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 №13**

**дисциплины «Основы программной инженерии»**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Выполнил:  Звездин Алексей Сергеевич  2 курс, группа ПИЖ-б-о-22-1,  09.03.04 «Программная инженерия», направленность (профиль) «Разработка и сопровождение программного обеспечения», очная форма обучения  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | |
|  | | Руководитель практики:  Воронкин Р. А., доцент кафедры инфокоммуникаций  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | |
|  | |  | |

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

Ставрополь, 2023 г.

**Ход работы**

1. Я изучил теоретический материал работы

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 1.1 – Изучение материала для лабораторной работы

1. Создал общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python

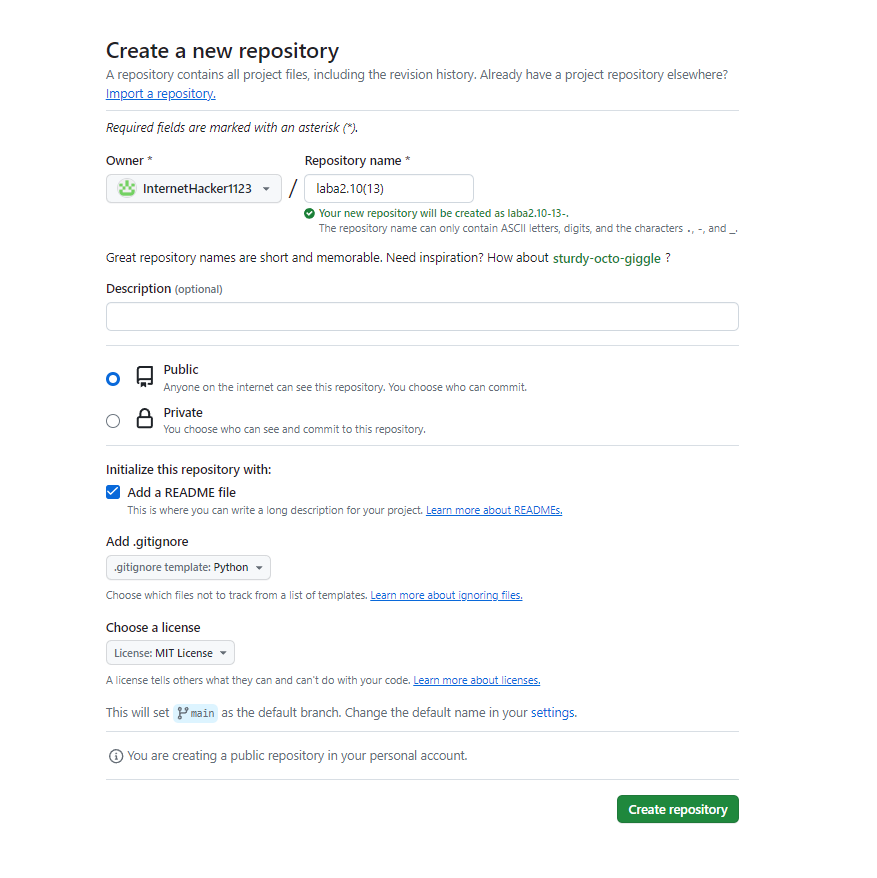


Рисунок 2.1 – Настройка репозитория

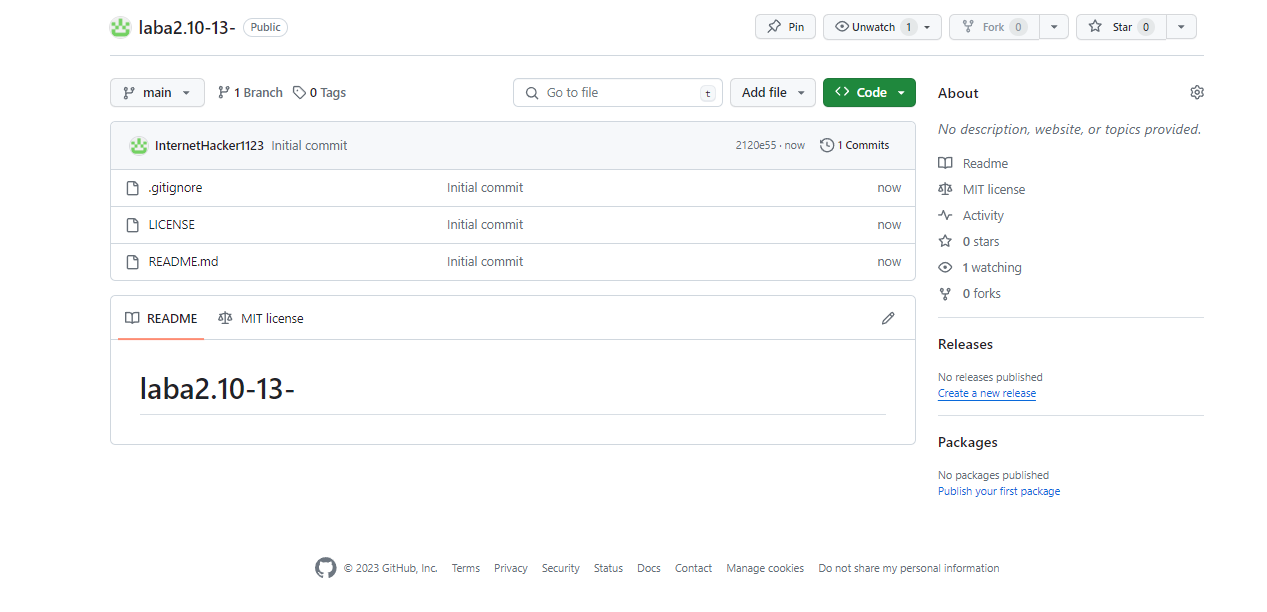


Рисунок 2.2 – Готовый репозиторий

1. Выполняю клонирование созданного репозитория

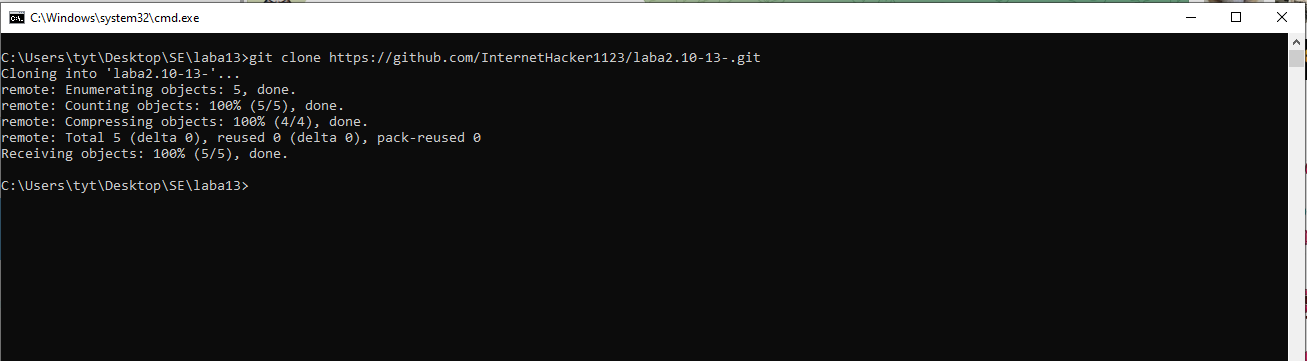


Рисунок 3.1 – Клонирование репозитория на локальный диск

1. Дополнил файл .gitignore необходимыми правилами для работы с VS Code

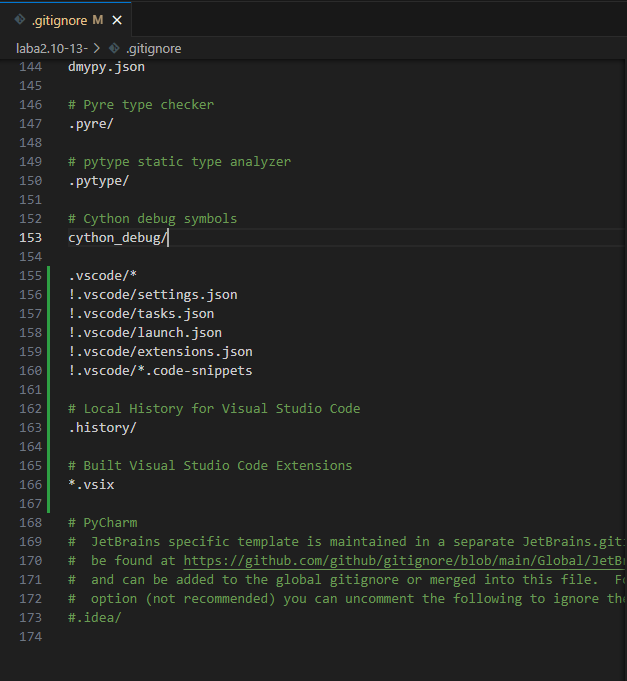


Рисунок 4.1 – .gitignore для VS Code

