

Федеральное государственное автономное образовательное учреждение
высшего образования «Северо-Кавказский федеральный университет»
Институт цифрового развития

ОТЧЕТ ПО ДИСЦИПЛИНЕ
ОСНОВЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ
Лабораторная работа №14

Выполнила:

Мирзаева Камилла Мирзаевна

2 курс, группа ПИЖ-б-о-20-1

Принял:

Воронкин Роман Александрович

Ставрополь, 2021 г.

Ход работы

```
def add_two(a):  
    x = 2  
    return a + x
```

Рисунок 1 – Пример работы с областью видимости local

```
def add_for(a):  
    x = 2  
  
    def add_some():  
        print("x = " + str(x))  
        return a + x  
  
    return add_some()
```

Рисунок 2 – Пример работы с областью видимости enclosing

```
def mul(a, b):  
    return a * b
```

Рисунок 3 – Пример работы с функцией

```
def mul5(a):  
    return mul5(5, a)
```

Рисунок 4 – Пример работы с функцией mul5

```
def mul(a):  
    def helper(b):  
        return a * b  
    return helper
```

Рисунок 5 – Пример замыкания

```
def fun1(a):
    x = a * 3

    def fun2(b):
        nonlocal x
        return b + x

    return fun2
```

Рисунок 6 – Пример функции с использованием локальных и глобальных переменных

```
>>> tpl = lambda a, b: (a, b)
>>> a = tpl(1, 2)
>>> a
(1, 2)
>>> b = tpl(3, a)
>>> c = tpl(a, b)
```

Рисунок 7 – Пример работы с замыканием, как средством для построения иерархических данных

Индивидуальное задание.

Вариант 15.

Используя замыкания функций, объявите внутреннюю функцию, которая принимает в качестве параметров фамилию и имя, а затем, заносит в шаблон эти данные. Сам шаблон – это строка, которая передается внешней функции и, например, может иметь такой вид: «Уважаемый %F%, %N%! Вы делаете работу по замыканиям функций.» Здесь %F% - это фрагмент куда нужно подставить фамилию, а %N% - фрагмент, куда нужно подставить имя. (Шаблон может быть и другим, вы это определяете сами). Здесь важно, чтобы внутренняя функция умела подставлять данные в шаблон, формировать новую строку и возвращать результат. Вызовите внутреннюю функцию замыкания и отобразите на экране результат ее работы.

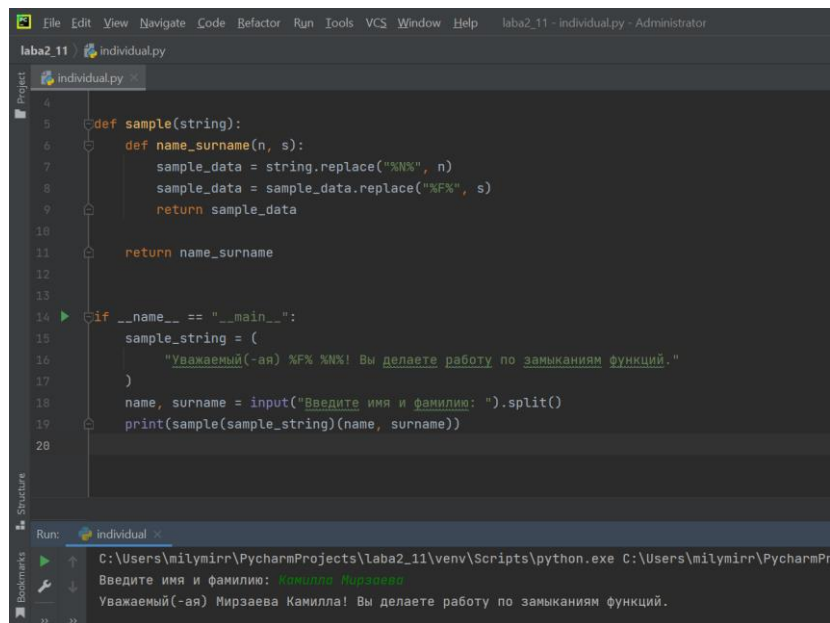


Рисунок 8 – Индивидуальное задание

Вопросы для защиты

1. Что такое замыкание?

Замыкание – это функция, в теле которой присутствуют ссылки на переменные, объявленные вне тела этой функции в окружающем коде и не являющиеся её параметрами.

2. Как реализованы замыкания в языке программирования Python?

```
>>> def mul(a):
    def helper(b):
        return a * b
    return helper
```

3. Что подразумевает под собой область видимости Local?

Эту область видимости имеют переменные, которые создаются и используются внутри функций.

4. Что подразумевает под собой область видимости Enclosing?

Суть данной области видимости в том, что внутри функции могут быть вложенные функции и локальные переменные, так вот локальная переменная функции для её вложенной функции находится в enclosing области видимости.

5. Что подразумевает под собой область видимости Global?

Переменные области видимости global – это глобальные переменные уровня модуля (модуль – это файл с расширением .py)

6. Что подразумевает под собой область видимости Build-in?

В рамках этой области видимости находятся функции open, len и т.п., также туда входят исключения. Эти сущности доступны в любом модуле Python и не

требуют предварительного импорта. Built-in – это максимально широкая область видимости.

7. Как использовать замыкания в языке программирования Python?

```
>>> def mul(a):  
    def helper(b):  
        return a * b  
    return helper
```

8. Как замыкания могут быть использованы для построения иерархических данных?

Это свойство позволяет строить иерархические структуры данных.

```
>>> union = lambda a, b: (a, b)
```

```
>>> x = union(7, 9)  
>>> x  
# (7, 9)
```

```
>>> y = union(5, a)  
>>> y  
# (5, (7, 9))
```

```
>>> union(x, y)  
# ((7, 9), (5, (7, 9)))
```