

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»**  
**Вариант 3**

Выполнил:  
Баканов Артем Вадимович  
1 курс, группа ИТС-6-0-22-1,  
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 со строками в языке Python.

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

**Ссылка:** <https://github.com/temacteklyannayapuwka/cross-platform-programming-v.2.5>

### Порядок выполнения работы:

1. Создал общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT.

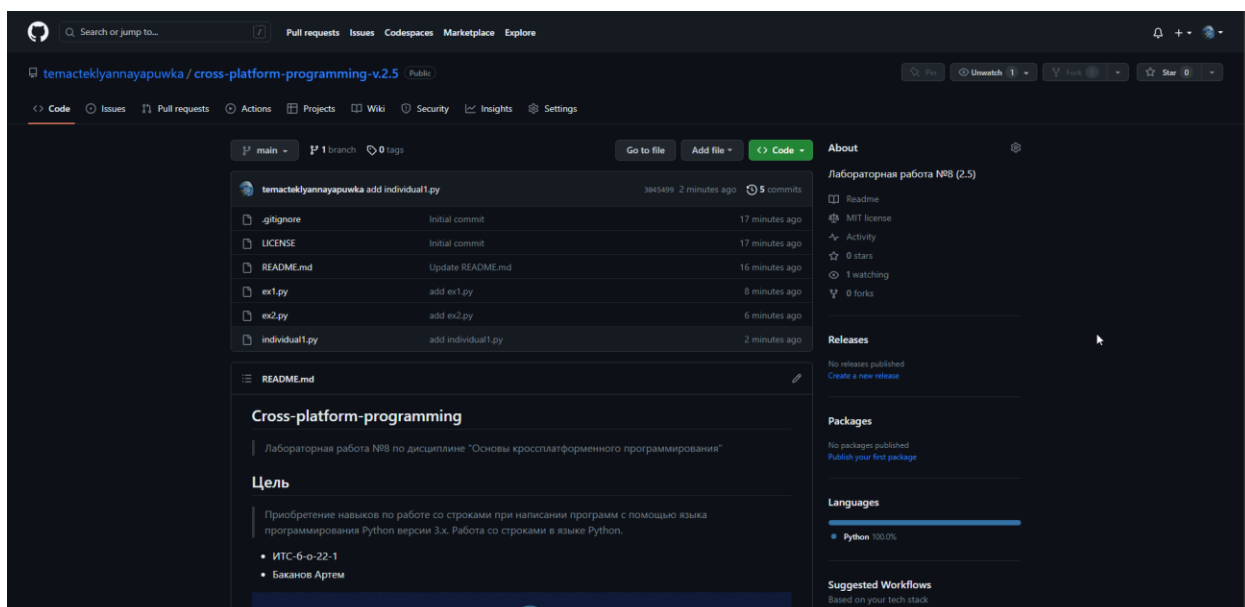
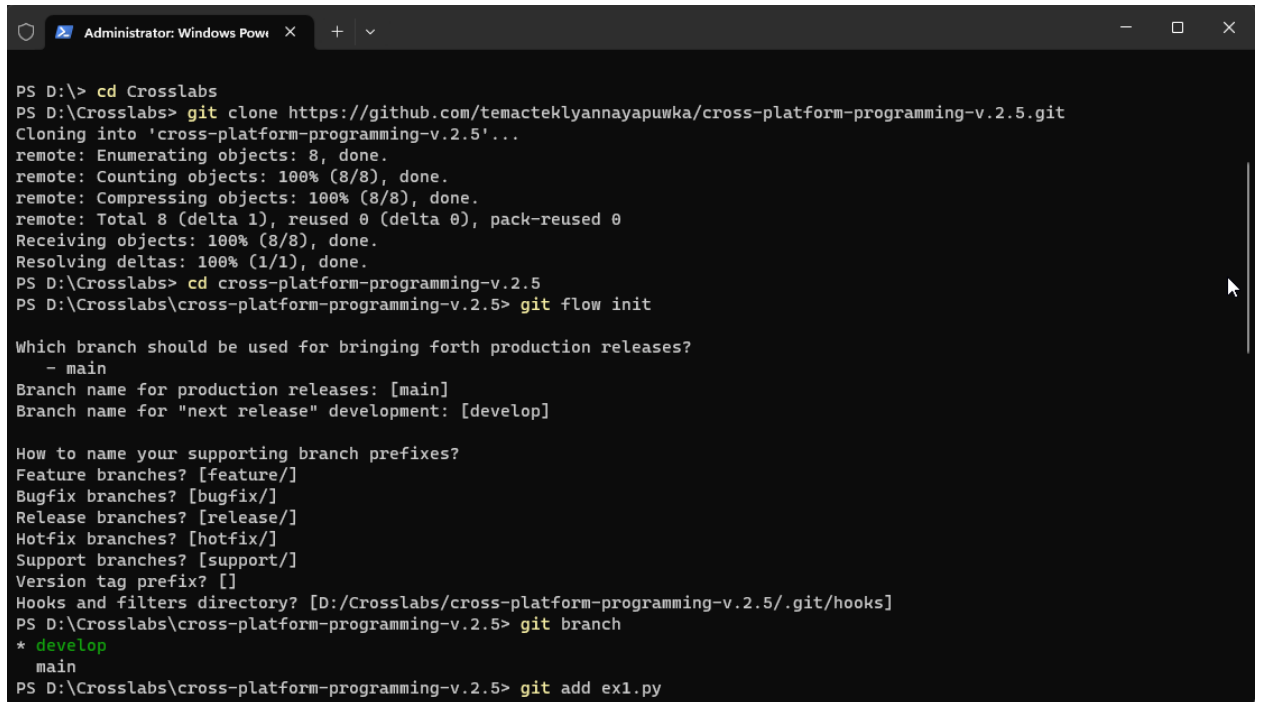


