# JavaScript学习

## 1、为什么学习JavaScript

### 一、学习的价值

1. 所有主流浏览器都支持JavaScript。

2. 目前，全世界大部分网页都使用JavaScript。

3. 它可以让网页呈现各种动态效果。

4. 做为一个Web开发师，如果你想提供漂亮的网页、令用户满意的上网体验，JavaScript是必不可少的工具。

### 二、易学行

1.学习环境无外不在，只要有文本编辑器，就能编写JavaScript程序。

2.我们可以用简单命令，完成一些基本操作。

### 三、从哪开始学习呢？

学习JavaScript的起点就是处理网页，所以我们先学习基础语法和如何使用DOM进行简单操作。

## 2、JavaScript应该放在哪里

### 一、放入html文件中

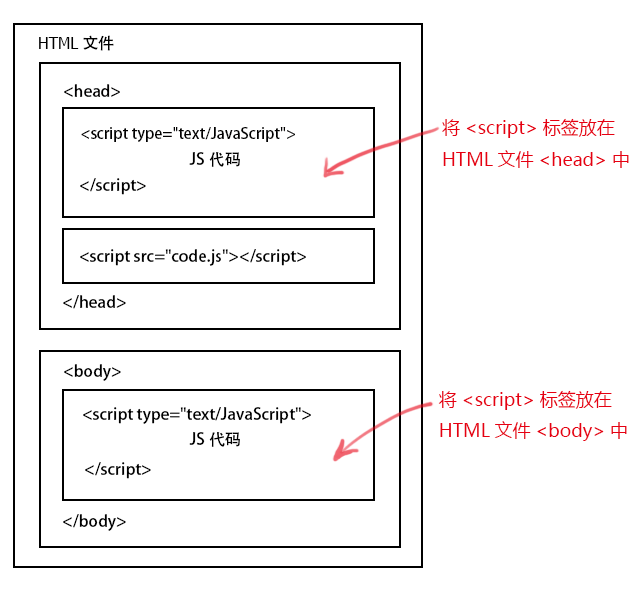
我们可以将JavaScript代码放在html文件中任何位置，但是我们一般放在网页的head或者body部分。

**放在<head>部分**

最常用的方式是在页面中head部分放置<script>元素，浏览器解析head部分就会执行这个代码，然后才解析页面的其余部分。

**放在<body>部分**

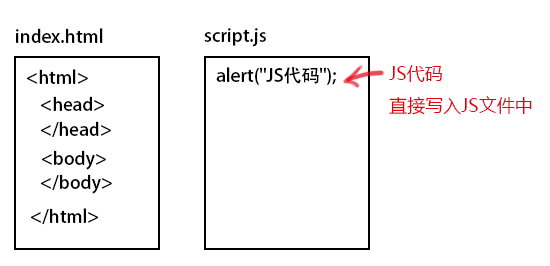
JavaScript代码在网页读取到该语句的时候就会执行。



注意: javascript作为一种脚本语言可以放在html页面中任何位置，但是浏览器解释html时是按先后顺序的，所以前面的script就先被执行。比如进行页面显示初始化的js必须放在head里面，因为初始化都要求提前进行（如给页面body设置css等）；而如果是通过事件调用执行的function那么对位置没什么要求的。

### 二、引用外部文件

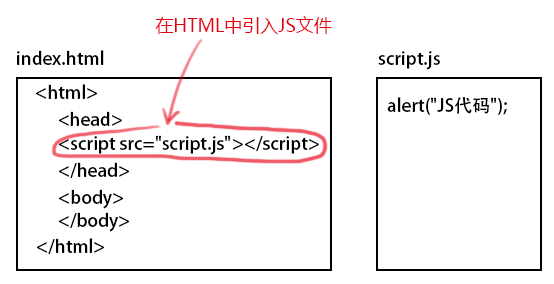
JavaScript代码只能写在HTML文件中吗?当然不是，我们可以把HTML文件和JS代码分开,并单独创建一个JavaScript文件(简称JS文件),其文件后缀通常为.js，然后将JS代码直接写在JS文件中。



