Python: How exactly can you take a string, split it, reverse it and join it back together again?

How exactly can you take a string, split it, reverse it and join it back together again without the brackets, commas, etc. using python?

python   string   join   split   reverse

asked Sep 2 '10 at 12:54
Tstrmwarrior
78 1 1 5

---

1   Which character are you splitting the string on? Space? – Manoj Govindan Sep 2 '10 at 12:57

2   is this homework? – msw Sep 2 '10 at 13:00

16   There's a Missy Elliott joke in there somewhere. – detly Sep 2 '10 at 13:12

@detly tiesreverdnatitilpsnwodgnirtsymtupi? generated via "iputmystringdownsplititandreverseit"[::-1] XD – setophaga Apr 26 at 19:33

3 Answers

```
>>> tmp = "a,b,cde"
>>> tmp2 = tmp.split(',')
>>> tmp2.reverse()
>>> "".join(tmp2)
'cdeba'
```

or simpler:

```
>>> tmp = "a,b,cde"
>>> ''.join(tmp.split(',')[::-1])
'cdeba'
```

The important parts here are the split function and the join function. To reverse the list you can use `reverse()`, which reverses the list in place or the slicing syntax `[::-1]` which returns a new, reversed list.

edited Sep 2 '10 at 13:10   answered Sep 2 '10 at 13:00
Mad Scientist
9,244 6 50 83

---

3   Alternatively, `''.join(reversed(tmp.split(',')))`, which is a little more explicit. – carl Sep 2 '10 at 16:49

Do you mean like this?

```python
import string
astr='a(b[c])d'

deleter=string.maketrans('()[]','    ')
print(astr.translate(deleter))
# a b c  d
print(astr.translate(deleter).split())
# ['a', 'b', 'c', 'd']
print(list(reversed(astr.translate(deleter).split())))
# ['d', 'c', 'b', 'a']
print(' '.join(reversed(astr.translate(deleter).split())))
# d c b a
```

answered Sep 2 '10 at 13:04

unutbu
421k   58   824   932

You mean this?

```python
from string import punctuation, digits

takeout = punctuation + digits

turnthis = "(fjskl) 234 = -345 089 abcdef"
turnthis = turnthis.translate(None, takeout)[::-1]
print turnthis
```

answered Sep 2 '10 at 13:06

Tony Veijalainen
3,424   13   27