Рис. 1. Новый репозиторий.

2. В ходе данной лабораторной работы работал с моделью ветвления gitflow.



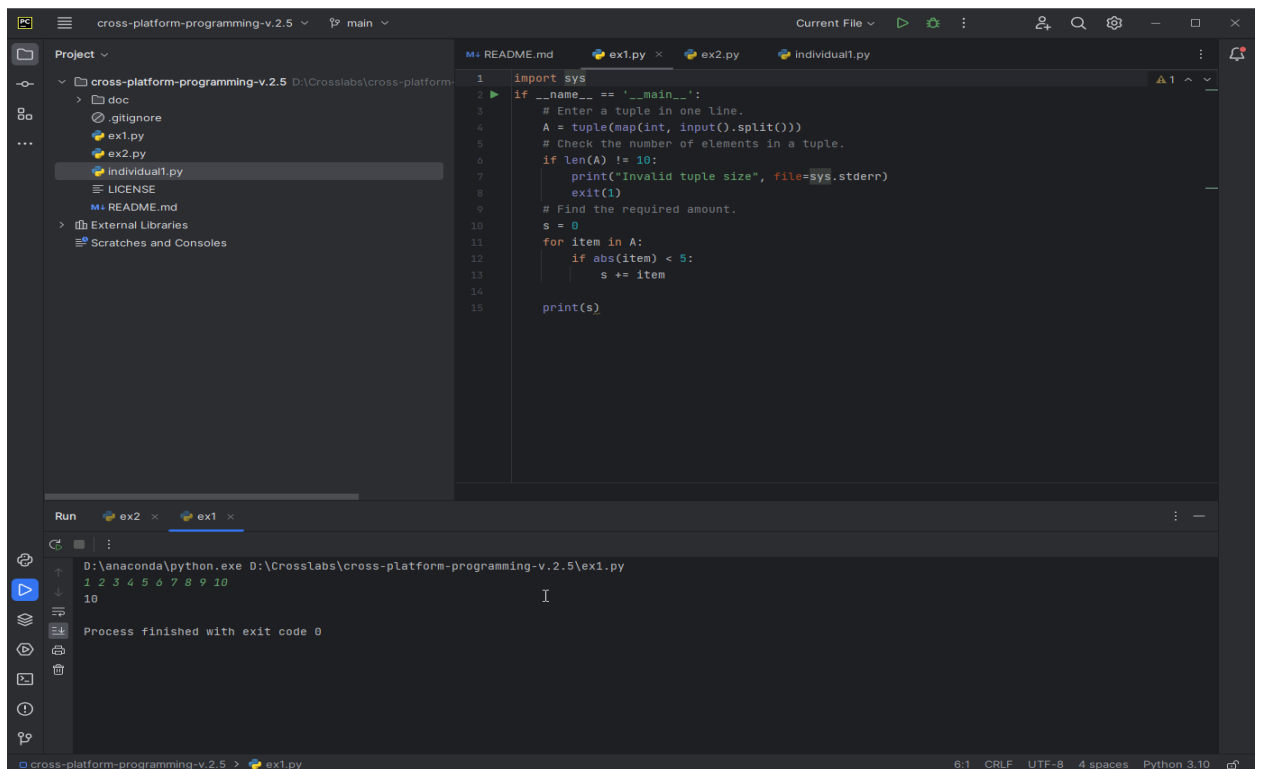
```
PS D:\> cd Crosslabs
PS D:\Crosslabs> git clone https://github.com/temacteklyannayapuwka/cross-platform-programming-v.2.5.git
Cloning into 'cross-platform-programming-v.2.5'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (8/8), done.
Resolving deltas: 100% (1/1), done.
PS D:\Crosslabs> cd cross-platform-programming-v.2.5
PS D:\Crosslabs\cross-platform-programming-v.2.5> 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? [D:/Crosslabs/cross-platform-programming-v.2.5/.git/hooks]
PS D:\Crosslabs\cross-platform-programming-v.2.5> git branch
* develop
  main
PS D:\Crosslabs\cross-platform-programming-v.2.5> git add ex1.py
```

