

第8章 Javascripts入门

笔记本: Javascript

创建时间: 6/5/2019 3:21 AM

更新时间: 5/19/2020 4:37 PM

作者: tanziqi1756@outlook.com

URL: <http://www.imooc.com/code/470>

写这篇文章是因为在知乎上看到有人问这个问题，在想怎么会有这种SB问题，不过想想当初SB的我貌似也搞不清两者的关系，认知还是需要一个过程。

然后看到比较经典的回答有：Java 和Javascript的关系就像雷锋和雷锋塔的关系，也像老婆和老婆饼的关系。

一般认为，当时 Netscape 之所以将 LiveScript 命名为 JavaScript，是因为 Java 是当时最流行的编程语言，**带有 "Java" 的名字有助于这门新生语言的传播。**

它们的相同之处包括：

它们的语法和 C 语言都很相似；

它们都是面向对象的（虽然实现的方式略有不同）；

JavaScript 在设计时参照了 Java 的命名规则；

它们的不同之处包括：

JavaScript 是动态类型语言，而 Java 是静态类型语言；

JavaScript 是弱类型的，Java 属于强类型；

JavaScript 的面向对象是基于原型的（prototype-based）实现的，Java 是基于类（class-based）的；

JavaScript 除了长得和 Java 比较像之外，语言风格相去甚远。

JavaScript 在设计时所参考的对象不包括 Java，而包括了像 Self 和 Scheme 这样的语言。

<https://www.runoob.com/w3cnote/java-and-javascript-relationship.html>

为什么学习JavaScript

JavaScript一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。

它的解释器被称为JavaScript引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在HTML（标准通用标记语言下的一个应用）网页上使用，用来给HTML网页增加动态功能。

一、你知道，为什么JavaScript非常值得我们学习吗？

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。

```
<body>
  <p id="p1">我是第一段文字</p>
  <p id="p2">我是第二段文字</p>

  <script type="text/javascript">
    document.write("hello");
    console.log("HelloWorld!");
    document.getElementById("p1").style.color="blue";
  </script>
</body>
```

也可以外部式引用JS

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

JS中可以定义变量

```
var mynum=8;
var mynum2;
mynum2 = 10;
```

数组

```
var animals = ['Dog', 'Cat']; // This is an array
// 也可以用new 关键字
animals.push('Pig'); // 在数组后面加一个元素
// 排序
animals.sort();
```

字典

```
var person = [];
person['name'] = 'John Doe';
person['age'] = 35;
```

```
var car = { make : 'Ford', model : 'Mustang', year : '1964'
};
```

条件语句

```
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。");}
```

```
var name = 'Adam';

switch(name){
  case 'Joe':
    console.log('My name in Joe');
    break;
  case 'Jeff':
    console.log('My name is Jeff');
    break;
  case 'Jack':
    console.log('My name is Jack');
    break;
  default:
    console.log('My name is '+name);
    break;
}
```

循环语句

```
for( i = 0; i < 10; i++) {
  console.log(i);
}
```

```
var i = 0;
while( i < 10) {
  console.log(i);
  i++;
}
```

函数的定义

```
function add2(){
  var sum = 3 + 2;
```

```
    alert(sum);
}

function sayHello(greeting, name = 'Jeff'){
    alert(greeting+ ' ' +name);
}

sayHello('Hello');
```

Alert()消息对话框

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>alert</title>
    <script type="text/javascript">
        function rec(){
            var mychar="I love JavaScript";
            alert(mychar);
        }
    </script>
</head>
<body>
    <input name="button" type="button"
onClick="rec()" value="点击我，弹出对话框" />
</body>
</html>
```

confirm()消息对话框

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>confirm</title>
    <script type="text/javascript">
        function rec(){
            var mymessage=confirm("你喜欢JavaScript吗?");
            if(mymessage==true)
            {
```

```

        document.write("你是女士!");
    }
    else
    {
        document.write("你是男士!");
    }
}
</script>
</head>
<body>
    <input name="button" type="button"
onClick="rec()" value="点击我，弹出确认对话框" />
</body>
</html>

