



猿编程

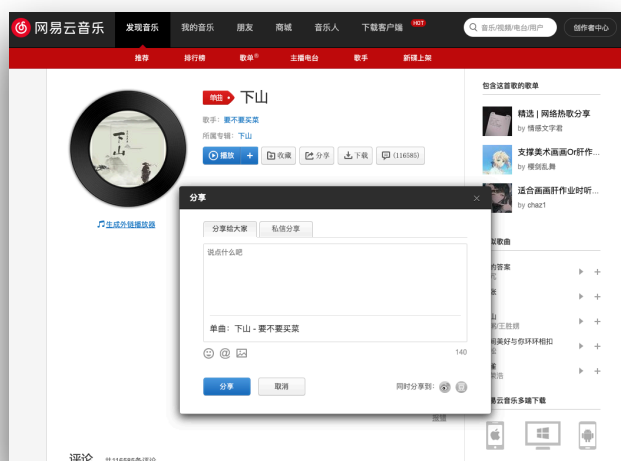
# 课程知识总结

Course knowledge summary

Unit05 JavaScript基础

## I JavaScript介绍

JavaScript简称JS，是一种编程语言，可用于开发网页和APP，为页面添加各种动态功能，实现用户与页面之间的交互。



## I JavaScript的用法

在html文件中使用JavaScript有两种方式：

- 1 在html文件中，引入单独的.js文件，使用被引入的.js文件中的JavaScript代码。  
例如：mui.min.js（mui资源文件）
- 2 在html文件中，插入JavaScript代码。





## 知识梳理

### I 设置.js路径并编写js代码

在html文件的head或 body中使用<script></script>，在<script>的src属性中设置 .js文件的路径。

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width,init
6   <title></title>
7   <link href="css/mui.min.css" rel="stylesheet"/>
8 </head>
9 <body>
10
11 <header class="mui-bar mui-bar-nav">
12   <h1 class="mui-title">JavaScript基础知识</h1>
13 </header>
14
15
16 <script src="js/mui.min.js"></script>
17
18 <script type="text/javascript" charset="utf-8">
19
20   // 初始化mui
21   mui.init()
22
23 </script>
24
25 </body>
26 </html>
```

在<script>和</script>之间编写JS代码。

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width,init
6   <title></title>
7   <link href="css/mui.min.css" rel="stylesheet"/>
8 </head>
9 <body>
10
11 <header class="mui-bar mui-bar-nav">
12   <h1 class="mui-title">JavaScript基础知识</h1>
13 </header>
14
15
16 <script src="js/mui.min.js"></script>
17
18 <script type="text/javascript" charset="utf-8">
19
20   // 初始化mui
21   mui.init()
22
23 </script>
24
25 </body>
26 </html>
```





## 知识梳理

### JavaScript数据类型

JavaScript中也存在基本数据类型，例如：

数据类型	示例	说明
字符串	"hello" , '小猿'	双引号，单引号
数值	10 , 4.5	包含整数，浮点数
null	null	表示“空”
...	...	...

### 警告框:alert()

alert()方法用于在页面中弹出一个警告框，该警告框内可以显示一条指定信息。

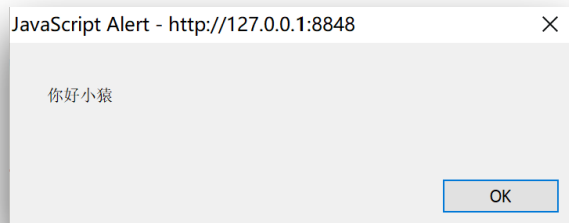
alert( message )



函数名



在警告框中要显示的信息可以是数字、字符串等



alert("你好小猿")



