

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 №6**  
**дисциплины «Основы программной инженерии»**

Выполнил:  
Яблоновский Дмитрий Николаевич  
2 курс, группа ПИЖ-б-о-22-1,  
09.03.04 «Программная инженерия»,  
направленность (профиль) «Разработка и  
сопровождение программного  
обеспечения», очная форма обучения

---

(подпись)

Руководитель практики:  
Богданов С.С., ассистент 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.

## Порядок выполнения работы

1. Создал репозиторий GitHub.

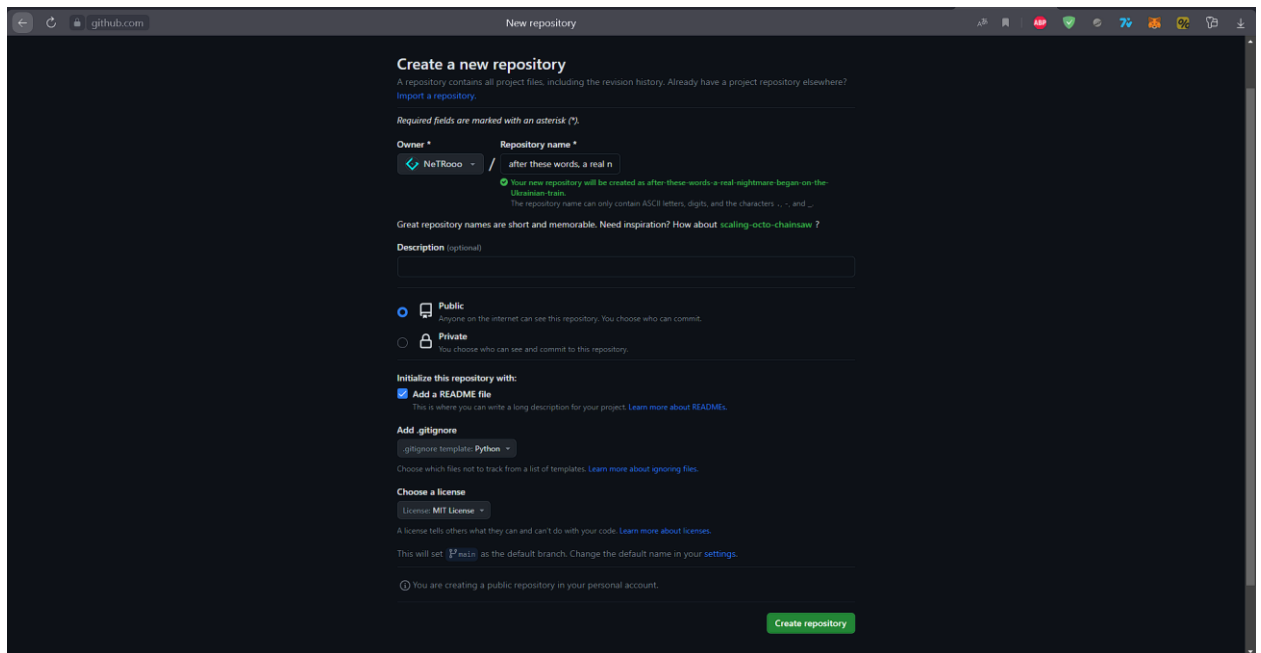


Рисунок 1- Создание репозитория

2. Проработал примеры из лабораторной работы.

