

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 №2.3 (6)
дисциплины
«Программирование на Python»

Выполнил:
Ибрагимов Муса Айнудинович
2 курс, группа ИВТ-6-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», очная
форма обучения

(подпись)

Руководитель:
Воронкин Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

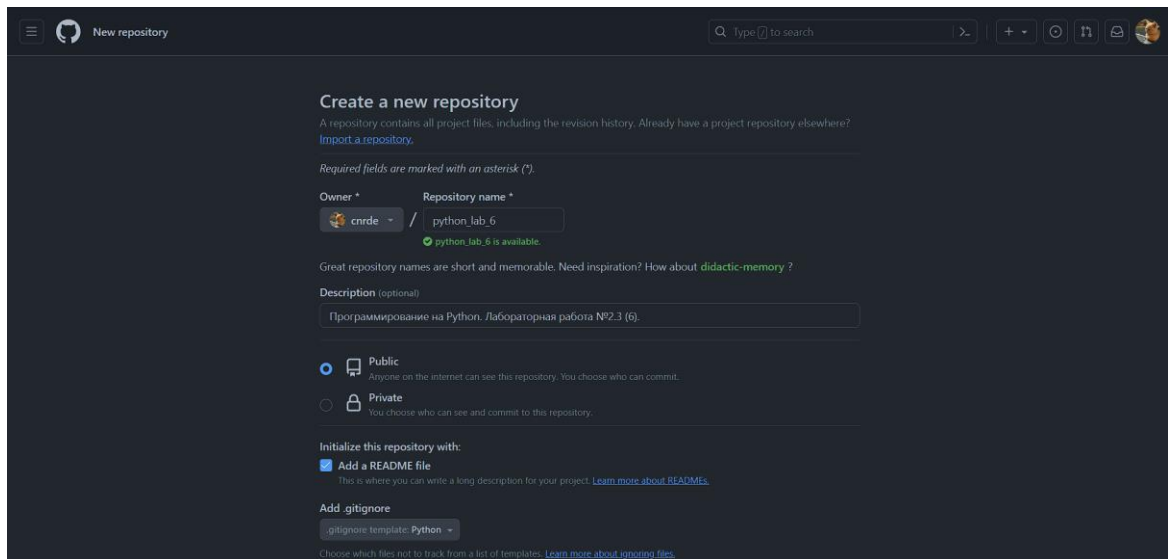
Ставрополь, 2024 г.

Тема: Работа со строками в языке Python.

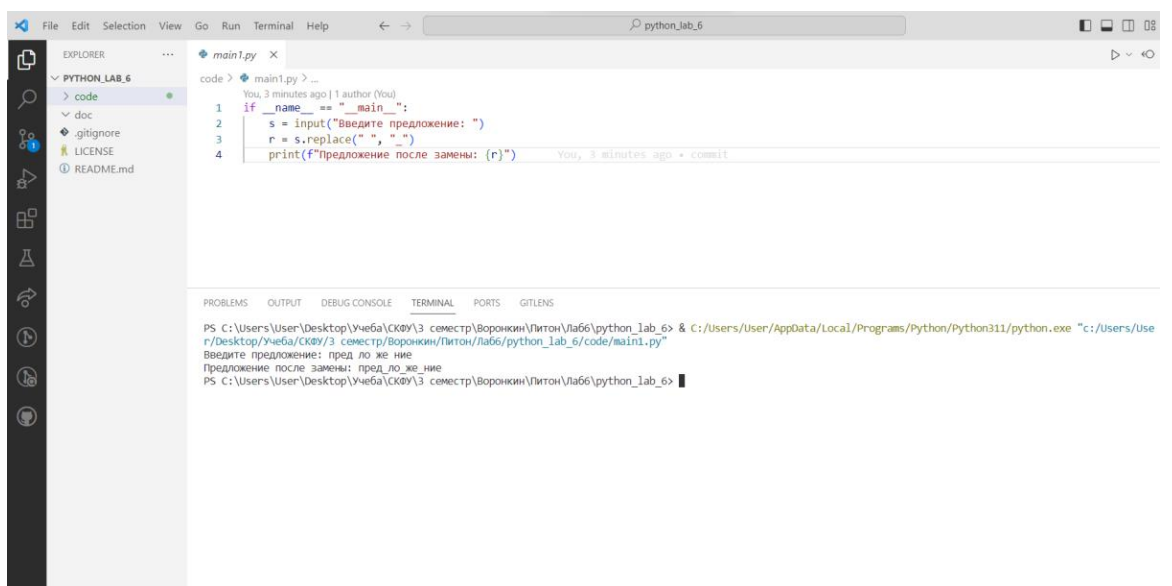
Цель: Приобретение навыков по работе со строками при написании программ с помощью языка программирования Python версии 3.x.