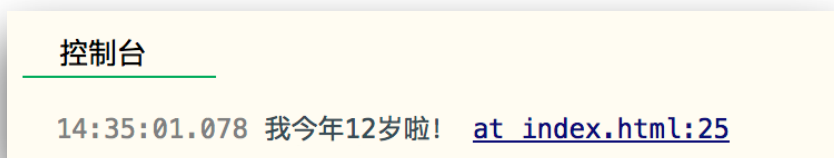


## 知识梳理

### I 控制台输出

`console.log()` 方法可以在控制台输出内容，常用于开发者测试程序的运行结果。

`console.log( message )`  
↓  
要输出的内容

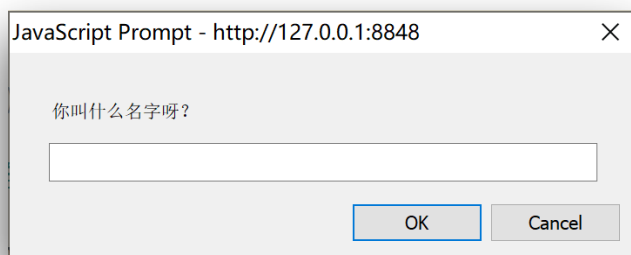


`console.log("我今年12岁啦!")`

### I 输入框:prompt()

`prompt()` 方法用于在页面中弹出一个输入框，该对话框可以提示用户输入信息。

`prompt( message )`  
↓      ↓  
函数名    在输入框中要  
          显示的提示文  
          字，是字符串



`prompt("你叫什么名字呀?")`

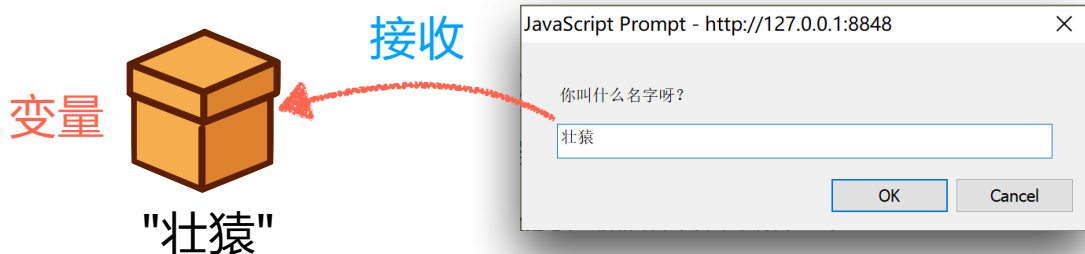




## 知识梳理

### 变量

使用变量接收输入框中的信息。



在JS中，定义变量的格式如下：

var   变量名   =   值

1                      2                      3

例如: `var name = "壮猿"`

- 1 `var`是variable(变量)的缩写，是JS中用于定义变量的关键字。
- 2 变量名由字母、数字、`_`和`$`组成，不能以数字开头，并且不能与其他关键字重名。
- 3 变量中存储的值可以是任意类型的。

给变量重新赋值，不需要使用var，例如：

```
var name = "小猿"
name = "壮猿"
```





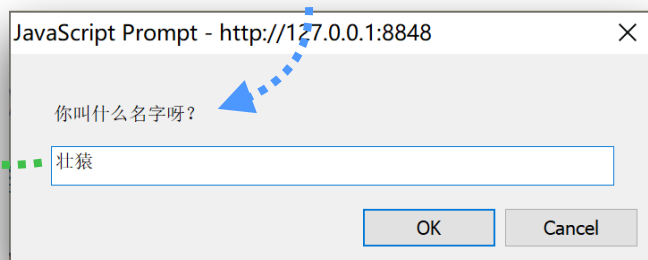
## 知识梳理

### I 接收输入框中的信息

接收输入框中的信息，返回的是字符串类型的数据。

```
var name = prompt("你叫什么名字呀?")
```

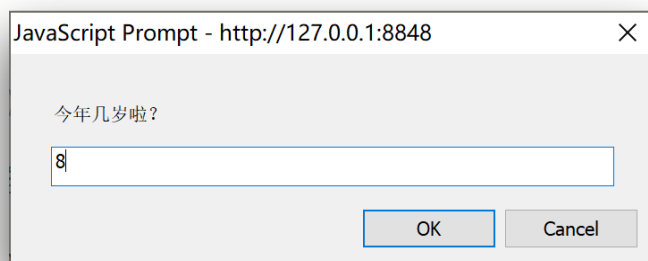
返回的是  
字符串'壮猿'



