

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт перспективной инженерии  
Департамент цифровых, робототехнических систем и электроники

**ОТЧЕТ**  
**ПО ЛАБОРАТОРНОЙ РАБОТЕ №6**  
**дисциплины**  
**«Основы кроссплатформенного программирования»**

Выполнил:  
Сопов Максим Игоревич  
2 курс, группа ИТС-б-о-23-1,  
11.03.02 «Инфокоммуникационные  
технологии и системы связи»,  
очная форма обучения

---

(подпись)

Проверил:  
Доцент департамента цифровых,  
робототехнических систем и  
электроники Воронкин Р.А.

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2024 г.

## ТЕМА: РАБОТА СО СТРОКАМИ В ЯЗЫКЕ PYTHON

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

### Порядок выполнения работы:

1. Изучил теоретический материал.
2. Приступил к выполнению заданий.
3. Создание репозитория.

**Ссылка на GitHub:** <https://github.com/MaxITS-kurwa/lb6>

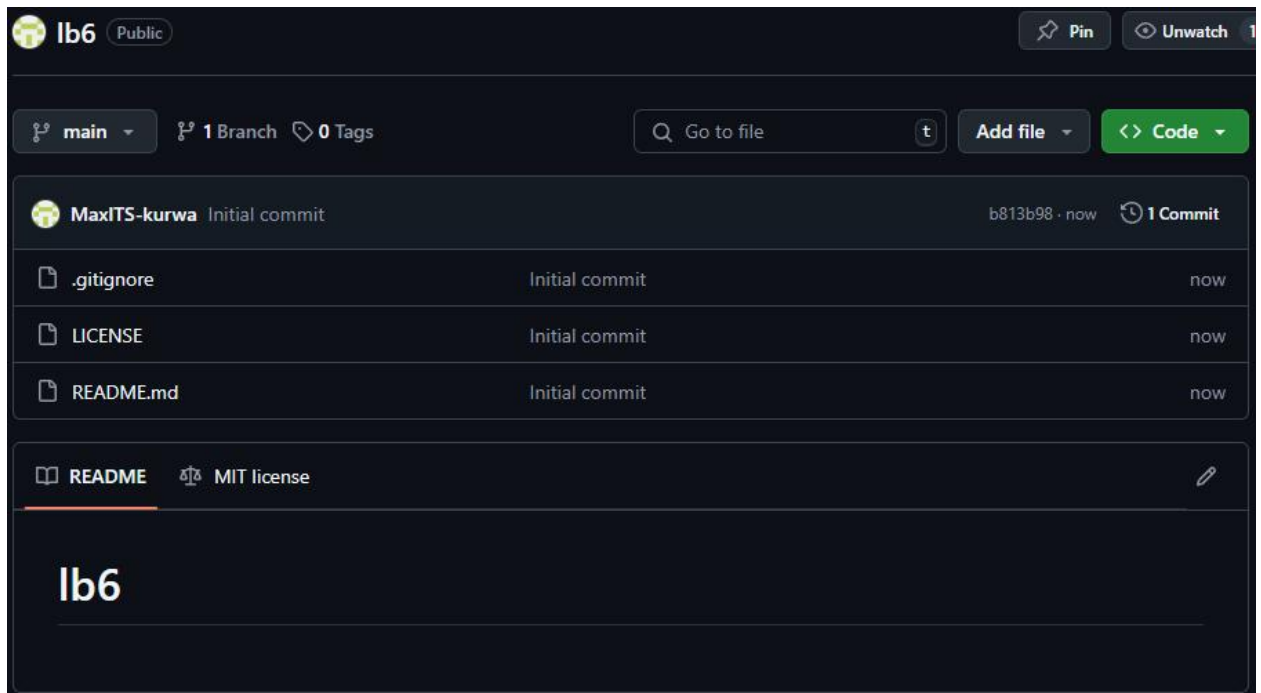


Рисунок 1. Репозиторий

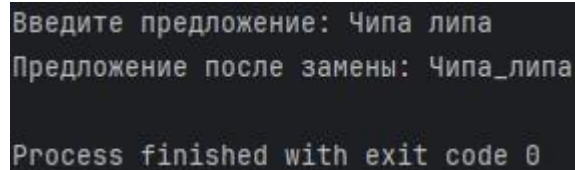
### 4. Клонирование репозитория.

```
C:\Users\sopov>git clone https://github.com/MaxITS-kurwa/lb6.git
Cloning into 'lb6'...
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 (from 0)
Receiving objects: 100% (5/5), done.
```

Рисунок 2. Клонирование

5. Приступил к выполнению примеров.

