

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

Выполнил:  
Хачатрян Владимир Владимирович  
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., доцент кафедры  
инфокоммуникаций

---

(подпись)

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

Ставрополь, 2023 г.

## Работа с кортежами в языке Python.

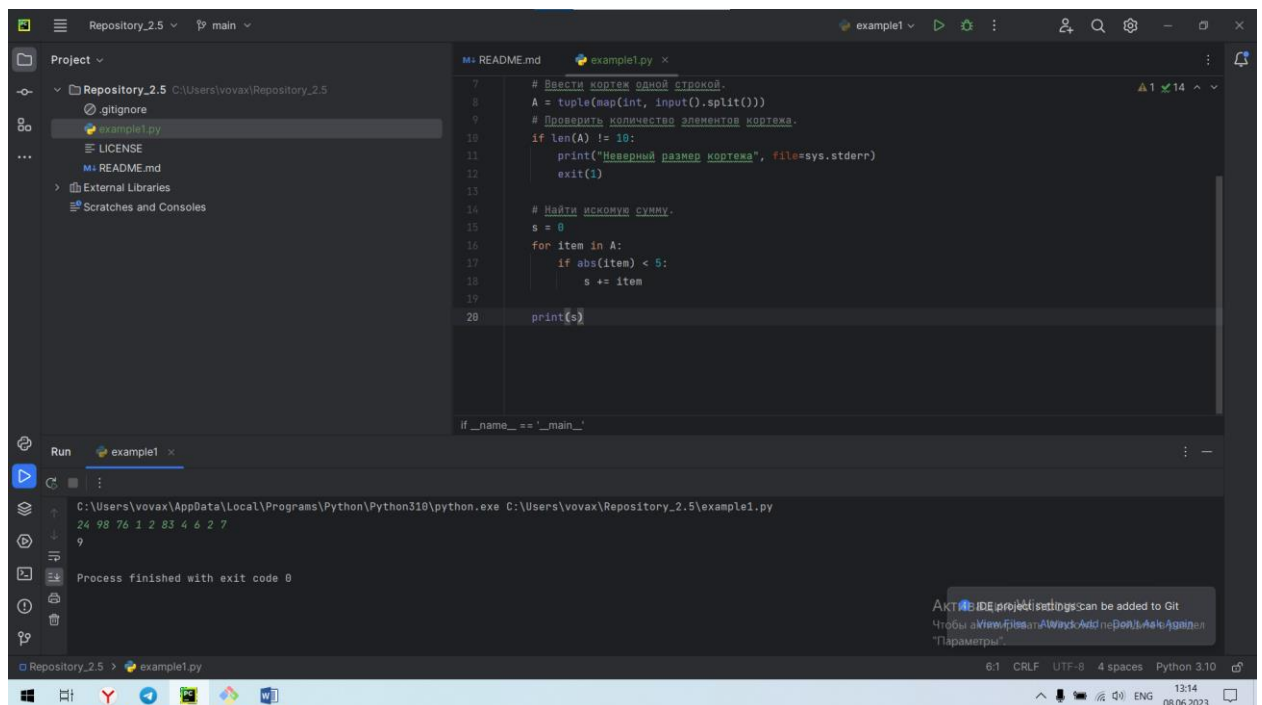
**Цель:** приобретение навыков по работе с кортежами при написании программ с помощью языка программирования Python версии 3.x.

### Ход работы:

Создал общедоступный репозиторий на GitHub, добавил файл .gitignore с необходимыми правилами. Клонировал репозиторий на свой компьютер.

```
c:\Users\vovax>git clone https://github.com/GitVolodya/Repository_2.5.git
Cloning into 'Repository_2.5'...
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.
```

**Пример 1:** ввести кортеж A из 10 элементов, найти сумму элементов, меньших по модулю 5, и вывести ее на экран. Использовать в программе вместо списков кортежи.



```
7 # Ввести кортеж одной строкой.
8 A = tuple(map(int, input().split()))
9 # Проверить количество элементов кортежа.
10 if len(A) != 10:
11     print("Неверный размер кортежа", file=sys.stderr)
12     exit(1)
13
14 # Найти искомую сумму.
15 s = 0
16 for item in A:
17     if abs(item) < 5:
18         s += item
19
20 print(s)

if __name__ == '__main__':
```

