Home (/) >> String Manipulation in Python

Apr. 09, 2013

Basics (/basics/) Strings (/python-strings/)

String Manipulation in Python

Overview

```
A string is a list of characters in order.

A character is anything you can type on the keyboard in one keystroke,
like a letter, a number, or a backslash.

Strings can have spaces: "hello world".

An empty string is a string that has 0 characters.

Python strings are immutable

Python recognize as strings everything that is delimited by quotation marks
(" " or ' ').
```

String Manipulation

```
To manipulate strings, we can use some of Pythons built-in methods.
```

```
Creation
```

```
word = "Hello World"

>>> print word
Hello World
```

```
Accessing

Use [ ] to access characters in a string
```

```
word = "Hello World"
letter=word[0]

>>> print letter
H
```

```
Length
```

```
word = "Hello World"

>>> len(word)
11
```

```
Finding
```

```
word = "Hello World"

>>> print word.count('l')       # count how many times l is in the string
3

>>> print word.find("H")         # find the word H in the string
0

>>> print word.index("World")   # find the letters World in the string
6
```

```
Count
```

```
s =  "Count, the number     of spaces"

>>> print s.count(' ')
8
```

```
Slicing
Use [ # : # ] to get set of letter

Keep in mind that python, as many other languages, starts to count from 0!!
```

```
word = "Hello World"

print word[0]          #get one char of the word
print word[0:1]        #get one char of the word (same as above)
print word[0:3]        #get the first three char
print word[:3]         #get the first three char
print word[-3:]        #get the last three char
print word[3:]         #get all but the three first char
print word[:-3]        #get all but the three last character
```

```
word = "Hello World"

word[start:end]         # items start through end-1
word[start:]            # items start through the rest of the list
word[:end]              # items from the beginning through end-1
word[:]                 # a copy of the whole list
```

```
Split Strings
```

```
word = "Hello World"

>>> word.split(' ')  # Split on whitespace
['Hello', 'World']
```

```
Startswith / Endswith
```

```
word = "hello world"

>>> word.startswith("H")
True

>>> word.endswith("d")
True

>>> word.endswith("w")
False
```

```
Repeat Strings
```

```
print "."* 10   # prints ten dots

>>> print "." * 10
..........
```

```
Replacing
```

```
word = "Hello World"

>>> word.replace("Hello", "Goodbye")
'Goodbye World'
```

```
Changing Upper and Lower Case Strings
```

```
string = "Hello World"

>>> print string.upper()
HELLO WORLD

>>> print string.lower()
hello world

>>> print string.title()
Hello World

>>> print string.capitalize()
Hello world

>>> print string.swapcase()
hELLO wORLD
```

```
Reversing
```

```
string = "Hello World"

>>> print ' '.join(reversed(string))
d l r o W   o l l e H
```

```
Strip

Python strings have the strip(), lstrip(), rstrip() methods for removing
any character from both ends of a string.

If the characters to be removed are not specified then white-space will be removed
```

```
word = "Hello World"
```

```
Strip off newline characters from end of the string
```

```
>>> print word.strip('
')
Hello World
```

```
strip()     #removes from both ends
lstrip()    #removes leading characters (Left-strip)
rstrip()    #removes trailing characters (Right-strip)

>>> word = "    xyz    "

>>> print word
    xyz

>>> print word.strip()
xyz

>>> print word.lstrip()
xyz

>>> print word.rstrip()
    xyz
```

Concatenation

To concatenate strings in Python use the "+" operator.

```
"Hello " + "World" # = "Hello World"
"Hello " + "World" + "!"# = "Hello World!"
```

Join

```
>>> print ":".join(word)  # #add a : between every char
H:e:l:l:o: :W:o:r:l:d

>>> print " ".join(word)  # add a whitespace between every char
H e l l o   W o r l d
```

Testing

A string in Python can be tested for truth value.

The return type will be in Boolean value (True or False)

```
word = "Hello World"

word.isalnum()         #check if all char are alphanumeric
word.isalpha()         #check if all char in the string are alphabetic
word.isdigit()         #test if string contains digits
word.istitle()         #test if string contains title words
word.isupper()         #test if string contains upper case
word.islower()         #test if string contains lower case
word.isspace()         #test if string contains spaces
word.endswith('d')     #test if string endswith a d
word.startswith('H')   #test if string startswith H
```

If you liked this article, please share it with your friends.

Recommended Python Training – DataCamp (https://www.datacamp.com/courses/tech:python?tap_a=5644-dce66f&tap_s=75426-9cf8ad)

For Python training (https://www.datacamp.com/courses/tech:python?tap_a=5644-dce66f&tap_s=75426-9cf8ad), our top recommendation is DataCamp.

Datacamp (https://www.datacamp.com/courses/tech:python?tap_a=5644-dce66f&tap_s=75426-9cf8ad) provides online interactive courses that combine interactive coding challenges with videos from top instructors in the field.

Datacamp has beginner to advanced Python training that programmers of all levels benefit from.

Tweet

Like 26 Share

Read more about:

- Basics (/basics/)
- Strings (/python-strings/)

21 Comments **Pythonforbeginners.com** 1 Login

♡ Recommend 3 Share Sort by Best

Join the discussion…

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Simon • 4 years ago

Very helpful to Python novice like me!

One note though:
print word[1:3] #get the first three char
This is incorrect, it'll get chars 2 & 3.

