

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение  
высшего образования «СЕВЕРОКАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ  
УНИВЕРСИТЕТ»

Кафедра инфокоммуникаций

Институт цифрового развития

ОТЧЁТ

по лабораторной работе №2.5

Дисциплина: «Основы кроссплатформенного программирования»

Тема: «Работа с кортежами в языке Python»

Выполнила: студентка 1 курса,  
группы ИВТ-б-о-21-1  
Диченко Дина Алексеевна

Ставрополь 2022

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

## Практическая часть:

### 1. Создала и клонировала репозиторий.

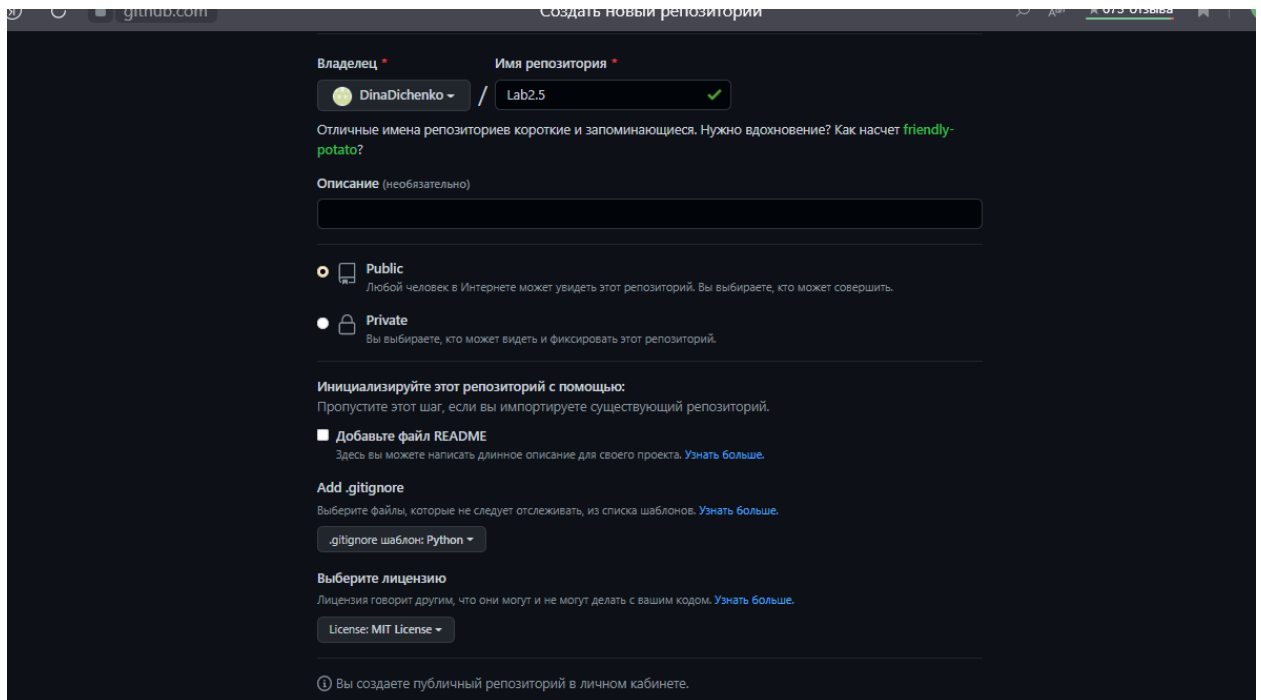


Рисунок 1. Создание репозитория

```
Microsoft Windows [Version 10.0.19043.1706]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование>git clone https://github.com/DinaDichenko/Lab2.5.git
Cloning into 'Lab2.5'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), done.

C:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование>
```

Рисунок 2. Клонирование репозитория

### 2. Изменила файл .gitignore.

```
# Создано https://www.toptal.com/developers/gitignore/api/python , pycharm
# Редактировать по адресу https://www.toptal.com/developers/gitignore?templates=python , pycharm

### PyCharm ###
# Охватывает IDE JetBrains: IntelliJ, RubyMine, PhpStorm, AppCode, PyCharm, CLion, Android Studio, WebStorm и Rider
# Ссылка: https://intellij-support.jetbrains.com/ru/en-us/articles/206544839

# Пользовательские вещи
.идея/**/workspace.xml
.идея/**/tasks.xml
.идея/**/usage.statistics.xml
.идея/**/словари
.идея/**/полка
.идея/

# AWS User-specific
.идея/**/aws.xml

# Сгенерированные файлы
.идея/**/contentModel.xml

# Чувствительные или файлы с высоким оттоком
.идея/**/Метаинформации/...
```

