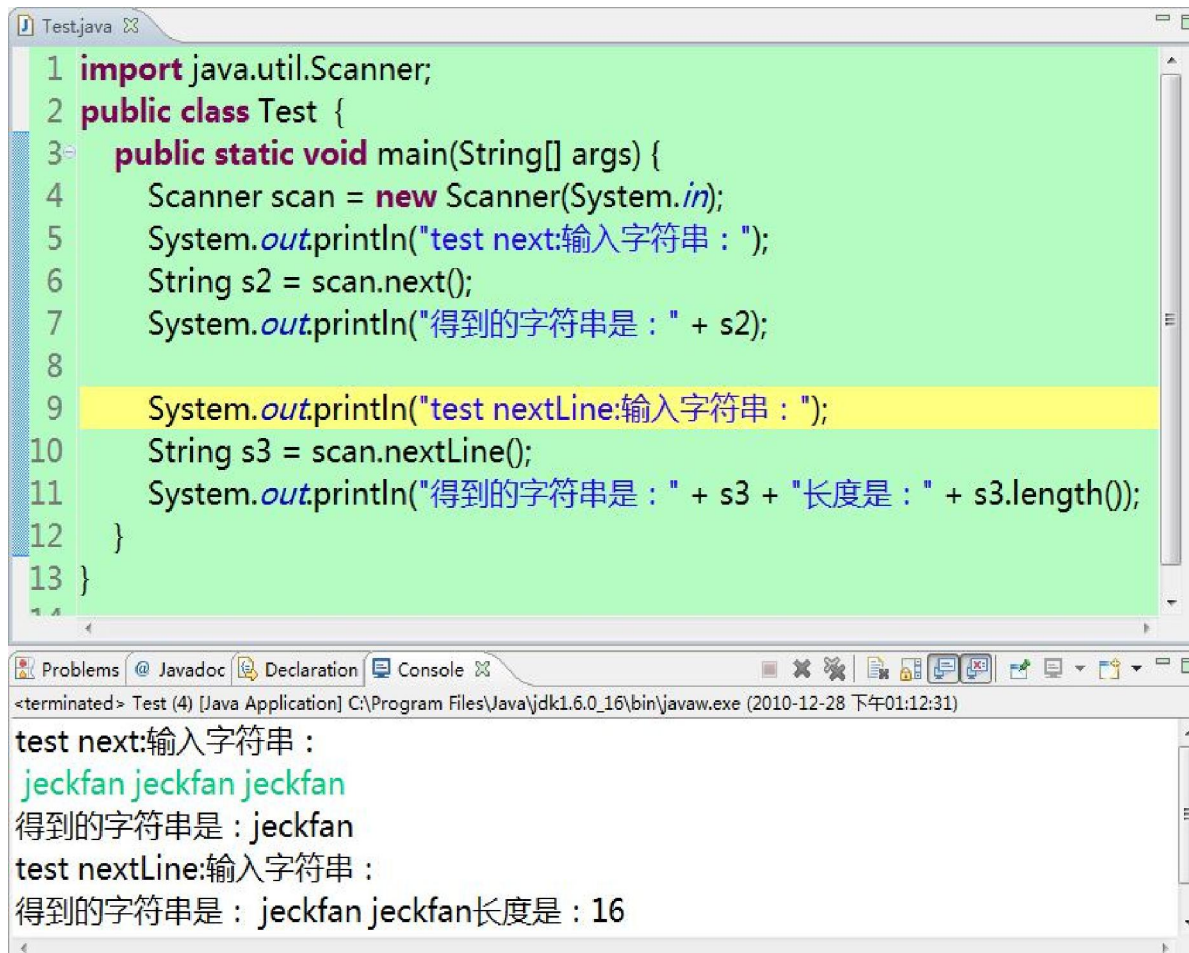


## 扈建峰 简略解析 java Scanner 中的 next() 方法和 nextLine() 方法：

让我们首先来看一个我编的例子吧：



```
1 import java.util.Scanner;
2 public class Test {
3     public static void main(String[] args) {
4         Scanner scan = new Scanner(System.in);
5         System.out.println("test next:输入字符串 :");
6         String s2 = scan.next();
7         System.out.println("得到的字符串是 : " + s2);
8
9         System.out.println("test nextLine:输入字符串 :");
10        String s3 = scan.nextLine();
11        System.out.println("得到的字符串是 : " + s3 + "长度是 : " + s3.length());
12    }
13 }
```

test next:输入字符串 :  
jeckfan jeckfan jeckfan  
得到的字符串是 : jeckfan  
test nextLine:输入字符串 :  
得到的字符串是 : jeckfan jeckfan长度是 : 16

说明：我输入的是“ jeckfan jeckfan jeckfan”，前面加了一个空格。

看到结果了吧？也许你会问，为什么当我调用 `next()`方法时，它让我输入字符串，我输入了三个 `jeckfan` 却只是打印出了一个；而当我再接着调用 `nextLine()`方法时，没用我输入它就自动打印出了上面的后两个 `jeckfan` 呢？下面，咱们就一起来探讨一下吧！

首先，我们要弄明白，`next` 方法的使用规则：当 `next` 方法遇见第一

一个有效字符（非空格、换行字符）时，开始执行扫描，而当遇见第一个分隔符或结束符（空格或换行符）时，结束扫描。这里，你看，当我一连串输入三个 `jeckfan` 时，其实 `next` 方法只执行到第一个 `jeckfan` 碰到第一个空格就结束了，不会再往下扫描，也就是它只接收了第一个 `jeckfan`。而从第一个 `jeckfan` 后的空格算起，就交给 `nextLine` 来扫描了，也就是说，`nextLine` 方法接收当前行剩下的所有的字符串，一直到你敲回车为止。

大家想到了吧？`Scanner` 中的 `next` 和 `nextLine` 方法最好不要联合使用，因为它们如果联合使用的话，它们就是一个整体了，你不要想当然的以为这两个方法一前一后调用时是各自孤立的。也就是说，在这里，`next` 方法和 `nextLine` 方法（`next` 在前，`nextLine` 在后）扫描接收的是同一行上的字符串，只不过接收的是不同部分罢了。不知道聪明的你是不是明白了我以上所说的内容。