

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 ЛАБОРАТОРНОЙ РАБОТЕ №2.11
дисциплины «Основы кроссплатформенного программирования»
Вариант ____

Выполнил:
Дмитриченко Марат Заурович
1 курс, группа ИТС-б-0-22-1,
11.03.02 «Инфокоммуникационные
технологии и системы связи»,
направленность (профиль)
«Инфокоммуникационные системы и
сети», очная форма обучения

(подпись)

Руководитель практики:
Воронкин Р. А., 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

Условие примера: функция `mul()` умножает два числа и возвращает полученный результат. Если мы ходим на базе нее решить задачу: “умножить число на пять”, то в самом простом случае, можно вызывать `mul()`, передавая в качестве первого аргумента пятерку.

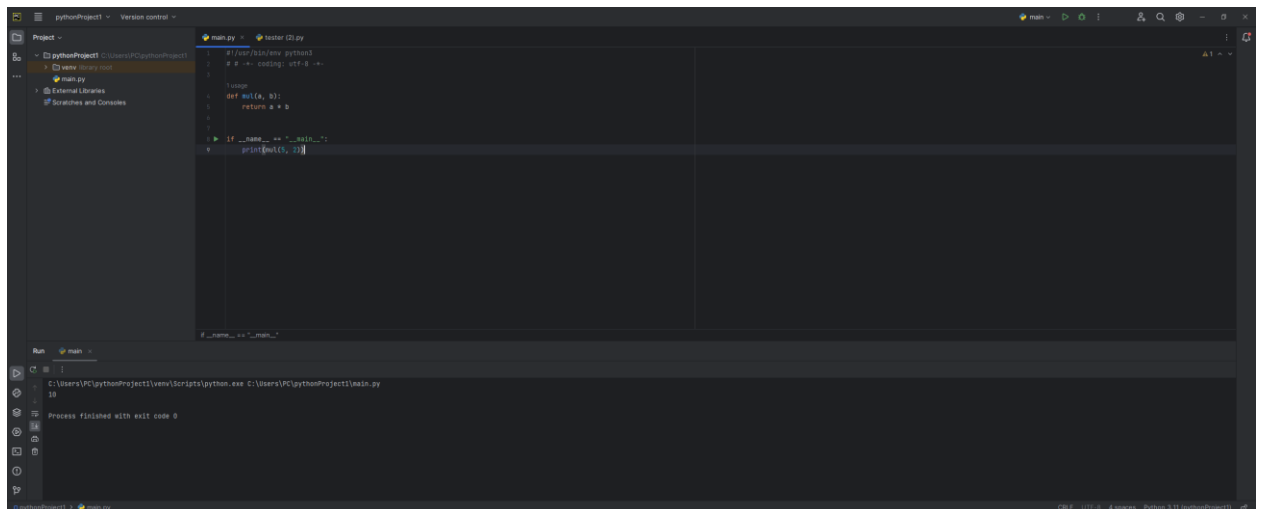


Рисунок 1. Реализация примера 1

Условие примера: на самом деле мы можем создать новую функцию, которая будет вызывать `mul()`, с пятеркой и ещё одним числом, которое она будет получать в качестве своего единственного аргумента.

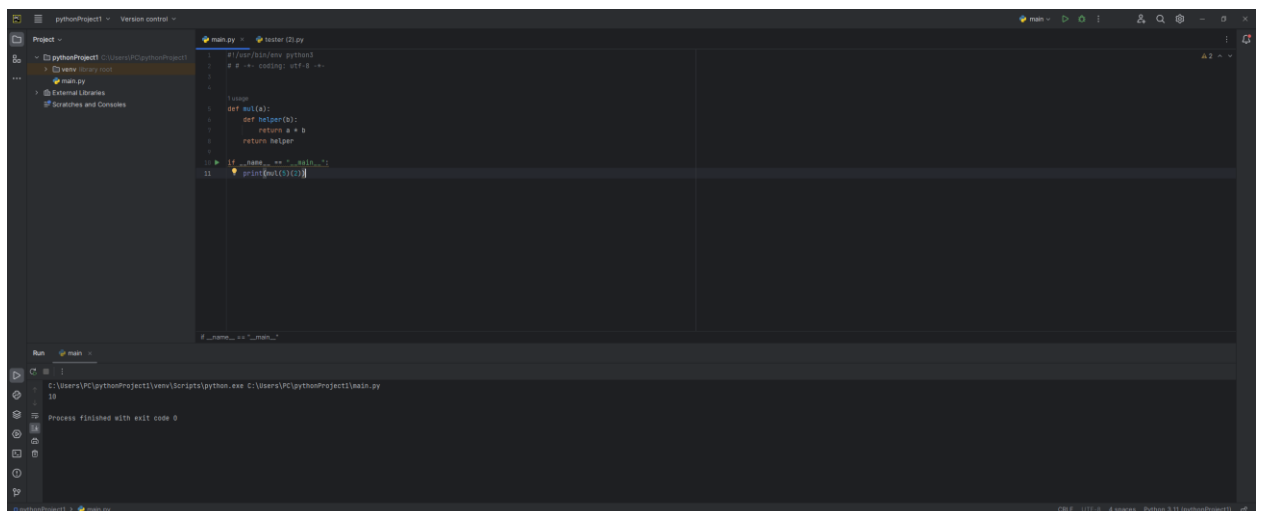


Рисунок 2. Реализация примера 2

Индивидуальное задание

5. 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.