1. Организовал свой репозиторий в соответствии с моделью ветвления git-flow

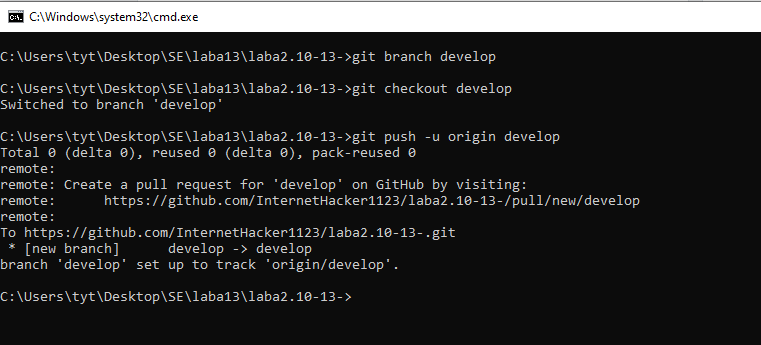


Рисунок 5.1 – Создание ветки develop от ветки main

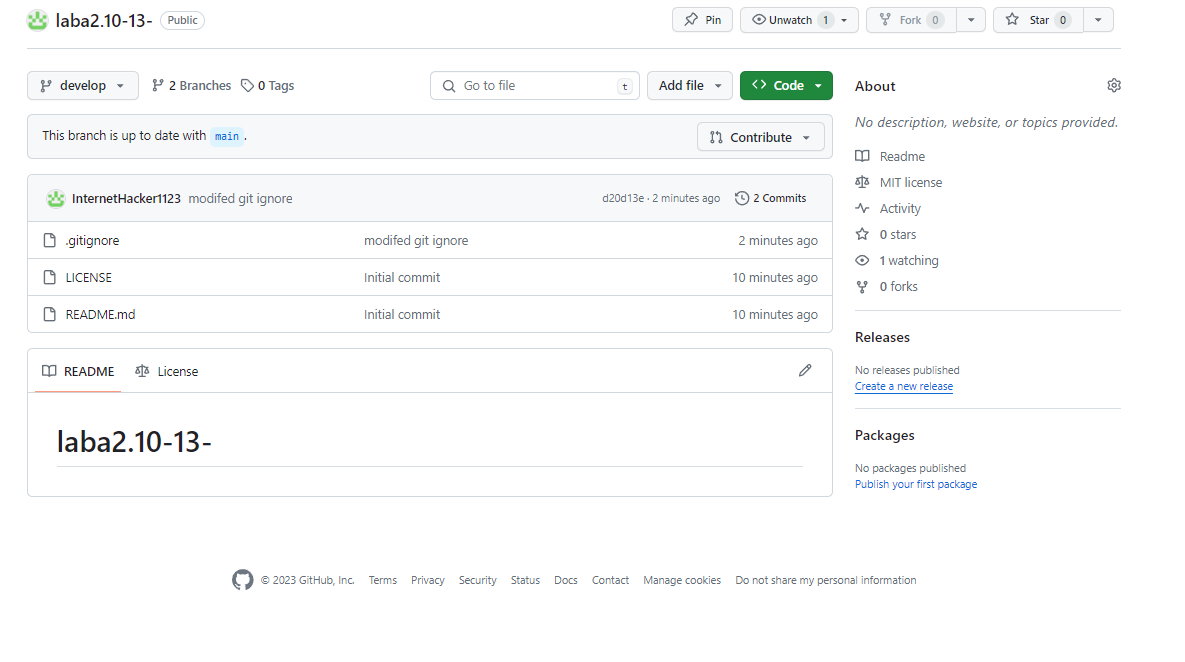


Рисунок 5.2 – Ветка develop на GitHub

1. Создал проект PyCharm в папке репозитория

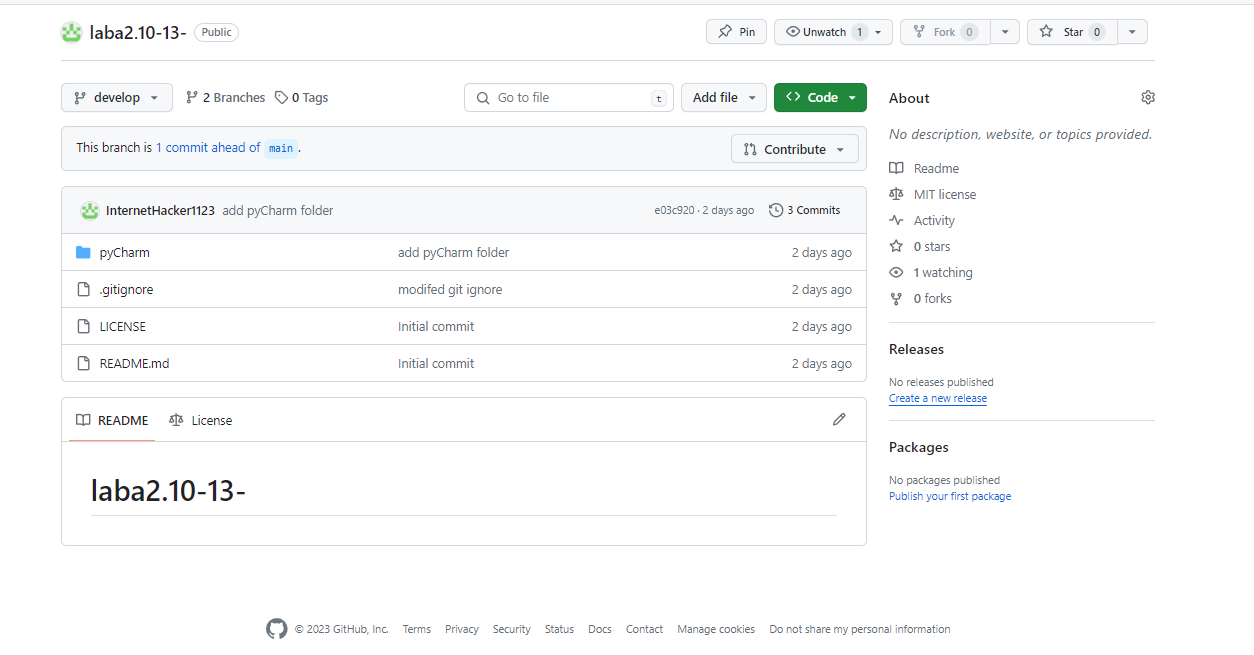


Рисунок 6.1 – Репозиторий с проектом PyCharm

1. Проработал примеры лабораторной работы.

