**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 №2.3**

Дисциплина: «Основы кроссплатформенного программирования»

Тема: «Работа со строками в языке Python»

Выполнил: студент 1 курса

группы ИВТ-б-о-21-1

Криворот Владимир Геннадьевич

Ставрополь 2022

**Выполнение работы:**

1. Создал репозиторий в GitHub «rep 2.2» в который добавил .gitignore, который дополнил правила для работы с IDE PyCharm с ЯП Python, выбрал лицензию MIT, клонировал его на лок. сервер и организовал в соответствии с моделью ветвления git-flow.

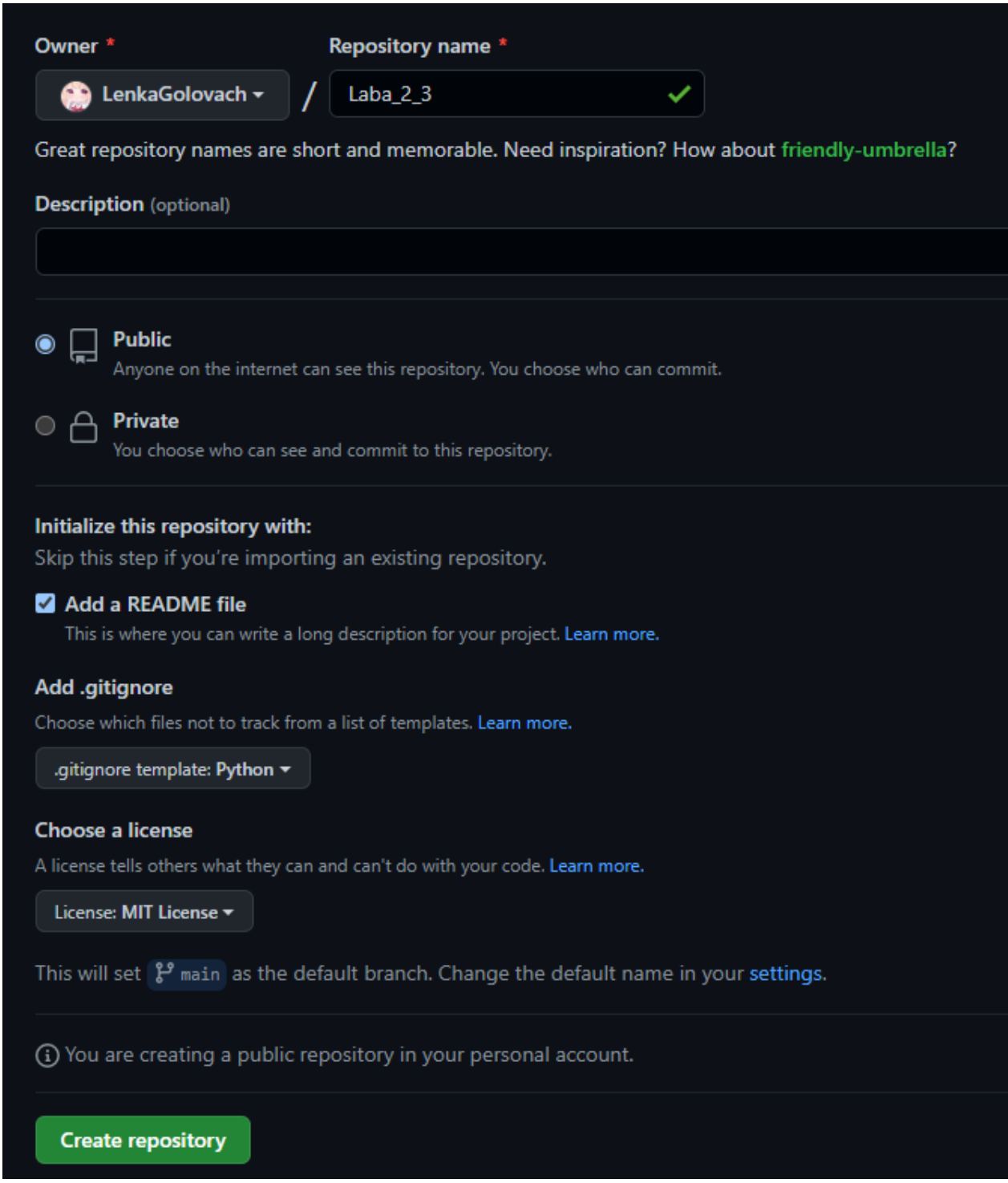


Рисунок 1.1 Создание репозитория

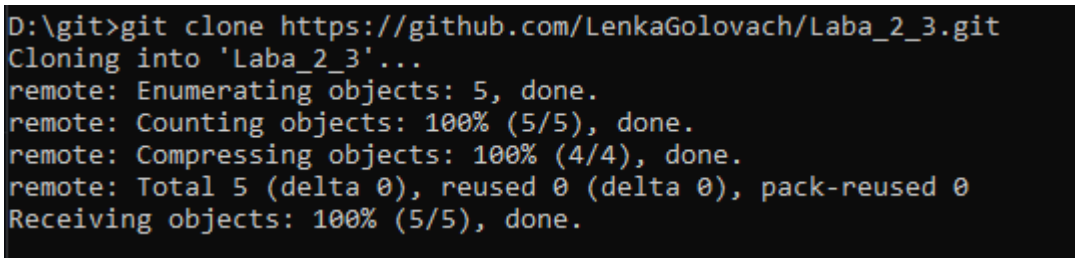


Рисунок 1.2 Клонирование репозитория

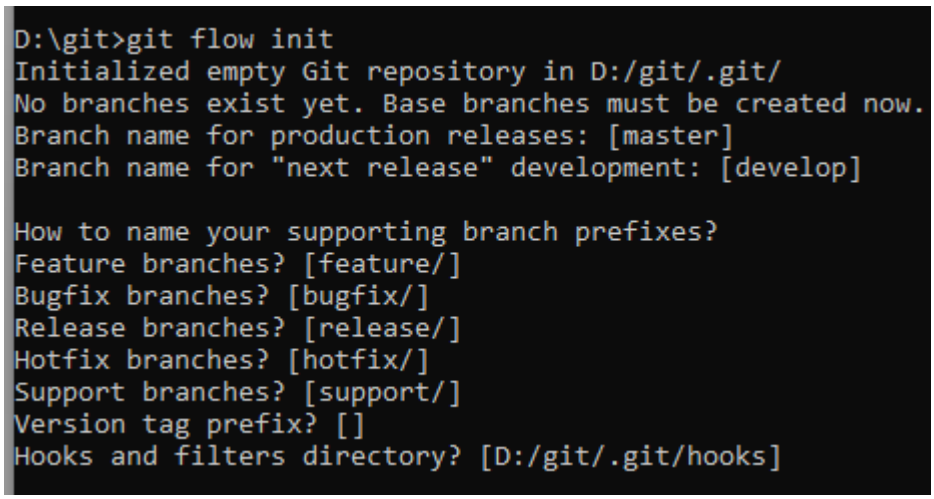


Рисунок 1.3 Организация репозитория в соответствии с моделью ветвления git-flow

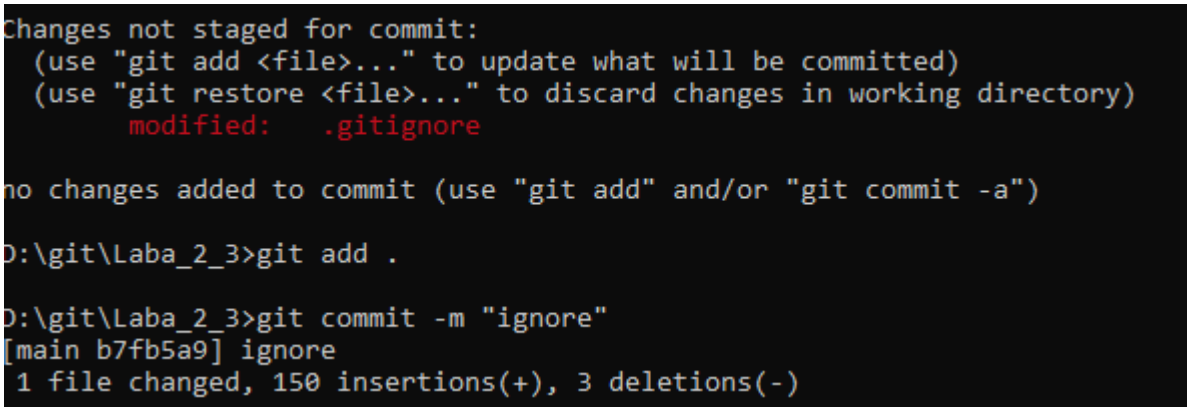


Рисунок 1.4 Изменение .gitignore

2. Создал проект PyCharm в папке репозитория, проработал примеры ЛР.

