

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций
«Работа со строками в языке Python»**

**Отчет по лабораторной работе № 2.3
по дисциплине «Основы кроссплатформенного
программирования»**

Выполнил студент группы ИВТ-б-о-21-1

Шайдеров Дмитрий Викторович.

«20 » мая 2022г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

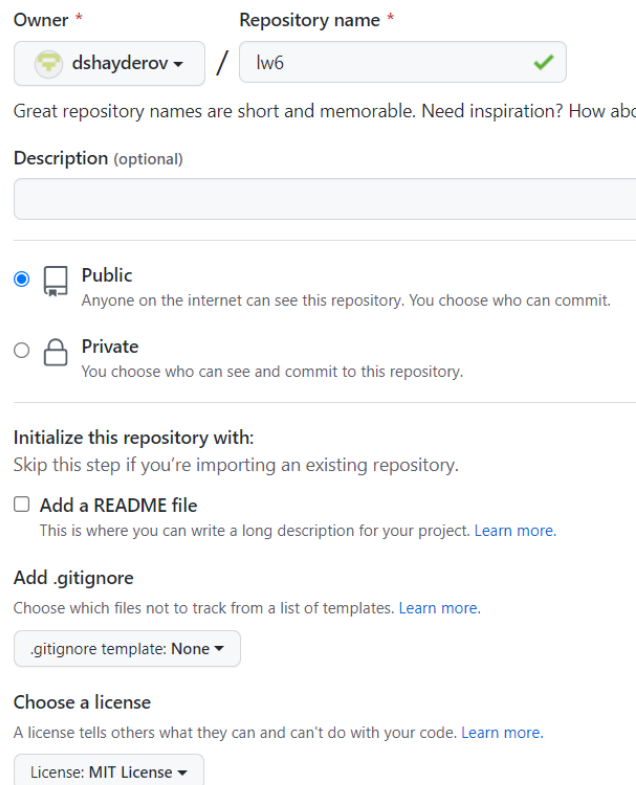
Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2022

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

Порядок выполнения работы:

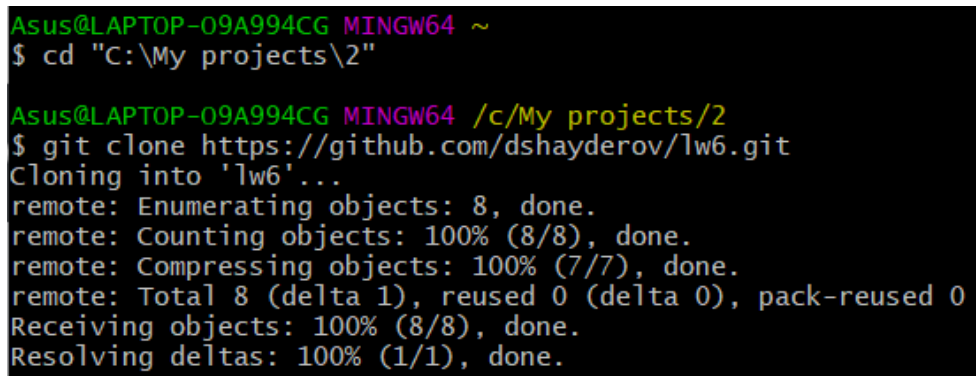
1. Создал общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python.



The screenshot shows the GitHub repository creation interface. At the top, the 'Owner' is set to 'dshayderov' and the 'Repository name' is 'lw6', with a green checkmark indicating it's valid. Below this, there's a note: 'Great repository names are short and memorable. Need inspiration? How abc'. The 'Description (optional)' field is empty. Under 'Visibility', the 'Public' option is selected with a radio button, and a note says 'Anyone on the internet can see this repository. You choose who can commit.' The 'Private' option is unselected. The 'Initialize this repository with:' section has a note: 'Skip this step if you're importing an existing repository.' There are two checkboxes: 'Add a README file' (unchecked) with a note 'This is where you can write a long description for your project. Learn more.' and 'Add .gitignore' (unchecked) with a note 'Choose which files not to track from a list of templates. Learn more.' Below these, there are two dropdown menus: '.gitignore template: None' and 'License: MIT License'.