Рис. 2. Клонирование и модель ветвления git-flow.

3. Пример №1. Условие примера: ввести кортеж А из 10 элементов, найти сумму элементов, меньших по модулю 5, и вывести ее на экран. Использовать в программе вместо списков кортежи.



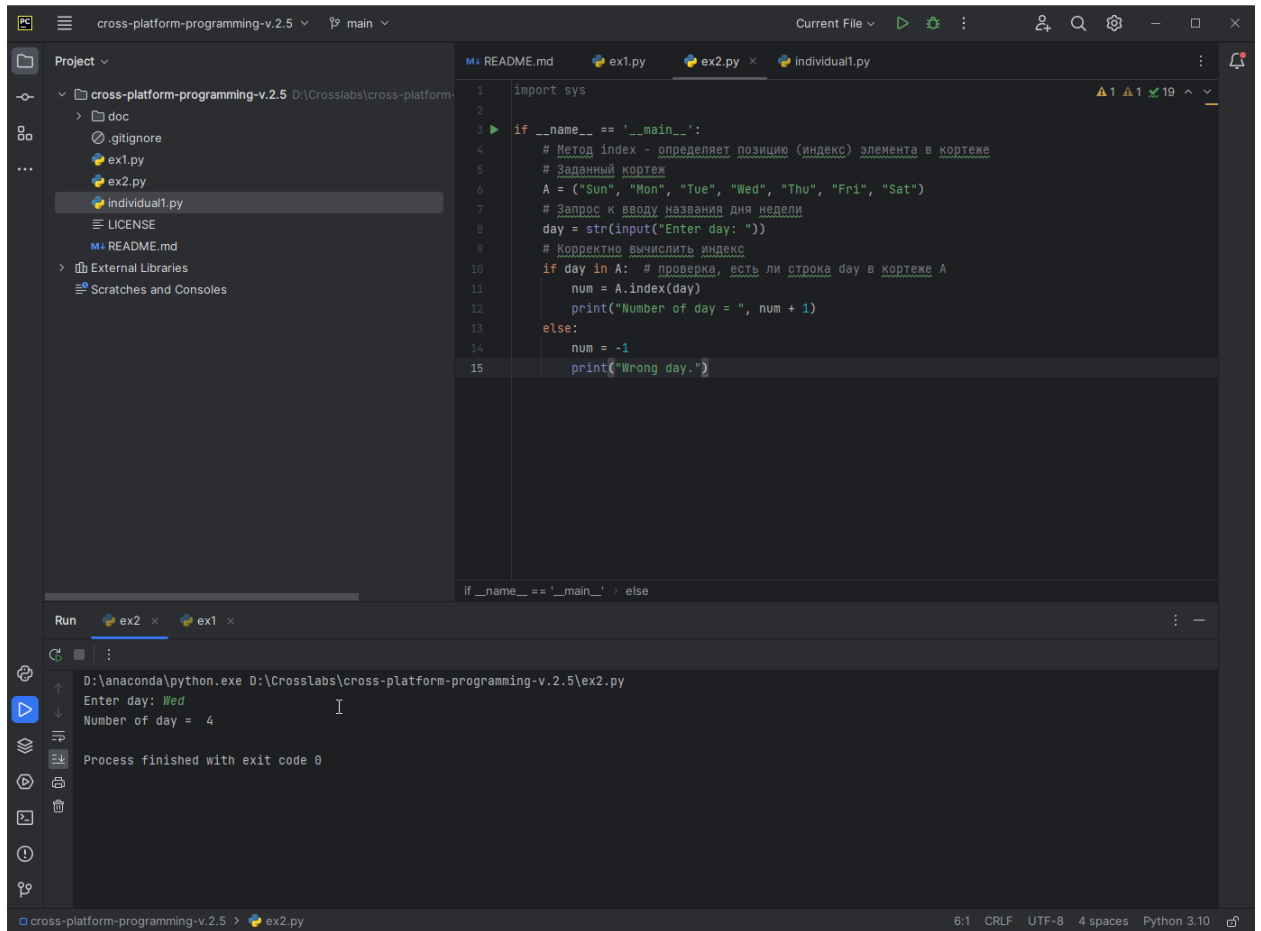
```
1 import sys
2 if __name__ == '__main__':
3     # Enter a tuple in one line.
4     A = tuple(map(int, input().split()))
5     # Check the number of elements in a tuple.
6     if len(A) != 10:
7         print("Invalid tuple size", file=sys.stderr)
8         exit(1)
9     # Find the required amount.
10    s = 0
11    for item in A:
12        if abs(item) < 5:
13            s += item
14
15    print(s)
```