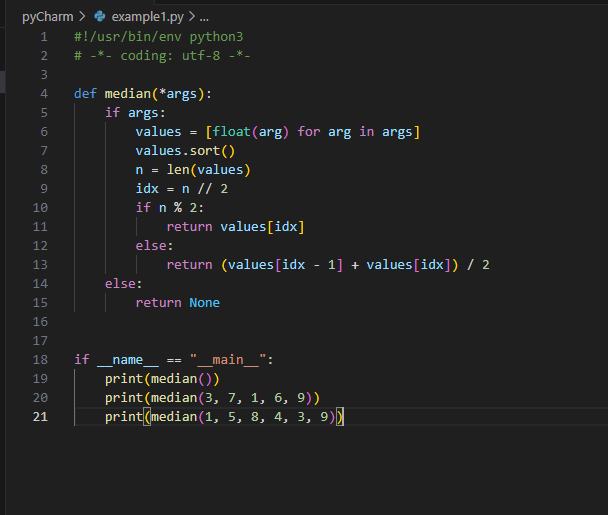


Рисунок 7.1 – Код программы example1.py

1. Решил поставленную задачу: написал функцию, вычисляющую среднее геометрическое своих аргументов.

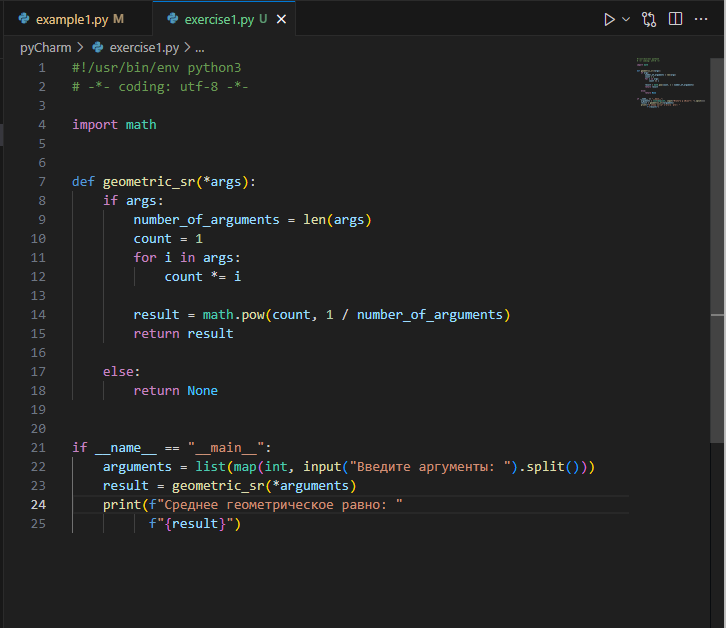


Рисунок 8.1 – Код программы exercise8.py

1. Решил поставленную задачу: написал функцию, вычисляющую среднее гармоническое своих аргументов.



Рисунок 9.1 – Код программы exercise2.py в IDE PyCharm

1. Привел в отчете скриншоты результатов выполнения примера при различных исходных данных, вводимых с клавиатуры.

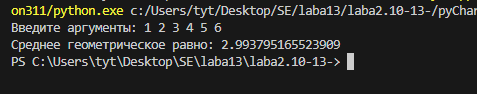


Рисунок 10.1 – Результат выполнения примера exercise1.py

1. Зафиксировал сделанные изменения в репозитории.