Рисунок 1 - Создание репозитория

2. Выполните клонирование созданного репозитория.



```
Asus@LAPTOP-09A994CG MINGW64 ~  
$ cd "C:\My projects\2"  
  
Asus@LAPTOP-09A994CG MINGW64 /c/My projects/2  
$ git clone https://github.com/dshayderov/lw6.git  
Cloning into 'lw6'...  
remote: Enumerating objects: 8, done.  
remote: Counting objects: 100% (8/8), done.  
remote: Compressing objects: 100% (7/7), 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.
```

Рисунок 2 - Клонирование репозитория

3. Организуйте свой репозиторий в соответствии с моделью ветвления git-flow.

```

Asus@LAPTOP-09A994CG MINGW64 /c/My projects/2
$ cd "C:\My projects\2\lw6"

Asus@LAPTOP-09A994CG MINGW64 /c/My projects/2/lw6 (main)
$ git checkout -b develop
Switched to a new branch 'develop'

Asus@LAPTOP-09A994CG MINGW64 /c/My projects/2/lw6 (develop)
$ |

```

Рисунок 3 - Ветвление по модели git-flow

4. Создайте проект PyCharm в папке репозитория.

Имя	Дата изменения	Тип	Размер
PyCharm Project	25.05.2022 0:41	Папка с файлами	
.gitignore	25.05.2022 0:38	Файл "GITIGNORE"	2 КБ
LICENSE	25.05.2022 0:38	Файл	2 КБ
README.md	25.05.2022 0:38	Файл "MD"	1 КБ

Рисунок 4 - Проект PyCharm

5. Проработайте примеры лабораторной работы. Создайте для каждого примера отдельный модуль языка Python. Зафиксируйте изменения в репозитории.

Пример 1. Дано предложение. Все пробелы в нем заменить символом «_».

```

example_1 x
"C:\My projects\2\lw6\PyCharm Project\venv\Scripts
Введите предложение: Сегодня - воскресенье
Предложение после замены: Сегодня_-_воскресение

Process finished with exit code 0

```

Рисунок 5 – Результат выполнения программы

Пример 2. Дано слово. Если его длина нечетная, то удалить среднюю букву, в противном случае – две средние буквы.

```
example_2 x
"C:\My projects\2\lw6\PyCharm Project
Введите слово: Генератор
Генеатор

Process finished with exit code 0
```

Рисунок 6 - Результат выполнения программы

Пример 3. Дана строка текста, в котором нет начальных и конечных пробелов. Необходимо изменить ее так, чтобы длина строки стала равна заданной длине (предполагается, что требуемая длина не меньше исходной). Это следует сделать путем вставки между словами дополнительных пробелов. Количество пробелов между отдельными словами должно отличаться не более чем на 1.

```
example_3 x
"C:\My projects\2\lw6\PyCharm Project\venv
Введите предложение: Весь день шел дождь
Введите длину: 30
Весьденьшелдождь

Process finished with exit code 0
```

Рисунок 7 - Результат выполнения программы

6. Выполните индивидуальные задания (Вариант 27).

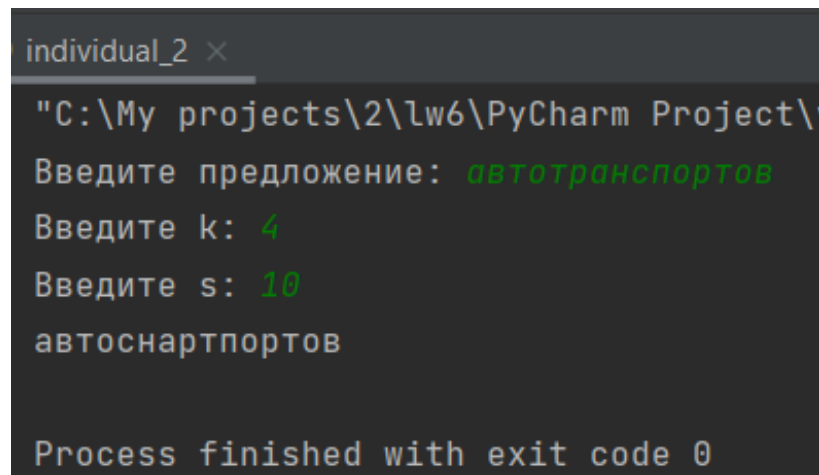
Задание 1: 27. Дано предложение. Определить, какая из букв – о или а – встречается в нем чаще (принять, что указанные буквы в строке есть).

```
individual_1 x
"C:\My projects\2\lw6\PyCharm Project\venv
Введите предложение: Море волнуется раз
Буква 'о' встречается в предложении чаще

Process finished with exit code 0
```