如果输入框中输入的是数字，返回的也是字符串类型的数字。

```
var name = prompt("今年几岁啦?")
```

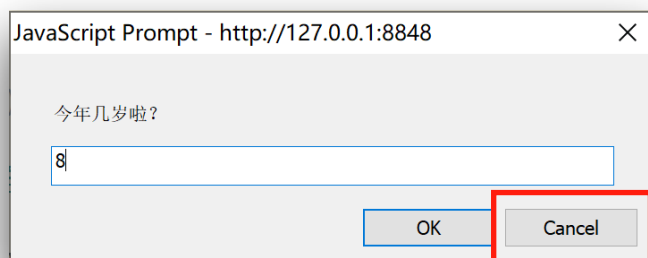
返回的是  
字符串'8'



如果点了Cancel，返回的是null。

```
var name = prompt("今年几岁啦?")
```

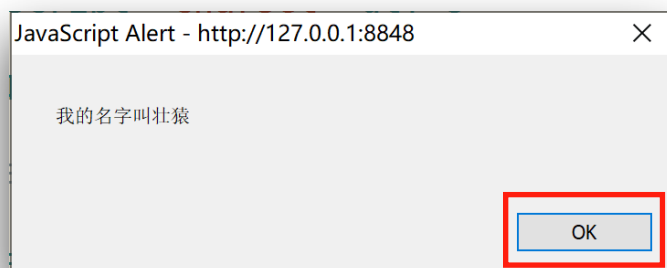
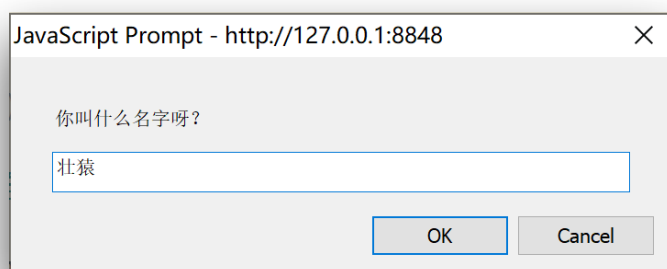
返回的是null，  
表示空



## 拼接字符串

如果要将接收到的输入框的信息（字符串）和其它字符串进行拼接并输出，需要用到“+”运算符。

例如：



```
<script>
  ...
  var xm = prompt("你叫什么名字呀? ")
  alert("我的名字叫" + xm)
</script>
```







## 知识梳理

### I "+"运算符总结

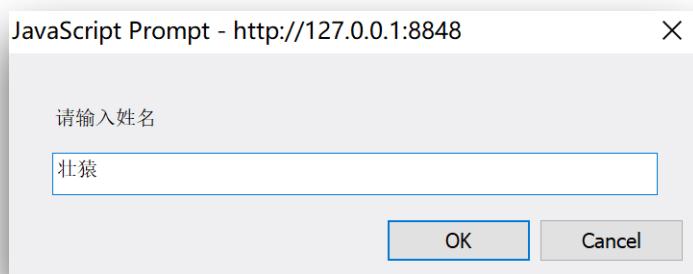
在JS中，"+"运算符可以用于字符串拼接、运算、以及运算和拼接的结合使用。

作用	举例	说明
拼接字符串	<pre>"壮猿" + "最帅" ==&gt; "壮猿最帅" "放假啦" + 666 ==&gt; "放假啦666" 666 + "55" ==&gt; "66655"</pre>	"+"只有一侧为字符串
加法运算	<pre>10 + 20 ==&gt; 30 100 + 666 ==&gt; 766</pre>	"+"左右两侧均为数值