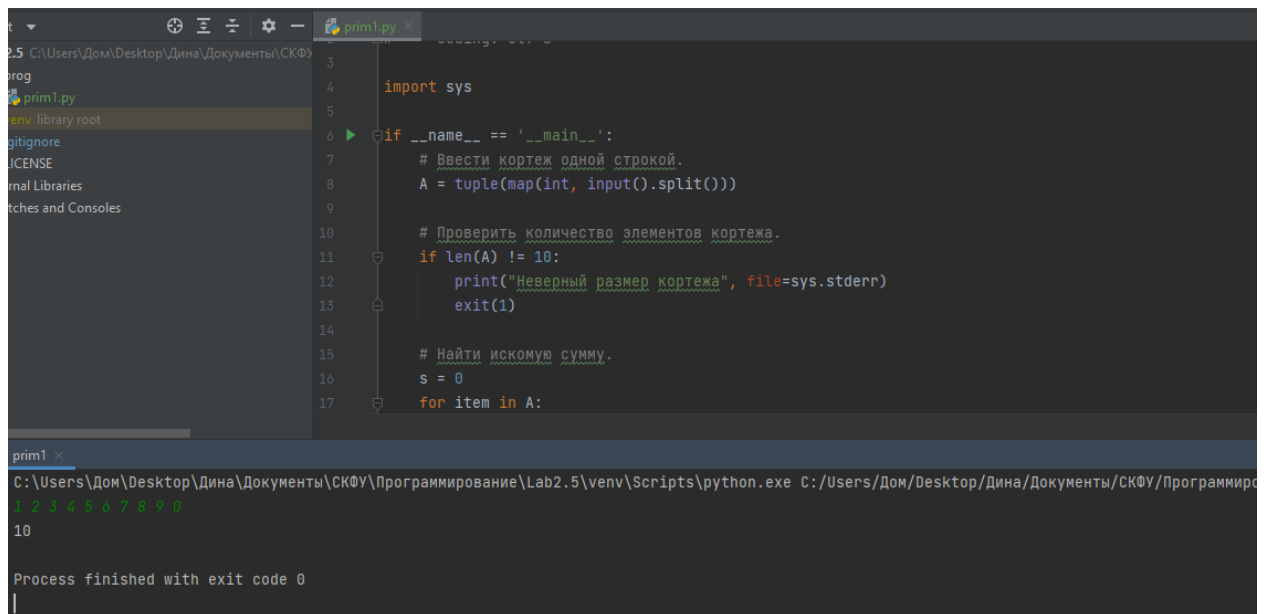
Рисунок 3. Изменение файла .gitignore

3. Организовала репозиторий в соответствии с моделью ветвления git-flow.

```
C:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование\Lab2.5>git add .
C:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование\Lab2.5>git commit -m "gitignore"
[develop acbb9a7] gitignore
1 file changed, 281 insertions(+), 129 deletions(-)
rewrite .gitignore (65%)
C:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование\Lab2.5>git flow init
Which branch should be used for bringing forth production releases?
- develop
- main
branch name for production releases: [main] develop
Which branch should be used for integration of the "next release"?
- main
branch name for "next release" development: [] main
How to name your supporting branch prefixes?
feature branches? [feature/] f
bugfix branches? [bugfix/] b
release branches? [release/] r
hotfix branches? [hotfix/] h
support branches? [support/] s
version tag prefix? [] p
hooks and filters directory? [C:/Users/Дом/Desktop/Дина/Документы/СКФУ/Программирование/Lab2.5/.git/hooks] hf
C:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование\Lab2.5>
```

Рисунок 4. Организация репозитория в соответствии с git-flow

4. Проработала примеры.