Практическая часть:

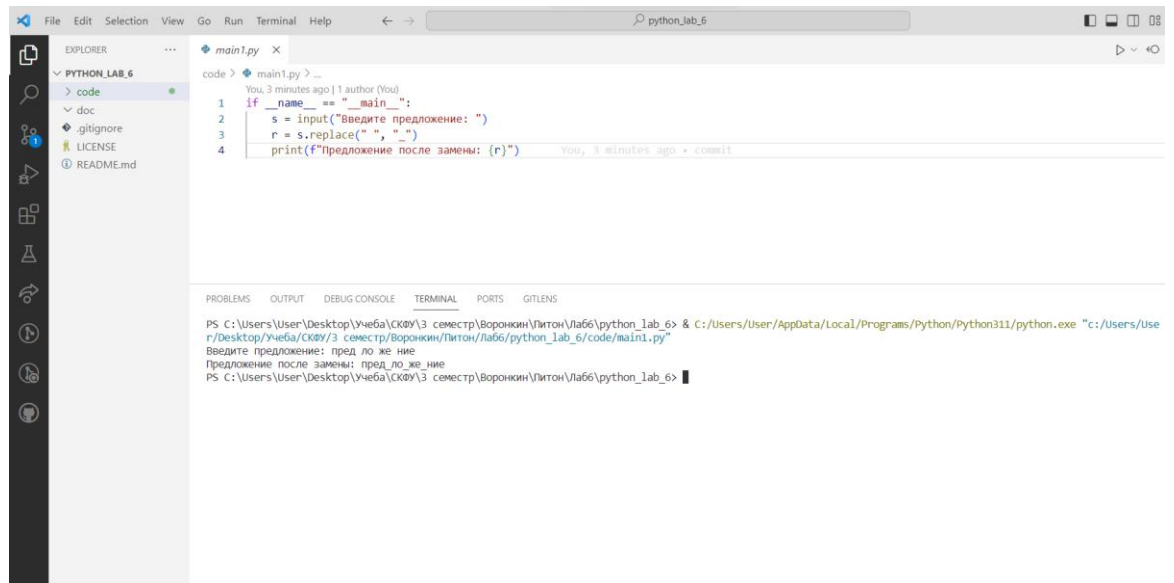
1. Изучил теоретический материал работы.
2. Создал общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python.



3. Проработал первый пример из лабораторной работы: Дано предложение. Все пробелы в нем заменить символом “_”.



4. Проработал второй пример из лабораторной работы: Дано слово. Если его длина нечетная, то удалить среднюю букву, в противном случае — две средние буквы.



