

# String类常用方法

## 1.判断功能方法

```
//创建字符串对象↵  
String s1 = "hello";↵  
String s2 = "hello";↵  
String s3 = "Hello";↵
```

`equals(Object obj)`

比较字符串的内容是否相同

```
//boolean equals(Object obj):比较字符串的内容是否相同↵  
System.out.println(s1.equals(s2));↵  
System.out.println(s1.equals(s3));↵
```

`equalsIgnoreCase(String anotherString)`

将此 String 与另一个 String 比较, 不考虑大小写。

```
//boolean equalsIgnoreCase(String str):比较字符串的内容是否相同, 忽略大  
System.out.println(s1.equalsIgnoreCase(s2));↵
```

`compareTo(String anotherString)`

按字典顺序比较两个字符串。

`compareToIgnoreCase(String str)`

按字典顺序比较两个字符串, 不考虑大小写。

`startsWith(String prefix)`

测试此字符串是否以指定的前缀开始。

```
//boolean startsWith(String str):判断字符串对象是否以指定的 str 开头↵  
System.out.println(s1.startsWith("he"));↵  
System.out.println(s1.startsWith("ll"));↵
```

`startsWith(String prefix, int toffset)`

测试此字符串从指定索引开始的子字符串是否以指定前缀开始。

`endsWith(String suffix)`

测试此字符串是否以指定的后缀结束。

## 2.获取功能方法

```
//创建字符串对象↵  
String s = "helloworld";↵
```

`charAt(int index)`

返回指定索引处的 char 值。

```
//char charAt(int index):获取指定索引处的字符↵  
System.out.println(s.charAt(0));↵  
System.out.println(s.charAt(1));↵
```

`length()`

返回此字符串的长度。

```
//int length():获取字符串的长度, 其实也就是字符个数↵  
System.out.println(s.length());↵
```

`indexOf(String str)`

返回指定子字符串在此字符串中第一次出现处的索引。

```
//int indexOf(String str):获取 str 在字符串对象中第一次出现的索引, 没有返回  
System.out.println(s.indexOf("l"));↵  
System.out.println(s.indexOf("owo"));↵
```

`indexOf(String str, int fromIndex)`

返回指定子字符串在此字符串中第一次出现处的索引, 从指定的索引开始

`String substring(int start)`

从start开始截取字符串到字符串结尾

```
//String substring(int start):从 start 开始截取字符串到字符串结尾↵  
System.out.println(s.substring(0));↵  
System.out.println(s.substring(5));↵
```

### String substring(int start,int end)

从start开始，到end结束截取字符串;包括start, 不包括end

```
//String substring(int start,int end):从 start 开始，到 end 结束截取字符  
包括 start, 不包括 end↵  
System.out.println(s.substring(0, s.length()));↵  
System.out.println(s.substring(3,8));↵
```

### replace(char oldChar, char newChar)

返回一个新的字符串，它是通过用 newChar 替换此字符串中出现的所有 oldChar 得到的。

### 3.转换功能

#### char[] toCharArray()

将此字符串转换为一个新的字符数组。

```
//char[] toCharArray():把字符串转换为字符数组↵  
char[] chs = s.toCharArray();↵  
for(intx=0; x<chs.length; x++) {↵  
    System.out.println(chs[x]);↵  
}↵
```

#### toLowerCase()

使用默认语言环境的规则将此 String 中的所有字符都转换为小写。

```
//String toLowerCase():把字符串转换为小写字母↵  
System.out.println("HelloWorld".toLowerCase());↵
```

#### toUpperCase()

使用默认语言环境的规则将此 String 中的所有字符都转换为大写。

```
//String toUpperCase():把字符串转换为大写字母↵  
System.out.println("HelloWorld".toUpperCase());↵
```

### 4.去空格和分割

#### trim()

返回字符串的副本，忽略前导空白和尾部空白。

#### String[] split(String str)

按照指定符号分割字符串。分割之后变成字符串数组。

```
//String[] split(String str)↵  
//创建字符串对象↵  
String s4 = "aa.bb.cc";↵  
String[] strArray = s4.split(",");↵  
for(intx=0; x<strArray.length; x++) {↵  
    System.out.println(strArray[x]);↵  
}↵
```