

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ
УНИВЕРСИТЕТ

Институт Цифрового Развития
Кафедра инфокоммуникаций

ОТЧЁТ

по лабораторной работе №6

Дисциплина: «Языки программирования»

Выполнил: студент 2 курса
группы ИТС-б-о-20-1

Харченко Екатерина Владимировна

Проверил:

К.т.н., доцент кафедры инфокоммуникаций

Воронкин Роман Александрович

Работа защищена с оценкой: _____

Ставрополь, 2021 г.

ДЕКОРАТОРЫ ФУНКЦИЙ В ЯЗЫКЕ PYTHON

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

2. Создадим общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python.
3. Выполним клонирование созданного репозитория.

```
C:\Users\Катя>git clone https://github.com/ekaterina533/2.6.git
Cloning into '2.6'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
```

4. Дополним файл .gitignore необходимыми правилами для работы с IDE PyCharm.
5. Организуем свой репозиторий в соответствии с моделью ветвления git-flow.
6. Создадим проект PyCharm в папке репозитория.
7. Проработаем примеры лабораторной работы.

```
Функция-обёртка!
Оборачиваемая функция: <function hello_world at 0x0000017F129268B0>
Выполняем обёрнутую функцию...
Hello world!
Выходим из обёртки
```

Рис. 1.1 – работа программы.

8. Выполним индивидуальное задание.

Вариант 8. Объявите функцию, которая вычисляет площадь круга и возвращает вычисленное значение. В качестве аргумента ей передается значение радиуса. Определите декоратор для этой функции, который выводит на экран сообщение: «Площадь круга равна = ». В строке выведите числовое значение с точностью до сотых. Примените декоратор к функции и вызовите декорированную функцию.

```
Введите значение радиуса:10  
Площадь круга равна  
= 314.0
```

```
Площадь круга равна  
= 19.62
```

Рис. 1.2 – работа программы.

9. Зафиксируем изменения в репозитории.

10. Добавим отчет по лабораторной работе в формате PDF в папку doc репозитория. Зафиксируйте изменения.

12. Отправьте сделанные изменения на сервер GitHub.

```
C:\Users\Катя\2.6>git push  
Enumerating objects: 5, done.  
Counting objects: 100% (5/5), done.  
Delta compression using up to 8 threads  
Compressing objects: 100% (4/4), done.  
Writing objects: 100% (4/4), 1007 bytes | 503.00 KiB/s, done.  
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0  
To https://github.com/ekaterina533/2.6.git  
f71d908..b552587 main -> main
```

Контрольные вопросы:

1. Что такое декоратор?

Декоратор — это структурный паттерн проектирования, который позволяет динамически добавлять объектам новую функциональность, оборачивая их в полезные «обёртки».

2. Почему функции являются объектами первого класса?

В Python функции являются объектами первого класса. Это означает, что функции можно передавать и использовать в качестве аргументов, как и любой другой объект

3. Каково назначение функций высших порядков?

Функция высшего порядка в Python – это функция, которая имеет другую функцию в качестве аргумента или функцию, которая возвращает

другую в качестве возврата на выходе. Функции высшего порядка работают с другими функциями, указанными в программе.

4. Как работают декораторы?

Декоратор является функцией высшего порядка, так как принимает функцию в качестве аргумента, а также возвращает функцию.

Внутри `decorator_function()` мы определили другую функцию, обёртку, так сказать, которая обёртывает функцию-аргумент и затем изменяет её поведение. Декоратор возвращает эту обёртку.

5. Какова структура декоратора функций?

Синтаксис функций очень простой. Она начинается с ключевого слова `def`. Далее идет уникальное имя, параметры и двоеточие. Также может быть документация (`docstring`), которая описывает назначение функции.

Все инструкции, которые идут дальше, составляют тело функции. Они должны иметь корректные отступы. В конце также может быть инструкция `return`.

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