**难度: 简单**

输入一个英文句子，翻转句子中单词的顺序，但单词内字符的顺序不变。为简单起见，标点符号和普通字母一样处理。例如输入字符串"I am a student. "，则输出"student. a am I"。

示例 1：

输入: "the sky is blue"

输出: "blue is sky the"

示例 2：

输入: "  hello world!  "

输出: "world! hello"

解释: 输入字符串可以在前面或者后面包含多余的空格，但是反转后的字符不能包括。

示例 3：

输入: "a good   example"

输出: "example good a"

解释: 如果两个单词间有多余的空格，将反转后单词间的空格减少到只含一个。

说明：

无空格字符构成一个单词。

输入字符串可以在前面或者后面包含多余的空格，但是反转后的字符不能包括。

如果两个单词间有多余的空格，将反转后单词间的空格减少到只含一个。

来源：力扣（LeetCode）

链接：https://leetcode-cn.com/problems/fan-zhuan-dan-ci-shun-xu-lcof

著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

思路:

1). 先按空格进行切割

2). 创建一个切割后长度的StringBuilder

3). 循环判断判断如果不是""就添加进去,并且加个空格

4). 删除前后所有的空格

|  |
| --- |
| public String reverseWords(String s) {  //按空格进行切割  String[] arr = s.split(" ");  //创建需要的长度StringBuilder  StringBuilder sb = new StringBuilder(arr.length);  //循环添加进去  for (int i = arr.length - 1; i >= 0; i--) {  //判断如果不是""就添加进去,并且加个空格  if (!arr[i].equals("")) {  sb.append(arr[i]).append(" ");  }  }  //删除前后所有的空格  return sb.toString().trim();  } |