

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:
Худяков Максим Дмитриевич
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:

Задание 1. Создал общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python, также добавил файл .gitignore с необходимыми правилами. Клонировал свой репозиторий на свой компьютер.

```
C:\GIT>git clone https://github.com/maxim-hoodyakov/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.
```

Рисунок 1. Клонирование репозитория

Задание 2. Организовал свой репозиторий в соответствии с моделью ветвления git-flow, появилась новая ветка develop в которой буду выполнять дальнейшие задачи.

```
C:\GIT\lab2.5>git flow init

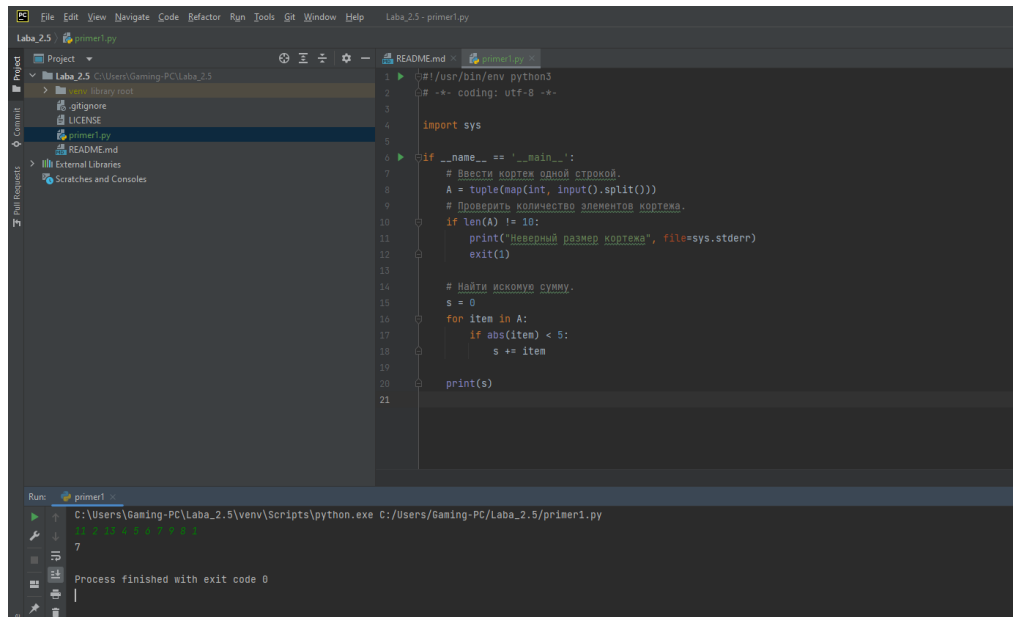
Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

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:/GIT/lab2.5/.git/hooks]
```

Рисунок 2. Модель ветвления git-flow

Задание 3. Создал проект PyCharm в папке репозитория. Приступил к работе с примером №1. Добавил новый файл primer1.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 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 import sys
5
6 if __name__ == '__main__':
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)
```

Run: primer1

C:\Users\Gaming-PC\Labo_2.5\venv\Scripts\python.exe C:\Users\Gaming-PC\Labo_2.5\primer1.py

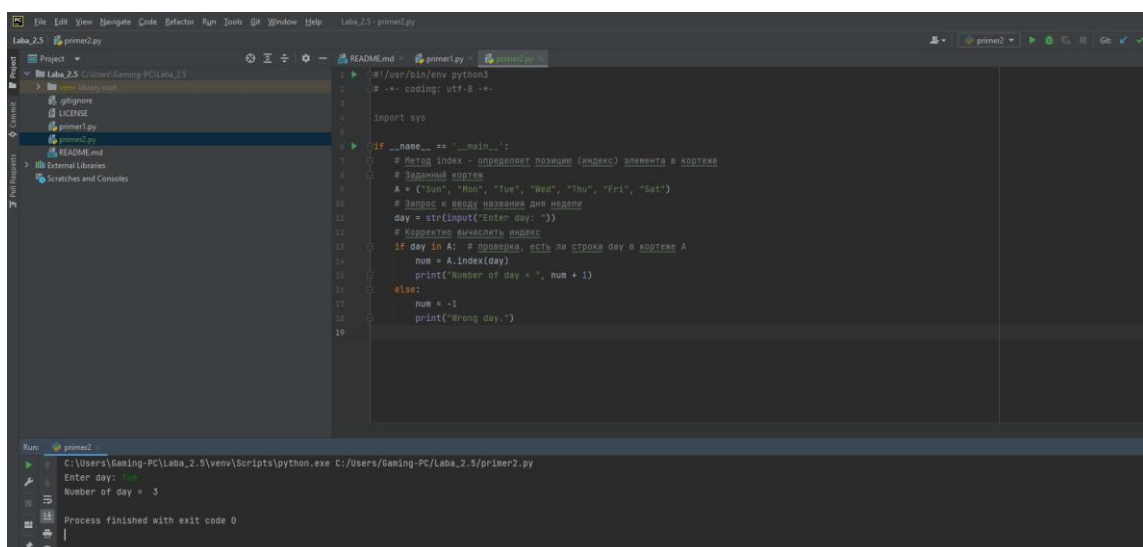
7

Process finished with exit code 0

Рисунок 3. Реализация первого примера

Задание 4. Создал новый файл под названием primer2.py. Приступил к работе с примером №2.

Условие примера: 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     # Инициализируем кортеж
9     A = ("Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat")
10    # Запрос + ввод 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.")
```

Run: primer2

C:\Users\Gaming-PC\Labo_2.5\venv\Scripts\python.exe C:\Users\Gaming-PC\Labo_2.5\primer2.py

Enter day: wed

Number of day = 3

Process finished with exit code 0

Рисунок 4. Реализация второго примера

Задание 5.

