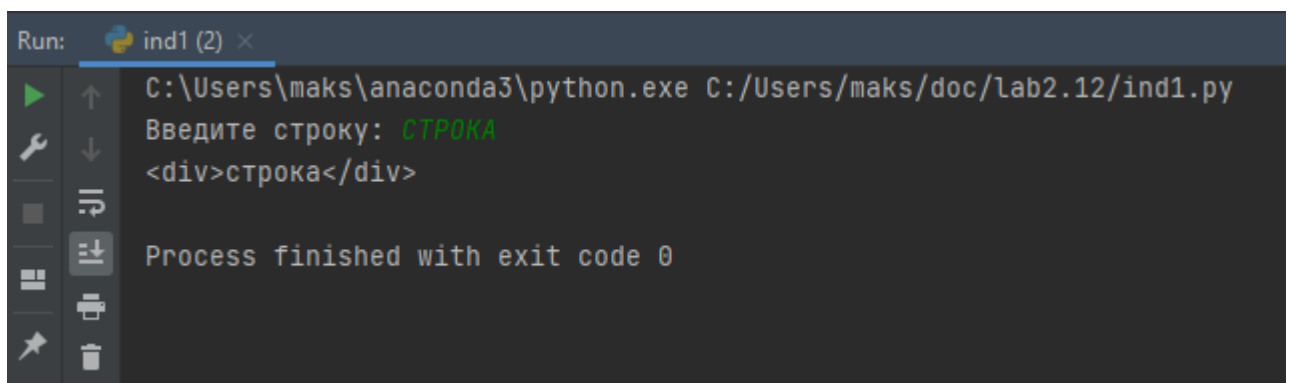


```
Run: 1 x
C:\Users\maks\anaconda3\python.exe C:/Users/maks/doc/lab2.12/1.py
Функция-обёртка!
Оборачиваемая функция: <function hello_world at 0x0000015042C0D430>
Выполняем обёрнутую функцию...
Hello world!
Выходим из обёртки

Process finished with exit code 0
```

Рисунок 1 – Пример

Выполнено индивидуальное задание варианта 6, рис. 2

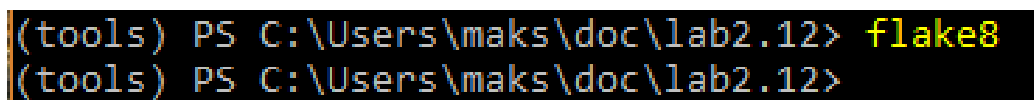


```
Run: ind1 (2) x
C:\Users\maks\anaconda3\python.exe C:/Users/maks/doc/lab2.12/ind1.py
Введите строку: СТРОКА
<div>строка</div>

Process finished with exit code 0
```

Рисунок 2 – Индивидуальное задание

Программы проверены на flake8, рис. 3



```
(tools) PS C:\Users\maks\doc\lab2.12> flake8
(tool) PS C:\Users\maks\doc\lab2.12>
```

Рисунок 3 – Проверка заданий

Контрольные вопросы:

1. Что такое декоратор?

Декораторы – один из самых полезных инструментов в Python, – это функция, которая позволяет обернуть другую функцию для расширения её функциональности без непосредственного изменения её кода.

2. Почему функции являются объектами первого класса?

В Python всё является объектом, этот факт открывает перед нами множество возможностей. Мы можем сохранять функции в переменные, передавать их в качестве аргументов и возвращать из других функций. Можно даже определить одну функцию внутри другой. Иными словами, функции – это объекты 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.

3. Каково назначение функций высших порядков?

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

Функции высших порядков, вроде дифференциального оператора, принимают на входе функцию и возвращают другую функцию, производную от исходной. Функции высших порядков в программировании работают точно так же – они либо принимают функцию(и) на входе и/или возвращают функцию(и).

4. Как работают декораторы?

Внутри функции-декоратора определяется другая функция, обёртка, так сказать, которая обёртывает функцию-аргумент и затем изменяет её поведение. Декоратор возвращает эту обёртку.

Просто добавив `@decorator` перед определением функции, её поведение меняется. Однако выражение с `@` является всего лишь синтаксическим сахаром. Иными словами, выражение `@decorator` вызывает функцию-декоратор с функцией в качестве аргумента.

5. Какова структура декоратора функций?

В начале объявляется функция в которой будет использоваться декорируемая функция. Далее объявляется декорируемая функция перед которой используется конструкция `@decorator`, где “decorator” имя функции, использующей декорируемую функцию.

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

Для этого после объявления декоратора в скобках нужно указать необходимые параметры: `@decorator(p1)`

Вывод: в ходе занятия были приобретены навыки по работе с декораторами функций при написании программ с помощью языка программирования Python версии 3.x