

In [6]:

```
S = "str"
print(S)
S = 'str'
print(S)
S = '"str"'
print(S)
S = "'str'"
print(S)
```

```
str
str
"str"
'str'
```

In [88]:

```
S = "Привет!\nТы\t\vizучаешь\nPython"
print(S) # используем функции экранирования
# \n - перевод на новую строку (запомним как new)
# \t - табуляция горизонтальная (так и запомним Табуляция)
# \v - табуляция вертикальная (так и запоминаем Вертикальная)
# \r - вернет корретку
```

```
Привет!
Ты      ъизучаешь
Python
```

In [11]:

```
S = "C:\temp\new"
print(S)
S = r"C:\temp\new" # подавляем функции экранирования с помощью символа r
print(S)
```

```
C:      emp
ew
C:\temp\new
```

In [185]:

```
# Конкатенация (сложение строк)
S1 = 'Hello' # строка 1
S2 = ' ' # строка 2
S3 = 'world' # строка 3
S4 = S1+S2+S3 # итоговая строка
print (S4)# вывод строки на экран
```

```
Hello world
```

In [14]:

```
# Конкатенация (сложение строк)
S1 = 'Hello' # строка 1
S2 = ' ' # строка 2
S3 = 'world' # строка 3
S4 = S1+S2+S3 # итоговая строка
print (S4)# вывод строки на экран
```

Hello world

In [17]:

```
#Повторение строки
S1 = "Hello"
#S1 = S1*3
print (S1)
S1*=3
print (S1)
```

Hello
HelloHelloHello

In [20]:

```
# Обращение по индексу в строке
S1 = "Hello"
print (S1[0]) # индексация начинается с 0 первый символ = H
print (S1[1])# обращение ко 2 символу в строке = e
print (S1[3])# обращение к 3 символу в строке = l
print (S1[2])# обращение к 4 символу в строке = l
print (S1[4])# обращение к 5 символу в строке = o
```

H
e
l
l
o

In [23]:

```
# Обращение по индексу в строке
# язык python поддерживает отрицательные индексы при обращении к элементам строки
# первый символ имеет индекс = длина строки * -1
# в нашем случае это число 5*-1 = -5
S1 = "Hello"
print (S1[-5]) # индексация начинается с 0 первый символ = H
print (S1[-4])# обращение ко 2 символу в строке = e
print (S1[-3])# обращение к 3 символу в строке = l
print (S1[-2])# обращение к 4 символу в строке = l
print (S1[-1])# обращение к 5 символу в строке = o
```

H
e
l
l
o

In [30]:

```
# измерим длину строки
S1 = "Hello world"
t = len(S1)
print (S1," длина строки равна = ", t)
t = len ("Вчера не читал лекции")
print ("У строки " + "Вчера не читал лекции" + "длина = ", t)
```

```
Hello world  длина строки равна =  11
У строки "Вчера не читал лекции"  длина =  21
```

In [43]:

```
# поиск подстроки в строке вернем индекс первого нахождения
S1 = "hello yellow"
print (S1.find("el")) # найти индекс с которого начинается подстрока в строке, без указ
ания начала поиска (во всей строке)
# использование начала поиска
S2 = "el"
print (S1.find(S2, 3))
# использование конца поиска
S2 = "el"
print (S1.find(S2, 0, 3))
# результат если не найдена подстрока будет -1
S2 = "tel"
print (S1.find(S2))
```

```
1
7
1
-1
```

In [44]:

```
# поиск подстроки в строке вернем индекс последнего нахождения
S1 = "hello yellow"
print (S1.rfind("el")) # найти индекс с которого начинается подстрока в строке, без ука
зания начала поиска (во всей строке)
# использование начала поиска
S2 = "el"
print (S1.rfind(S2, 3))
# использование конца поиска
S2 = "el"
print (S1.rfind(S2, 0, 3))
# результат если не найдена подстрока будет -1
S2 = "tel"
print (S1.rfind(S2))
```

```
7
7
1
-1
```

In [45]:

```
# поиск подстроки в строке вернем индекс последнего нахождения
S1 = "hello yellow"
print (S1.rindex("el")) # найти индекс с которого начинается подстрока в строке, без ук
азания начала поиска (во всей строке)
# использование начала поиска
S2 = "el"
print (S1.rindex(S2, 3))
# использование конца поиска
S2 = "el"
print (S1.rindex(S2, 0, 3))
# результат если не найдена подстрока будет ValueError
S2 = "tel"
print (S1.rindex(S2))
```

