# 网站开发相关笔记

## 一、Javascript篇

### 1.1元素定位

document.getElementById("demo")；返回唯一对象

document.getElementByTagName();

document.getElementByClassName();

document.getElementByName(); 返回一个结果列表

以上使用时需注意位置，必须保证元素已经在前面出现过，即已经定义过了；否则取到的可能是空值

Element.parentNode();

element.childNodes();

element.firstChild();lastChild();

Document.body();documetnElement();

模糊定位 element.src**.match**("bulbon")；

多个元素 var x=document.getElementByTagName(“p”); x[0].innerHTML=””;

### 1.2元素编辑

增加元素

element.insertBefore(new,old);

Element.appendChild();创建element的子节点

document.createElement("strong'); 创建一组strong标签

document.createAttribute();创建属性节点

document.createTextNode("New text ");创建一段文本

Document.write();写入HTML或javascript代码，若在文档加载完成之后执行此操作，整个页面将被覆盖。

element.getAttribute();

element.setAttribute();

移除元素

Element.removeChild();

child.parentNode.removeChild(child);

element.replaceChild(new,old);

Element.innerHTML vs element.innerText

前者指element标签内包含的所有代码内容，含标签；后者指element标签内包含代码内容解释之后的结果；需要注意的是，innerText并非W3C标准属性，因此可能存在浏览器兼容性问题。

**元素属性**

Img.src

Element.style.color

Location.hostname;lpathname;.port;protocal;href;assign

Document.cookie

nodeName();nodeValue();nodeType();

### 1.3事件

Onclick();

Onload();onunload(); 属于window对象，不属于document，如果用body.onload则必须在<body>标签出现之后才使用;两个脚本里同时出现onload事件之后，先读入的会失效，[解决方案](http://bolg.malu.me/html/2011/1113.html)

Onchange();

Onreadystatechange();

Onmouseover();onmouseout();onmousedown();onmouseup();

Window.open();window.close();window.resizeTo();window.replace();window.location=url

|  |
| --- |
| 添加图像映射，特定区域事件  <img src="/i/eg\_planets.jpg" border="0" usemap="#planetmap" alt="Planets" />  <map name="planetmap" id="planetmap">  <area shape="circle" coords="180,139,14" onMouseOver="dosomething" href ="url" target ="\_blank" alt="Venus" />  </map> |

### 1.4实用函数

判断是否数字isNaN()

Date();getDay();getTime();自1970.1.1至今毫秒数;getHours();getMinutes();getSeconds();

排序sort();

合并数组concat();

数组元素组成字符串join();

四舍五入Round();

随机数0到1之间Random();

返回前一页History.back();forward();

消息框alert();confirm();prompt();

计时setTimeout();clearTimeout();

匹配字符串match();

### 1.5安全性

HTML编码，其实就是将字符转换为HTML实体，这是防止脚本注入的重要手段之一。

google提供了许多javascript库的CDN，比如JQuery()：

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js"></script>

主要的 jQuery 函数是 $() 函数（jQuery 函数）。如果您向该函数传递 DOM 对象，它会返回 jQuery 对象，带有向其添加的 jQuery 功能。

### 1.6 组件化编程

通过对象间的树型关系来组织，因为js中构造函数本身就是一个对象，所以构造函数可以加入对象树。这颗对象树的root就是浏览器对象window，所有全局函数或对象在浏览器内部都是作为window的成员挂靠的。

|  |
| --- |
| var someclass=function(options){  this.mem1=options.meml;  this.mem2=’ ’;  }  someclass.prototype={  showmem1:function(){  alert(this.mem1);  }  } |
| var someobj=new someclass({mem1:’123’});  someobj.showmem1();//输出：123 |

如下所示，someclass和config成为了someModule的一员。

|  |
| --- |
| If(!window.someModul)//若someModule已定义，直接扩展模块，js对象成员支持动态扩展  window.someModule={};  window.someModule.SomeClass=function(){...};  window.someModule.config={option1:’’,option2:’’;}; |

自执行函数，即定义后立即执行，通过自执行函数，可以实现模块间解耦，将对象所依赖的外部模块作为参数输入。

如下例，在function($){}定义完成之后即进行调用，并将jQuery作为实参，传递给$这一形参，在函数体内将someclass设置为$的成员，实现了jQuery的扩展。

|  |
| --- |
| （function($){  Var \_someclass=function(options){...};  \_someclass.prototype={showMem1:...};  $.someclass=\_someclass;  })(jQuery); |
| Var someobj=new jQuery.someclass({mem1:’123’});  someobj.showMem1(); |

