

# JavaScript 注释

JavaScript 注释可用于提高代码的可读性。

## JavaScript 注释

JavaScript 不会执行注释。

我们可以添加注释来对 JavaScript 进行解释，或者提高代码的可读性。

单行注释以 `//` 开头。

本例用单行注释来解释代码：

### 实例

```
// 输出标题：  
document.getElementById("myH1").innerHTML="欢迎来到我的主页";  
// 输出段落：  
document.getElementById("myP").innerHTML="这是我的第一个段落。";
```

[尝试一下 »](#)

## JavaScript 多行注释

多行注释以 `/*` 开始，以 `*/` 结尾。

下面的例子使用多行注释来解释代码：

### 实例

```
/*  
  下面的这些代码会输出  
  一个标题和一个段落  
  并将代表主页的开始  
*/  
document.getElementById("myH1").innerHTML="欢迎来到我的主页";  
document.getElementById("myP").innerHTML="这是我的第一个段落。";
```

[尝试一下 »](#)

## 使用注释来阻止执行

在下面的例子中，注释用于阻止其中一条代码行的执行（可用于调试）：

### 实例

```
// document.getElementById("myH1").innerHTML="欢迎来到我的主页";  
document.getElementById("myP").innerHTML="这是我的第一个段落。";
```

[尝试一下 »](#)

在下面的例子中，注释用于阻止代码块的执行（可用于调试）：

### 实例

```
/*
document.getElementById("myH1").innerHTML="欢迎来到我的主页";
document.getElementById("myP").innerHTML="这是我的第一个段落。";
*/
```

[尝试一下 »](#)

## 在行末使用注释

在下面的例子中，我们把注释放到代码行的结尾处：

### 实例

```
var x=5; // 声明 x 并把 5 赋值给它
var y=x+2; // 声明 y 并把 x+2 赋值给它
```

[尝试一下 »](#)[← JavaScript 语句](#)[JavaScript 变量 →](#)**2 篇笔记**[📝 写笔记](#)

养成一个好的习惯就是写注释。

方便你二次阅读，维护你的代码。

如果后面项目转手，也方便别人来理解你的代码和维护你的代码。

**Answer** 2年前 (2017-06-30)



### 应用注释符号验证浏览器是否支持 JavaScript 脚本功能

如果用户不能确定浏览器是否支持JavaScript脚本，那么可以应用HTML提供的注释符号进行验证。HTML注释符号是以 <!-- 开始以 --> 结束的。如果在此注释符号内编写 JavaScript 脚本，对于不支持 JavaScript 的浏览器，将会把编写的 JavaScript 脚本作为注释处理。

使用 JavaScript 脚本在页面中输出一个字符串，将 JavaScript 脚本编写在 HTML 注释中，如果浏览器支持 JavaScript 将输出此字符串，如果不支持将不输出此字符串，代码如下：

```
<script>
<!--
document.write("您的浏览器支持JavaScript脚本!");
```

```
//-->  
</script>
```

**注意：**注释行结尾处的两条斜杠 `//` 是 JavaScript 注释符号。这可以避免 JavaScript 执行 `-->` 标签。

**islander** 2年前 (2017-08-17)