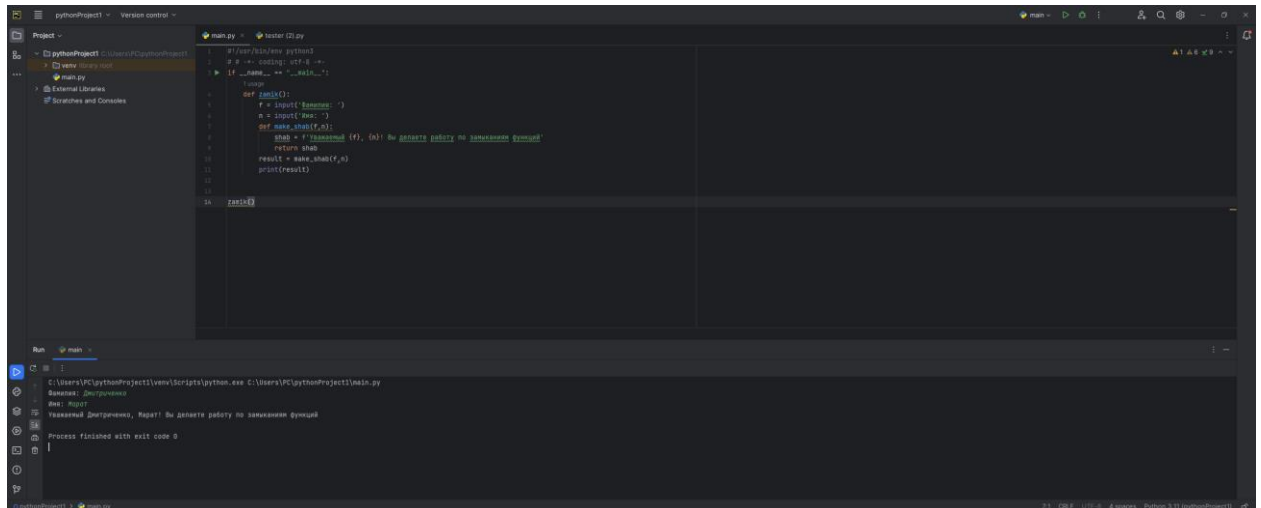


Рисунок 3. Программа индивидуального задания

Ответы на контрольные вопросы

1. Что такое замыкание?

Замыкание - это функция, которая имеет доступ к переменным из внешней функции, в которой она была определена, даже после того, как внешняя функция завершила свое выполнение. Это позволяет создавать более гибкие и мощные функции в Python.

2. Как реализованы замыкания в языке программирования Python?

В языке Python замыкания реализуются путем определения внутренней функции внутри внешней функции, и возвращения этой внутренней функции в качестве результата выполнения внешней функции. Это позволяет сохранять контекст вызова, чтобы его можно было использовать при последующих вызовах внутренней функции.

3. Что подразумевает под собой область видимости Local?

Область видимости Local - это область видимости, ограниченная внутри текущей функции, где переменные могут быть определены и использованы только внутри этой функции.

4. Что подразумевает под собой область видимости Enclosing?

Область видимости Enclosing - это область видимости, в которой определена текущая функция, а также область видимости любых внешних функций, в которых эта функция была определена, но не в глобальной области видимости.

5. Что подразумевает под собой область видимости Global?

Область видимости Global предполагает, что переменная объявлена за пределами функции или класса и доступна везде.

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

Область видимости Build-in - это область видимости, содержащая функции и объекты, встроенные в Python, которые могут быть использованы в любом месте в программе без импорта. В ней находятся все стандартные имена модулей и функций, например: `print ()`, `len ()`, `str ()`, `list ()`.

7. Как использовать замыкания в языке программирования Python?

Для использования замыканий в Python нужно определить внутреннюю функцию внутри внешней функции и вернуть эту внутреннюю функцию в качестве результата выполнения внешней функции. Затем можно вызвать возвращенную функцию, и она будет иметь доступ к переменным из внешней функции, даже после того, как внешняя функция завершила свое выполнение.

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

Замыкания могут быть использованы для построения иерархических данных в языке программирования Python, например, для создания древовидных структур данных, таких как деревья решений. Один из способов использования замыканий для построения иерархических данных - это создание функций-генераторов.

Вывод: приобрел навыки по работе с замыканиями при написании программ с помощью языка программирования Python версии 3.x.