

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 РАБОТЕ №2.5**  
**дисциплины «Основы кроссплатформенного программирования»**

Выполнила:  
Дудова Мира Сергеевна  
1 курс, группа ИТС-б-о-22-1,  
11.03.02 «Инфокоммуникационные  
технологии и системы связи»,  
направленность (профиль)  
«Инфокоммуникационные системы и  
сети», очная форма обучения

---

(подпись)

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

---

(подпись)

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

Ставрополь, 2023 г.

Тема: работа с кортежами в языке Python.

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

Порядок выполнения работы:

1. Создала новый репозиторий и клонировала его на свой компьютер

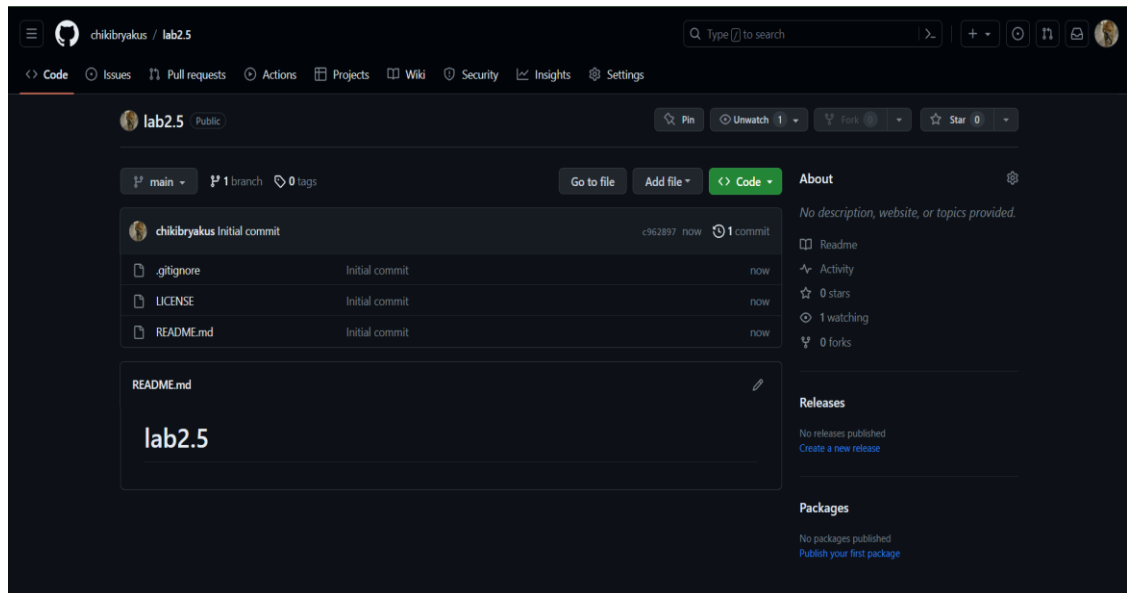


Рисунок 1 – Новый репозиторий

2. Проклонировала репозиторий на свой компьютер. В ходе данной лабораторной работы работала с моделью ветвления git-flow.

```
C:\Users\Hp>git clone https://github.com/chikibryakus/lab2.5.git
Cloning into 'lab2.5'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.

