

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ПРАКТИЧЕСКОЙ РАБОТЕ №2.11
дисциплины «Программирование на Python»
Вариант №2

Выполнила:
Беседина Инга Олеговна
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника», очная
форма обучения

(подпись)

Руководитель практики:
Воронкин Р. А., канд. технических
наук, доцент, доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема: Замыкания в языке Python

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

Ход работы

Индивидуальное задание:

2. Используя замыкания функций, объявите внутреннюю функцию, которая заключает строку `s` (`s` – строка, параметр внутренней функции) в произвольный тег, содержащийся в переменной `tag` – параметре внешней функции. Далее, на вход программы поступает две строки: первая с тегом, вторая с некоторым содержимым. Вторую строку нужно поместить в тег из первой строки с помощью реализованного замыкания. Результат выведите на экран.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

def tag_wrapper(tag):
    def inner_function(s):
        return f"{tag}{s}{tag}"
    return inner_function

if __name__ == '__main__':
    tag = input('тег - ')
    s = input('строка - ')

    tag_wrapp = tag_wrapper(tag)
    print(tag_wrapp(s))
```

```
тег - /
строка - text
|text|
```

Рисунок 1. Результат работы программы

Контрольные вопросы:

1. Замыкание - это функция, которая запоминает окружение, в котором она была создана, и имеет доступ к переменным этого окружения даже после того, как окружающая функция завершила свою работу.

2. В Python замыкания реализуются путем определения функции внутри другой функции. Внутренняя функция имеет доступ к переменным внешней функции, даже после того, как внешняя функция завершила свою работу.

3. Область видимости Local относится к переменным, определенным внутри текущей функции. Эти переменные видны только внутри этой функции.

4. Область видимости Enclosing относится к переменным, определенным в объемлющей (внешней) функции. Внутренняя функция имеет доступ к этим переменным.

5. Область видимости Global относится к переменным, определенным на уровне модуля или в глобальной области видимости. Эти переменные видны во всем модуле.

6. Область видимости Build-in относится к встроенным функциям и переменным Python, которые всегда доступны в любой области видимости.

7. Для использования замыканий в Python нужно определить внутреннюю функцию внутри внешней функции и вернуть эту внутреннюю функцию как результат выполнения внешней функции.

8. Замыкания могут быть использованы для построения иерархических данных путем создания функций, которые возвращают другие функции или объекты данных в зависимости от контекста или параметров. Например, замыкания могут быть использованы для создания деревьев или структур данных с вложенными элементами.

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