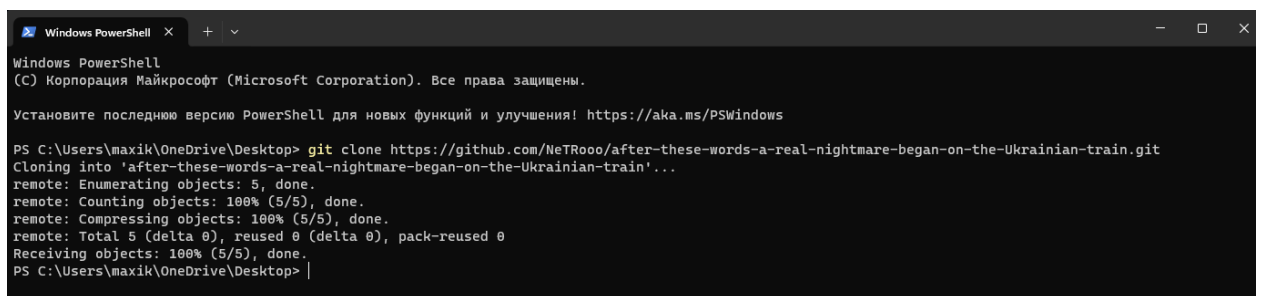


Рисунок 2 – Выполнил клонирование созданного репозитория



```
ind1.py U X
ind1.py > ...
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 def print_occurrences(sentence, target_char):
5     for index, char in enumerate(sentence):
6         if char == target_char:
7             print(f'Символ "{char}" найден в позиции {index + 1}')
8
9 if __name__ == '__main__':
10     input_sentence = input('Введите предложение: ')
11     target_character = input('Введите символ для поиска: ')
12     print_occurrences(input_sentence, target_character)
13
14
ПРОБЛЕМЫ Выходные данные КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ ПОРТЫ
PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> & C:\Users\maxik\AppData\Local\Programs\Python\Python39\python.exe c:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train\ind1.py
Введите предложение: are-began-on-the-Ukrainian-train/ind1.py
Введите символ для поиска: o
Символ "o" найден в позиции 11
Символ "o" найден в позиции 18
Символ "o" найден в позиции 47
Символ "o" найден в позиции 68
Символ "o" найден в позиции 92
Символ "o" найден в позиции 97
Символ "o" найден в позиции 102
Символ "o" найден в позиции 119
Символ "o" найден в позиции 128
Символ "o" найден в позиции 131
Символ "o" найден в позиции 159
PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train>
```

Рисунок 6 – Дано предложение. Составить программу, которая печатает «столбиком» все вхождения в предложение некоторого символа. (инд. задание №1)

```
ind1.py U ind2.py U X
ind2.py > ...
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 def check_and_correct(sequence):
5     words = sequence.split()
6
7     for i in range(len(words)):
8         word = words[i]
9
10        if "ua" in word:
11            corrected_word = word.replace("ча", "ча")
12            words[i] = corrected_word
13        elif "ua" in word:
14            corrected_word = word.replace("щя", "ща")
15            words[i] = corrected_word
16
17    corrected_sequence = ' '.join(words)
18    return corrected_sequence
19
20 if __name__ == '__main__':
21     sequence = input('Введите последовательность: ')
22     corrected_sequence = check_and_correct(sequence)
23
24     print("Исходная последовательность:", sequence)
25     print("Исправленная последовательность:", corrected_sequence)
26
ПРОБЛЕМЫ Выходные данные КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ ПОРТЫ
PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> & C:\Users\maxik\AppData\Local\Programs\Python\Python39\python.exe c:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train\ind2.py
Введите последовательность: часть шивель чары чива
Исходная последовательность: часть шивель чары чива
Исправленная последовательность: часть шивель чары чива
PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train>
```

Рисунок 7 – Дана последовательность слов. Проверить, правильно ли в ней записаны буквосочетания ча и ща. Исправить ошибки. (инд. задание №2)

```
ind3.py U ind2.py U ind3.py U X
ind3.py > ind3.py
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 def insert_letter_before_last_i(sentence, letter_to_insert):
5     last_i_index = sentence.rfind("и")
6
7     modified_sentence = sentence[:last_i_index] + letter_to_insert + sentence[last_i_index:]
8
9     return modified_sentence
10
11 if __name__ == '__main__':
12     original_sentence = input("Введите предложение: ")
13     letter_to_insert = "X"
14     result_sentence = insert_letter_before_last_i(original_sentence, letter_to_insert)
15
16     print(result_sentence)
17
```

ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ КОНСОЛЬ ОТЛАДКИ **ТЕРМИНАЛ** ПОРТЫ

PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> & C:/Users/maxik/AppData/Local/Programs/Python/Python39/python.exe c:/Users/maxik/OneDrive/Desktop/after-these-wo  
rd5-a-real-nightmare-began-on-the-Ukrainian-train/ind3.py  
Введите предложение: Это предложение оканчивается символом «и».  
Это предложение оканчивается символом «и».  
PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train>

Рисунок 8 – Дано предложение, оканчивающееся символом «.». Вставить заданную букву перед последней буквой и. (инд. задание №3)

```
ind4.py U X ind1.py
ind4.py > ...
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 def common_letters(word1, word2, word3):
5     unique_letters_word1 = set(word1)
6     unique_letters_word2 = set(word2)
7     unique_letters_word3 = set(word3)
8
9     common = unique_letters_word1.intersection(unique_letters_word2, unique_letters_word3)
10
11     print("Общие буквы:", common)
12
13 if __name__ == '__main__':
14     word1 = input("Введите первое слово: ")
15     word2 = input("Введите второе слово: ")
16     word3 = input("Введите третье слово: ")
17
18     common_letters(word1, word2, word3)

```

ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ КОНСОЛЬ ОТЛАДКИ **ТЕРМИНАЛ** ПОРТЫ

PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> & C:/Users/maxik/AppData/Local/Programs/Python/Python39/python.exe c:/Users/maxik/OneDrive/Desktop/after-these-words-a-real-nightma  
re-began-on-the-Ukrainian-train/ind4.py  
Введите первое слово: Школа  
Введите второе слово: Школьник  
Введите третье слово: Уроки  
Общие буквы: {'и', 'о'}

Рисунок 9 – Даны три слова. Напечатать их общие буквы. Повторяющиеся буквы каждого слова не рассматривать. (задание повышенной сложности)

```
ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ ПОРТЫ
PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> git add .
PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> git commit -m "Who's here?"
[dev cb6cca8] who's here?
6 files changed, 117 insertions(+)
create mode 100644 ind1.py
create mode 100644 ind2.py
create mode 100644 ind3.py
create mode 100644 task1.py
create mode 100644 task2.py
create mode 100644 task3.py
PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> git merge dev
Updating fd1a849..cb6cca8
Fast-forward
 ind1.py | 13 ++++++++
 ind2.py | 25 ++++++++
 ind3.py | 16 ++++++++
 task1.py | 6 +++++
 task2.py | 12 ++++++++
 task3.py | 45 ++++++++
 6 files changed, 117 insertions(+)
create mode 100644 ind1.py
create mode 100644 ind2.py
create mode 100644 ind3.py
create mode 100644 task1.py
create mode 100644 task2.py
create mode 100644 task3.py
```

Рисунок 10 – Создание коммита

```
ПРОБЛЕМЫ  ВЫХОДНЫЕ ДАННЫЕ  КОНСОЛЬ ОТЛАДКИ  ТЕРМИНАЛ  ПОРТЫ

● PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> git push
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 2.44 KiB | 2.44 MiB/s, done.
Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/NeTRooo/after-these-words-a-real-nightmare-began-on-the-Ukrainian-train.git
fd1a849..cb6cca8 dev -> dev
● PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)
● PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> git push
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/NeTRooo/after-these-words-a-real-nightmare-began-on-the-Ukrainian-train.git
fd1a849..cb6cca8 main -> main
○ PS C:\Users\maxik\OneDrive\Desktop\after-these-words-a-real-nightmare-began-on-the-Ukrainian-train> |
```

Рисунок 11 – Отправка коммита

Ссылка на репозиторий: <https://github.com/NeTRooo/after-these-words-a-real-nightmare-began-on-the-Ukrainian-train/tree/main>

## Ответы на контрольные вопросы

1. Что такое строки в языке Python?