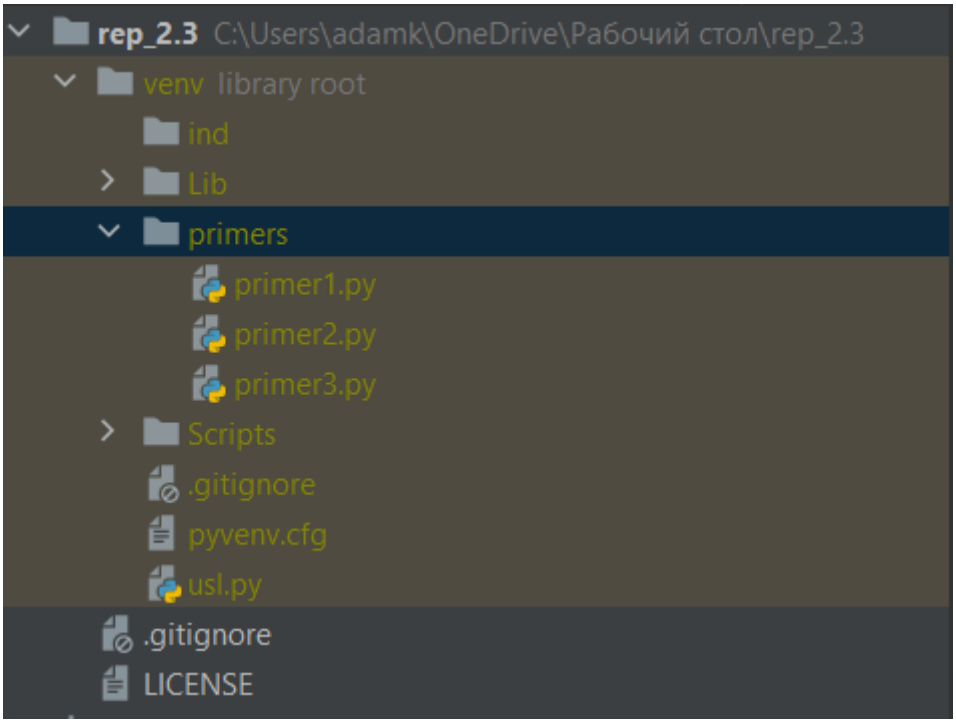


Рисунок 2.1 Создание проекта в PyCharm

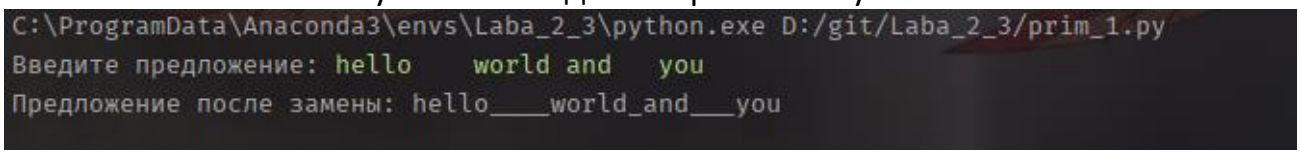


Рисунок 2.2 Рез-т выполнения программы

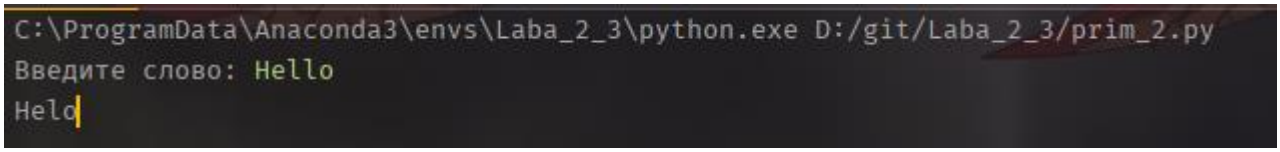


Рисунок 2.3 Рез-т выполнения программы

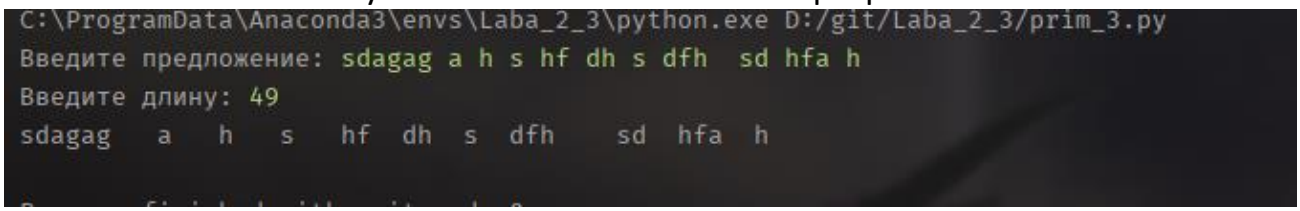


Рисунок 2.4 Рез-т выполнения программы

3. (16 вариант). Выполнил 3 индивидуальных задания и задание повышенной сложности.

**Задание 1.** Дано предложение. Определить число пробелов в нем.

```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

"""
16. Дано предложение. Определить число пробелов в нем.
"""

if __name__ == '__main__':
    s = str(input("Введите предложение - "))
    prob = 0
    for i in range(0, len(s)):
        if s[i] == ' ':
            prob += 1
    print("Кол-во пробелов равно - ", prob)
```

Рисунок 3.1 Листинг программы

```
C:\ProgramData\Anaconda3\envs\Laba_2_3\python.e
Введите предложение - Привет как твои дела
Кол-во пробелов равно -  3
```

Рисунок 3.2 Выполнение программы

**Задание 2.** Дано предложение. Все его символы, стоящие на третьем, шестом, девятом и т. д. местах, заменить буквой а.

```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

"""
Дано предложение. Все его символы, стоящие на третьем, шестом, девятом и т. д. местах, заменить буквой а.
"""

if __name__ == '__main__':
    s = str(input("Введите предложение - "))
    edit = ' '
    for i in range(0, len(s)):
