

24 занятие

Школа::Кода

«Основы программирования на языке
Python»

Проверка присутствия



Встроенные функции Python для строк

<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>

Встроенные функции Python для строк

<code>S.replace(шаблон, замена)</code>	Замена шаблона
<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

Встроенные функции Python для строк

<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 ('\f'), "новая строка" ('\n'), "перевод каретки" ('\r'), "горизонтальная табуляция" ('\t') и "вертикальная табуляция" ('\v'))
<code>S.istitle()</code>	Начинаются ли слова в строке с заглавной буквы
<code>S.upper()</code>	Преобразование строки к верхнему регистру

Встроенные функции Python для строк

<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

Встроенные функции Python для строк

<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 отцентрованную 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

Встроенные функции Python для строк

<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

Встроенные функции Python для строк

<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 width, по необходимости заполняя первые символы нулями
<code>S.ljust(width, fillchar=" ")</code>	Делает длину строки не меньшей width, по необходимости заполняя последние символы символом fillchar
<code>S.rjust(width, fillchar=" ")</code>	Делает длину строки не меньшей width, по необходимости заполняя

Встроенные функции Python для СПИСКОВ

<code>list.append(x)</code>	Добавляет элемент в конец списка
<code>list.extend(L)</code>	Расширяет список list, добавляя в конец все элементы списка L
<code>list.insert(i, x)</code>	Вставляет на i-ый элемент значение x
<code>list.remove(x)</code>	Удаляет первый элемент в списке, имеющий значение x. ValueError, если такого элемента не существует
<code>list.pop([i])</code>	Удаляет i-ый элемент и возвращает его. Если индекс не указан, удаляется последний элемент

Встроенные функции Python для СПИСКОВ

<code>list.index(x, [start [, end]])</code>	Возвращает положение первого элемента со значением x (при этом поиск ведется от start до end)
<code>list.count(x)</code>	Возвращает количество элементов со значением x
<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

Интересный код на Python

```
s = input()
print(len(s))
t = input()
number = int(t)
u = str(number)
print(s * 3)
print(s + ' ' + u)
```

Интересный код на Python

```
s = 'abcdefg'  
print(s[1])  
print(s[-1])  
print(s[1:3])  
print(s[1:-1])  
print(s[:3])  
print(s[2:])  
print(s[:-1])  
print(s[::2])  
print(s[1::2])  
print(s[::-1])
```

Интересный код на Python

```
s = 'abcdefghijklm'  
print(s[0:10:2])  
for i in range(0, 10, 2):  
    print(i, s[i])
```

Интересный код на Python

```
S = 'Hello'  
print(S.find('e'))  
# вернёт 1  
print(S.find('ll'))  
# вернёт 2  
print(S.find('L'))  
# вернёт -1
```

Интересный код на Python

```
S = 'Hello'  
print(S.find('l'))  
# вернёт 2  
print(S.rfind('l'))  
# вернёт 3
```


Интересный код на Python

```
print('Hello'.replace('l', 'L'))  
# вернёт 'HeLLo'
```

Интересный код на Python

```
print('Abrakadabra'.replace('a', 'A', 2))  
# вернёт 'AbrAkAdabra'
```

Интересный код на Python

```
print('Abracadabra'.count('a'))  
# вернёт 4  
print(('a' * 10).count('aa'))  
# вернёт 5
```

Интересный код на Python

Перевернуть строку — Вводится строка и программа ее переворачивает и распечатывает на экран.

```
reverse_string = ""
num = len(string) - 1
for i in string:
    reverse_string += string[num]
    num-=1
print (reverse_string)
```

Интересный код на Python

Счетчик гласных — Вводится строка и программа считает количество гласных в тексте. Для усложнения задачи можно генерировать отчет о том, сколько раз какая гласная была найдена.

```
def count_vowels(string):
    vowels = ['a', 'e', 'i', 'o', 'u']
    lower_string = string.lower()
    vowel_count = 0
    for char in lower_string:
        for vowel in vowels:
            if char == vowel:
                vowel_count+=1
    return vowel_count
print count_vowels("This is a sentence with vowels")
print count_vowels("There are 6 vowels")
```