```

prompt()消息对话框

```

<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>prompt</title>
<script type="text/javascript">
function rec(){
    var score; //score变量，用来存储用户输入的成绩值。
    score = prompt("请输入你的成绩");
    if(score>=90)
    {
        document.write("你很棒!");
    }
    else if(score>=75)
    {
        document.write("不错吆!");
    }
    else if(score>=60)
    {
        document.write("要加油!");
    }
    else
    {
        document.write("要努力了!");
    }
}

```

```

    </script>
</head>
<body>
    <input name="button" type="button"
onClick="rec()" value="点击我，对成绩做评价!" />
</body>
</html>

```

window.open()打开新窗口 **用window.close()关闭窗口对象**

```

<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>window.open</title>
<script type="text/javascript">
    function Wopen(){
        window.open('http://www.imooc.com', '_blank', 'width=600,
height=400, menubar=no, left=0');
    }
</script>
</head>
<body>
    <input name="button" type="button"
onClick="Wopen()" value="点击我，打开新窗口!" />
</body>
</html>

```

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）。

HTML文档可以说由节点构成的集合，三种常见的DOM节点：

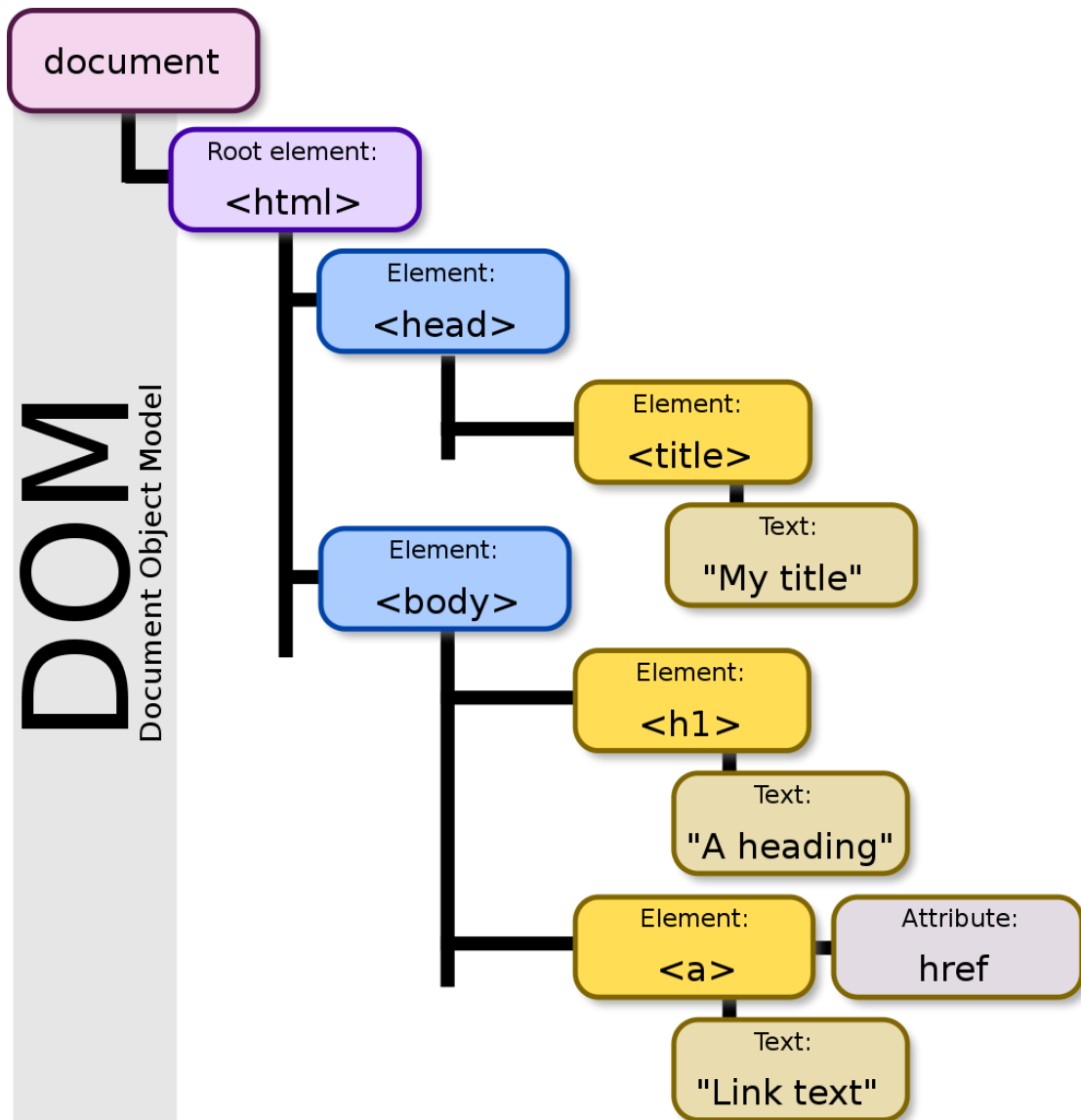
1. 元素节点：上图中<html>、<body>、<p>等都是元素节点，即标签。