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

prim1 x

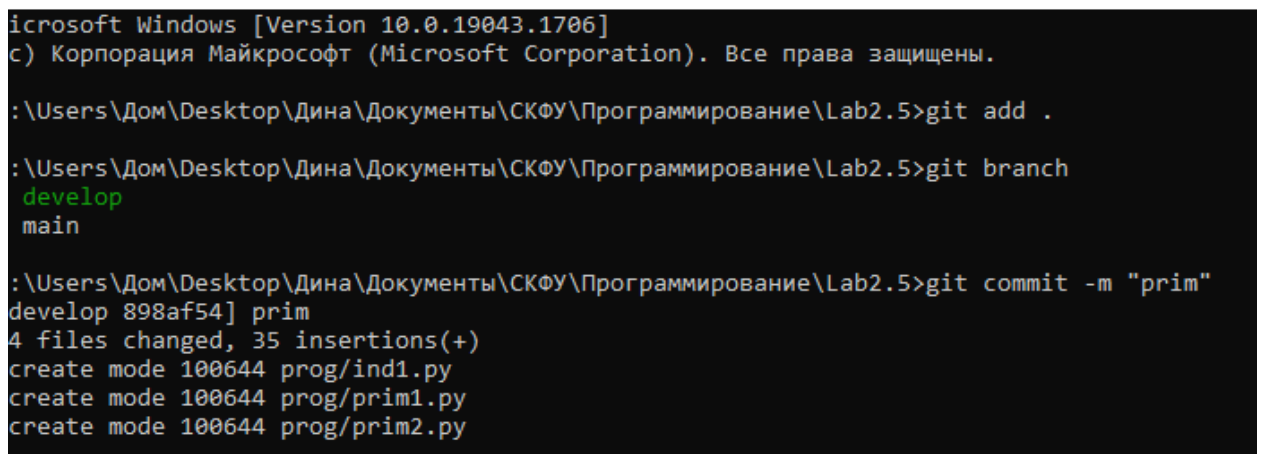
C:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование\Lab2.5\env\Scripts\python.exe C:/Users/Дом/Desktop/Дина/Документы/СКФУ/Программиро

1 2 3 4 5 6 7 8 9 0

10

Process finished with exit code 0

Рисунок 5. Проработка 1 примера



```
icrosoft Windows [Version 10.0.19043.1706]
с) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование\Lab2.5>git add .

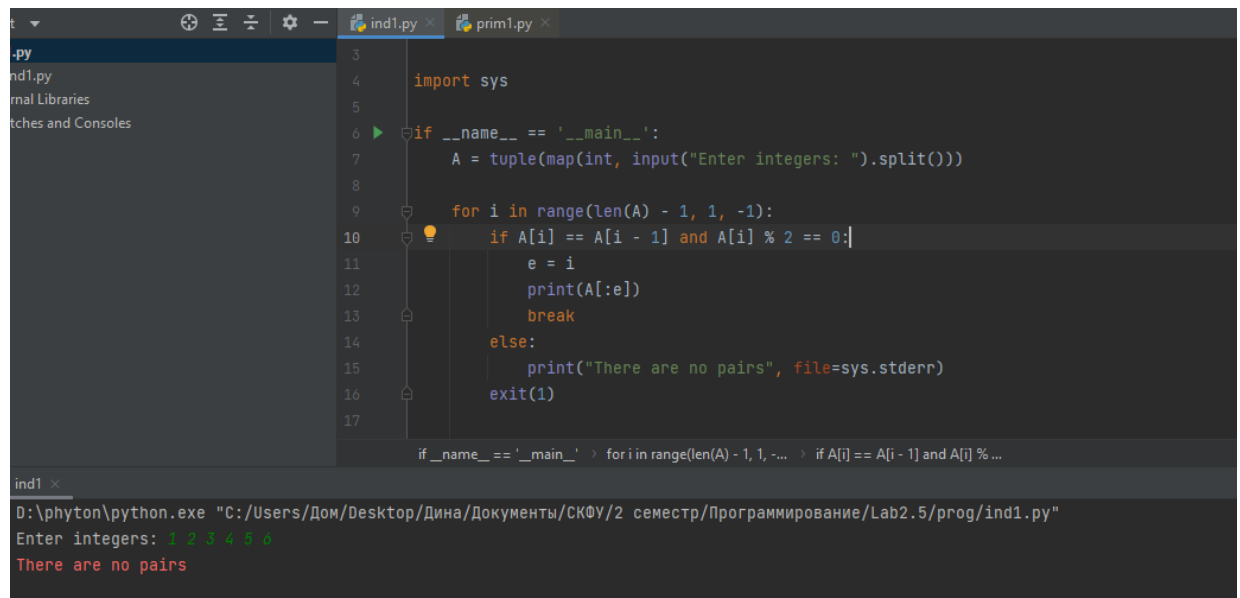
:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование\Lab2.5>git branch
develop
main

:\Users\Дом\Desktop\Дина\Документы\СКФУ\Программирование\Lab2.5>git commit -m "prim"
develop 898af54] prim
4 files changed, 35 insertions(+)
create mode 100644 prog/ind1.py
create mode 100644 prog/prim1.py
create mode 100644 prog/prim2.py
```

