

Javascript基础

脚本编写



2016-11-11

JavaScript笔记

貴社ますますご盛栄のこととお慶び申し上げます。平素は格別のご高配を賜り、厚く御礼申し上げます。

### 目录页

目录

[1 语法 2](#_Toc466647195)

[1.1空的链接 2](#_Toc466647196)

[1.2 引用js脚本 2](#_Toc466647197)

# 1 语法

## 1.1空的链接

<script>

<a href=”JavaScript:alert(“hello kitty”)”>点击</a>

</script> innerHTML= “小星星”;

## 1.2 引用js脚本

1. <script src = "move.js" type="text/javascript"></script>
2. <a href="javascript:go()">click here</a>

## 1.3 变量及数据类型

\* 隐式定义(使用前要赋值)

\*显示定义 Var temp； \_ $ 数字 字母

var i,j=”hello”;

* 换行

var temp = “hello \n kitty”;

var temp = “hello\n”+

“kitty”; NaN == not a number

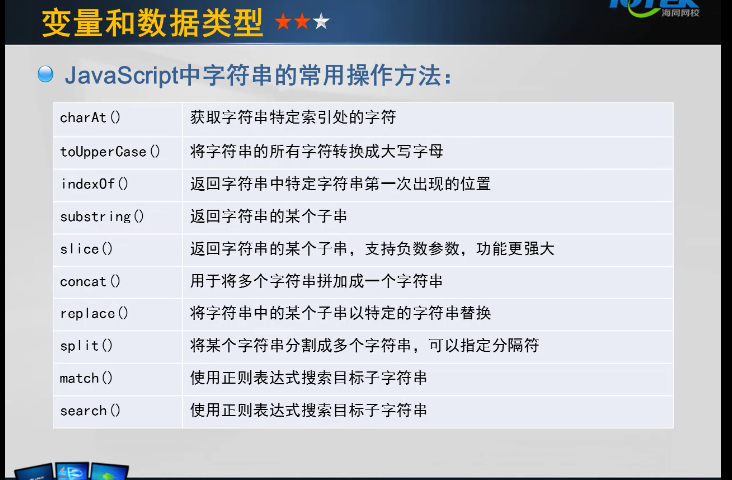
### 1.3.1 类型转换

var a = 3; var c = a.toString(); var c += a;

parseInt(“67red”) ==67; parseInt(“0xA”)//16进制可以转换

parseFloat(“35chen”) == 35.00 **无16进制转换**

### 1.3.2 字符串操作



返回下标索引

### 1.3.3 正则表达式

对字符串操作的逻辑公式，组成‘ **规则字符串**’；

特点：**灵活，逻辑性，功能性强，复杂；**

1. var reg = /chen/;
2. var reg = new RegExp(‘chen’);

exec(str) 返回找到的值；

reg.exec(str);

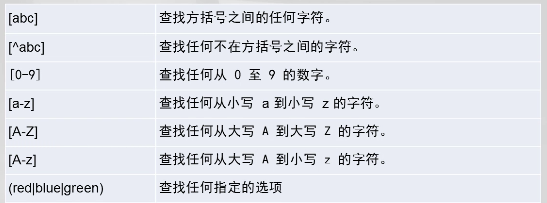
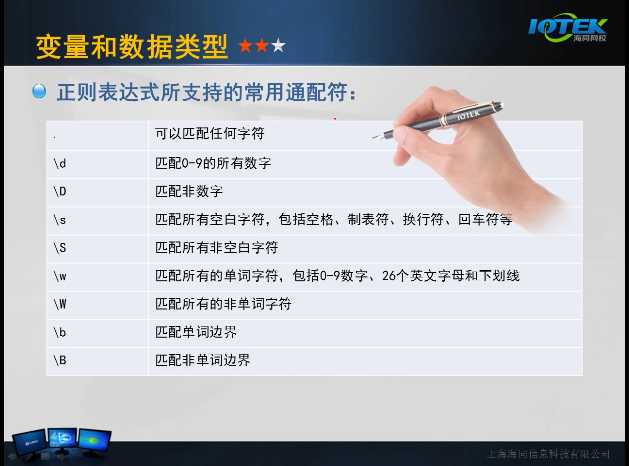
test(str) 返回 true；

通配符：

. 任意;

\d数字;\s空白字符;\w单词下划线;

o\b单词边界；（**判断有无，不在乎位置**）



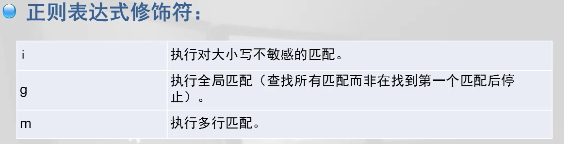
### 1.3.4 正则表达式量词



/[0-9]+/ i

邮箱表达式：

var reg = /^[0-9a-zA-Z\_-]+@[0-9a-zA-Z\_]+\.[com|cn|org]$/;