7
7
1

```
-----
-
ValueError                                Traceback (most recent call las
t)
<ipython-input-45-4f60c8ae9041> in <module>
     10 # результат если не найдена подстрока будет -1
     11 S2 = "tel"
--> 12 print (S1.rindex(S2))
```

ValueError: substring not found

In [46]:

```
# поиск подстроки в строке вернем индекс первого нахождения
S1 = "hello yellow"
print (S1.index("el")) # найти индекс с которого начинается подстрока в строке, без учета
                        # начала поиска (во всей строке)
# использование начала поиска
S2 = "el"
print (S1.index(S2, 3))
# использование конца поиска
S2 = "el"
print (S1.index(S2, 0, 3))
# результат если не найдена подстрока будет ValueError
S2 = "tel"
print (S1.index(S2))
```

```
1
7
1
```

```
-----
-
ValueError                                Traceback (most recent call last)
<ipython-input-46-9d7a5bcd1654> in <module>
    10 # результат если не найдена подстрока будет ValueError
    11 S2 = "tel"
--> 12 print (S1.index(S2))
```

ValueError: substring not found

In [48]:

```
# Замена шаблона в строке .replace(шаблон,замена)
S1 = "hello yellow"
print (S1.replace("el","tel"))
# через переменные
S2 = "ll"
S3 = "LL"
print (S1.replace(S2,S3))
```

```
htello ytellow
heLLo yeLLow
```

In [55]:

```
# разделение строки
S1 = "Коровы овцы лошади собаки"
print (S1.split(' ')) # разбиваем строку по пробелу
print (S1.split('o')) # разбиваем строку по 'o'
# укажем количество строк
print (S1.split('o', 2)) # разбиваем два раза, т.е. пока не встретится два раза символ
'o'
print (S1.split('o', 4)) # разбиваем пока не встретится четыре раза символ 'o'
```

```
['Коровы', 'овцы', 'лошади', 'собаки']
['К', 'р', 'вы ', 'вцы л', 'шади с', 'баки']
['К', 'р', 'вы овцы лошади собаки']
['К', 'р', 'вы ', 'вцы л', 'шади собаки']
```

In [57]:

```
# проверим состоит ли строка из цифр
S1 = '34567o'
print(S1.isdigit())
S1 = '34567'
print(S1.isdigit())
```

False

True

In [60]:

```
# проверим состоит ли строка из букв
S1 = '1 корова'
print(S1.isalpha())
S1 = 'одна корова'
print(S1.isalpha())
S1 = 'однакорова'
print(S1.isalpha())
```

False

False

True

In [62]:

```
# проверим состоит ли строка из букв и цифр
S1 = '1 корова'
print(S1.isalnum())
S1 = 'одна корова'
print(S1.isalnum())
S1 = '1корова'
print(S1.isalnum())
```

False

False

True

In [83]:

```

# удаление пробелов в строке
S1 = '      1 корова      !'
k1 = len(S1)
print('Начальная строка:', S1, 'всего символов - ', k1)
S2= S1.strip() # удадим пробелы в начале и конце строки
k2 = len(S2)
print ('Удалил пробелы в начале и конце строки результат:', S2, 'осталось символов - ', k2)
S1 = '      !"1 корова      '
k1 = len(S1)
print('Начальная строка:', S1, 'всего символов - ', k1)
S2 = S1.lstrip() # удадим пробелы в начале строки
k2 = len(S2)
print ('Удалил пробелы в начале строки результат:', S2, 'осталось символов - ', k2)
S1 = '      !"1 корова"      '
k1 = len(S1)
print('Начальная строка:', S1, 'всего символов - ', k1)
S2 = S1.rstrip() # удадим пробелы в конце строки
k2 = len(S2)
print ('Удалил пробелы в конце строки результат:', S2, 'осталось символов - ', k2)
S1 = '      !  !"1 корова"  ! * '
k1 = len(S1)
print('Начальная строка:', S1, 'всего символов - ', k1)
S2 = S1.replace(' ', '')
k2 = len(S2)
print ('Удалил все пробелы результат:', S2, 'осталось символов - ', k2)