2. 文本节点：向用户展示的内容，如...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>"。

文档对象模型（英语：**Document Object Model**，缩写**DOM**），是W3C组织推荐的处理**可扩展置标语言**的标准编程接口。

Document Object Model的历史可以追溯至1990年代后期微软与Netscape的“浏览器大战”（browser wars），双方为了在JavaScript与JScript一决生死，于是大规模的赋予浏览器强大的功能。微软在网页技术上加入了不少专属事物，既有VBScript、ActiveX、以及微软自家的DHTML格式等，使不少网页使用非微软平台及浏览器无法正常显示。DOM即是当时酝酿出来的杰作。



通过ID获取元素

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
```

```

<title>document.getElementById</title>
</head>
<body>
<p id="con">JavaScript</p>
<script type="text/javascript">
    var mychar=document.getElementById("con");
    document.write("结果:"+mychar); //输出获取的P标签。
</script>
</body>
</html>

```

innerHTML属性用于获取或替换 HTML 元素的内容。

```

<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>innerHTML</title>
</head>
<body>
<h2 id="con">javascript</H2>
<p> JavaScript是一种基于对象、事件驱动的简单脚本语言，
嵌入在HTML文档中，由浏览器负责解释和执行，在网页上产生动态
的显示效果并实现与用户交互功能。</p>
<script type="text/javascript">
    var mychar=document.getElementById("con");
    document.write("原标题:"+mychar.innerHTML+"<br>");
//输出原h2标签内容
    mychar.innerHTML = "new text";
    document.write("修改后的标题:"+mychar.innerHTML);
//输出修改后h2标签内容
</script>
</body>
</html>

```

改变HTML的CSS样式。

```

<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>style样式</title>

```



```

</head>
<body>
  <h2 id="con">I love JavaScript</H2>
  <p> JavaScript使网页显示动态效果并实现与用户交互功能。</p>
  <script type="text/javascript">
    var mychar= document.getElementById("con");
    mychar.style.color = "red";
    mychar.style.backgroundColor="#CCC";
    mychar.style.width = "300px";
  </script>
</body>
</html>

```

属性↗	描述↗
background-color↗	设置元素的背景颜色↗
height↗	设置元素的高度↗
width↗	设置元素的宽度↗
color↗	设置文本的颜色↗
font↗	在一行设置所有的字体属性↗
fontFamily↗	设置元素的字体系列。↗
fontSize↗	设置元素的字体大小。↗

控制CSS的display属性。

