

Функции и методы для работы со строками

<code>len(S)</code>	Длина строки
<code>S.find(str, [start],[end])</code>	Поиск подстроки в строке. Возвращает номер первого вхождения или -1
<code>S.rfind(str, [start],[end])</code>	Поиск подстроки в строке. Возвращает номер последнего вхождения или -1
<code>S.index(str, [start],[end])</code>	Поиск подстроки в строке. Возвращает номер первого вхождения или вызывает <code>ValueError</code>
<code>S.rindex(str, [start],[end])</code>	Поиск подстроки в строке. Возвращает номер последнего вхождения или вызывает <code>ValueError</code>
<code>S.replace(шаблон, замена[, maxcount])</code>	Замена шаблона на замену. <code>maxcount</code> ограничивает количество замен
<code>S.split(символ)</code>	Разбиение строки по разделителю
<code>S.isdigit()</code>	Состоит ли строка из цифр
<code>S.isalpha()</code>	Состоит ли строка из букв
<code>S.isalnum()</code>	Состоит ли строка из цифр или букв
<code>S.islower()</code>	Состоит ли строка из символов в нижнем регистре
<code>S.isupper()</code>	Состоит ли строка из символов в верхнем регистре
<code>S.isspace()</code>	Состоит ли строка из неотображаемых символов (пробел, символ перевода страницы (<code>\f</code>), "новая строка" (<code>\n</code>), "перевод каретки" (<code>\r</code>), "горизонтальная табуляция" (<code>\t</code>) и "вертикальная табуляция" (<code>\v</code>))

<code>S.istitle()</code>	Начинаются ли слова в строке с заглавной буквы
<code>S.upper()</code>	Преобразование строки к верхнему регистру
<code>S.lower()</code>	Преобразование строки к нижнему регистру
<code>S.startswith(str)</code>	Начинается ли строка S с шаблона str
<code>S.endswith(str)</code>	Заканчивается ли строка S шаблоном str
<code>S.join(список)</code>	Сборка строки из списка с разделителем S
<code>ord(символ)</code>	Символ в его код ASCII
<code>chr(число)</code>	Код ASCII в символ
<code>S.capitalize()</code>	Переводит первый символ строки в верхний регистр, а все остальные в нижний
<code>S.center(width, [fill])</code>	Возвращает отцентрированную строку, по краям которой стоит символ fill (пробел по умолчанию)
<code>S.count(str, [start],[end])</code>	Возвращает количество непересекающихся вхождений подстроки в диапазоне [начало, конец] (0 и длина строки по умолчанию)
<code>S.expandtabs([tabsize])</code>	Возвращает копию строки, в которой все символы табуляции заменяются одним или несколькими пробелами, в зависимости от текущего столбца. Если TabSize не указан, размер табуляции полагается равным 8 пробелам
<code>S.lstrip([chars])</code>	Удаление пробельных символов в начале строки
<code>S.rstrip([chars])</code>	Удаление пробельных символов в конце строки

<code>S.strip([chars])</code>	Удаление пробельных символов в начале и в конце строки
<code>S.partition(шаблон)</code>	Возвращает кортеж, содержащий часть перед первым шаблоном, сам шаблон, и часть после шаблона. Если шаблон не найден, возвращается кортеж, содержащий саму строку, а затем две пустых строки
<code>S.rpartition(sep)</code>	Возвращает кортеж, содержащий часть перед последним шаблоном, сам шаблон, и часть после шаблона. Если шаблон не найден, возвращается кортеж, содержащий две пустых строки, а затем саму строку
<code>S.swapcase()</code>	Переводит символы нижнего регистра в верхний, а верхнего – в нижний
<code>S.title()</code>	Первую букву каждого слова переводит в верхний регистр, а все остальные в нижний
<code>S.zfill(width)</code>	Делает длину строки не меньшей <code>width</code> , по необходимости заполняя первые символы нулями
<code>S.ljust(width, fillchar=" ")</code>	Делает длину строки не меньшей <code>width</code> , по необходимости заполняя последние символы символом <code>fillchar</code>
<code>S.rjust(width, fillchar=" ")</code>	Делает длину строки не меньшей <code>width</code> , по необходимости заполняя первые символы символом <code>fillchar</code>

Функции и методы для работы со списками

<code>len(list)</code>	Количество элементов в списке
<code>list.append(x)</code>	Добавляет элемент в конец списка
<code>list.extend(L)</code>	Расширяет список <code>list</code> , добавляя в конец все элементы списка <code>L</code>
<code>list.insert(i, x)</code>	Вставляет на <code>i</code> -ый элемент значение <code>x</code>
<code>list.remove(x)</code>	Удаляет первый элемент в списке, имеющий значение <code>x</code> . <code>ValueError</code> , если такого элемента не существует
<code>list.pop([i])</code>	Удаляет <code>i</code> -ый элемент и возвращает его. Если индекс не указан, удаляется последний элемент
<code>list.index(x, [start [, end]])</code>	Возвращает положение первого элемента со значением <code>x</code> (при этом поиск ведется от <code>start</code> до <code>end</code>)
<code>list.count(x)</code>	Возвращает количество элементов со значением <code>x</code>
<code>list.sort([key=функция])</code>	Сортирует список на основе функции
<code>list.reverse()</code>	Разворачивает список
<code>list.copy()</code>	Поверхностная копия списка
<code>list.clear()</code>	Очищает список

Методы словарей

<code>dict.clear()</code>	Очищает словарь
<code>dict.copy()</code>	Возвращает копию словаря
<code>classmethod dict.fromkeys(seq[, value])</code>	Создает словарь с ключами из <code>seq</code> и значением <code>value</code> (по умолчанию <code>None</code>)

<code>dict.get(key[, default])</code>	Возвращает значение ключа, но если его нет, не бросает исключение, а возвращает <code>default</code> (по умолчанию <code>None</code>)
<code>dict.items()</code>	Возвращает пары (ключ, значение)
<code>dict.keys()</code>	Возвращает ключи в словаре
<code>dict.values()</code>	Возвращает значения в словаре
<code>dict.pop(key[, default])</code>	Удаляет ключ и возвращает значение. Если ключа нет, возвращает <code>default</code> (по умолчанию бросает исключение)
<code>dict.popitem()</code>	Удаляет и возвращает пару (ключ, значение). Если словарь пуст, бросает исключение <code>KeyError</code> . Помните, что словари неупорядочены
<code>dict.setdefault(key[, default])</code>	Возвращает значение ключа, но если его нет, не бросает исключение, а создает ключ со значением <code>default</code> (по умолчанию <code>None</code>)
<code>dict.update([other])</code>	Обновляет словарь, добавляя пары (ключ, значение) из <code>other</code> . Существующие ключи перезаписываются. Возвращает <code>None</code> (не новый словарь!)