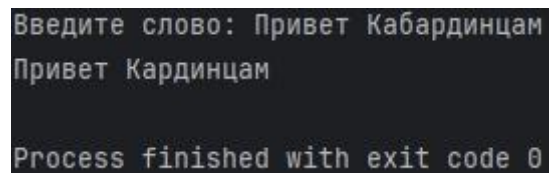
```
if __name__ == '__main__':  
    s = input("Введите предложение: ")  
    r = s.replace(' ', '_')  
    print("Предложение после замены:", r)
```



```
Введите предложение: Чипа липа  
Предложение после замены: Чипа_липа  
  
Process finished with exit code 0
```

Рисунок 3. Пример 1.

```
if __name__ == '__main__':  
    word = input("Введите слово: ")  
    idx = len(word) // 2  
    if len(word) % 2 == 1:  
        r = word[:idx] + word[idx+1:]  
    else:  
        r = word[:idx-1] + word[idx+1:]  
    print(r)
```



```
Введите слово: Привет Кабардинцам  
Привет Кардинцам  
  
Process finished with exit code 0
```

Рисунок 4. Пример 2.

```
import sys
```

```
if __name__ == "__main__":  
    s = input("Введите предложение: ")  
    n = int(input("Введите длину: "))
```

```
if len(s) >= n:  
    print("Заданная длина должна быть больше длины предложения", file=sys.stderr)  
    exit(1)
```

```
words = s.split()
```

```
if len(words) < 2:  
    print("Предложение должно содержать несколько слов", file=sys.stderr)  
    exit(1)
```

```
delta = n - len(s)
```

```
for word in words:  
    delta -= len(word)
```

```
w, r = divmod(delta, len(words) - 1)
```

```
lst = []
```

```
for i, word in enumerate(words):  
    lst.append(word)
```

```
if i < len(words) - 1:  
    width = w
```

```
if r > 0:  
    width += 1  
r -= 1
```

```
if width > 0:  
    lst.append(' ' * width)
```

```
print("".join(lst))
```

```
Введите предложение: Привет Кабардинцам  
Введите длину: 20  
  
Process finished with exit code 1
```

Рисунок 5. Пример 3.

6. Зафиксировал изменения в репозитории.

```

C:\Users\sopov>cd lb6
C:\Users\sopov\lb6>git add .
C:\Users\sopov\lb6>git commit -m "Добавил примеры"
[main 2616959] Добавил примеры
3 files changed, 51 insertions(+)
create mode 100644 1.py
create mode 100644 2.py
create mode 100644 3.py
C:\Users\sopov\lb6>git push
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 1.09 kiB | 558.00 kiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/MaxITS-kurwa/lb6.git
b813b98..2616959 main -> main

```

Рисунок 6. Изменения

7. Привел скриншоты результатов каждой из программы.

8. Приступил к выполнению индивидуального задания.

```

def calculate_percentage_of_a(sentence):
    total_letters = sum(1 for char in sentence if char.isalpha())
    count_a = sentence.lower().count('a') # Количество букв "a"

    return (count_a / total_letters) * 100 if total_letters > 0 else 0

```

```
sentence = input("Введите предложение: ")
```

```

result = calculate_percentage_of_a(sentence)
print(f'Доля букв 'a' в предложении: {result:.2f}%')

```

```

Введите предложение: Максим Игоревич
Доля букв 'a' в предложении: 7.14%

Process finished with exit code 0