Run example1

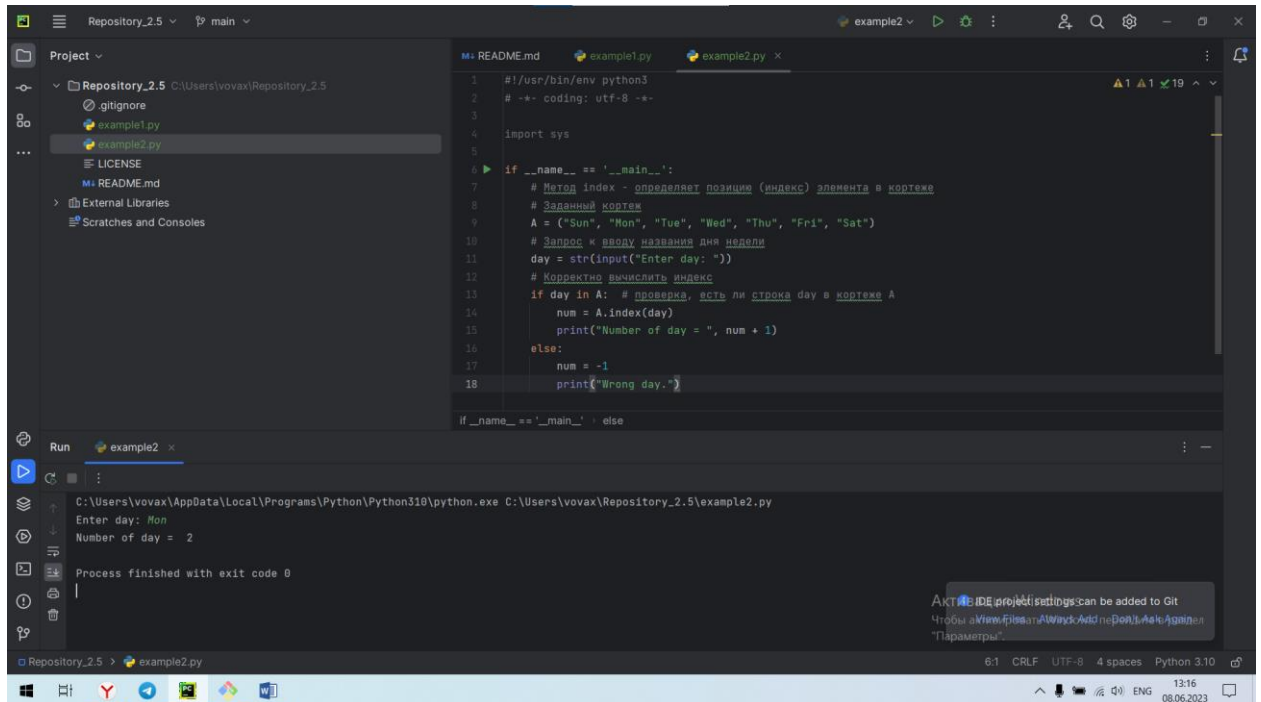
C:\Users\vovax\AppData\Local\Programs\Python\Python310\python.exe C:\Users\vovax\Repository\_2.5\example1.py

24 98 76 1 2 83 4 6 2 7

9

Process finished with exit code 0

**Пример 2:** в перечне названий дней недели вычисляется порядковый номер дня.



```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 import sys
5
6 if __name__ == '__main__':
7     # Метод index - определяет позицию (индекс) элемента в кортеже
8     # Заданный кортеж
9     A = ("Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat")
10    # Запрос к вводу названия дня недели
11    day = str(input("Enter day: "))
12    # Корректно вычислить индекс
13    if day in A: # проверка, есть ли строка day в кортеже A
14        num = A.index(day)
15        print("Number of day = ", num + 1)
16    else:
17        num = -1
18        print("Wrong day.")
19
20 if __name__ == '__main__': else
```

Run example2 x

C:\Users\vovax\AppData\Local\Programs\Python\Python310\python.exe C:\Users\vovax\Repository\_2.5\example2.py