Run ex2 ex1

```
D:\anaconda\python.exe D:\Crosslabs\cross-platform-programming-v.2.5\ex1.py
1 2 3 4 5 6 7 8 9 10
10
Process finished with exit code 0
```

Рис. 3. Программа и ее результат.

4. Пример №2. Условие примера: в перечне названий дней недели вычисляется порядковый номер дня.



The screenshot shows a code editor with a project named "cross-platform-programming-v.2.5". The project files include "doc", "gitignore", "ex1.py", "ex2.py", "individual1.py", "LICENSE", "README.md", "External Libraries", and "Scratches and Consoles". The active file is "ex2.py", which contains the following Python code:

```
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.")
```

The code is executed, and the output is shown in the Run console:

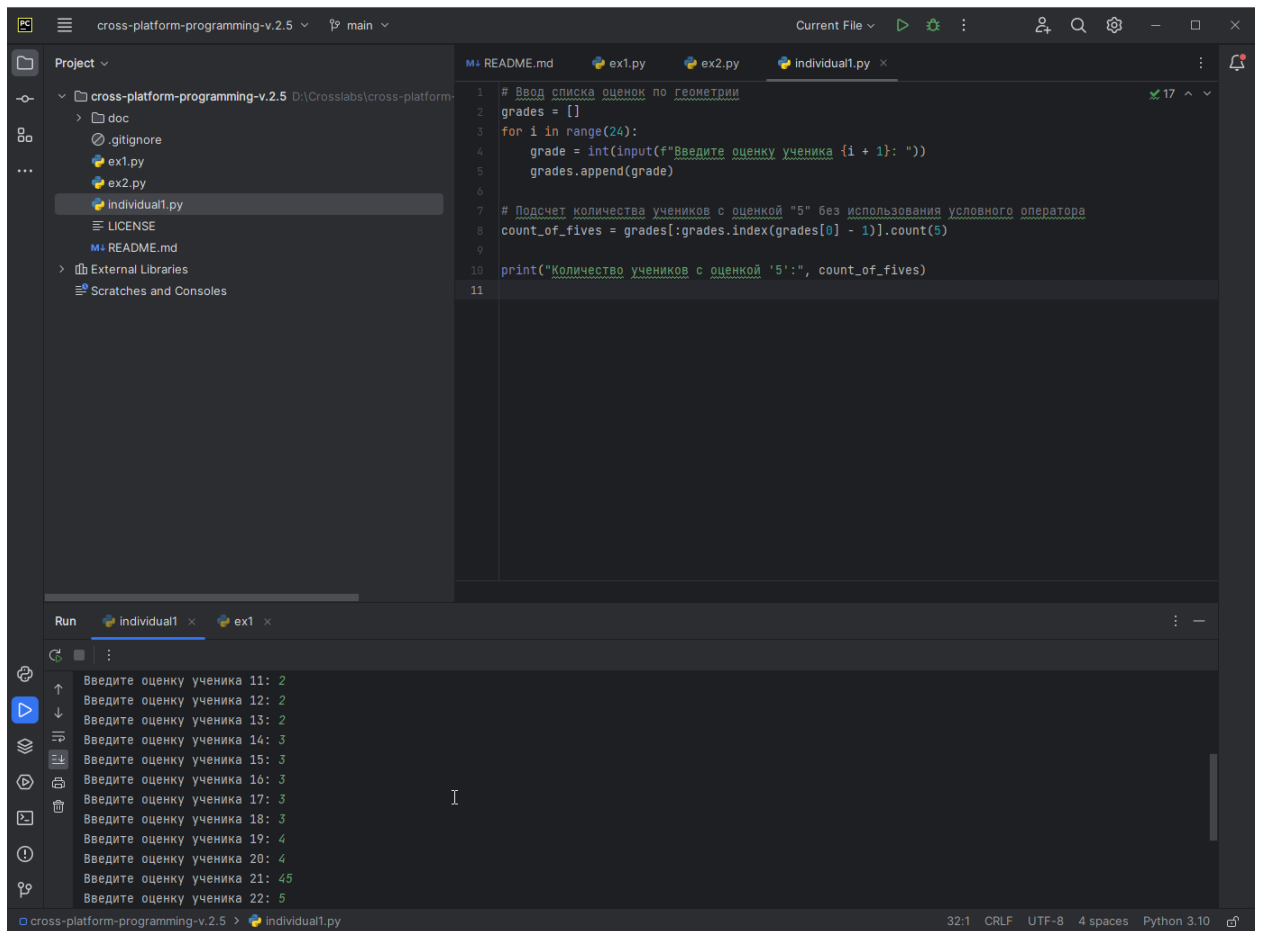
```
D:\anaconda\python.exe D:\Crosslabs\cross-platform-programming-v.2.5\ex2.py
Enter day: Wed
Number of day = 4
Process finished with exit code 0
```

