

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 ПРАКТИЧЕСКОЙ РАБОТЕ №8**  
**дисциплины «Программирование на Python»**  
**Вариант 23**

Выполнил:  
Мотовилов Вадим Борисович  
2 курс, группа ИВТ-б-о-22-1,  
09.03.01 «Информатика и  
вычислительная техника»,  
направленность (профиль)  
«Информатика и вычислительная  
техника», очная форма обучения

---

(подпись)

Руководитель практики:  
Воронкин Роман Александрович

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2023 г.

Порядок выполнения работы:

1. Создал репозиторий и скопировал его

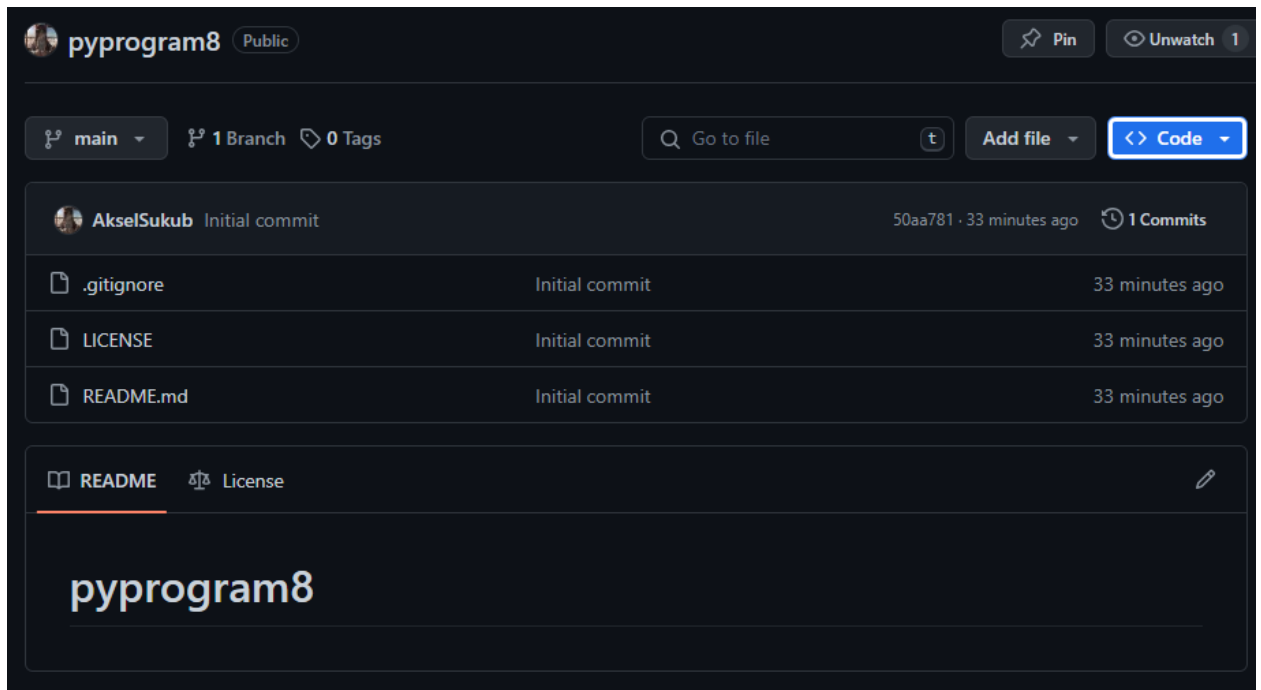


Рисунок 1. Созданный репозиторий

2. Изменил файл .gitignore и README.rm и добавил git flow

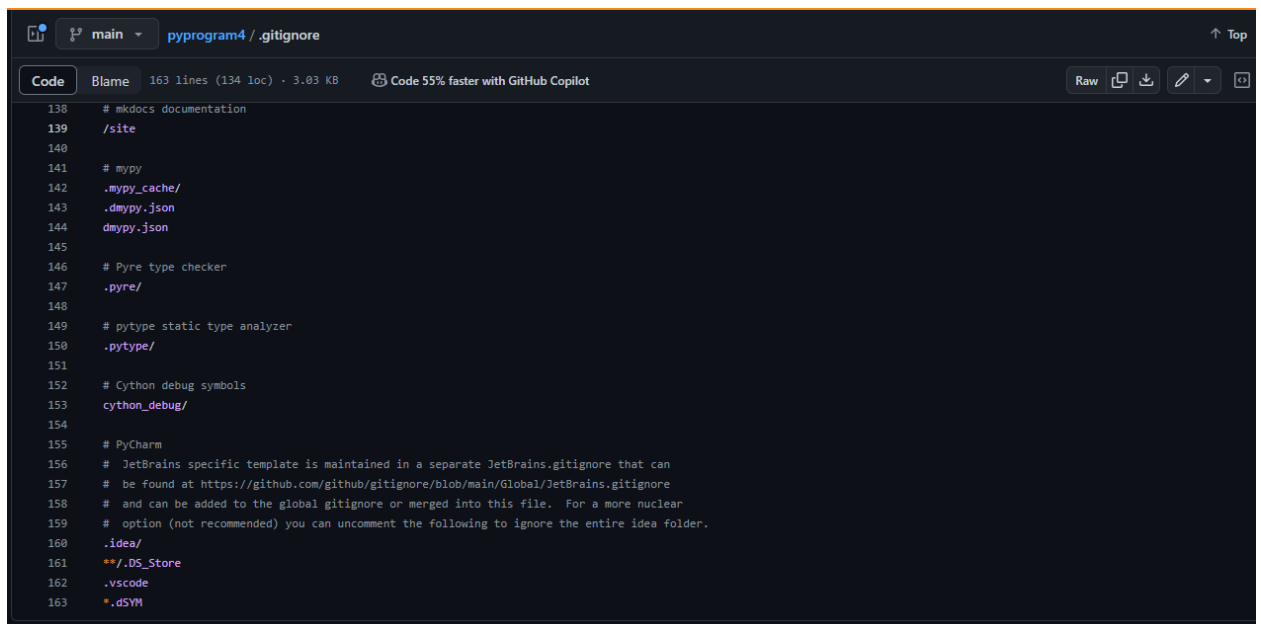


Рисунок 2. Измененный файл .gitignore

3. Выполнил задания

Из элементов кортежа  $m$  сформировать кортеж  $n$  того же размера по правилу: если номер  $i$  элемента кортежа  $m$  нечетный, то  $n_i = i \times m_i$ , в противном случае  $n_i = m_i/i$ .

```
zadanie1.py > ...
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  def process_tuple(m):
5      n = tuple(i * m_i if i % 2 != 0 else m_i / i for i, m_i in enumerate(m, 1))
6      return n
7
8  tuple_m = (1, 2, 3, 4, 5)
9  tuple_n = process_tuple(tuple_m)
10 print(tuple_n)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\1\pyprogram8\prog> & "C:/Program Files/Python311/python.exe" c:/Users/1/pyprogram8
(1, 1.0, 9, 1.0, 25)
```

Рисунок 3 . Выполнение 1 задания

Ответы на контрольные вопросы:

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

В Python списки представляют собой упорядоченные изменяемые коллекции элементов. Элементы могут быть различных типов данных.

2. **Каково назначение кортежей в языке Python?**

Кортежи в Python - это упорядоченные неизменяемые коллекции элементов. Они используются для хранения данных, которые не должны изменяться.