C:\Users\Hp>cd C:\Users\Hp\lab2.5
C:\Users\Hp\lab2.5>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]

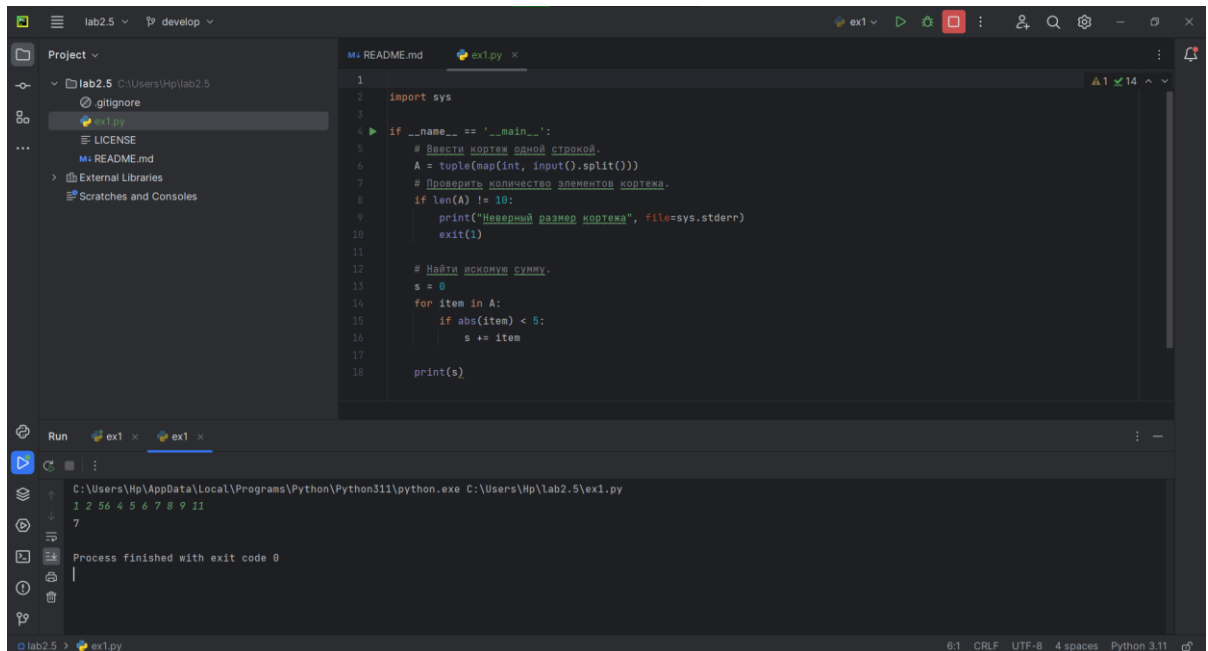
How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/Hp/lab2.5/.git/hooks]
```

Рисунок 2 – Клонирование и модель ветвления git-flow

Реализовывала примеры и индивидуальные задания на основе ветки develop.

### 3. Пример №1. Добавила новый файл ex1.py.

**Условие примера:** ввести кортеж A из 10 элементов, найти сумму элементов, меньших по модулю 5, и вывести ее на экран. Использовать в программе вместо списков кортежи



```
1 import sys
2
3 if __name__ == '__main__':
4     # Ввести кортеж одной строкой.
5     A = tuple(map(int, input().split()))
6     # Проверить количество элементов кортежа.
7     if len(A) != 10:
8         print("Неверный размер кортежа", file=sys.stderr)
9         exit(1)
10
11     # Найти искомую сумму.
12     s = 0
13     for item in A:
14         if abs(item) < 5:
15             s += item
16
17     print(s)
```