Строки в языке Python представляют собой последовательность символов и являются неизменяемым типом данных.

2. Какие существуют способы задания строковых литералов в языке Python?

Способы задания строковых литералов: в одинарных, двойных или тройных кавычках.

3. Какие операции и функции существуют для строк?

Операции и функции для строк: конкатенация (+), умножение (\*), len(), upper(), lower(), find(), replace(), и другие.

4. Как осуществляется индексирование строк?

Индексирование строк осуществляется с использованием квадратных скобок, например, str[0] для первого символа.

5. Как осуществляется работа со срезами для строк?

Работа со срезами осуществляется через указание начального, конечного индексов и шага: str[start:stop:step].

6. Почему строки Python относятся к неизменяемому типу данных?

Строки относятся к неизменяемым типам данных, потому что после создания строку нельзя изменить.

7. Как проверить то, что каждое слово в строке начинается с заглавной буквы?

Можно воспользоваться методом istitle().

8. Как проверить строку на вхождение в неё другой строки?

Используется оператор in: "подстрока" in "строка".

9. Как найти индекс первого вхождения подстроки в строку?

Индекс первого вхождения подстроки в строку можно найти с помощью метода find().

10. Как подсчитать количество символов в строке.

Для подсчета количества символов в строке используется функция len()

11. Как подсчитать то, сколько раз определённый символ встречается в строке?

Методом `count()`.

12. Что такое f-строки и как ими пользоваться?

F-строки — это способ форматирования строк, позволяющий встраивать значения переменных в строки.

13. . Как найти подстроку в заданной части строки?

Для поиска подстроки в заданной части строки используется метод `find()` или оператор `in`.

14. Как вставить содержимое переменной в строку, воспользовавшись методом `format()`?

"строка {0}".format(x).

15. Как узнать о том, что в строке содержатся только цифры?

Метод `isdigit()`.

16. Как разделить строку по заданному символу?

`split()`.

17. Как проверить строку на то, что она составлена только из строчных букв?

Метод `islower()`.

18. Как проверить то, что строка начинается со строчной буквы?

Чтобы проверить, что строка начинается со строчной буквы, используйте метод `islower()`.

19. Можно ли в Python прибавить целое число к строке?

Нельзя прибавить целое число к строке напрямую, нужно сначала преобразовать число в строку.

20. Как «перевернуть» строку?

Для "переворота" строки можно использовать срез с шагом -1: `str[::-1]`.

21. Как объединить список строк в одну строку, элементы которой разделены дефисами?

Метод `join()`.



22. Как привести всю строку к верхнему или нижнему регистру?

Методы `upper()` и `lower()`.

23. Как преобразовать первый и последний символы строки к верхнему регистру?

Методы `capitalize()` и `title()`.

24. Как проверить строку на то, что она составлена только из прописных букв?

Метод `isupper()`.

25. В какой ситуации вы воспользовались бы методом `splitlines()` ?

Метод `splitlines()` используется, чтобы разделить строку на список строк по символам новой строки.

26. Как в заданной строке заменить на что-либо все вхождения некоей подстроки?

Метод `replace()`.

27. Как проверить то, что строка начинается с заданной последовательности символов, или заканчивается заданной последовательностью символов?

Методы `startswith()` и `endswith()`.

28. Как узнать о том, что строка включает в себя только пробелы?

Метод `isspace()`.

29. Что случится, если умножить некую строку на 3?

Умножение строки на число создает новую строку, повторенную указанное количество раз.

30. Как привести к верхнему регистру первый символ каждого слова в строке?

Метод `title()`.

31. Как пользоваться методом `partition()` ?

Метод `partition()` используется для деления строки на три части по первому вхождению заданного разделителя.

32. В каких ситуациях пользуются методом `rfind()` ?

Метод `rfind()` используется для поиска последнего вхождения подстроки в строку.