```

Рисунок 7. Задание 1

```
sentence = input("Введите предложение: ")
```

```
modified_sentence = sentence.replace('да', 'не').replace('Да', 'Не')
```

```
print("Измененное предложение:", modified_sentence)
```

```
Введите предложение: да пикачу
Измененное предложение: не пикачу
Process finished with exit code 0
```

Рисунок 10. Задание 2

```
word = input("Введите слово: ")
```

```
s = int(input("Введите индекс s (s < k): "))
```

```
k = int(input("Введите индекс k (k > s): "))
```

```
modified_word = word[:s] + word[s+1:k] + word[s] + word[k+1:]
```

```
print("Измененное слово:", modified_word)
```

```
Введите слово: пикачу
Введите индекс s (s < k): 2
Введите индекс k (k > s): 3
Измененное слово: пикчу
Process finished with exit code 0
```

Рисунок 11. Задание 3

9. Зафиксировал изменения в репозитории.

```
C:\Users\sopov\lb6>git add .
C:\Users\sopov\lb6>git commit -m "Добавил файлы с заданиями"
[main f344733] Добавил файлы с заданиями
3 files changed, 24 insertions(+)
create mode 100644 z1.py
create mode 100644 z2.py
create mode 100644 z3.py
C:\Users\sopov\lb6>git push
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 956 bytes | 478.00 KiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/MaxITS-kurwa/lb6.git
2616959..f344733 main -> main
```

Рисунок 12. Изменения

## **Ответы на контрольные вопросы:**

### **1. Что такое строки в языке Python?**

Строки в Python — строки в языке Python это последовательности символов, используемые для хранения текстовой информации. Они могут содержать буквы, цифры, пробелы и специальные символы.

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

Строки можно задавать с помощью одинарных, двойных, тройных одинарных и тройных двойных кавычек.

### **3. Какие операции и функции существуют для строк?**

Операции включают конкатенацию, повторение, а также функции, такие как `upper`, `lower`, `replace`, `split`.

### **4. Как осуществляется индексирование строк?**

Индексирование строк осуществляется с помощью квадратных скобок. Индексы начинаются с 0 для первого символа, -1 для последнего и тд.

### **5. Как осуществляется работа со срезами для строк?**

Срезы позволяют извлекать подстроки. Синтаксис: `s` - start - начальный индекс, а `end` - конечный индекс.

### **6. Почему строки Python относятся к неизменяемому типу данных?**

Строки являются неизменяемыми, потому что после создания их содержимое нельзя изменить. Любые операции, которые изменяют строку, создают новую строку.

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

`istitle`, `s.istitle`.

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

Оператор `in`: `substring in s`.

### **9. Как найти индекс первого вхождения подстроки в строку?**

Через `find`: `s.find`.

### **10. Как подсчитать количество символов в строке?**

Через функцию len: len(s).

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

Через метод count.

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

Форматированные строки позволяют вставлять выражения в строку.

**13. Как найти подстроку в заданной части строки?**

Через метод find(start, end)

**14. Как вставить содержимое переменной в строку, воспользовавшись методом format?**

Через format: "Пример, {}".format(name).

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

Через isdigit

**16. Как разделить строку по заданному символу?**

Через метод split(,).

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

Через метод islower: s.islower.

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

Через islower для первого символа.

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

Нет. Строки и числа нельзя складывать напрямую.

**20. Как «перевернуть» строку?**

Через срез

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

через join

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

через методы upper и lower



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

Через upper

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

Через метод isupper

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

Метод splitlines() полезен для разделения строки на строки по символам новой строки.

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

Через метод replace.

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

Через методы startswith() и endswith()

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

Используйте метод isspace: s.isspace.

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

Строка будет повторена трижды.

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

Через метод title: s.title.

**31. Как пользоваться методом partition()?**

Метод partition() разделяет строку на три части: до, разделитель и после.

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

Метод rfind() используется для поиска последнего вхождения подстроки в строке

**Вывод:** в ходе работы исследовал базовые возможности системы контроля версий Git для работы с локальными репозиториями.