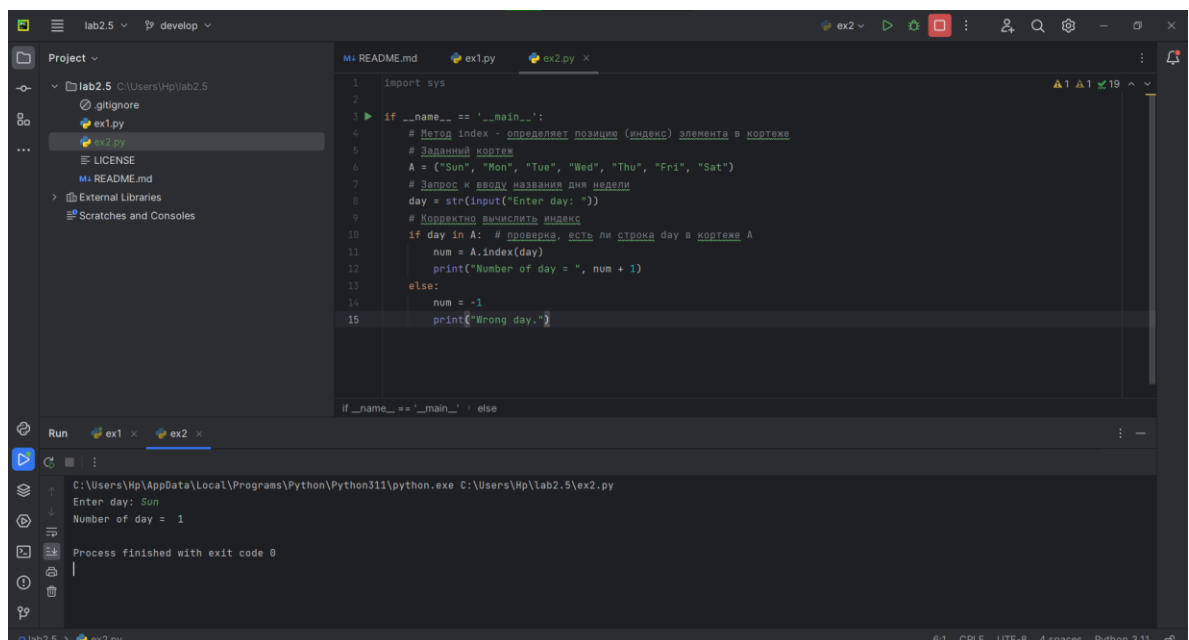
Run console output:

```
C:\Users\Hp\AppData\Local\Programs\Python\Python311\python.exe C:\Users\Hp\lab2.5\ex1.py
1 2 56 4 5 6 7 8 9 11
7
Process finished with exit code 0
```

Рисунок 3 – Реализация первого примера

### 4. Пример №2. Создала новый файл под названием ex2.py

**Условие примера:** в перечне названий дней недели вычисляется порядковый номер дня.



```
1 import sys
2
3 if __name__ == '__main__':
4     # Метод index - определяет позицию (индекс) элемента в кортеже
5     # Заданный кортеж
6     A = ("Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat")
7     # Запрос к вводу названия дня недели
8     day = str(input("Enter day: "))
9     # Корректно вычислить индекс
10     if day in A: # проверка, есть ли строка day в кортеже A
11         num = A.index(day)
12         print("Number of day = ", num + 1)
13     else:
14         num = -1
15         print("Wrong day.")
```

Run console output:

```
C:\Users\Hp\AppData\Local\Programs\Python\Python311\python.exe C:\Users\Hp\lab2.5\ex2.py
Enter day: Sun
Number of day = 1
Process finished with exit code 0
```

Рисунок 4 – Реализация второго примера

5. Индивидуальное задание №1. Создала новый файл под названием ind1.py

Вариант 8.

**Условие задания:** определить, есть ли в кортеже хотя бы одна тройка соседних чисел, в которой средний элемент больше своих «соседей», т. е. предшествующего и последующего. В случае положительного ответа определить номера элементов первой из таких троек.