如果将上面例子的jQuery换成window，那么someclass类就可以通过window.someclass或者直接全局调用了，而内部函数不需要做任何改变——这就是模块之间的解耦。

### 1.7node.js——服务器端的javascript

不同于传统的javascript代码，node.js并非运行与前端的脚本，而是提供服务器端脱离浏览器独立运行javascript代码的环境。

特点：事件驱动，更快；异步，事件发生时产生函数回调；封装了常用模块，如”http”、“url”、“querystring”等；单线程，通过事件轮询实现并行非阻塞，克服多线程吃掉更多内存的缺点，比如apache；包管理器可以方便地进行外部模块的安装扩展；命令行工具，通过javascript 与文件系统和网络系统交互；

优点：

缺点：

典型的代码

Var x=require(“modelName”);

X.method();

通过exports关键词导出方法即可成为模块；

## 二、HTML篇

### 2.1常用属性

innerText：标签内部文本，比如<span>hehe</span>

导入外部js:

2.2

中文乱码

**<head><**meta http-equiv="Content-Type" content="text/html; charset=utf-8" />**</head>**

**单选按钮**

|  |
| --- |
| <label><input type="radio" name="**indoor-outdoor**">indoor</input></label>  <label><input type="radio" name="**indoor-outdoor**">outdoor</input></label> |

**复选按钮**

|  |
| --- |
| <label><input type="checkbox" name="**personality**"></label>  <label><input type="checkbox" name="**personality**"></label>  <label><input type="checkbox" name="**personality**"></label> |

**超链接新标签打开：**target=”\_blank”

## CSS篇

### 3.1 元素定位

根据标签

p h1 h2{property1:value;property2:value;}

根据类

<p class=”classname”></p> .classname{property1:value;property2:value;}

注意一个元素可以属于多个类 <img class="class1 class2">

根据id

<p id=”name”></p> #name{property1:value;property2:value;}

### 3.2 常用属性

长度 length

宽度 width

字体颜色color

字体大小font-size 候选字体设置font-family: Lobster, Monospace;

字体类型 font-family

引入外部字体

<link href="http://fonts.googleapis.com/css?family=Lobster" rel="stylesheet" type="text/css">

图片 <img src="http://www.your-image-source.com/your-image.jpg">

图片来源 src

图片提示 alt

边框颜色 border-color

边框宽度 border-width

边框线型 border-style

边框圆角 border-radius 50%是个圆

内间距 padding padding-left padding-right padding-top padding-bottom

外边距 margin 可为负值，亦可四个方向分别设置

边框 border

背景色 background-color

文本框提示 placeholder

表单 action

提交按钮 type=”submit”

必须填写 <input type="text" **required**>

单选复选默认选中 <label><input type="checkbox" name="personality" **checked**> Loving</label>

Important 关键字确保属性不会被覆盖

|  |
| --- |
| .pink-text {  color: pink **!important**;  } |

**Bootstrap自适应框架**

Bootstrap是Twitter推出的一个用于前端开发的开源工具包

<link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/bootstrap/3.3.1/css/bootstrap.min.css"/>

## jQuery篇

所有的jQurery语句以“$”符开始

元素定位

$(“.className”)

$(“#idName”)

函数

document.ready() 保证整个网页的HTML代码渲染完毕之后才执行函数体

.addClass() 动态为元素添加类标识

.removeClass() 动态为元素移除标识

.css() 动态改变元素样式 $("#target1").css("color", "blue");

.prop() 动态改变元素属性 $("button").prop("disabled", true);

.html() 动态改变标签（如<h3>）内html代码

.text() 只能改变标签内文本内容

.remove() 移除元素

.appendTo() 移动一个元素到另一个位置

.clone() 复制一个元素 $("#target2").clone().appendTo("#right-well");

.parent() 获取当前元素的父元素

.children() 获取当前元素的所有子元素

.classname:nth-child(3) 获取每组第三个类名为classname的元素

.classname:odd 获取每组第奇数个类名为classname的元素

## ASP.NET篇