Рис. 4. Программа и ее результат.

## 9. Индивидуальное задание №1.

Вариант 3 (по списку группы). Условие задания:

Известны оценки по геометрии каждого из 24 учеников класса. В начале списка перечислены все пятерки, затем все остальные оценки. Сколько учеников имеет по геометрии оценку «5»? Условный оператор не использовать.



The screenshot shows a code editor with a project named 'cross-platform-programming-v.2.5'. The file explorer on the left shows a directory structure with files like 'ex1.py', 'ex2.py', and 'individual1.py'. The main editor window displays the code for 'individual1.py'. The code is in Russian and uses a for loop to collect grades for 24 students. It then counts the number of '5' grades using the count method. The output window at the bottom shows the execution results, where the user has entered grades for 22 students, and the program has calculated the count of '5' grades as 4.

```
1 # Ввод списка оценок по геометрии
2 grades = []
3 for i in range(24):
4     grade = int(input(f"Введите оценку ученика {i + 1}: "))
5     grades.append(grade)
6
7 # Подсчет количества учеников с оценкой "5" без использования условного оператора
8 count_of_fives = grades[grades.index(grades[0] - 1)].count(5)
9
10 print("Количество учеников с оценкой '5':", count_of_fives)
11
```

Run individual1 x ex1 x

Введите оценку ученика 11: 2  
Введите оценку ученика 12: 2  
Введите оценку ученика 13: 2  
Введите оценку ученика 14: 3  
Введите оценку ученика 15: 3  
Введите оценку ученика 16: 3  
Введите оценку ученика 17: 3  
Введите оценку ученика 18: 3  
Введите оценку ученика 19: 4  
Введите оценку ученика 20: 4  
Введите оценку ученика 21: 45  
Введите оценку ученика 22: 5

32:1 CRLF UTF-8 4 spaces Python 3.10

Рис. 9. Программа и ее результат.

12. Слил ветку develop с веткой main и отправил на удаленный сервер – Github.

```
Administrator: Windows Powe x + v
PS D:\Crosslabs\cross-platform-programming-v.2.3> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
PS D:\Crosslabs\cross-platform-programming-v.2.3> git merge develop
Updating 4a104b2..4bc9e43
Fast-forward
 ex1.py      | 5  +++++
 ex2.py      | 12 ++++++++
 ex3.py      | 54 +++++++++++++++++++++++++++++++++++++++++++++++++++++
 individual1.py | 10 +++++
 individual2.py | 11 +++++
 individual3.py | 10 +++++
 6 files changed, 102 insertions(+)
 create mode 100644 ex1.py
 create mode 100644 ex2.py
 create mode 100644 ex3.py
 create mode 100644 individual1.py
 create mode 100644 individual2.py
 create mode 100644 individual3.py
PS D:\Crosslabs\cross-platform-programming-v.2.3> git push
Enumerating objects: 22, done.
Counting objects: 100% (22/22), done.
Delta compression using up to 4 threads
Compressing objects: 100% (21/21), done.
Writing objects: 100% (21/21), 2.93 KiB | 750.00 KiB/s, done.
Total 21 (delta 7), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (7/7), done.
To https://github.com/temacteklyannayapuwka/cross-platform-programming-v.2.3.git
 4a104b2..4bc9e43  main -> main
PS D:\Crosslabs\cross-platform-programming-v.2.3> |
```

Рис. 12. Слияние веток.

### 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 нескольким 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.

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. 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 приобрел навыки по работе с кортежами при написании программ с помощью языка программирования Python версии 3.x.