3. **Как осуществляется создание кортежей?**

Кортежи создаются с использованием круглых скобок () или функции tuple().  
Пример: my\_tuple = (1, 2, 3).

**4. Как осуществляется доступ к элементам кортежа?**

Доступ к элементам кортежа осуществляется по индексу, например, `element = my_tuple[0]`.

**5. Зачем нужна распаковка (деструктуризация) кортежа?**

Распаковка позволяет присвоить значения элементов кортежа отдельным переменным. Например, `a, b, c = my_tuple`.

**6. Какую роль играют кортежи в множественном присваивании?**

Кортежи позволяют одновременно присваивать значения нескольким переменным в одной строке кода.

**7. Как выбрать элементы кортежа с помощью среза?**

Срезы в кортежах работают так же, как и в списках. Например, `subset = my_tuple[1:3]`.

**8. Как выполняется конкатенация и повторение кортежей?**

Кортежи могут быть объединены с использованием оператора `+`, их можно повторять с использованием оператора `*`.

**9. Как выполняется обход элементов кортежа?**

Обход элементов кортежа можно выполнить с использованием цикла `for`. Например:

```
10. for element in my_tuple:  
    print(element)
```

**11. Как проверить принадлежность элемента кортежу?**

Используйте оператор `in`. Пример: `element in my_tuple`.

**12. Какие методы работы с кортежами Вам известны?**

Кортежи не имеют множества методов, но они поддерживают основные операции, такие как `count()` и `index()`.

**13. Допустимо ли использование функций агрегации таких как `len()`, `sum()` и т. д. при работе с кортежами?**  
Да, кортежи поддерживают функции агрегации, такие как `len()`, `sum()`, `max()`, `min()`, и т. д.

#### 14. Как создать кортеж с помощью спискового включения.

В Python нет концепции "спискового включения" для кортежей, как для списков, но можно использовать генераторы кортежей:

```
my_tuple = tuple(x * 2 for x in range(5))
```