Рисунок 8 - Результат выполнения программы

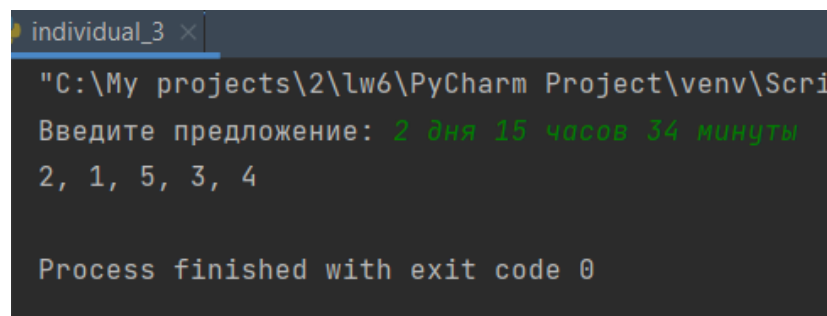
Задание 2: Дано слово из 15 букв. Переставить в обратном порядке буквы, расположенные между k-й и s-й буквами (т. е. с (k+1)-й по (s-1)-ю). Значения k и s вводятся с клавиатуры, $k < s$.



```
individual_2 x
"C:\My projects\2\lw6\PyCharm Project\venv\Scripts\python.exe"
Введите предложение: автотранспортиров
Введите k: 4
Введите s: 10
автоснартпиров
Process finished with exit code 0
```

Рисунок 9 - Результат выполнения программы

Задание 3: Дан текст. Напечатать все имеющиеся в нем цифры.

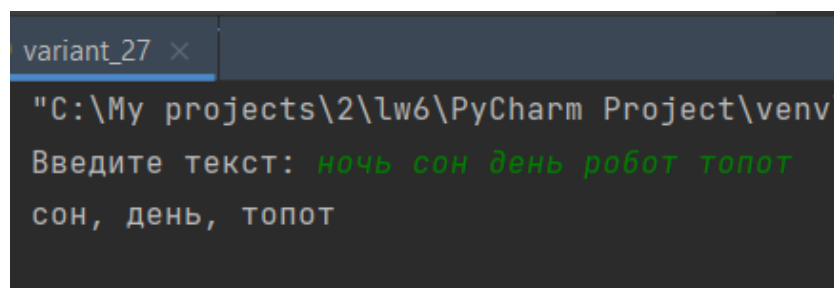


```
individual_3 x
"C:\My projects\2\lw6\PyCharm Project\venv\Scripts\python.exe"
Введите предложение: 2 дня 15 часов 34 минуты
2, 1, 5, 3, 4
Process finished with exit code 0
```

Рисунок 10 - Результат выполнения программы

Задание повышенной сложности: Дана последовательность слов. Напечатать те слова последовательности, которые отличны от первого слова и удовлетворяют следующему свойству:

- в слове нет повторяющихся букв;
- слово симметрично.



```
variant_27 x
"C:\My projects\2\lw6\PyCharm Project\venv\Scripts\python.exe"
Введите текст: ночь сон день робот топот
сон, день, топот
Process finished with exit code 0
```

Рисунок 11 - Результат выполнения программы

Вывод: приобрел навыки по работе со строками при написании программ с помощью языка программирования Python версии 3.x.