Индивидуальное задание

Вариант 9

Создал новый файл под названием individual.py.

Условие задания: если в кортеже есть хотя бы одна тройка соседних чисел, в которой средний элемент больше своих «соседей», т. е. предшествующего и последующего, то напечатать все элементы, предшествующие элементам последней из таких троек.

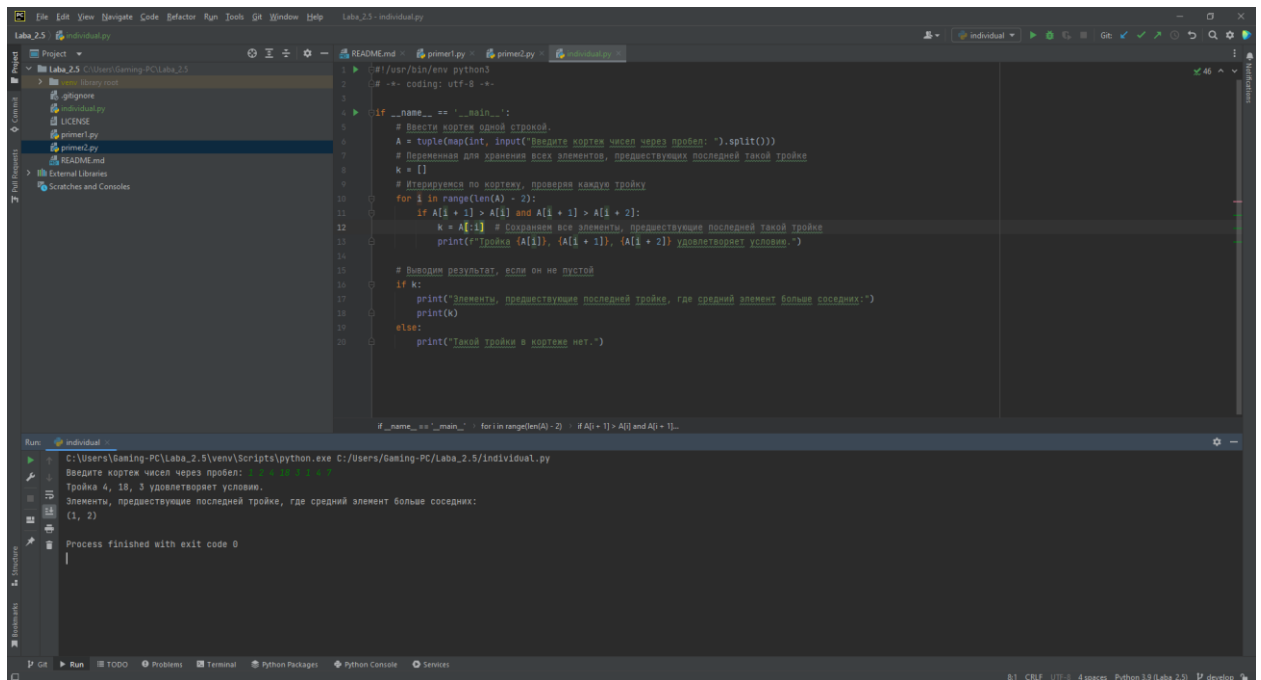


Рисунок 5. Программа индивидуального задания

Задание 6.

После выполнения работы на ветке develop, слил ее с веткой main и отправил изменения на удаленный сервер.

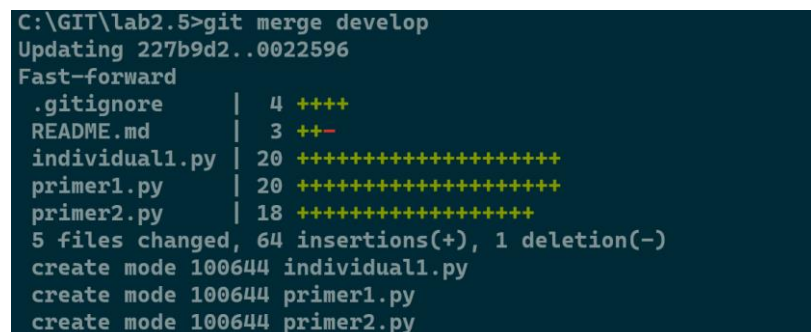


Рисунок 6. Слияние ветки develop и main

Ссылка: <https://github.com/maxim-hoodyakov/lab2.5.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: приобрел навыки по работе с кортежами при написании программ с помощью языка программирования Python версии 3.x.