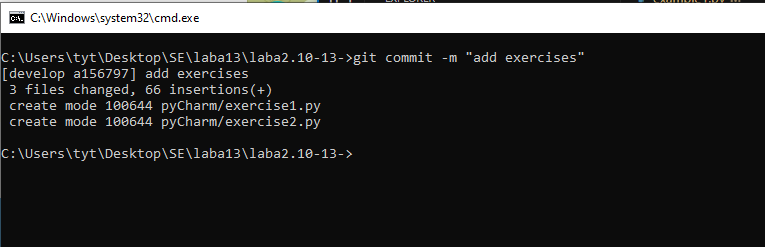


Рисунок 11.1 – Коммит файлов в репозитории git

1. Решил индивидуальное задание согласно своему варианту.



Рисунок 12.1 – Код программы individual.py

1. Самостоятельно подобрал или придумал задачу с переменным числом именованных аргументов. Привел решение этой задачи.

Задача: Напишите функцию, которая принимает переменное количество именованных аргументов (например, предметы и их цены) и выводит их на экран в виде списка.

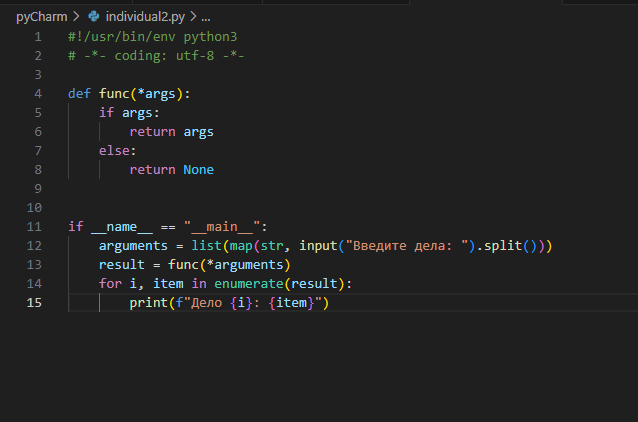


Рисунок 13.1 – Код программы individual.py

1. Зафиксировал изменения в репозитории.

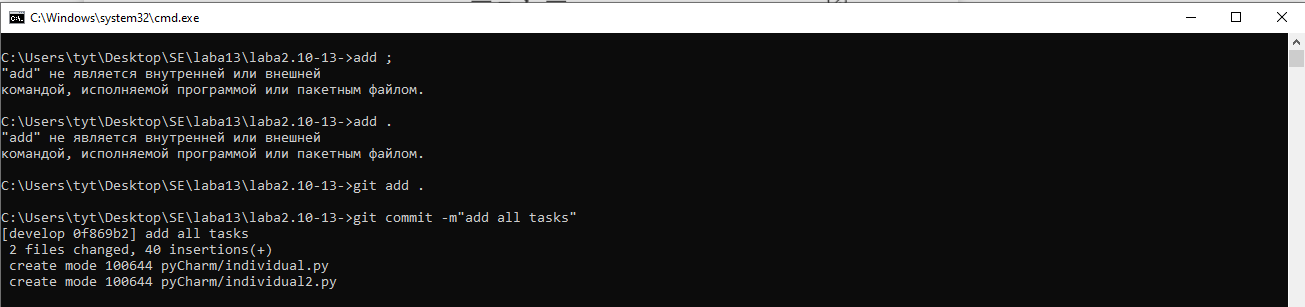


Рисунок 14.1 – Коммит файлов в репозитории git

**Контрольные вопросы**

1. Какие аргументы называются позиционными в Python?

Позиционные аргументы — это аргументы, которые передаются в функцию в определенном порядке, и их значения связываются с параметрами функции в том же порядке.

1. Какие аргументы называются именованными в Python?

Именованные аргументы — это аргументы, которые передаются в функцию с указанием имени параметра.

1. Для чего используется оператор \*?

Этот оператор позволяет «распаковывать» объекты, внутри которых хранятся некие элементы.

1. Каково назначение конструкций \*args и \*\*kwargs?

Каждая из этих конструкций используется для распаковки аргументов соответствующего типа, позволяя вызывать функции со списком аргументов переменной длины