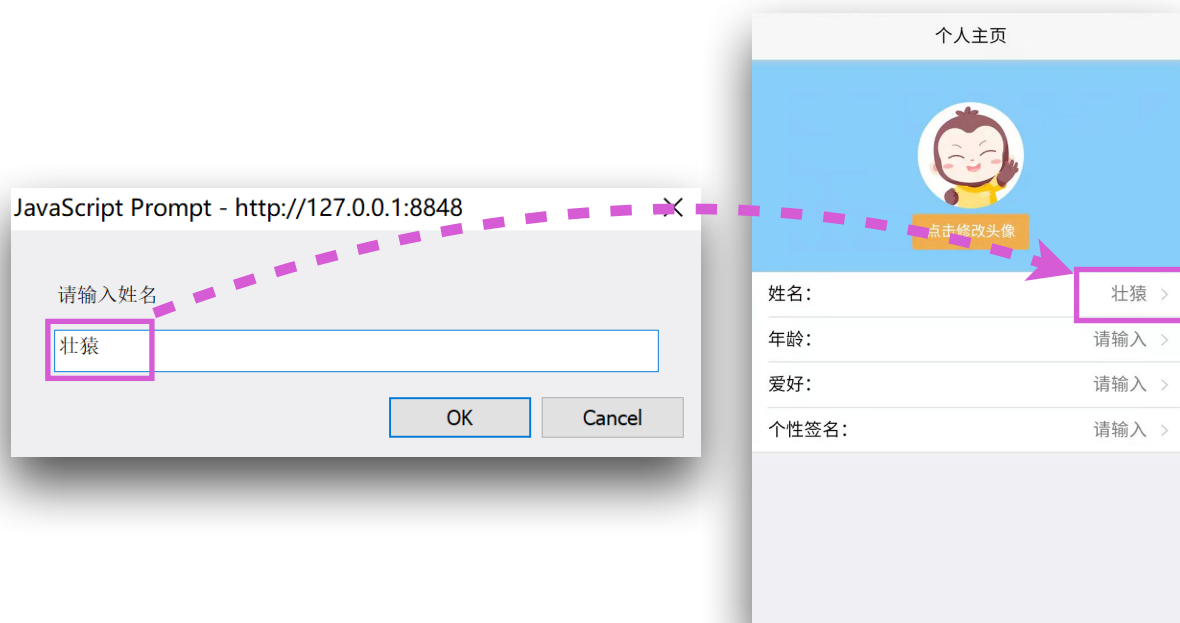
## I 需求分析

在<script>标签中添加代码，实现弹出输入弹框，让用户输入姓名。



```
<script>
...
//输入姓名
var xmText = prompt("请输入姓名:")
</script>
```

接下来，还需要将输入的姓名显示在个人主页上。



## I 添加<span>标签

要让个人信息显示到右侧，需要先在<a>标签中添加<span>标签，用于显示个人信息。



要将<span>设置到列表右边，只需要按照mui组件库的说明，设置类的值为“mui-pull-right”即可。

```
<!-- 个人信息列表部分 -->
<ul class="mui-table-view">
  <li class="mui-table-view-cell">
    <a class="mui-navigate-right">
      姓名: <span id="xm" class="mui-pull-right">请输入</span>
    </a>
  </li>
  <li class="mui-table-view-cell">
    <a class="mui-navigate-right">
      年龄: <span id="nl" class="mui-pull-right">请输入</span>
    </a>
  </li>
  <li class="mui-table-view-cell">
    <a class="mui-navigate-right">
      爱好: <span id="ah" class="mui-pull-right">请输入</span>
    </a>
  </li>
  <li class="mui-table-view-cell">
    <a class="mui-navigate-right">
      个性签名: <span id="qm" class="mui-pull-right">请输入</span>
    </a>
  </li>
</ul>
```

