**难度: 简单**

URL化。编写一种方法，将字符串中的空格全部替换为%20。假定该字符串尾部有足够的空间存放新增字符，并且知道字符串的“真实”长度。（注：用Java实现的话，请使用字符数组实现，以便直接在数组上操作。）

示例1:

输入："Mr John Smith ", 13

输出："Mr%20John%20Smith"

示例2:

输入：" ", 5

输出："%20%20%20%20%20"

提示：

字符串长度在[0, 500000]范围内。

来源：力扣（LeetCode）

链接：https://leetcode-cn.com/problems/string-to-url-lcci

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:**

1). 创建一个StringBuilder

2). 循环length的长度

3). 判断字符是否等于空格, 如果等于空格就直接把%20加入到StringBuilder里面,不等于空格则把字符添加到StringBuilder里面,

4). 最后返回StringBuilder的toString

**注意:** 不要直接使用String类型一个一个加 例如: str += “%20”, 这样会超出时间限制

|  |
| --- |
| public String replaceSpaces(String S, int length) {  StringBuilder str = new StringBuilder();  for (int i = 0; i < length; i++) {  if (S.charAt(i) == ' ') {  str.append("%20");  }else {  str.append(S.charAt(i));  }  }  return str.toString();  } |

**思路2:**

1). 先将字符串转换成字符数组

2). 创建一个新的字符数组, 大小为length的长度\*3, 因为如果是空格的话,需要将空格转换成%20

3). 定义一个int 类型的变量, 用来标记最后添加了多少个到新的数组中

4). 循环遍历字符串转换的数组,判断是否是空格

1.如果是, 依次添加%20,

2.如果不是,直接将哪个字符添加进去新的字符数组就行了

注意: 定义的int类型变量,每次都要++;

5) 最后返回新的数组就行了

|  |
| --- |
| public String replaceSpaces2(String S, int length) {  //转换成字符数组  char[] chars = S.toCharArray();  //创建新的数组  char[] newChar = new char[length \* 3];  int count = 0; //标记newChar最终的下标    //循环遍历  for (int i = 0; i < length; i++) {  //判断是不是空格  if (chars[i] == ' ') { //是  newChar[count++] = '%';  newChar[count++] = '2';  newChar[count++] = '0';  }else {  //不是  newChar[count++] = chars[i];  }  }  return new String(newChar,0,count);  } |