Рисунок 6. Сохранение изменений

## 5. Выполнила индивидуальное задание.

Дан кортеж целых чисел. Если в нем есть хотя бы одна пара соседних четных чисел, то напечатать все элементы, предшествующие элементам последней из таких пар.



```
3
4 import sys
5
6 if __name__ == '__main__':
7     A = tuple(map(int, input("Enter integers: ").split()))
8
9     for i in range(len(A) - 1, 1, -1):
10         if A[i] == A[i - 1] and A[i] % 2 == 0:
11             e = i
12             print(A[:e])
13             break
14         else:
15             print("There are no pairs", file=sys.stderr)
16             exit(1)
17
```

ind1 x

D:\phyton\python.exe "C:/Users/Дом/Desktop/Дина/Документы/СКФV/2 семестр/Программирование/Lab2.5/prog/ind1.py"

Enter integers: 1 2 3 4 5 6

There are no pairs

Рисунок 7. Результат работы индивидуального задания

## Ответы на вопросы

### 1. Что такое кортежи в языке Python?

Кортеж (tuple) – это неизменяемая структура данных, которая по своему подобию очень похожа на список.

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

Существует несколько причин, по которым стоит использовать кортежи вместо списков. Одна из них – это обезопасить данные от случайного изменения. Если мы получили откуда-то массив данных, и у нас есть желание поработать с ним, но при этом непосредственно менять данные мы не собираемся, тогда, это как раз тот случай, когда кортежи придутся как нельзя кстати.

Используя их в данной задаче, мы дополнительно получаем сразу несколько бонусов – во-первых, это экономия места. Дело в том, что кортежи в памяти занимают меньший объем по сравнению со списками.

### 3. Как осуществляется создание кортежей?

Для создания пустого кортежа можно воспользоваться следующими командами:  $a = ()$ ,  $b = \text{tuple}()$ . При желании можно воспользоваться функцией  $\text{tuple}()$ .

#### 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?

Таким способом принято получать и сразу разбирать значения, которые возвращает функция (если таковая возвращает несколько значений, конечно).

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

Используя множественное присваивание, можно провернуть интересный трюк: обмен значениями между двумя переменными.

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

С помощью операции взятия среза можно получить другой кортеж. Общая форма операции взятия среза для кортежа следующая:  $T2 = T1[i:j]$ .

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

Для кортежей можно выполнять операцию конкатенации, которая обозначается символом  $+$ . В простейшем случае для конкатенации двух кортежей общая форма операции следующая:  $T3 = T1 + T2$ .

Кортеж может быть образован путем операции повторения, обозначаемой символом  $*$ . При использовании в выражении общая форма операции следующая:  $T2 = T1 * n$ .

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

Элементы кортежа можно последовательно просмотреть с помощью операторов цикла while или for.

10.Как проверить принадлежность элемента кортежу?

С помощью операции in.

11.Какие методы работы с кортежами Вам известны?

Метод index() - поиск позиции элемента в кортеже.

Метод count() - количество вхождений элемента в кортеж.

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

Да.

13.Как создать кортеж с помощью спискового включения.

A = list(map(int, input().split())).

**Вывод:** в результате выполнения работы были приобретены навыки по работе с кортежами при написании программ с помощью языка программирования Python версии 3.x.