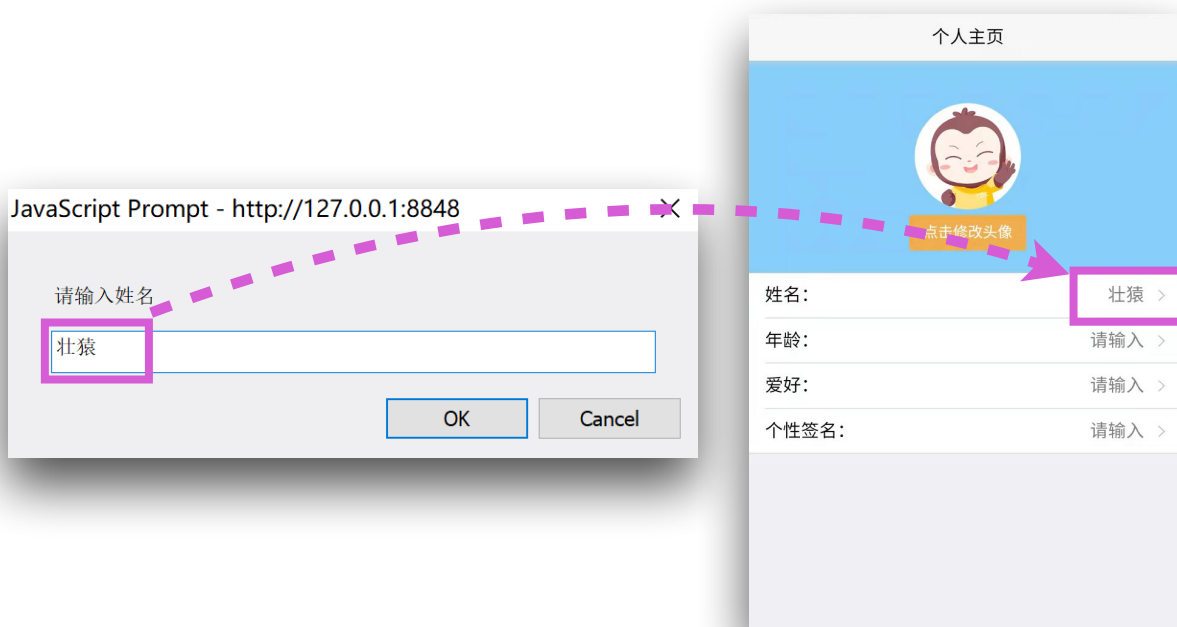




## 知识梳理

### 需求分析

将输入的内容显示到页面，实际上就是要修改`<span>`标签中的内容为输入框返回的值。



```
<!-- 个人信息列表部分 -->
<ul class="mui-table-view">
  <li class="mui-table-view-cell">
    <a class="mui-navigate-right">
      姓名: <span id="xm" class="mui-pull-right">请输入</span>
    </a>
  </li>
  ...
</ul>
</div>

<script src="js/mui.min.js"></script>
<script type="text/javascript" charset="utf-8">
  mui.init();

  //输入姓名
  var xmText = prompt("请输入姓名: ")
```





## 知识梳理

### I 修改元素的内容

使用JS代码，修改标签内容，需要以下两个步骤：

- 1 查找页面中指定的标签（获取元素）
- 2 修改标签（元素）内容

要想实现这两个步骤，需要用到JS中的对象。

### I 生活中的对象

在生活中，对象表示一个具体的事物，例如一个人就是一个对象。每个对象都包含：特征和功能。



对象

姓名：lili  
年龄：12  
身高：150cm

特征

唱歌  
跳舞  
弹琴

功能

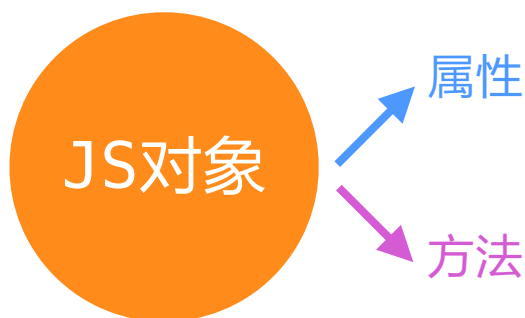




## 知识梳理

### I JS中的对象

JS中对象是一种数据，用于表示程序中的某个事物。每个对象都可以包含属性（特征）和方法（功能）。



当一个html页面载入浏览器后，程序就会自动生成document对象。document对象提供了一些属性和方法，可以用来操作当前html页面。



页面的标题  
属性: 页面最后被修改的日期和时间  
.....

