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

Институт перспективной инженерии
Департамент цифровых, робототехнических систем и электроники

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №8
дисциплины
«Основы кроссплатформенного программирования»

Выполнил:
Пугачев Кирилл Дмитриевич
2 курс, группа ИТС-б-о-23-1,
11.03.02«Инфокоммуникационные технологии и системы связи», очная форма обучения

_____
(подпись)

Проверил:
Доцент департамента цифровых, робототехнических систем и электроники Воронкин Р.А.

_____
(подпись)

Отчет защищен с оценкой _____ Дата защиты_____

Ставрополь, 2024 г.

ТЕМА: РАБОТА С КОРТЕЖАМИ В ЯЗЫКЕ PYTHON

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

Порядок выполнения работы:

1. Изучил теоретический материал.

2. Приступил к выполнению заданий.

3. Создание репозитория

Ссылка на GitHub https://github.com/chillkirill/LABA8.git

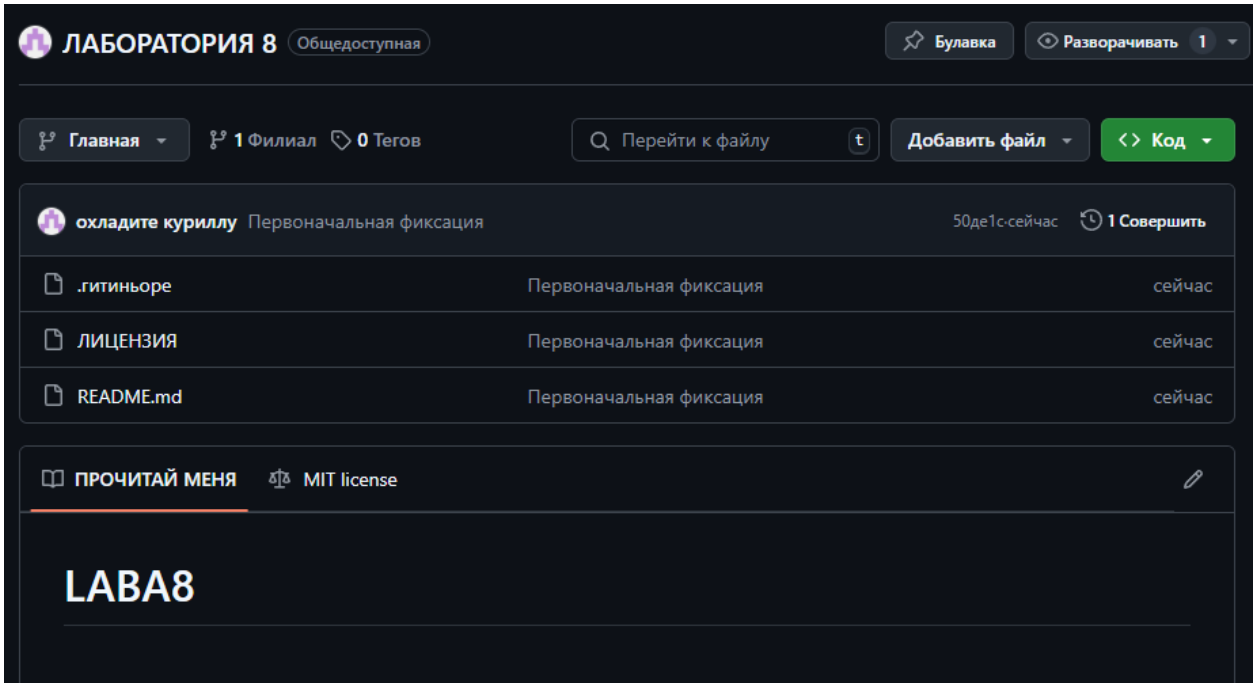

Рисунок 1. Репозиторий

4. Проработал примеры

```
import random

def sum_elements_less_than_5(tuple_A):
    """
    Вычисляет сумму элементов кортежа, модуль которых меньше 5.

    Args:
      tuple_A: Кортеж чисел.

    Returns:
      Сумма элементов, модуль которых меньше 5. Возвращает 0, если кортеж пуст.
    """
    if not tuple_A: # проверка на пустой кортеж
        return 0
    sum_elements = sum(x for x in tuple_A if abs(x) < 5)
    return sum_elements

#Пример использования:
# Создаем кортеж из 10 случайных чисел
tuple_A = tuple(random.randint(-10, 10) for _ in range(10))
print("Кортеж:", tuple_A)

# Вычисляем сумму элементов
sum_less_than_5 = sum_elements_less_than_5(tuple_A)
print("Сумма элементов, меньших 5 по модулю:", sum_less_than_5)

#Пример с пустым кортежем
tuple_B = ()
sum_less_than_5_B = sum_elements_less_than_5(tuple_B)
print("Сумма элементов для пустого кортежа:", sum_less_than_5_B)
```

Рисунок 2. Код к 1 примеру

```
Кортеж: (-8, -1, -6, 6, 3, 4, -5, 0, 7, -3)
Сумма элементов, меньших 5 по модулю: 3
Сумма элементов для пустого кортежа: 0
```

