

**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 №2.11**

**Замыкания 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)

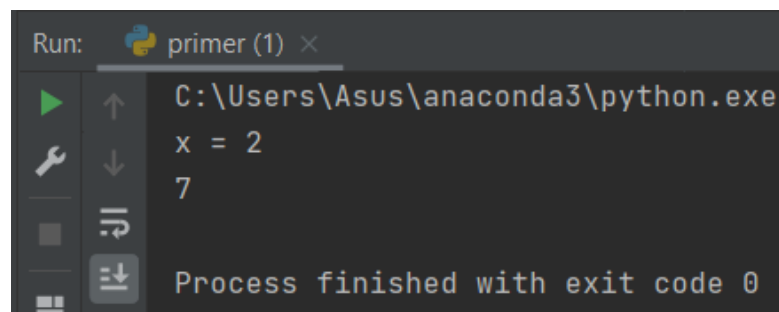
Ставрополь 2021

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

**Ссылка на репозиторий** - <https://github.com/ShveczovaKS/7lab2k>

**Ход работы:**

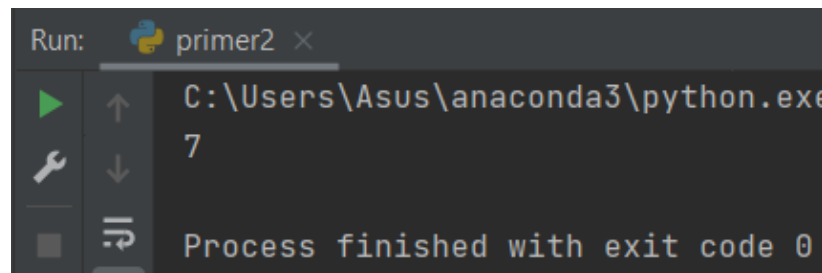
**Пример 1.** В данном случае переменная `x` имеет область видимости `enclosing` для функции `add_some()`.



```
Run: primer (1) x
C:\Users\Asus\anaconda3\python.exe
x = 2
7
Process finished with exit code 0
```

Рисунок 1. Результат выполнения программы

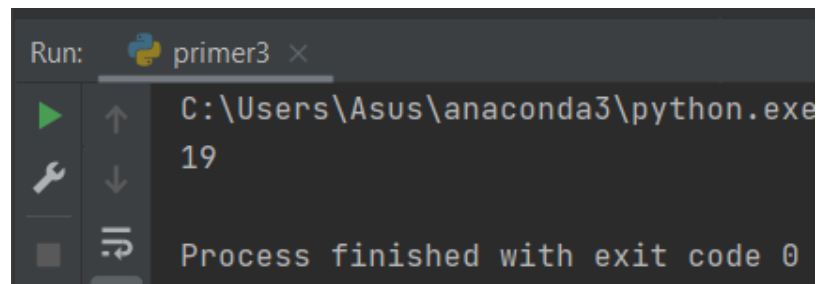
**Пример 2.** Переменная `x` – это `global` переменная. Доступ к ней можно получить из любой функции объявленной в данном модуле.



```
Run: primer2 x
C:\Users\Asus\anaconda3\python.exe
7
Process finished with exit code 0
```

Рисунок 2. Результат выполнения программы

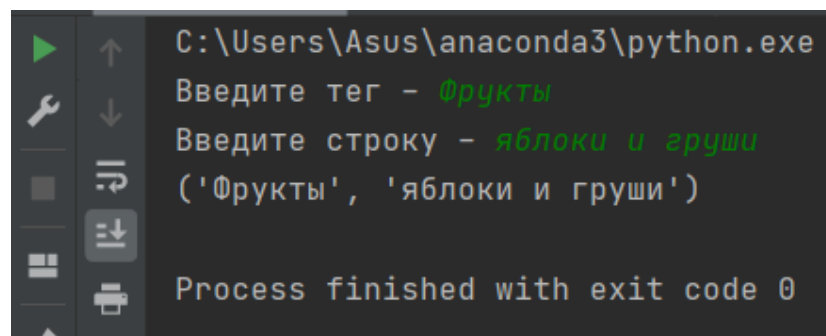
**Пример 3.** В функции `fun1()` объявлена локальная переменная `x`, значение которой определяется аргументом `a`. В функции `fun2()` используются эта же переменная `x`, `nonlocal` указывает на то, что эта переменная не является локальной, следовательно, ее значение будет взято из ближайшей области видимости, в которой существует переменная с таким же именем.



```
Run: primer3 x
C:\Users\Asus\anaconda3\python.exe
19
Process finished with exit code 0
```

Рисунок 3. Результат выполнения программы

**Индивидуальное задание. 2 вариант.** 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.



```
C:\Users\Asus\anaconda3\python.exe
Введите тег - Фрукты
Введите строку - яблоки и груши
('Фрукты', 'яблоки и груши')
Process finished with exit code 0
```

Рисунок 4. Результат выполнения программы

### Контрольные вопросы:

1. Что такое замыкание?

Замыкание (closure) в программировании — это функция, в теле которой присутствуют ссылки на переменные, объявленные вне тела этой функции в окружающем коде и не являющиеся ее параметрами.

2. Как реализованы замыкания в языке программирования Python?

В Python выделяют четыре области видимости для переменных: *local*, *enclosing*, *global*, *build-in*.

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 ее 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?

Уровень Python интерпретатора. 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 замыкания могут быть использованы для построения иерархических данных?

В общем случае, операция комбинирования объектов данных обладает свойством замыкания в том случае, если результаты соединения объектов с помощью этой операции сами могут соединяться этой же операцией. Это свойство позволяет строить иерархические структуры данных.

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