

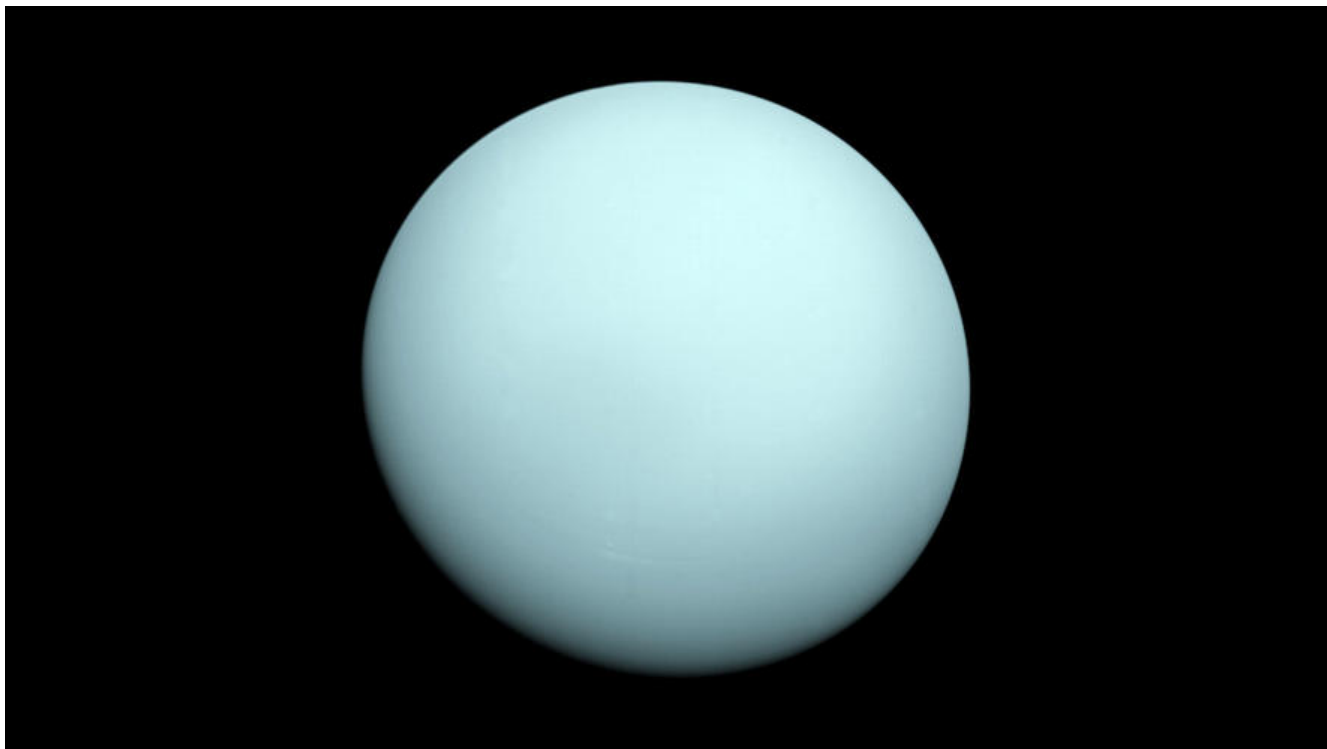
Spring 类中的一些常用方法

作者: [howie404](#)

原文链接: <https://ld246.com/article/1566581448663>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



1 length () 字符串的长度

```
1 public static void main(String[] args) {  
2     String str = "Good Boy";  
3     System.out.println(str.length());  
4 }
```

输出的结果是字符串长度为8。

2 substring () 截取字符串

```
1 public static void main(String[] args) {  
2     String str = "Good Boy";  
3     System.out.println(str.substring(5));  
4     System.out.println(str.substring(0, 5));  
5 }
```

输出的结果第一条为 "Boy" ,参数5 (beginIndex) 是开始截取的位置。

输出的结果第二条是 "Good (此处有一空格)" ,第一个参数0 (beginIndex) 是开始截取的位置, 第二个参数5 (endIndex) 是截取结束的位置 (含头不含尾) 。