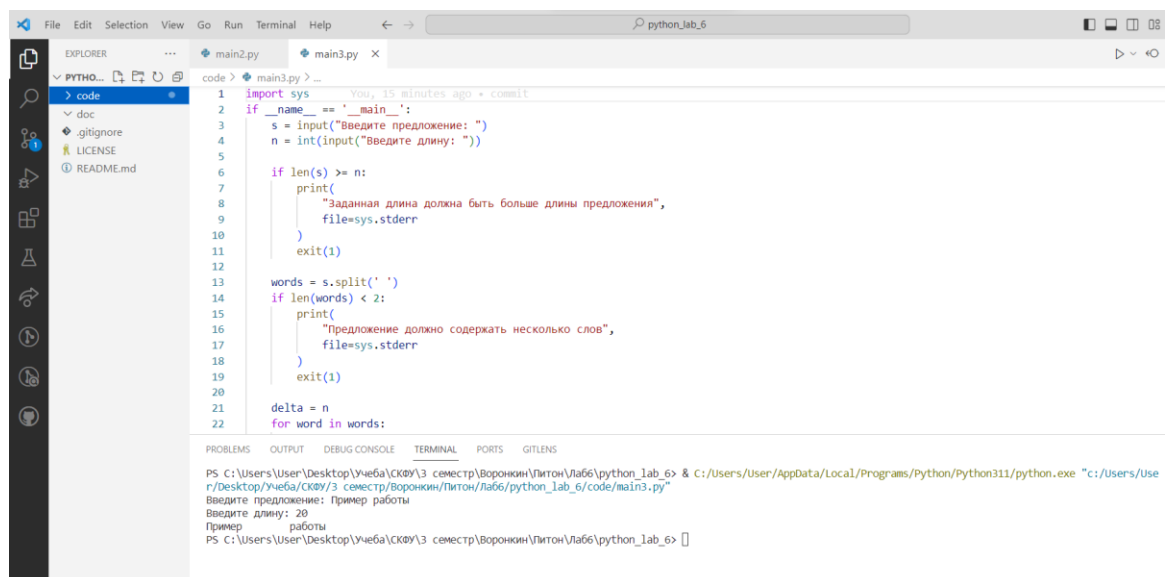
The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows a project named 'PYTHON_LAB_6' with a 'code' folder. The main editor displays a file named 'main1.py' with the following Python code:

```
1 if __name__ == "__main__":
2     s = input("Введите предложение: ")
3     r = s.replace(" ", "")
4     print(f"Предложение после замены: {r}")
```

The TERMINAL pane at the bottom shows the command prompt output:

```
PS C:\Users\User\Desktop\Учеба\СКФУ\3 семестр\Воронкин\Питон\Лаба6\python_lab_6> & C:/Users/User/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/User/Desktop/Учеба/СКФУ/3 семестр/Воронкин/Питон/Лаба6/python_lab_6/code/main1.py"
Введите предложение: пред ло же ние
Предложение после замены: предложени
```

5. Проработал третий пример из лабораторной работы: Дана строка текста, в котором нет начальных и конечных пробелов. Необходимо изменить её 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.



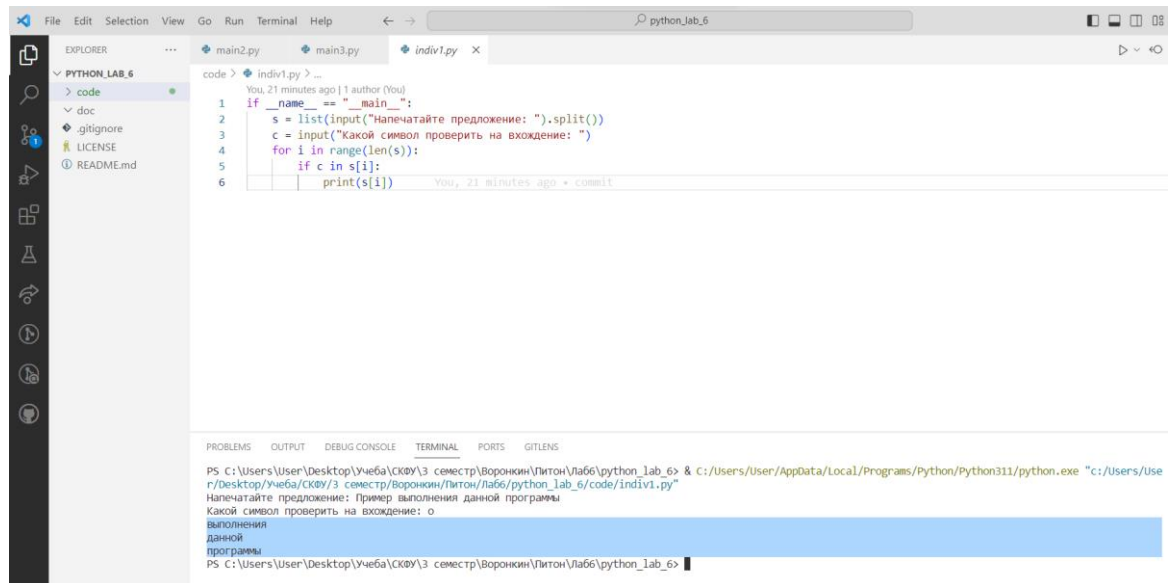
The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows a project named 'PYTHON_LAB_6' with a 'code' folder. The main editor displays a file named 'main3.py' with the following Python code:

```
1 import sys
2 if __name__ == "__main__":
3     s = input("Введите предложение: ")
4     n = int(input("Введите длину: "))
5
6     if len(s) >= n:
7         print(
8             "Заданная длина должна быть больше длины предложения",
9             file=sys.stderr
10        )
11        exit(1)
12
13    words = s.split(' ')
14    if len(words) < 2:
15        print(
16            "Предложение должно содержать несколько слов",
17            file=sys.stderr
18        )
19        exit(1)
20
21    delta = n
22    for word in words:
```