```

Начальная строка: 1 корова ! всего символов - 34

Удалил пробелы в начале и конце строки результат: 1 корова

! осталось символов - 26

Начальная строка: !"1 корова " ! всего символов - 37

Удалил пробелы в начале строки результат: !"1 корова " ! осталось символов - 29

Начальная строка: !"1 корова" всего символов - 24

Удалил пробелы в конце строки результат: !"1 корова" осталось символов - 19

Начальная строка: ! !"1 корова" ! * всего символов - 27

Удалил все пробелы результат: !!!"1корова"!* осталось символов - 13

In [99]:

```

# проверим состоит ли в строке неотображаемых символов
S1 = 'Привет мир'
print (S1)
print(S1.isspace())
S1 = ' '
print (S1)
print(S1.isspace())
S1 = '\n'
print (S1)
print(S1.isspace())
# проверим состоит ли строка из символов в нижнем регистре
S1 = 'привет мир'
print (S1)
print(S1.islower())
S1 = 'Привет мир'
print (S1)
print(S1.islower())
# проверим состоит ли строка из символов в верхнем регистре
S1 = 'Привет мир'
print (S1)
print(S1.isupper())
S1 = 'ПРИВЕТ МИР'
print (S1)
print(S1.isupper())
# проверим начинается ли строка с символа в верхнем регистре
S1 = 'привет мир' # все слова в строке с нижнего регистра результат False
print (S1)
print(S1.istitle())
S1 = 'Привет мир' # только первое слово в строке с верхнего регистра результат False
print (S1)
print(S1.istitle())
S1 = 'Привет Мир'
print (S1)
print(S1.istitle()) # оба слова с верхнего регистра результат True

```

Привет мир
False

True

True
привет мир
True
Привет мир
False
Привет мир
False
ПРИВЕТ МИР
True
привет мир
False
Привет мир
False
Привет Мир
True

In [102]:

```
# преобразовать строку к верхнему регистру
S1 = "привет мир"
print(S1.upper())
# преобразовать строку к нижнему регистру
S1 = "Привет мир"
print(S1.lower())
```

ПРИВЕТ МИР

привет мир

In [106]:

```
# начинается ли строка с заданного шаблона
S1 = "привет мир"
print(S1.startswith("пр"))
S1 = "привет мир"
print(S1.startswith("Пр"))

# заканчивается ли строка с заданным шаблоном
S1 = "Привет мир"
print(S1.endswith("ир"))
S1 = "Привет мир"
print(S1.endswith("иР"))
```

True

False

True

False

In [116]:

```
# сборка строки из списка с разделителем
S1 = '-'
pos1 = ('Быстрее', 'Выше', 'Сильнее')
print (S1.join(pos1))
```

Быстрее-Выше-Сильнее

In [110]:

```
# переводим символы верхнего регистра в нижний и нижнего в верхний
S1 = 'прИвЕт МиР'
S2 = S1.swapcase()
print ('Начальная строка', S1)
print ('Измененная строка', S2)
# первая буква каждого слова с символа в верхнем регистре
S1 = 'иванов иван иванович'
S2 = S1.title()
print ('Начальная строка', S1)
print ('Измененная строка', S2)
```

Начальная строка прИвЕт МиР

Измененная строка ПРиВеТ мИР

Начальная строка иванов иван иванович

Измененная строка Иванов Иван Иванович

In [153]:

```
# работа со срезами строк
S1 = '0123456789'
print('Строка для среза: ',S1)
print ('<----->')
# выполним срез начиная с первого элемента по последний с шагом 2, т.е. каждый второй элемент
# синтаксис "строка[начало среза:конец среза:шаг]". 0 - начало строки, 17 последний индекс
print ('Срез с 1 элемента по 5 с шагом 2', S1[0:5:3])
print ('Срез с первого по последний элемент с шагом 2', S1[::2])
# выполним срез начиная со второго элемента с шагом 2, не забываем, что индексация в строке начинается с 0,
# т.е. второй элемент имеет индекс 1
print ('Срез без указания шага во всей строке', S1[::])
print ('Срез без указания шага во всей строке', S1[-10:0:])
print ('Срез со второго по элемент с шагом 2',S1[1:3:2])
print (S1[1::2])
```

```
Строка для среза: 0123456789
<----->
Срез с 1 элемента по 5 с шагом 2 03
Срез с первого по последний элемент с шагом 2 02468
Срез без указания шага во всей строке 0123456789
Срез без указания шага во всей строке
Срез со второго по элемент с шагом 2 1
13579
```

In [171]:

```
S1 = '0123456789'
print(S1[-1:6:-2])
print(S1[-2::-2])
print(S1[::3])
```

```
97
86420
9630
```

In [184]:

```
S1 = 'Пусть наступит новый год, он нам радость принесет!!!!'
print(S1[6:14:1])
print (S1[6:-39:1])
print (S1[-40:-48:-1])
print(S1[-40:5:-1])
print (S1[::3])
print(S1[::2])
```

```
наступит
наступит
типутсан
типутсан
Птнтиныг,нароьрет!
!!тснр тоа а о,о ью иусньсП
```

In []: