**РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное**

**образовательное учреждение высшего**

**образования**

**«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**«Перегрузка операторов в языке Python»»**

**Отчет по лабораторной**

**работе по дисциплине**

**«Объектно-ориентированное**

**программирование»**

Выполнил студент группы ИВТ-б-о-21-1

Кочкаров Умар Ахматович.

«17» ноября 2023г.

Подпись студента_____

Работа защищена «  »_____20__г.

Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2023

**Цель работы:** приобретение навыков по перегрузке операторов 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, и клонировал его.

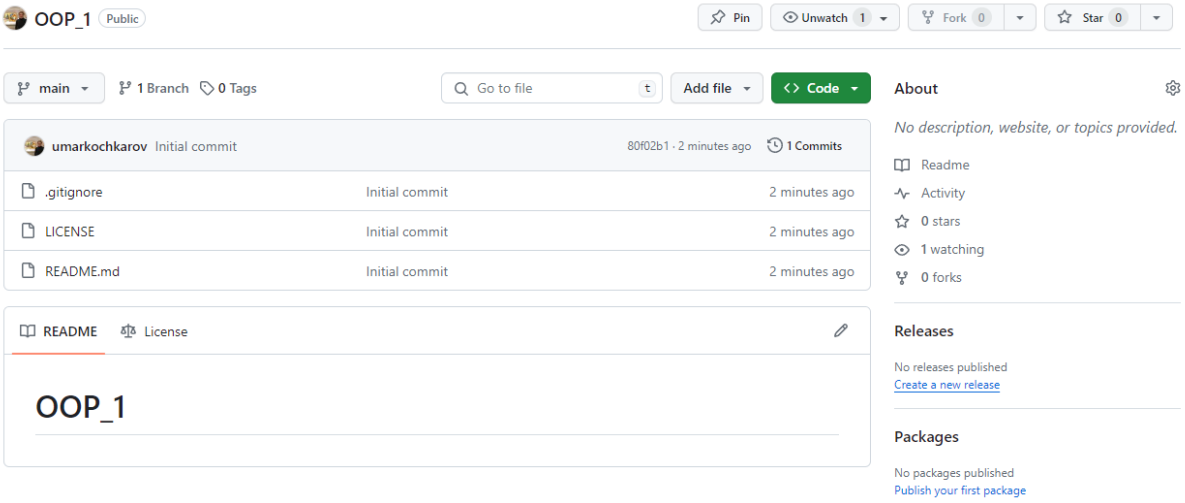


Рисунок 1. Создание репозитория

2. Организовать репозиторий в соответствии с Git-Flow init.

```
erken@LAPTOP-ESTC6OGF MINGW64 ~/Desktop/python/OOP_1 (main)
$ git flow init

Which branch should be used for bringing forth production releases?
   - main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/erken/Desktop/python/OOP_1/.git/hooks]
```

Рисунок 2. Организация в соответствии с Git-Flow init

3. Проработка примеров лабораторной работы:

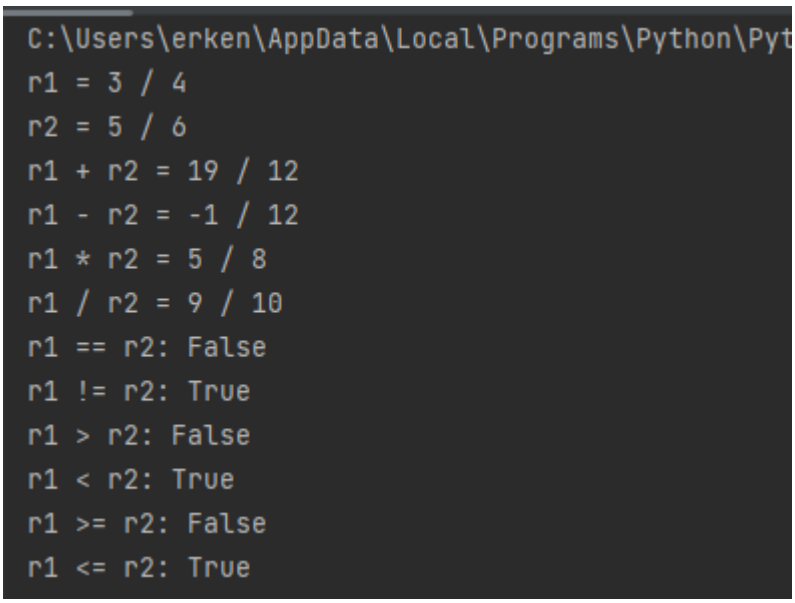


Рисунок 3. Пример

4. Индивидуальные задания (вариант 7):

**Задание 1.** Поле first — дробное число; поле second — целое число, показатель степени. Реализовать метод power() — возведение числа first в степень second. Метод должен правильно работать при любых допустимых значениях first и second.

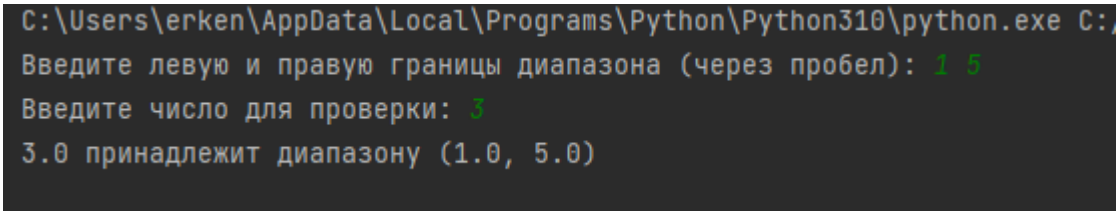


Рисунок 4. Индивидуальное задание 1

**Задание 2.** Реализовать класс Rational, используя два списка из 100 элементов типа int для представления числителя и знаменателя. Каждый элемент является десятичной цифрой. Младшая цифра имеет меньший индекс (единицы — в нулевом элементе списка). Реальный размер списка задается как аргумент конструктора инициализации.

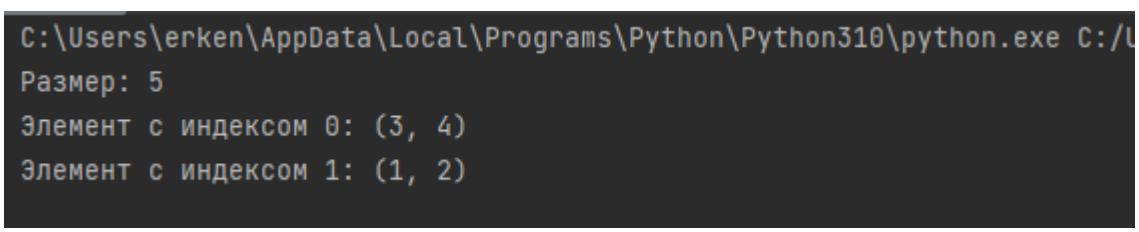


Рисунок 5. Индивидуальное задание 2

**Контрольные вопросы:**

**1. Какие средства существуют в Python для перегрузки операций?**

В python имеются методы, которые не вызываются напрямую, а вызываются встроенными функциями или операторами. С их помощью можно перегрузить операции.

**2. Какие существуют методы для перегрузки арифметических операций и операций отношения в языке Python?**

Пример: ___add___- сложение, __sub___- вычитание, __mul___- умножение.

**3. В каких случаях будут вызваны следующие методы: __add__, ____Iadd и ____radd__?**

____add_- вызывается при сложении двух чисел оператором «+». В случае, если это сделать не удаётся, вызываются ____iadd__и ____radd__, они

делают то же самое, что и арифметические операторы, перечисленные выше, но для аргументов, находящихся справа, и только в случае, если для левого операнда не определён соответствующий метод.

**4. Для каких целей предназначен метод __new__? Чем он отличается от метода ___init__?**

Управляет созданием экземпляра. В качестве обязательного аргумента принимает класс (не путать с экземпляром). Должен возвращать экземпляр класса для его последующей его передачи методу ___init___

**5. Чем отличаются методы __str___и __repr__?**

___str___- вызывается функциями str, print и format. Возвращает строковое представление объекта.

___repr___- вызывается встроенной функцией repr; возвращает "сырые" данные, использующиеся для внутреннего представления в python.

**Вывод:** приобретены навыки по перегрузке операторов при написании программ с использованием языка программирования Python версии 3.x.