The TERMINAL pane at the bottom shows the command prompt output:

```
PS C:\Users\User\Desktop\Учеба\СКФУ\3 семестр\Воронкин\Питон\Лаба6\python_lab_6> & C:/Users/User/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/User/Desktop/Учеба/СКФУ/3 семестр/Воронкин/Питон/Лаба6/python_lab_6/code/main3.py"
Введите предложение: Пример работы
Введите длину: 20
Пример работы
PS C:\Users\User\Desktop\Учеба\СКФУ\3 семестр\Воронкин\Питон\Лаба6\python_lab_6>
```

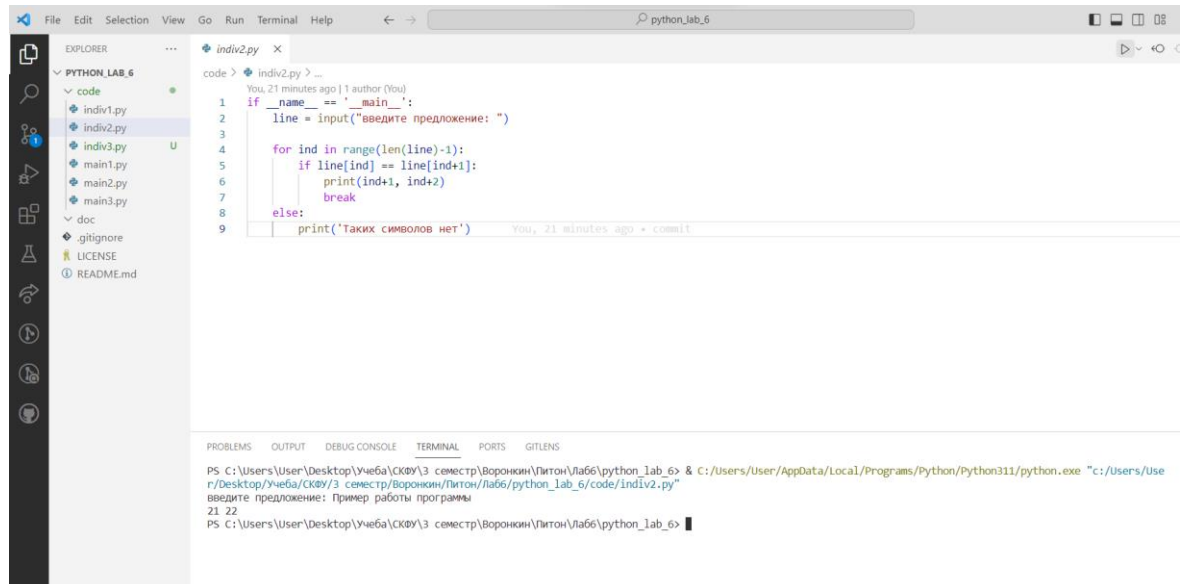
6. Проработал первое индивидуальное задание: Дано предложение. Составить программу, которая печатает “столбиком” все вхождения в предложение некоторого символа.



```
code > indiv1.py > ...
You, 21 minutes ago | 1 author (You)
1 if __name__ == '__main__':
2     s = list(input("Напечатать предложение: ").split())
3     c = input("Какой символ проверить на вхождение: ")
4     for i in range(len(s)):
5         if c in s[i]:
6             print(s[i])
You, 21 minutes ago • commit

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS
PS C:\Users\User\Desktop\Учеба\СКОУ\3 семестр\Воронкин\Литон\Лаб6\python_lab_6> & C:/Users/User/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/User/Desktop/Учеба/СКОУ/3 семестр/Воронкин/Литон/Лаб6/python_lab_6/code/indiv1.py"
Напечатать предложение: Пример выполнения данной программы
Какой символ проверить на вхождение: о
П
р
и
м
е
р
 
в
ы
п
о
л
н
е
н
и
я
 
д
а
н
н
о
й
 
п
р
о
г
р
а
м
м
ы
PS C:\Users\User\Desktop\Учеба\СКОУ\3 семестр\Воронкин\Литон\Лаб6\python_lab_6>
```