4 ⌃ | ⌄ • Reply • Share ›

pythonforbeginners Mod ➦ Simon • 4 years ago

Thank you. The article was a bit confusing. Indeed, to get the first three characters you can use word[0:3]:

```
word="hello"

print word[1:3] #will show "el"
print word[0:3] #will show "hel"
```

3 ⌃ | ⌄ • Reply • Share ›

emad ➦ pythonforbeginners • 4 years ago

we are lerning python in high school
and our teacher share this article to us it is very helpful
we are in iran

12 ⌃ | ⌄ • Reply • Share ›

huntkil ➦ Simon • 4 years ago

it will show char 1 & 2 not 2 & 3

2 ⌃ | ⌄ • Reply • Share ›

AnonXcode • 3 years ago

anyone here has idea,how to remove the second word in a sentence,like
its red
then the output is one of them
thanks

2 ⌃ | ⌄ • Reply • Share ›

Bui Phuoc • 5 months ago

I'm making the guessing letter game .
The user will type 1 letter to guess.
At the original value, the result is "_ _ _ _ _" for example
I want to replace the third "_" and I make this:

```
Word = "heilo"

answer = '_' * len(Word)

print(answer)

for attempt in range(0,20):

guess = input("guess the letter:")

for i in range(0,len(Word)):

if guess == i:

answer = answer.replace(answer[i], guess)
```

answer = answer.replace(answer[i], guess)

see more

⌃ | ⌄ • Reply • Share ›

настя корн • 6 months ago

This is very helpful, easy and clear to understand, thank you so much

⌃ | ⌄ • Reply • Share ›

Barry B • a year ago

Hi very good explanations.. I am new to python and want to manipulate a string. Say stringstr="2016-06-09 12:13:14 ABCDEFGH" I want to pull say the date datestr=stringstr[8:2]+stringstr[5:2]+stringstr[2:2] which I was hopeing would give me 090616 but it only grabs the first 2 chars. Can anyone help....

⌃ | ⌄ • Reply • Share ›

Deb ➦ Barry B • 6 months ago

You can do -
datestr=stringstr[8:10] + stringstr[5:7] + stringstr[2:4]
print datestr

This will give you the output 090616

⌃ | ⌄ • Reply • Share ›

Silverium • a year ago

You have a little mistake here. Documentation says:
str.isalnum()
Return true if all characters in the string are alphanumeric and there is at least one character, false otherwise.

reference: https://docs.python.org/2/l...

⌃ | ⌄ • Reply • Share ›

pythonforbeginners Mod ➦ Silverium • a year ago

Fixed ~ :)

⌃ | ⌄ • Reply • Share ›

Arun Sharma • a year ago

how to print a characher like 'a' in (a)

⌃ | ⌄ • Reply • Share ›

Karan Singh • 2 years ago

Such a nice and helpful post. Thanks

⌃ | ⌄ • Reply • Share ›

Satdhruti Paul • 2 years ago

Instead of reversed(string), you could use string[::-1]. it's a slicing technique
I'm a fan of code golfing.

⌃ | ⌄ • Reply • Share ›

Graham Yapp • 2 years ago

<---You kids got no stile

⌃ | ⌄ • Reply • Share ›

Graham Yapp • 2 years ago

You tink u can face the yappmeister, I will yappinate you!

⌃ | ⌄ • Reply • Share ›

Graham Yapp • 2 years ago

Amateurs, I AM THE GRAHAM YAPP! you think u kids r so cool; but ik all de language of python fam.

⌃ | ⌄ • Reply • Share ›

AnonXcode • 3 years ago

Hey !!! i find this site is very helpfull !!!

⌃ | ⌄ • Reply • Share ›

Uncommon • 4 years ago

I have been asked to take the phrase "ItTakesOneToKnowOne" and add whitespace between each word in this string so it now reads "It Takes One To Know One". You mention how to add whitespace between each character, anyway to code so it only places whitespace before an uppercase character? (I assume it has something to do with ch.isupper, but I'm at a loss as to how to proceed.) Thanks!

^ | v • Reply • Share ›

pythonforbeginners Mod ➦ Uncommon • 4 years ago

```
# For Latin characters, this can be a way:
import re
string = "ItTakesOneToKnowOne"
print re.sub( "(?<!^)(?=[A-Z])", " ", string )
```

For non-Latin, check this solution:
http://stackoverflow.com/qu...

^ | v • Reply • Share ›

Uncommon ➦ pythonforbeginners • 4 years ago

Thanks!

^ | v • Reply • Share ›

✉ **Subscribe** **Add Disqus to your siteAdd DisqusAdd** **Privacy**

Disclosure of Material Connection: Some of the links in the post above are "affiliate links." This means if you click on the link and purchase the item, I will receive an affiliate commission. Regardless, PythonForBeginners.com only recommend products or services that we try personally and believe will add value to our readers.

Search SEARCH

Categories

Basics (/basics/)

Cheatsheet (/cheatsheet/)

Code snippets (/code-snippets-source-code/)

Development (/development/)

Dictionary (/dictionary/)

Error Handling (/error-handling/)

Lists (/lists/)

Loops (/loops/)

Modules (/modules-in-python/)

Strings (/python-strings/)

System & OS (/systems-programming/)

Web & Internet (/python-on-the-web/)