### 1.3.5 数组

var arr = [2,5,6];

var arr = [ ];

var arr = new Array(); **document.write(arr [i]+”<br/>”);**

**“10”==10; “10” !== 10;**

var a = **void**(b=0,c=3) ; undefine

string == **typeof** a; a **instanceof** Object == 1;

## 1.4 流程控制 ---for in

1 for (var index in arry){ } //输出的是赋值的数组下标。

2 for(var properties in navigator){ document.write(properties+":"+navigator[properties]+"<br/>");

}

throw new **Error**(“error”);

try{ if (1) { throw new **Error**(“error”); }}

catch(e){ document.write(e.message);}

finally{document.write(‘error’)};

## 1.5 函数

1 function show(){ } //可以先执行，后调用

2 var f = function (name){ } f(“bowen”); //匿名函数，要先定义后执行；

**(function (name))(‘bowen’);**

3 var f = new Function**(‘name’,’age’**,”alert(name, age)**;”**) ;

**函数默认是window的属性。**

调用：

1 window.fun();

2 fun.**call**(window,name,age ) ;

3 fun.**apply(** window, [‘zhangsan’,30] );

function show (arr,func){

**func.call(window,arr);**

}

show([1,2,3],function(arr){

for(i in arr){ document.write(arr[i]); }

})

## 1.6 对象

new Date(); new Date(month dd,yyy hh:mm:ss);

new Date(yyy,mth,dd//,hh,mm.ss) new Date(ms);

getDate(); getDay(); getMonth(); 0-11

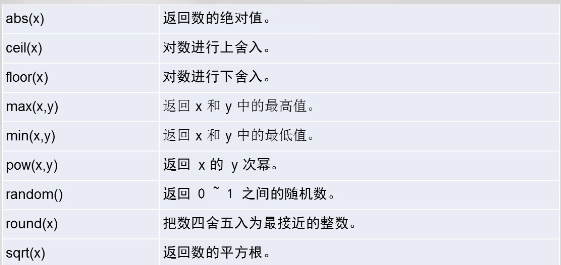
getFullYear(); getHours();0-23

getTime() 1970-1-1到现在的毫秒数

var d1 = new Date(); d1.toString();

var d2 = new Date(“2009-12-12 12:20:20”)

document .write(d2.toString());



对象创建：

new Object() JSON: {“name”:”zhangsan”,”age”:30}

new:一般通过构造函数添加属性；

Object：动态添加属性。（少）

Json：对象里，添加json对象数组

var Js = {

name:"zhangsan",age:30,

**son:[**

{name:"bowne",age:10},

{name:"nat",age:10}

**]**,

info:function(){document.write(this.name+":"+this.age+"<br/>");

for(var child in this.son){ document.write("son:"+this.son[child].name+","+this.son[child].age+"<br/>"); //document.write(child);}

}

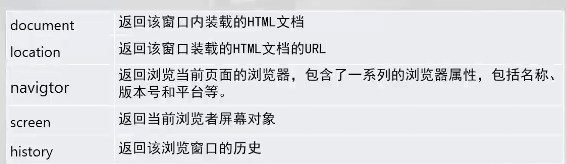
}

Js.info();

# 2 浏览器对象模型 BOM

## 2.1 对象类型

### Window对象：

navigator

flag = confirm(“是否退出”)

flag=prompt(“密码”,”123”)

**moveBy(50,50)表示：+**

scrollBy(50,50)x,y

Tip:

**<pre>域格式化**

window.open(<http://www.baidu.com,”baidu”,”left=20”);//>多次打开，会重新加载

**location=no,toolbar=no,resizable=no; 地址栏 状态栏 大小（不推荐）**

var t = window.setInterval(“showTime()”,100); window调用方法：

window.clearInteral(t); **定时器，及清除定时器 窗口位置，滚动条，弹框，定时器**

### History对象

back();

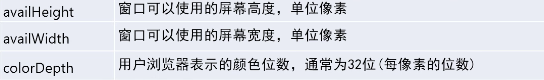
forward()//前进

go(intValue); go(-1) == back(); **history.back(); <br/> &nbsp**

### location 对象



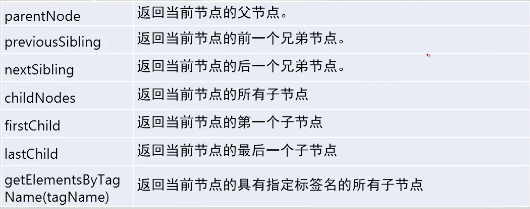
### screen对象



### navigator 对象



# 3 文档对象Dom



节点访问

form.submit()表单提交

form表单操作



表单元素查找3种

<form id="myform" action="http://www.baidu.com" method="get" target="\_blank">

<input name="username" type="text" value="nat"/><br/>

<input name="password" type="password" value="123456"/><br/>

<**select** name="city">

<option value="shanghai" selected="select">上海</option>

<option value="hefei">合肥</option>

</select><br/>

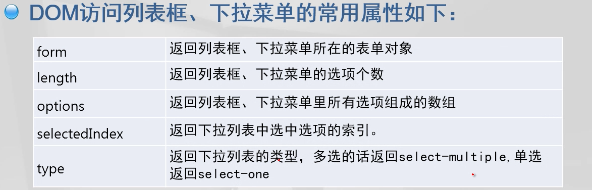
<input type="button" value="获取第一个表单" onclick="alert(document.getElementById('myform').elements[3].value)"

onclick="alert(document.getElementById('myform').elements[‘city’].value)"

onclick="alert(document.getElementById('myform').city.value)"/>

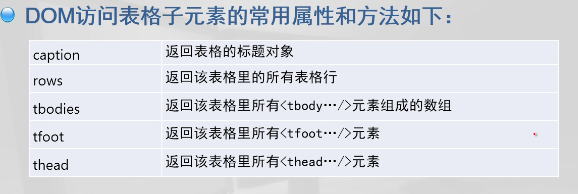
</form>

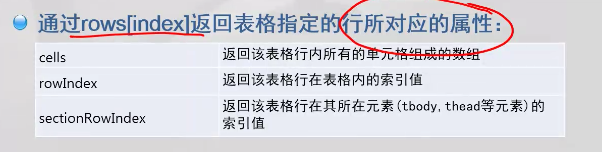
## 3.1 列表框，下拉菜单操作：



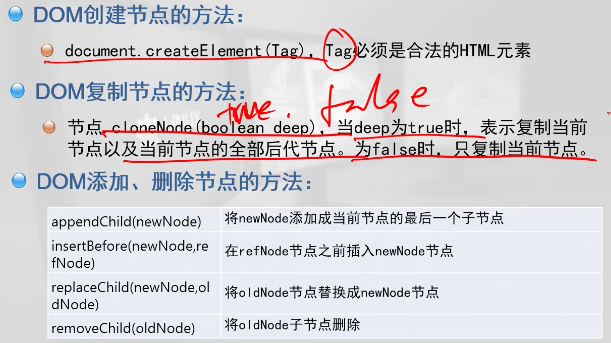


## 3.2 访问表格

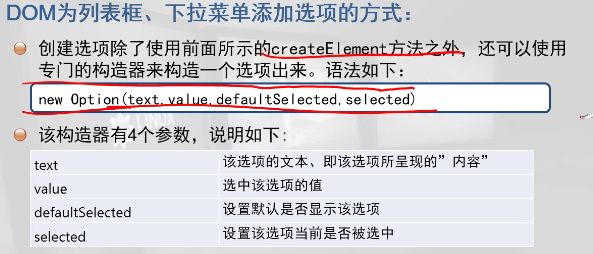




## 3.3 DOM对元素的增删改（IE不支持）



## 3.4 New列表框、下拉菜单



## 3.5 动态修改表格

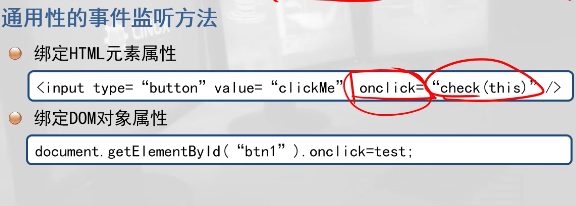




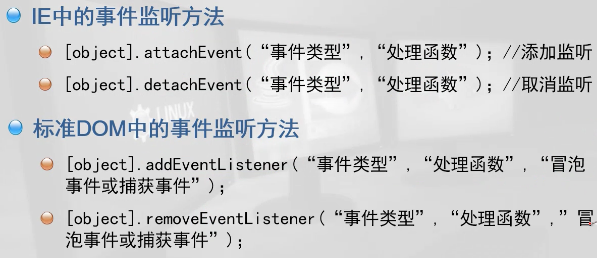
# 4 事件

事件：冒泡，捕获型事件（上到下，IE不支持）

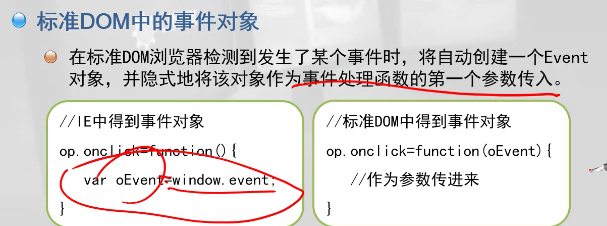
## 4.1 添加事件

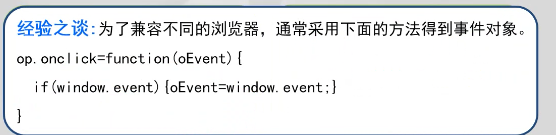


## 4.2 监听事件











## 4.3 常用的事件





