

# Java方向编程题答案

## 第八周

### day44

#### 846 骆驼命名法

链接: <https://www.nowcoder.com/questionTerminal/aed1c7bbc2604e7c9661a2348b0541b8?pos=46&mutiTagIds=579&orderByHotValue=1>

##### 【题目解析】:

题目描述比较好理解, 仔细读题即可.

##### 【解题思路】:

首先一个字符一个字符的读取内容

1. 遇到 \_ 就直接跳过
2. 如果上一个字符是 \_ 则下一个字符转大写字母

```
// 思路很简单.
// 首先一个字符一个字符的读取内容
// 1. 遇到 _ 就直接跳过
// 2. 如果上一个字符是 _ 则下一个字符转为大写字母

import java.util.*;

public class Main{
    public static void main(String[] args){
        Scanner scanner=new Scanner(System.in);
        while(scanner.hasNext()){
            String str=scanner.nextLine();
            // 先将字符串按 _ 切分
            String[] words=str.split("_");
            String result=words[0];
            for(int i=1;i<words.length;i++){
                // 将首字母转为大写即可.
                result+=(char)(words[i].charAt(0)-'a'+'A');
                for(int j=1;j<words[i].length();j++){
                    result+=words[i].charAt(j);
                }
            }
            System.out.println(result);
        }
    }
}
```

#### 36855 单词倒排

链接: <https://www.nowcoder.com/questionTerminal/81544a4989df4109b33c2d65037c5836>

### 【题目解析】：

这个问题是包含了字符串常见操作的 切分 和 合并 虽然曾经 C 语言课上讲过类似的问题, 但是这个问题的变化是对单词分割符不设限制, 只要非字母, 都需要当成分割符, 并且分割符的数目不受限制.

### 【解题思路】：

我们整体的思路仍然是

1. 先切分(切分前先对分割符做处理, 统一分割符), 此处使用 split 进行切分, 用正则表达式作为分割符
2. 再合并(对切分结果逆序合并) 直接字符串拼接合并即可.

```
// 这个问题是包含了字符串常见操作的 切分 和 合并
// 核心是在于 split 的用法
import java.util.*;
public class Main{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        while(sc.hasNext()){
            String str=sc.nextLine();
            // 使用正则表达式作为分割符.
            String []strArray=str.split("[^a-zA-Z]+");
            for(int i=strArray.length-1;i>=2;i--){
                System.out.print(strArray[i]+' ');
            }
            if(strArray[0].length()==0)//如果字符串数组的第一个元素是空串, 那么下标为1的元素就是最后一个
            要输出的元素, 末尾不要再加空格
                System.out.println(strArray[1]);
            else
                System.out.println(strArray[1]+' '+strArray[0]);
        }
    }
}
```