注意:在JS文件中，不需要<script>标签,直接编写JavaScript代码就可以了。

JS文件不能直接运行，需嵌入到HTML文件中执行，我们需在HTML中添加如下代码，就可将JS文件嵌入HTML文件中。

<script src="script.js"></script>



## 3、JavaScript基础中的基础

### 一、JavaScript-认识语句和符号

JavaScript语句是发给浏览器的命令。这些命令的作用是告诉浏览器要做的事情。

每一句JavaScript代码格式: 语句;

先来看看下面代码

<script type="text/javascript">

alert("hello!");

</script>

例子中的alert("hello!");就是一个JavaScript语句。

一行的结束就被认定为语句的结束，通常在结尾加上一个分号";"来表示语句的结束。

看看下面这段代码,有三条语句，每句结束后都有";"，按顺序执行语句。

<script type="text/javascript">

document.write("I");

document.write("love");

document.write("JavaScript");

</script>

注意:

1. “;”分号要在英文状态下输入，同样，JS中的代码和符号都要在英文状态下输入。

2. 虽然分号“;”也可以不写，但我们要养成编程的好习惯，记得在语句末尾写上分号。

### 二、注释

注释的作用是提高代码的可读性，帮助自己和别人阅读和理解你所编写的JavaScript代码，注释的内容不会在网页中显示。注释可分为单行注释与多行注释两种。

我们为了方便阅读，注释内容一般放到需要解释语句的结尾处或周围。

单行注释，在注释内容前加符号 “//”。

<script type="text/javascript">

document.write("单行注释使用'//'"); // 我是注释，该语句功能在网页中输出内容

</script>

多行注释以"/\*"开始，以"\*/"结束。

<script type="text/javascript">

document.write("多行注释使用/\*注释内容\*/");

/\*

多行注释

养成书写注释的良好习惯

\*/

</script>

### 三、变量

什么是变量? 从字面上看，变量是可变的量；从编程角度讲，变量是用于存储某种/某些数值的存储器。我们可以把变量看做一个盒子，为了区分盒子，可以用BOX1,BOX2等名称代表不同盒子，BOX1就是盒子的名字（也就是变量的名字）。

定义变量使用关键字var,语法如下：

var 变量名

变量名可以任意取名，但要遵循命名规则:

1.变量必须使用字母、下划线(\_)或者美元符($)开始。

2.然后可以使用任意多个英文字母、数字、下划线(\_)或者美元符($)组成。

3.不能使用JavaScript关键词与JavaScript保留字。

变量要先声明再赋值，如下：

var mychar;

mychar="javascript";

var mynum = 6;

变量可以重复赋值，如下：

var mychar;

mychar="javascript";

mychar="hello";

### 四、if…else…判断语句

if...else语句是在指定的条件成立时执行代码，在条件不成立时执行else后的代码。

语法:

if(条件)

{ 条件成立时执行的代码 }

else

{ 条件不成立时执行的代码 }

假设我们通过年龄来判断是否为成年人，如年龄大于等于18岁，是成年人，否则不是成年人。代码表示如下:

<script type="text/javascript">

var myage = 18;

if(myage>=18) //myage>=18是判断条件

{ document.write("你是成年人。");}

else //否则年龄小于18

{ document.write("未满18岁，你不是成年人。");}

</script>

### 五、函数

是完成某个特定功能的一组语句。如没有函数，完成任务可能需要五行、十行、甚至更多的代码。这时我们就可以把完成特定功能的代码块放到一个函数里，直接调用这个函数，就省重复输入大量代码的麻烦。

如何定义一个函数呢？基本语法如下:

function 函数名()

{

函数代码;

}

说明:

1. function定义函数的关键字。

2. "函数名"你为函数取的名字。

3. "函数代码"替换为完成特定功能的代码。

我们来编写一个实现两数相加的简单函数,并给函数起个有意义的名字：“add2”，代码如下：

function add2(){

var sum = 3 + 2;

alert(sum);

}

函数调用:

函数定义好后，是不能自动执行的，所以需调用它,只需直接在需要的位置写函数就ok了,代码如下:



