Лабораторная работа №4

Тема: Работа со строками в Python

Цель: приобрести навыки работы со встроенными функциями для работы со строками в Python

теоретическая часть

Строка состоит из последовательности символов.

Тип строки str.

>>> type ( '2')

<Class 'str'>

В языке Питон нет отдельного символьного типа. Символ - это просто строка длины 1.

Длина строк в Python не ограничена или, строго говоря, ограничена объемом выделенной оперативной памяти.

Строка из нескольких подряд литералов будет неявно конкатенованим даже при отсутствии знака +:

s = 'Infinite' "blue" "sky"

Строки в тройных кавычках могут содержать несколько строк текста. Пробелы от начала строки входят в текст:

s = "" "Happy birthday to you,

           My darling Findus!

                     Your hostess "." "

Можно добиться того же эффекта и в одинарных кавычках, оставляя в конце каждой строки обратный слеш (но рекомендуется так не делать, после \ не должно быть пробела в конце строки):

s = "Happy birthday to you, \

My darling Findus! \

Your hostess ".

базовые операции

Конкатенация (сложение)

>>>

>>> S1 = 'spam'

>>> S2 = 'eggs'

>>> print (S1 + S2)

"Spameggs"

• Дублирование строки

>>>

>>> print ( 'spam' \* 3)

spamspamspam

• Длина строки

>>>

>>> len ( 'spam')

4

• Доступ по индексу

>>>

>>> S = 'spam'

>>> S [0]

"S"

>>> S [2]

"A"

>>> S [-2]

"A"

Как видно из примера, в Python есть возможность доступа по отрицательного индекса, при этом отсчет идет от конца строки.

• Получение среза

Оператор извлечения среза: [X: Y]. X - начало среза, а Y - окончание; символ с номером Y в срез не входит. По умолчанию первый индекс равен 0, а второй - длине строки.

>>>

>>> s = 'spameggs "

>>> s [3: 5]

"Me"

>>> s [2: -2]

"Ameg"

>>> s [6]

"Spameg"

>>> s [1:]

"Pameggs"

>>> s [:]

"Spameggs"

Кроме того, можно задать шаг, с которым нужно извлекать срез.

>>>

>>> s [:: - 1]

"Sggemaps"

>>> s [3: 5: -1]

''

>>> s [2 :: 2]

"Aeg"

Другие функции и методы строк

При вызове методов необходимо помнить, что строки в Python относятся к категории неизменяемых последовательностей, то есть все функции и методы могут только создавать новую строку.

>>>

>>> s = 'spam'

>>> s [1] = 'b' # Ошибка!

Traceback (most recent call last):

  File "", line 1, in

    s [1] = 'b'

TypeError: "str" ​​object does not support item assignment

>>> s = s [0] + 'b' + s [2:]

>>> s

"Sbam"

Поэтому все срочные методы возвращают новую строку, который затем следует присвоить переменной.

Таблица "Функции и методы строк"

Функция или метод Назначение

S = 'str'; S = "str"; S = '' 'str' '; S = "" "str" ​​"" литералы строк

S = "s \ np \ ta \ nbbb" экранированные последовательности

S = r "C: \ temp \ new" неформатированные строки (подавляют экранирование)

S = b "byte" строка байтов

S1 + S2 конкатенация (добавление строк)

S1 \* 3 повторения строки

S [i] обращение по индексу

S [i: j: step] извлечение среза

len (S) длина строки

S.find (str [start], [end]) Поиск подстроки в строке. Возвращает номер первого вхождения или -1

S.rfind (str [start], [end]) Поиск подстроки в строке. Возвращает номер последнего вхождения или -1

S.index (str [start], [end]) Поиск подстроки в строке. Возвращает номер первого вхождения или вызывает ValueError

S.rindex (str [start], [end]) Поиск подстроки в строке. Возвращает номер последнего вхождения или вызывает ValueError

S.replace (шаблон, замена) Замена шаблона

S.split (символ) Разбивка строки по разделителю

S.isdigit () Составляется строку из цифр

S.isalpha () Состоит строку из цифр букв

S.isalnum () Состоит строку из цифр или букв

S.islower () Состоит строку из символов в нижнем регистре

S.isupper () Состоит строку из символов в верхнем регистре

S.isspace () Состоит строку с невидображаючихся символов (пробел, символ перевода страницы ( '\ f'), "новая строка" ( '\ n'), "перевод каретки" ( '\ r'), "горизонтальная табуляция" ( '\ t') и "вертикальная табуляция" ( '\ v'))

S.istitle () или начинаются слова в строке с большой буквы

S.upper () Преобразование строки к верхнему регистру

S.lower () Преобразование строки к нижнему регистру

S.startswith (str) Начинается строку S с шаблона str

S.endswith (str) ли закончится строку S с шаблона str

S.join (список) Сборник строки из списка с разделителем S

ord (символ) код символа в ASCII

chr (число) код ASCII в символ

S.capitalize () переводит первый символ строки в верхний регистр, а все остальные в нижней

S.center (width [fill]) Возвращает центробежные строку, по краям которого стоит символ fill (пробел по умолчанию)

S.count (str [start], [end]) Возвращает количество пересекающихся вхождений подстроки в диапазоне [начало, конец] (0 и длина строки по умолчанию)

S.expandtabs ([tabsize]) Возвращает копию строки, в которой все символы табуляции заменяются одним или несколькими пробелами, в зависимости от текущего столбца. Если TabSize не указано, размер табуляции считается равным 8 пробелов

S.lstrip ([chars]) Удаление символов пробелов в начале строки

S.rstrip ([chars]) Удаление символов пробелов в конце строки

S.strip ([chars]) Удаление символов пробелов в начале и в конце строки

S.partition (шаблон) Возвращает кортеж, содержащий часть перед первым шаблоном, сам шаблон, и часть после шаблона. Если шаблон не найден