



一、字符串基础 (String)



二、常用字符串方法汇总 (重点掌握)

方法名	作用	常见场景
<code>.length</code>	获取字符串长度	统计字数、判断是否为空
<code>.charAt(n)</code>	获取指定位置的字符	提取某个字符
<code>.slice(start, end)</code>	截取子字符串	提取某段文字、裁剪用户输入
<code>.substring(start, end)</code>	同上	功能和 <code>.slice()</code> 类似
<code>.substr(start, length)</code>	按长度截取	裁剪固定长度的用户名
<code>.indexOf(str)</code>	查找子字符串的位置	判断某个词是否存在
<code>.includes(str)</code>	是否包含子字符串	搜索关键词
<code>.toUpperCase()</code>	转为大写	统一大小写、验证邮箱等
<code>.toLowerCase()</code>	转为小写	同上
<code>.trim()</code>	去除前后空格	表单清理、用户登录验证
<code>.replace(a, b)</code>	替换内容	敏感词过滤、关键词替换
<code>.split("分隔符")</code>	拆分为数组	按逗号分割、解析标签、读取 CSV
<code>.startsWith()</code>	是否以某个字符串开头	判断标题、校验格式
<code>.endsWith()</code>	是否以某个字符串结尾	判断文件名、后缀名

三、示例讲解（每个方法详细讲解）

1 `.length`

作用： 返回字符串的长度（字数）



```
let name = "Alice";  
console.log(name.length); // 输出 5
```

 **场景：** 判断用户名是否太短或太长。

2 `.charAt(index)`

作用： 获取指定位置的字符（从0开始）



```
let str = "Hello";  
console.log(str.charAt(1)); // e
```

 **场景：** 获取验证码第一个字符是否为数字。

3 `.slice(start, end)`

作用： 从 `start` 到 `end`（不包含end）截取子字符串



```
let msg = "Welcome to JavaScript";  
console.log(msg.slice(0, 7)); // Welcome
```

💡 **场景：** 截取标题前缀或片段。

4 `.indexOf(str)`

作用： 查找字符串是否包含某内容，返回位置，找不到返回 `-1`



```
let text = "I love JavaScript!";  
console.log(text.indexOf("love")); // 2  
console.log(text.indexOf("hate")); // -1
```

💡 **场景：** 搜索关键字、判断是否出现。

5 `.includes(str)`

作用： 是否包含某字符串，返回布尔值 (`true/false`)



```
let msg = "Today is sunny";  
console.log(msg.includes("sunny")); // true
```

💡 **场景：** 判断评论中是否出现违规词。

6 `.toUpperCase()` / `.toLowerCase()`

作用： 将字符串转成全大写 / 全小写



```
let email = "User@Example.Com";  
console.log(email.toLowerCase()); // user@example.com
```

💡 **场景：**邮箱大小写不敏感比对、统一格式。

7 `.trim()`

作用：去除字符串前后空格



```
let input = "  Hello world!  ";  
console.log(input.trim()); // "Hello world!"
```

💡 **场景：**用户登录时输入有多余空格，登录失败。

8 `.replace(old, new)`

作用：替换字符串中的内容（只替换第一个）



```
let sentence = "I like JS and I like coding";  
console.log(sentence.replace("like", "love")); // I love JS and I  
like coding
```

💡 **场景：**内容审查、关键词替换。

9 `.split("分隔符")`

作用：按照分隔符将字符串拆成数组



```
let tags = "red,green,blue";  
console.log(tags.split(",")); // ["red", "green", "blue"]
```

💡 **场景：** 标签管理、CSV 数据处理。

10

.startsWith() / **.endsWith()**

作用： 判断是否以某个字符串开头或结尾



```
let filename = "report.pdf";  
console.log(filename.endsWith(".pdf")); // true
```

💡 **场景：** 校验文件类型、URL 规则判断。