### 六、打印输出

document.write() 可用于直接向 HTML 输出流写内容。简单的说就是直接在网页中输出内容。

第一种:输出内容用""括起，直接输出""号内的内容。

<script type="text/javascript">

document.write("I love JavaScript！"); //内容用""括起来，""里的内容直接输出。

</script>

第二种:通过变量，输出内容

<script type="text/javascript">

var mystr="hello world!";

document.write(mystr); //直接写变量名，输出变量存储的内容。

</script>

第三种:输出多项内容，内容之间用+号连接。

<script type="text/javascript">

var mystr="hello";

document.write(mystr+"I love JavaScript"); //多项内容之间用+号连接

第四种:输出HTML标签，并起作用，标签使用""括起来。

<script type="text/javascript">

var mystr="hello";

document.write(mystr+"<br>");//输出hello后，输出一个换行符

document.write("JavaScript");

</script>

第五种 输出空格

在写JS代码的时候，大家可以会发现这样现象:

document.write(" 1 2 3 ");

结果: 1 2 3

无论在输出的内容中什么位置有多少个空格，显示的结果好像只有一个空格。

这是因为浏览器显示机制，对手动敲入的空格，将连续多个空格显示成1个空格。

解决方法:

1. 使用输出html标签&nbsp;来解决

document.write("&nbsp;&nbsp;"+"1"+"&nbsp;&nbsp;&nbsp;&nbsp;"+"23");

结果: 1 23

2. 使用CSS样式来解决

document.write("<span style='white-space:pre;'>"+" 1 2 3 "+"</span>");

结果: 1 2 3

在输出时添加“white-space:pre;”样式属性。这个样式表示"空白会被浏览器保留"

### 七、警告（alert 消息对话框）

我们在访问网站的时候，有时会突然弹出一个小窗口，上面写着一段提示信息文字。如果你不点击“确定”，就不能对网页做任何操作，这个小窗口就是使用alert实现的。

语法:

alert(字符串或变量);

看下面的代码:

<script type="text/javascript">

var mynum = 30;

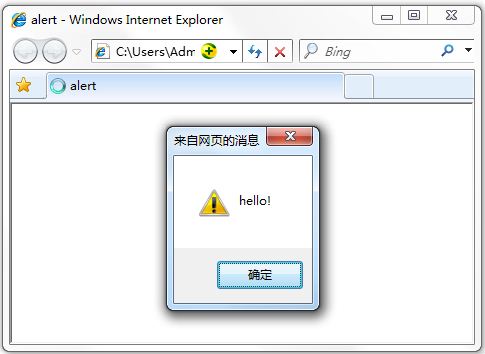
alert("hello!");

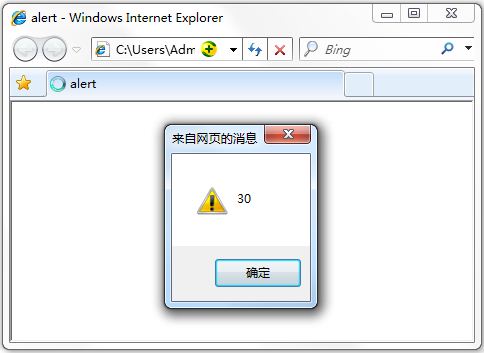
alert(mynum);

</script>

注:alert弹出消息对话框(包含一个确定按钮)。

结果:按顺序弹出消息框





注意:

1. 在点击对话框"确定"按钮前，不能进行任何其它操作。

2. 消息对话框通常可以用于调试程序。

3. alert输出内容，可以是字符串或变量，与document.write 相似。

### 八、确认（confirm 消息对话框）

confirm 消息对话框通常用于允许用户做选择的动作，如：“你对吗？”等。弹出对话框(包括一个确定按钮和一个取消按钮)。

语法:

confirm(str);

参数说明:

str：在消息对话框中要显示的文本

返回值: Boolean值

返回值:

当用户点击"确定"按钮时，返回true

当用户点击"取消"按钮时，返回false

注: 通过返回值可以判断用户点击了什么按钮

看下面的代码:

