

## 按单词下标区间翻转文章内容

题目描述：给定一段英文文章片段，由若干单词组成，单词间以空格间隔，单词下标从0开始。  
请翻转片段中指定区间的单词顺序并返回翻转后的内容。

例如给定的英文文章片段为"I am a developer"，翻转区间为[0,3]，则输出"developer a am I"。

```
String reverseWords(String s, int start, int end)
```

输入描述：使用换行隔开三个参数，第一个参数为英文文章内容即英文字符串，第二个参数为待翻转内容起始单词下标，第三个参数为待翻转内容最后一个单词下标。

输出描述：翻转后的英文文章片段所有单词之间以一个半角空格分隔进行输出

补充说明：英文文章内容首尾无空格

### 示例1

输入：I am a developer

1

2

输出：I a am developer

说明：

### 示例2

输入：hello world

-1

1

输出：world hello

说明：下标小于0时，从第一个单词开始

### 示例3

输入：I am a developer

0

5

输出：developer a am I

说明：下标大于实际单词个数，则按最大下标算

### 示例4

输入：I am a developer

-2

-1

输出：I am a developer

说明：翻转区间无效时，不做翻转

```
1 import java.util.Scanner;
2 import java.util.StringJoiner;
3 // 注意类名必须为 Main, 不要有任何 package xxx 信息
4 public class Main {
5     public static void main(String[] args) {
6         Scanner in = new Scanner(System.in);
7         String[] words = in.nextLine().split(" ");
8         int start = Integer.parseInt(in.nextLine());
9         int end = Integer.parseInt(in.nextLine());
10        System.out.println(reverseWords(words,start,end));
11    }
12
13    private static String reverseWords(String[] words, int start, int end) {
14        int len = words.length;
15        //如果起止下标均为有效值, 交换范围内的字符串位置
16        if(start < len && end >= 0 && start < end){
17            //下标越界的处理
18            start = Math.max(start,0);
19            end = Math.min(end,len-1);
20            //交换
21            while(start < end){
22                String temp = words[start];
23                words[start] = words[end];
24                words[end] = temp;
25                start++; end--;
26            }
27        }
28        //将数据拼接为结果返回
29        StringJoiner sj = new StringJoiner(" ");
30        for (String word : words) {
31            sj.add(word);
32        }
33        return sj.toString();
34    }
35 }
```