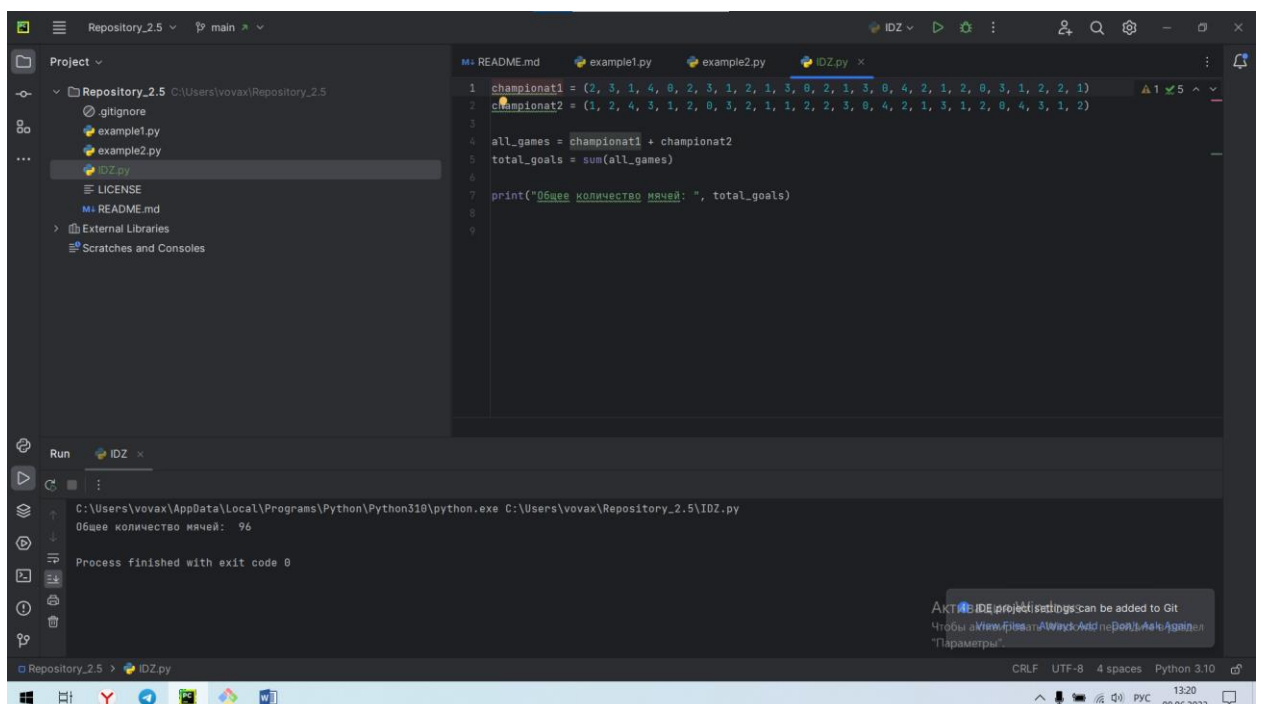
Enter day: Mon

Number of day = 2

Process finished with exit code 0

## Индивидуальное задание

**Условие задания:** Известно количество мячей, забитых футбольной командой за каждую игру в двух чемпионатах, которое хранится в двух кортежах. В каждом из чемпионатов команда сыграла 26 игр. Найти общее количество мячей, забитых командой в двух чемпионатах.



```
1 championat1 = (2, 3, 1, 4, 0, 2, 3, 1, 2, 1, 3, 0, 2, 1, 3, 0, 4, 2, 1, 2, 0, 3, 1, 2, 2, 1)
2 championat2 = (1, 2, 4, 3, 1, 2, 0, 3, 2, 1, 1, 2, 2, 3, 0, 4, 2, 1, 3, 1, 2, 0, 4, 3, 1, 2)
3
4 all_games = championat1 + championat2
5 total_goals = sum(all_games)
6
7 print("Общее количество мячей: ", total_goals)
8
9
```

Run IDZ x

C:\Users\vovax\AppData\Local\Programs\Python\Python310\python.exe C:\Users\vovax\Repository\_2.5\IDZ.py

Общее количество мячей: 96

Process finished with exit code 0

Ссылка на репозиторий: [https://github.com/GitVolodya/Repository\\_2.5.git](https://github.com/GitVolodya/Repository_2.5.git)

### **Ответы на контрольные вопросы:**

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

Список (list) – это структура данных для хранения объектов различных типов. В нем можно хранить объекты различных типов. Размер списка не статичен, его можно изменять. Список по своей природе является изменяемым типом данных. 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.

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

Деструктуризация кортежа позволяет извлекать отдельные элементы кортежа и присваивать их значения переменным. 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 играют важную роль в множественном присваивании, потому что они могут быть использованы для одновременного присваивания нескольких переменных из кортежа.

#### *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?*

A. count(x) - возвращает количество вхождений элемента x в кортеж.

B. index(x) - возвращает индекс первого вхождения элемента x в кортеже.

C. len(t) - возвращает длину кортежа t.

D. sorted() - возвращает новый отсортированный список из элементов кортежа.

12. *Допустимо ли использование функций агрегации таких как len(), sum() и т. д. при работе с кортежами?*

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

13. *Как создать кортеж с помощью спискового включения.*

Для создания кортежа с помощью спискового включения можно использовать функцию tuple(), которая преобразует список в кортеж.

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