<script type="text/javascript">

var mymessage=confirm("你喜欢JavaScript吗?");

if(mymessage==true)

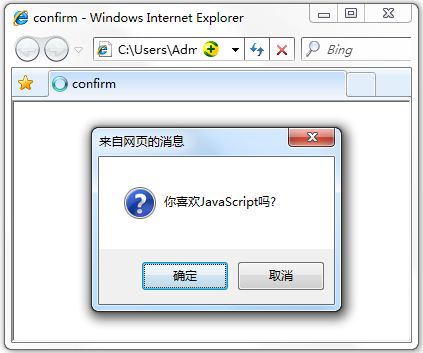
{ document.write("很好,加油!"); }

else

{ document.write("JS功能强大，要学习噢!"); }

</script>

结果:



**注: 消息对话框是排它的，即用户在点击对话框按钮前，不能进行任何其它操作。**

### 九、提问（prompt 消息对话框）

prompt弹出消息对话框,通常用于询问一些需要与用户交互的信息。弹出消息对话框（包含一个确定按钮、取消按钮与一个文本输入框）。

语法:

prompt(str1, str2);

参数说明：

str1: 要显示在消息对话框中的文本，不可修改

str2：文本框中的内容，可以修改

返回值:

1. 点击确定按钮，文本框中的内容将作为函数返回值

2. 点击取消按钮，将返回null

看看下面代码:

var myname=prompt("请输入你的姓名:");

if(myname!=null)

{ alert("你好"+myname); }

else

{ alert("你好 my friend."); }

结果:

注:在用户点击对话框的按钮前，不能进行任何其它操作。

### 十、打开和关闭窗口（window.open）

**打开窗口：**

open() 方法可以查找一个已经存在或者新建的浏览器窗口。

语法：

window.open([URL], [窗口名称], [参数字符串])

参数说明:

URL：可选参数，在窗口中要显示网页的网址或路径。如果省略这个参数，或者它的值是空字符串，那么窗口就不显示任何文档。

窗口名称：可选参数，被打开窗口的名称。

1.该名称由字母、数字和下划线字符组成。

2."\_top"、"\_blank"、"\_self"具有特殊意义的名称。

\_blank：在新窗口显示目标网页

\_self：在当前窗口显示目标网页

\_top：框架网页中在上部窗口中显示目标网页

3.相同 name 的窗口只能创建一个，要想创建多个窗口则 name 不能相同。

4.name 不能包含有空格。

参数字符串：可选参数，设置窗口参数，各参数用逗号隔开。

参数表:



例如:打开http://www.imooc.com网站，大小为300px \* 200px，无菜单，无工具栏，无状态栏，有滚动条窗口：

<script type="text/javascript"> window.open('http://www.imooc.com','\_blank','width=300,height=200,menubar=no,toolbar=no, status=no,scrollbars=yes')

</script>

注意：运行结果考虑浏览器兼容问题。

**关闭窗口：**

close()关闭窗口

用法：

window.close(); //关闭本窗口

或

<窗口对象>.close(); //关闭指定的窗口

例如:关闭新建的窗口。

<script type="text/javascript">

var mywin=window.open('http://www.imooc.com'); //将新打的窗口对象，存储在变量mywin中

mywin.close();

</script>

注意:上面代码在打开新窗口的同时，关闭该窗口，看不到被打开的窗口。

### 十一、DOM操作

1、认识DOM

文档对象模型DOM（Document Object Model）定义访问和处理HTML文档的标准方法。DOM 将HTML文档呈现为带有元素、属性和文本的树结构（节点树）。

先来看看下面代码:



将HTML代码分解为DOM节点层次图:



HTML文档可以说由节点构成的集合，三种常见的DOM节点:

1. 元素节点：上图中<html>、<body>、<p>等都是元素节点，即标签。

2. 文本节点:向用户展示的内容，如<li>...</li>中的JavaScript、DOM、CSS等文本。

3. 属性节点:元素属性，如<a>标签的链接属性href="http://www.imooc.com"。

看下面代码:

<a href="http://www.imooc.com">JavaScript DOM</a>