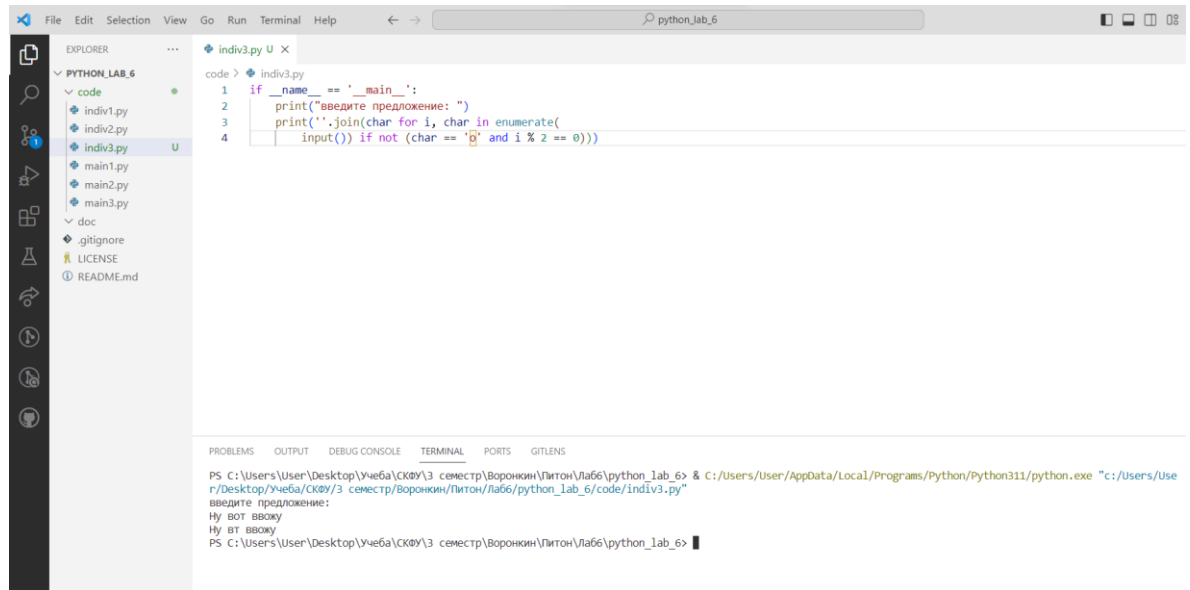
7. Проработал второе индивидуальное задание: Определить порядковые номера первой пары одинаковых соседних символов. Если таких символов нет, то должно быть напечатано соответствующее сообщение.



```
code > indiv2.py > ...
You, 21 minutes ago | 1 author (You)
1 if __name__ == '__main__':
2     line = input("Введите предложение: ")
3
4     for ind in range(len(line)-1):
5         if line[ind] == line[ind+1]:
6             print(ind+1, ind+2)
7             break
8     else:
9         print('Таких символов нет')
You, 21 minutes ago • commit

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS
PS C:\Users\User\Desktop\Учеба\СКОУ\3 семестр\Воронкин\Литон\Лаб6\python_lab_6> & C:/Users/User/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/User/Desktop/Учеба/СКОУ/3 семестр/Воронкин/Литон/Лаб6/python_lab_6/code/indiv2.py"
Введите предложение: Пример работы программы
21 22
PS C:\Users\User\Desktop\Учеба\СКОУ\3 семестр\Воронкин\Литон\Лаб6\python_lab_6>
```

8. Проработал третье индивидуальное задание: Удалить из него все буквы О, стоящие на нечетных местах.