3 equals()和equalsIgnoreCase()比较两个字符串是否相等 前者区分大小写, 后者不区分大小写

```
1 public static void main(String[] args) {  
2     String a = "Good Boy";  
3     String b = "good boy";  
4     System.out.println(a.equals(b));  
5     System.out.println(a.equalsIgnoreCase(b));  
}
```

```
6 }
```

输出的结果为第一条为false，第二条为true。

4 startsWith()和endsWith()判断字符串是不是以特定的字开头或结束（区分大小写）

```
1 public static void main(String[] args) {
2     String a = "Good Boy";
3     String b = "good boy";
4     System.out.println(a.startsWith("G"));
5     System.out.println(a.startsWith("g"));
6     System.out.println(a.startsWith("B", 5));
7     System.out.println(b.endsWith("Y"));
8     System.out.println(b.endsWith("y"));
9 }
```

输出结果依次为：

truefalsetruefalsetrue

a.startsWith(String prefix,int toffset)用于判断特定的下标是否为指定字符串中开始查找的位置

5 indexOf()和lastIndexOf()前者是查找字符或字符串第一次出现的地方，后者是查找字符或字符串最后一次出现的地方

```
1 public static void main(String[] args) {
2     String a = "Good Boy";
3     System.out.println(a.indexOf("o"));
4     System.out.println(a.lastIndexOf("o"));
5 }
```

输出结果：

16

6 getChars () 截取多个字符并由其他字符串接收

```
1 public static void main(String[] args) {
2     String a = "Good Boy";
3     char[] b = new char[10];
4     char[] c = new char[10];
5     a.getChars(0, 3, b, 0);
6     a.getChars(0, 4, c, 0);
7     System.out.println(b);
8     System.out.println(c);
9 }
```

输出的结果：

Goo

Good

其中第一个参数0是要截取的字符串的初始下标 (int sourceStart) , 第二个参数是要截取的字符串结束后的下一个下标 (int sourceEnd) 也就是实际截取到的下标是int sourceEnd-1, 第三个参数是收的字符串 (char target[]) , 最后一个参数是接收的字符串开始接收的位置。

7 getBytes()将字符串变成一个byte数组

```
1 public static void main(String[] args) {  
2     String a = "Good Boy";  
3     byte b [] = a.getBytes();  
4     System.out.println(new String(b));  
5 }
```

输出的结果为Good Boy的byte数组。

8 toUpperCase()和toLowerCase()将字符串转换为大写或小写

```
1 public static void main(String[] args) {  
2     String a = "Good Boy";  
3     System.out.println(a.toLowerCase());  
4     System.out.println(a.toUpperCase());  
5 }
```

输出的结果第一条为 "good boy" , 第二条为 "GOOD BOY" 。

9 concat() 连接两个字符串

```
1 public static void main(String[] args) {  
2     String a = "Good Boy";  
3     String b = "You Are A ";  
4     System.out.println(b.concat(a));  
5 }
```

输出的结果为 "You Are A Good Boy" 。

10 trim()去掉起始和结束的空格

```
1 public static void main(String[] args) {  
2     String a = " Good Boy ";  
3     System.out.println(a.trim());  
4 }
```

输出的结果为 "Good Boy" 。

11 charAt () 截取一个字符

```
1 public static void main(String[] args) {  
2     String a = "Good Boy";
```

```
3     System.out.println(a.charAt(1));
4 }
```

输出的结果是字符串a的下标为1的字符o。

12 toCharArray()将字符串变成一个字符数组

```
1 public static void main(String[] args) {
2     String a = "Good Boy";
3     char b[] = a.toCharArray();
4     System.out.println(b);
5 }
```

输出结果： Good Boy

13 split 字符串分割

```
1 public static void main(String[] args) {
2     String a = "Good Boy";
3     String b[] = a.split(" ");
4     System.out.println(b.length);
5     for (int i = 0; i < b.length; i++) {
6         System.out.println(b[i]);
7     }
8 }
```

输出结果：

2GoodBoy

14 replace() 替换

```
1 public static void main(String[] args) {
2     String a = "Good Boy";
3     String b = "好孩子";
4     System.out.println(a.replace(a, b));
5     System.out.println(b.replace("好", "你是个"));
6 }
```

输出的结果：

好孩子你是个孩子