**String cleaning**

23789% of 250636 of1,612[mkelty](http://www.codewars.com/users/mkelty)

Python

* [TRAIN AGAIN](http://www.codewars.com/kata/string-cleaning/train/python)
* [NEXT KATA](http://www.codewars.com/trainer/python)

Details

[Solutions](http://www.codewars.com/kata/string-cleaning/solutions/python)

[Forks (7)](http://www.codewars.com/kata/string-cleaning/forks/python)

[Discourse (13)](http://www.codewars.com/kata/string-cleaning/discuss/python)

* Add to Collection
* |
* Share this kata:

Your boss decided to save money by purchasing some cut-rate optical character recognition software for scanning in the text of old novels to your database. At first it seems to capture words okay, but you quickly notice that it throws in a lot of numbers at random places in the text. For example:

string\_clean('! !') == '! !'

string\_clean('123456789') == ''

string\_clean('This looks5 grea8t!') == 'This looks great!'

Your harried co-workers are looking to you for a solution to take this garbled text and remove all of the numbers. Your program will take in a string and clean out all numeric characters, and return a string with spacing and special characters ~#$%^&!@\*():;"'.,? all intact.

<http://www.codewars.com/kata/string-cleaning/python>

def **string\_clean**(s):

concat = *""*

for i in range(len( s)):

ch = s[i]

if not (*'0'* <= ch and ch <= *'9'*):

concat += ch

return concat

print string\_clean(*'This looks5 grea8t!'*)