The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows a project named 'PYTHON_LAB_6' with a 'code' folder containing several Python files. The file 'indiv3.py' is selected and open in the editor. The code in 'indiv3.py' is as follows:

```
1 if __name__ == '__main__':
2     print("Введите предложение: ")
3     print(''.join(char for i, char in enumerate(
4         input()) if not (char == 'О' and i % 2 == 0)))
```

The bottom pane shows the 'TERMINAL' output, which displays the command used to run the script and its output:

```
PS C:\Users\User\Desktop\Учеба\СКОУ\3 семестр\Воронкин\Питон\Лаб6\python_lab_6> & C:/Users/User/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/User/Desktop/Учеба/СКОУ/3 семестр/Воронкин/Питон/Лаб6/python_lab_6/code/indiv3.py"
Введите предложение:
Ну вот ввожу
PS C:\Users\User\Desktop\Учеба\СКОУ\3 семестр\Воронкин\Питон\Лаб6\python_lab_6>
```

Ответы на контрольные вопросы:

1. Что такое строки в языке Python?

Строки в Python - упорядоченные последовательности символов, используемые для хранения и представления текстовой информации, поэтому с помощью строк можно работать со всем, что может быть представлено в текстовой форме.

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

Строки в апострофах и в кавычках, экранированные последовательности, "сырые" строки, строки в тройных апострофах или кавычках

3. Какие операции и функции существуют для строк?

Сложение, дублирование, длина строки, извлечение среза и т. д.

4. Как осуществляется индексирование строк?

Доступ к символам в строках основан на операции индексирования — после строки или имени переменной, ссылающейся на строку, в квадратных скобках указываются номера позиций необходимых символов.

5. Как осуществляется работа со срезами для строк?

Есть три формы срезов. Самая простая форма среза: взятие одного символа строки, а именно, `S[i]` — это срез, состоящий из одного символа, который имеет номер `i`, при этом считая, что нумерация начинается с числа 0. То есть если `S = 'Hello'`, то `S[0]=='H'`, `S[1]=='e'`, `S[2]=='l'`, `S[3]=='l'`, `S[4]=='o'`. Если указать отрицательное значение индекса, то номер будет отсчитываться с конца, начиная с номера -1. Срез с двумя параметрами: `S[a:b]` возвращает подстроку из `b-a` символов, начиная с символа с индексом `a`, то есть до символа с индексом `b`, не включая его.

6. Почему строки Python относятся к неизменяемому типу данных?

Строки — один из типов данных, которые Python считает неизменяемыми, что означает невозможность их изменять. Python дает возможность изменять (заменять и перезаписывать) строки.

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

```
string.istitle()
```

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

```
string.find()
```

9. Как найти индекс первого вхождения подстроки в строку?

```
s.partition(<sep>)
```

10. Как подсчитать количество символов в строке?

```
len(s)
```

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

```
s.count(<sub>)
```

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

Эти строки улучшают читаемость кода, а также работают быстрее чем другие способы форматирования. F-строки задаются с помощью литерала «f» перед кавычками. Пример: `print(f'Меня зовут {name} Мне {age} лет.')`

13. Как найти подстроку в заданной части строки?

s.find(значение, начало, конец)

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

print('{}'.format(s))

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

s.isdigit()

16. Как разделить строку по заданному символу?

str.split()

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

s.isalpha()

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

s.istitle()

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

Нет

20. Как «перевернуть» строку?

s.reverse()

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

str.split('-')

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

s.upper()

s.lower

23. Как преобразовать первый символ строки к верхнему регистру?

s.capitalize()

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

s.isupper()

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

`s.splitlines()` делит `s` на строки и возвращает их в списке. Любой из следующих символов или последовательностей символов считается границей строки.

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

`s.replace(old, new)`

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

`str.startswith()` и `str.endswith()`

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

`s.isspace()`

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

`Asd*3 = AsdAsdAsd`

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

`s.title()`

31. Как пользоваться методом `partition()`?

Метод `partition()` разбивает строку при первом появлении строки аргумента и возвращает кортеж, содержащий часть перед разделителем, строку аргумента и часть после разделителя.

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

`s.rfind(<sub>)` возвращает индекс последнего вхождения подстроки `<sub>` в `s`, который соответствует началу `<sub>`

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