        if (i + 1) % 3 == 0:
            edit += 'a'
        else:
            edit += s[i]
    print(edit)
```

Рисунок 4.1 Листинг программы

```
Введите предложение - привет как дела
правеа как аела
```

Рисунок 4.2 Выполнение программы

**Задание 3.** Дано слово. Переставить его первую букву на место -й. При этом вторую, третью, ..., -ю буквы сдвинуть влево на одну позицию.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

"""
Дано слово. Переставить его первую букву на место -й. При этом
буквы сдвинуть влево на одну позицию.
"""

if __name__ == '__main__':
    w = str(input('Введите слово: '))
    k = int(input('Введите k: '))
    tmp = list(w)

    s = tmp[0]
    for i in range(k - 1, len(w) - 1):
        tmp[i] = tmp[i + 1]
    tmp[k - 1] = s

    w = ''.join(tmp)
    print(w)
```

Рисунок 4.3 Листинг программы

```
Введите слово: привет
Введите k: 2
ппвет
```

Рисунок 4.4 Выполнение программы

**Усложненное задание**. Дано предложение. Напечатать все его слова в порядке неубывания их длин.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

if __name__ == '__main__':
    s = str(input('Введите слово: '))
    n = 0
    p = 0
    for i in range(0, len(s) - 1):
        if s[i] == "н" and (s[i + 1] == ' ' or i == len(s)):
            n += 1
        if s[i] == "р" and (s[i + 1] == ' ' or i == len(s)):
            p += 1
    print("На н - ", n, "\nОканчивающихся на р - ", p)
```

