

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 №14
дисциплины «Программирование на Python»

Выполнил:
Говоров Егор Юрьевич
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», очная
форма обучения

(подпись)

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

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема: Замыкания в языке Python

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

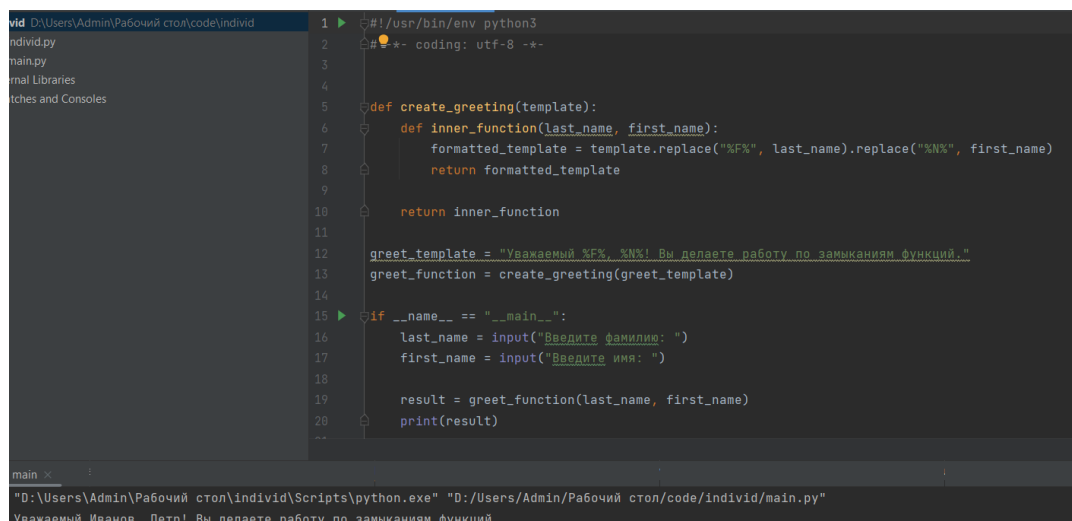
Ход работы

1. Создал общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python. Выполнил клонирование созданного репозитория.

2. Дополнил файл .gitignore необходимыми правилами.

3. Организовал созданный репозиторий в соответствие с моделью ветвления git-flow.

Выполнил индивидуальное задание, согласно варианту 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.



```
1 #!/usr/bin/env python3
2 -*- coding: utf-8 -*-
3
4
5 def create_greeting(template):
6     def inner_function(last_name, first_name):
7         formatted_template = template.replace("%F%", last_name).replace("%N%", first_name)
8         return formatted_template
9
10    return inner_function
11
12 greet_template = "Уважаемый %F%, %N%! Вы делаете работу по замыканиям функций."
13 greet_function = create_greeting(greet_template)
14
15 if __name__ == "__main__":
16     last_name = input("Введите фамилию: ")
17     first_name = input("Введите имя: ")
18
19     result = greet_function(last_name, first_name)
20     print(result)
```

main x

"D:\Users\Admin\Рабочий стол\indiv\Scripts\python.exe" "D:/Users/Admin/Рабочий стол/code/indiv/main.py"

Уважаемый Иванов, Петр! Вы делаете работу по замыканиям функций.

Рисунок 1. Результат работы программы из индивидуального задания

Контрольные вопросы

1. Что такое замыкание?

Функция, которая запоминает окружение, в котором она была создана, и может получать доступ к переменным из этого окружения, даже после того, как оно было вызвано и покинуло свою область видимости.

2. Как реализованы замыкания в языке программирования Python?

В языке программирования Python замыкания реализованы путем вложения функций, то есть функция внутри другой функции. Внутренняя функция имеет доступ к переменным внешней функции, что позволяет создавать и использовать замыкания.

3. Что подразумевает под собой область видимости Local?

Означает, что переменная определена внутри функции и доступна только внутри этой функции. Это означает, что переменная имеет локальную область видимости

4. Что подразумевает под собой область видимости Enclosing?

Означает, что переменная определена во внешней функции и доступна внутри вложенной функции. Это означает, что переменная имеет область видимости, охватывающую внешнюю функцию и все вложенные функции.

5. Что подразумевает под собой область видимости Global?

Означает, что переменная определена в основной программе или модуле и доступна в любом месте программы. Это означает, что переменная имеет глобальную область видимости.

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

Подразумевает, что переменная является встроенной в язык Python и доступна в любом месте программы. Это означает, что переменная имеет встроенную область видимости.

7. Как использовать замыкания в языке программирования Python?

Для использования замыканий в Python необходимо определить внутреннюю функцию внутри внешней функции и вернуть эту внутреннюю функцию в качестве результата внешней функции. Это позволит внутренней функции сохранить доступ к переменным внешней функции даже

после завершения работы внешней функции

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

Замыкания могут быть использованы для построения иерархических данных путем создания функций, которые возвращают другие функции или объекты данных. Это позволяет создавать иерархию функций или объектов, которые могут иметь доступ к переменным и состоянию своих родительских функций или объектов. Такой подход позволяет эффективно управлять иерархическими структурами данных в программировании

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