Рисунок 3. Пример

```
# Ввод данных: осадки и температура для каждого дня месяца
precipitation = list(map(float, input("Введите количество осадков (мм) для каждого дня
temperature = list(map(float, input("Введите температуру воздуха (°C) для каждого дня м

# Инициализация переменных для суммы осадков снега и дождя
snow_precipitation = 0
rain_precipitation = 0

# Пробегаем по дням и вычисляем сумму осадков снега и дождя
for i in range(len(precipitation)):
    if temperature[i] <= 0:  # Если температура ниже или равна 0, то осадки - снег
        snow_precipitation += precipitation[i]
    else:  # Если температура выше 0, то осадки - дождь
        rain_precipitation += precipitation[i]

# Вывод результатов
print(f"Общее количество осадков в виде снега: {snow_precipitation} мм")
print(f"Общее количество осадков в виде дождя: {rain_precipitation} мм")
```

Рисунок 4. Код к 1 заданию

```
Введите количество осадков (мм) для каждого дня месяца через пробел: 5 10 3 0 0 8 12 4
Введите температуру воздуха (°C) для каждого дня месяца через пробел: -2 5 0 4 -1 -3 2 6
Общее количество осадков в виде снега: 16.0 мм
Общее количество осадков в виде дождя: 26.0 мм
```

Рисунок 5. Задание

5. Зафиксировал изменения

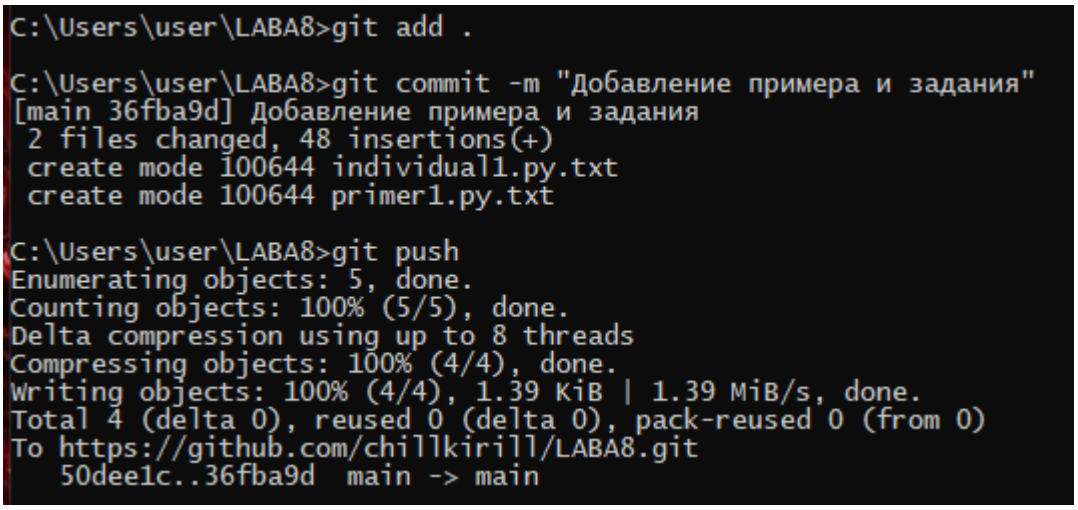

Рисунок 6. Изменения

Ответы на контрольные вопросы:

1. **Что такое кортежи в языке Python?**

Кортежи — это неизменяемые последовательности, которые могут содержать элементы различных типов. Они используются для хранения коллекций данных, аналогично спискам, но с тем отличием, что их содержимое нельзя изменить после создания.

2. Каково назначение кортежей в языке Python?

Кортежи используются для хранения фиксированных наборов данных, которые не должны изменяться. Они могут быть полезны для группировки связанных данных и передачи их в функции.

3. Как осуществляется создание кортежей?

Кортежи создаются с помощью круглых скобок (), также можно создать кортеж без скобок, просто перечислив элементы через запятую. Для создания кортежа с одним элементом необходимо добавить запятую.

4. Как осуществляется доступ к элементам кортежа?

Доступ к элементам кортежа осуществляется с помощью индексов, аналогично спискам

5. Зачем нужна распаковка (деструктуризация) кортежа?

Распаковка кортежа позволяет присвоить значения его элементам переменным в одном выражении, что делает код более читаемым и удобным.

6. Какую роль играют кортежи в множественном присваивании?

Кортежи позволяют удобно присваивать несколько значений нескольким переменным одновременно, что упрощает код и делает его более понятным.

7. Как выбрать элементы кортежа с помощью среза?

Срезы для кортежей работают так же, как и для списков: my tuple[start:end] вернет элементы с индексами от start до end-1.

8. Как выполняется конкатенация и повторение кортежей?

Конкатенация осуществляется с помощью оператора +, а повторение — с помощью оператора *

9. Как выполняется обход элементов кортежа?

Обход элементов кортежа можно осуществить с помощью цикла for.

10. Как проверить принадлежность элемента кортежу?

Используя оператор in if element in my tuple

11. Какие методы работы с кортежами Вам известны?

Кортежи имеют ограниченное количество методов, count — подсчитывает количество вхождений элемента. index — возвращает индекс первого вхождения элемента.

12. Допустимо ли использование функций агрегации таких как len, sum и тд при работе с кортежами?

Да, функции агрегации, такие как len, sum, min, max, могут использоваться с кортежами так же, как и со списками.

13. Как создать кортеж с помощью спискового включения?

Кортеж можно создать с помощью генератора кортежей, используя круглые скобки

Вывод: в ходе работы исследовал базовые возможности системы контроля версий Git для работы с локальными репозиториями.