Рисунок 4.5 Листинг программы

```
C:\ProgramData\Anaconda3\envs\Laba_2_3\python.exe D:/git/Laba_2_3/pov.py
Введите слово: рр нн рр
На н -  1
Оканчивающихся на р -  2
```

Рисунок 4.6 Выполнение программы

4. Сделал коммит, выполнил слияние с веткой main, и запушил изменения в уд. репозиторий.

```
D:\git\Laba_2_3>git add .

D:\git\Laba_2_3>git commit -m
error: switch `m' requires a value

D:\git\Laba_2_3>
D:\git\Laba_2_3>git commit -m "Выполненные задания"
[develop e9e953d] Выполненные задания
```

Рисунок 4.1 Фиксация и коммит файлов

```
D:\git\Laba_2_3>git merge develop
Merge made by the 'ort' strategy.
 ind_1.py  | 14 ++++++++++++++
 ind_2.py  | 15 +++++++++++++++
 ind_3.py  | 20 ++++++++++++++++++++
 pov.py    | 13 +++++++++++++
 prim_1.py |  7 +++++++
 prim_2.py | 13 +++++++++++++
 prim_3.py | 57 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 7 files changed, 139 insertions(+)
 create mode 100644 ind_1.py
 create mode 100644 ind_2.py
 create mode 100644 ind_3.py
 create mode 100644 pov.py
 create mode 100644 prim_1.py
 create mode 100644 prim_2.py
 create mode 100644 prim_3.py
```

Рисунок 4.2 Слияние ветки develop с main

```
D:\git\Laba_2_3>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 657 bytes | 657.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/LenkaGolovach/Laba_2_3.git
   e71c96f..0b7bab8  main -> main
```

Рисунок 4.3 Пуш коммитов

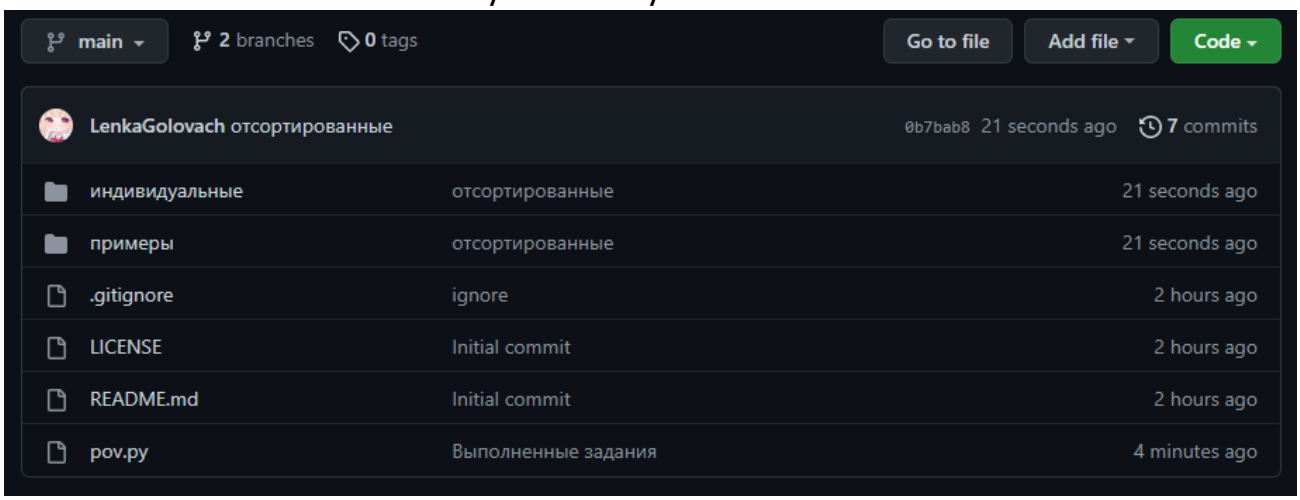


Рисунок 4.4 Изменения на уд. сервере

**Контр. вопросы и ответы на них:**

**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, 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>.