向页面写入内容  
方法: 查找页面中指定的标签（获取元素）  
.....





## 知识梳理

### 获取对象中属性的值

获取对象中属性的值，格式：**对象.属性名**

例如，获取页面的标题：

```
var text = document.title
```

**对象**

**属性名：该属性存储当前页面的标题**

注意，不能访问对象中不存在的属性。

### 修改对象中属性的值

修改对象中属性的值，格式：**对象.属性名 = 新值**

例如，修改页面标题：

```
document.title = "新标题"
```

**对象**

**属性名：该属性存储当前页面的标题**

**新值**





## 知识梳理

### 调用对象中的方法

调用对象中的方法，格式：对象.方法名()



调用方法时，必须使用 ()

例如，获取页面中指定标签（获取元素）：

document.getElementById("标签的id属性值")



对象



获取标签的方法名



标签的id属性值

使用getElementById()方法返回的是指定id的元素对象，元素对象也有属性和方法。

以下语句中，获取的是id为“xm”的标签元素，该标签对象被保存在xmSpan变量中。

```
var xmSpan = document.getElementById("xm")
```







## 知识梳理

### I 获取<span>元素

在<script>标签中补充代码，获取id为“xm”的<span>元素。

姓名：	<input type="text" value="请输入"/>	<span>
年龄：	<input type="text" value="请输入"/>	
爱好：	<input type="text" value="请输入"/>	
个性签名：	<input type="text" value="请输入"/>	

```
<ul class="mui-table-view">
  <li class="mui-table-view-cell">
    <a class="mui-navigate-right">
      姓名: <span id="xm" class="mui-pull-right">请输入</span>
    </a>
  </li>
  ...
</ul>
<script>
  ...
  //输入姓名
  var xmText = prompt("请输入姓名:")
  //获取id为"xm"的<span>元素
  var xmSpan = document.getElementById("xm")
</script>
```

获取到<span>元素之后，接下来就需要将元素的内容修改为输入框中输入的内容，即xmText变量中保存的内容。





## 知识梳理

### I 修改元素的内容

通过元素对象的innerHTML属性，可以获取该元素的内容，格式如下：

元素.innerHTML

例： var c = xmSpan.innerHTML

变量c中保存的值为字符串“请输入”。

如果要修改元素的内容，只需要将新的值赋值给该元素即可，格式如下：

元素.innerHTML = 新值

例： xmSpan.innerHTML = "壮猿"

修改后，xmSpan中保存的内容就变为“壮猿”。





## 知识梳理

### I 修改元素的内容

在<script>标签中补充代码，修改元素的值。

修改为

姓名:	请输入 >
年龄:	请输入 >
爱好:	请输入 >
个性签名:	请输入 >

姓名:	壮猿 >
年龄:	请输入 >
爱好:	请输入 >
个性签名:	请输入 >

```
<ul class="mui-table-view">
  <li class="mui-table-view-cell">
    <a class="mui-navigate-right">
      姓名: <span id="xm" class="mui-pull-right">请输入</span>
    </a>
  </li>
  ...
</ul>
<script>
  ...
  //输入姓名
  var xmText = prompt("请输入姓名:")
  //获取id="xm"的<span>元素
  var xmSpan = document.getElementById("xm")
  //修改xmSpan元素对象的内容为输入的信息
  xmSpan.innerHTML = xmText
</script>
```





## 词汇总结

单词	音标	含义
variable	['veriəbl]	变量，易变的
alert	[ə'leɜ:rt]	警告，警示，警惕的
console	['kɑ:nsoʊl]	控制台
log	[lɔ:g]	日志，记录
prompt	[pra:mpɪt]	提示，题词
document	['dɑ:kjʊmənt]	文档，文件
element	['elɪmənt]	元素
inner	['ɪnər]	内部，内部的