```

index.html
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=gb2312">
5 <title>display</title>
6 <script type="text/javascript">
7   function hidetext() {
8     var mychar = document.getElementById("con");
9     mychar.style.display = "none";
10    // 隐藏内容
11  }
12
13   function showtext() {
14     var mychar = document.getElementById("con");
15     mychar.style.display = "block";
16     // 显示内容
17   }
18 </script>
19 </head>
20 <body>
21 <h1>JavaScript</h1>
22 <p id="con">做为一个Web开发工程师来说，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。</p>
23 <form>
24   <input type="button" onclick="hidetext()" value="隐藏内容" />
25   <input type="button" onclick="showtext()" value="显示内容" />
26 </form>
27 </body>
28 </html>

```

JavaScript

做为一个Web开发工程师来说，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。

隐藏内容

显示内容

控制HTML标签的class.

```
index.html
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=gb2312">
5 <title>className属性</title>
6 <style>
7     body{ font-size:16px;}
8     .one{
9         border:1px solid #eee;
10        width:230px;
11        height:50px;
12        background:#ccc;
13        color:red;
14    }
15    .two{
16        border:1px solid #ccc;
17        width:230px;
18        height:50px;
19        background:#9CF;
20        color:blue;
21    }
22 </style>
23 </head>
24 <body>
25 <p id="p1" > JavaScript使网页显示动态效果并实现与用户交互功能。</p>
26 <input type="button" value="添加样式" onclick="add()"/>
27 <p id="p2" class="one">JavaScript使网页显示动态效果并实现与用户交互功能。</p>
28 <input type="button" value="更改外观" onclick="modify()"/>
29
30 <script type="text/javascript">
31     function add(){
32         var p1 = document.getElementById("p1");
33         p1.className = "one";
34     }
35     function modify(){
36         var p2 = document.getElementById("p2");
37         p2.className = "two";
38     }
39 </script>
40 </body>
41 </html>
```

重置代码 自动运行 提交

JavaScript使网页显示动态效果并实现与用户交互功能。

添加样式

JavaScript使网页显示动态效果并实现与用户交互功能。

更改外观

RemoveAttribute("style")

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" Content="text/html;
charset=utf-8" />
<title>javascript</title>
<style type="text/css">
body{font-size:12px;}
#txt{
    height:400px;
    width:600px;
    border:#333 solid 1px;
    padding:5px;}
p{
    line-height:18px;
    text-indent:2em;}
</style>
</head>
<body>
<h2 id="con">JavaScript课程</H2>
```

```

<div id="txt">
  <h5>JavaScript为网页添加动态效果并实现与用户交互的功能。</h5>
  <p>1. JavaScript入门篇，让不懂JS的你，快速了解JS。</p>
  <p>2. JavaScript进阶篇，让你掌握JS的基础语法、函数、数组、事件、内置对象、BOM浏览器、DOM操作。</p>
  <p>3. 学完以上两门基础课后，在深入学习JavaScript的变量作用域、事件、对象、运动、cookie、正则表达式、ajax等课程。</p>
</div>
<form>
  <!-- 当点击相应按钮，执行相应操作，为按钮添加相应事件 -->
  <input type="button" onClick="changeColor()" value="改变颜色" >
  <input type="button" value="改变宽高" >
  <input type="button" value="隐藏内容" >
  <input type="button" value="显示内容" >
  <input type="button" onClick="removeSetting()" value="取消设置" >
</form>
<script type="text/javascript">
  var text = document.getElementById("txt");
  //定义"改变颜色"的函数
  function changeColor() {

    text.style.color = "red";
  }
  //定义"取消设置"的函数
  function removeSetting(){
    var mymessage = confirm("是否取消设置");
    if( mymessage == true) {
      txt.removeAttribute("style");
    }
  }
</script>
</body>
</html>

```

