

ГЛАВА 3. СТРОКИ И ОПЕРАЦИИ НАД НИМИ

Python часто используют для обработки текстов: поиска в тексте, замены отдельных частей текста и т.д. Для работы с текстом в Python предусмотрен специальный строковый тип данных `str`.

Python создает строковые объекты, если текст поместить в одинарные или двойные кавычки:

```
>>> 'hello'
'hello'
>>> "Hello"
'Hello'
>>>
```

Без кавычек Python расценит текст как переменную и попытается вывести на экран ее содержимое (если такая переменная была создана):

```
>>> hello
Traceback (most recent call last):
  File "<pyshell#2>", line 1, in <module>
    hello
NameError: name 'hello' is not defined
>>>
```

Можно создать пустую строку:

```
>>> ''
''
>>>
```

Для работы со строками в Python предусмотрено большое число встроенных функций, например, `len`. Эта функция определяет длину строки, которая передается ей в качестве аргумента.

```
>>> help(len)
Help on built-in function len in module builtins:

len(obj, /)
    Return the number of items in a container.

>>> len('Привет!')
7
>>>
```

К примеру, если мы хотим объединить несколько строк в одну, Python позволяет это сделать с помощью операции конкатенации (обычный символ `+` для строк):

```
>>> 'Привет, ' + 'земляне!'
'Привет, земляне!'
>>>
```