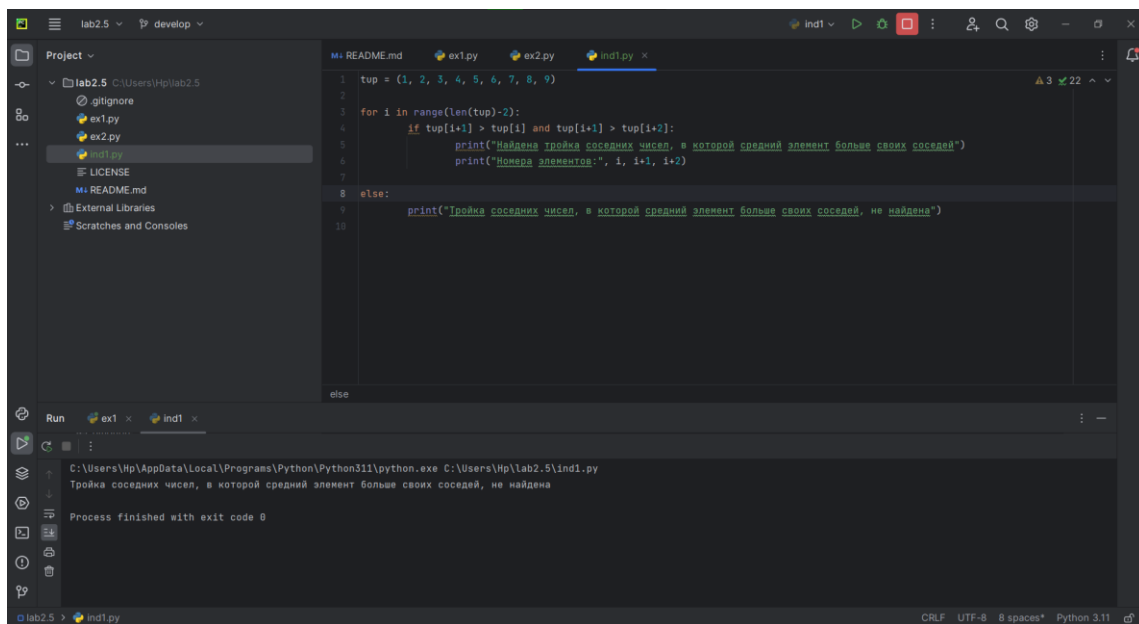


Рисунок 5 – Программа и ее результат

6. Слила ветку develop с веткой main и отправил на удаленный сервер.

```
C:\Users\Hp\lab2.5>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

C:\Users\Hp\lab2.5>git merge develop
Updating c962897..75ec1ea
Fast-forward
 ex1.py | 18 ++++++
 ex2.py | 15 ++++++
 ind1.py | 9 ++++++
 3 files changed, 42 insertions(+)
 create mode 100644 ex1.py
 create mode 100644 ex2.py
 create mode 100644 ind1.py
C:\Users\Hp\lab2.5>
```

Рисунок 6 – Слияние веток develop в main

**Ссылка:** <https://github.com/chikibryakus/lab2.5>

## **Ответы на контрольные вопросы:**

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

Списки - это коллекции указателей на элементы. В Python не обязательно, чтобы все элементы списка были одного типа

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

Кортеж в Python — это более быстрый и неизменяемый аналог списка. Он очень часто используется для защиты хранимых данных приложения от незапланированных или непреднамеренных изменений.

### **3. Как осуществляется создание кортежей?**

Кортеж создается путем помещения всех элементов (элементов) в круглые скобки (), разделенных запятыми.

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

Доступ к элементам кортежа осуществляется также как к элементам списка – через указание индекса. Но, как уже было сказано – изменять элементы кортежа нельзя!

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

Часто кортежи содержат значения разных типов. Сложно запомнить, каким индексом обозначается каждое значение. Чтобы упростить работу, можно разобрать кортеж.

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

Кортежи в Python играют важную роль в множественном присваивании. Множественное присваивание позволяет присваивать значения нескольким переменным за один раз. Кортежи могут быть использованы для присваивания нескольких переменным значений за один раз.

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

Элементы кортежа можно выбрать с помощью среза, указав индексы начала и конца нужного диапазона через двоеточие в квадратных скобках.

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

Конкатенация (объединение) кортежей в Python выполняется с помощью оператора "+". Для повторения кортежа используется оператор "\*".

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

Обход элементов кортежа может быть выполнен с помощью цикла for.

10. Как проверить принадлежность элемента кортежу?

Для проверки нахождения элемента в кортеже можно использовать оператор in.

11. Какие методы работы с кортежами Вам известны?

1. count(x) - возвращает количество вхождений элемента x в кортеж.
2. index(x) - возвращает индекс первого вхождения элемента x в кортеже.
3. len(t) - возвращает длину кортежа t.
4. Допустимо ли использование функций агрегации таких как len() , sum() и т. д. при работе с кортежами?

12. Да, использование функций агрегации, таких как len(), sum(), min(), max(), и т.д. допустимо при работе с кортежами, так как они позволяют получать информацию о элементах кортежа и выполнять различные математические операции над ними.

13. Как создать кортеж с помощью спискового включения.

Кортеж можно создать с помощью спискового включения с использованием круглых скобок.

**Вывод:** в ходе данной лабораторной работы, я приобрела навыки по работе с кортежами при написании программ с помощью языка программирования Python версии 3.x

