

## js语法大全

一.JavaScript 有六种数据类型。主要的类型有 number、string、object 以及 Boolean 类型, 其他两种类型为 null 和 undefined。

- String 字符串类型:字符串是用单引号或双引号来说明的。(使用单引号来输入包含引号的字符串。)如:“The cow jumped over the moon.”
- 数值数据类型:JavaScript 支持整数和浮点数。整数可以为正数、0 或者负数;浮点数可以包含小数点、也可以包含一个“e”(大小写均可,在科学记数法中表示“10的幂”)、或者同时包含这两项。
- Boolean 类型:可能的 Boolean 值有 true 和 false。这是两个特殊值,不能用作 1 和 0。
- Undefined 数据类型:一个为 undefined 的值就是指在变量被创建后,但未给该变量赋值以前所具有的值。
- Null 数据类型:null 值就是没有任何值,什么也不表示。
- object类型:除了上面提到的各种常用类型外,对象也是JavaScript中的重要组成部分,这部分将在后面章节详细介绍。

在 JavaScript 中变量用来存放脚本中的值,这样在需要用这个值的地方就可以用变量来代表,一个变量可以是一个数字,文本或其它一些东西。

JavaScript是一种对数据类型变量要求不太严格的语言,所以不必声明每一个变量的类型,变量声明尽管不是必须的,但在使用变量之前先进行声明是一种好的习惯。可以使用 var 语句来进行变量声明。如:var men = true; // men 中存储的值为 Boolean 类型。

变量命名:JavaScript 是一种区分大小写的语言,因此将一个变量命名为 computer 和将其命名为 Computer是不一样的。

另外,变量名称的长度是任意的,但必须遵循以下规则:

- 1.第一个字符必须是一个字母(大小写均可)、或一个下划线(\_)或一个美元符(\$)。
- 2.后续的字符可以是字母、数字、下划线或美元符。
- 3.变量名称不能是保留字。

## 二.js语句及语法

JavaScript所提供的语句分为以下几大类:

- 1.变量声明,赋值语句:var。

语法如下:var 变量名称 [=初始值]

例:var computer = 32 //定义computer是一个变量,且有初值为32。

- 2.函数定义语句:function, return。

语法如下: function 函数名称 ( 函数所带的参数 )

{

函数执行部分

}

return 表达式 //return语句指明将返回的值。

例: function square ( x )

```
{  
    return x*x  
}
```

### 3.条件和分支语句：if...else，switch。

if...else语句完成了程序流程块中分支功能：如果其中的条件成立，则程序执行紧接着条件的语句或语句块；否则程序执行else中的语句或语句块。

语法如下：if (条件)

```
{  
    执行语句1  
} else{  
    执行语句2  
}
```

例：if (result == true)

```
{  
    response = “你答对了！”  
} else{  
    response = “你错了！”  
}
```

分支语句switch可以根据一个变量的不同取值采取不同的处理方法。

语法如下：switch (expression)

```
{  
    case label1: 语句串1；  
    case label2: 语句串2；  
    case label3: 语句串3；  
    ...  
    default: 语句串3；  
}
```

如果表达式取的值同程序中提供的任何一条语句都不匹配，将执行default 中的语句。

### 4. 循环语句：for，for...in，while,break,continue。

for语句的语法如下：for (初始化部分；条件部分；更新部分)

```
{  
    执行部分...  
}
```

只要循环的条件成立，循环体就被反复的执行。

for...in语句与for语句有一点不同，它循环的范围是一个对象所有的属性或是一个数组的所有元素。

for...in语句的语法如下：for (变量 in 对象或数组)

```
{
```

语句...

}

while语句所控制的循环不断的测试条件，如果条件始终成立，则一直循环，直到条件不再成立。

语法如下： while (条件)

{

    执行语句...

}

break语句结束当前的各种循环，并执行循环的下一条语句。

continue语句结束当前的循环，并马上开始下一个循环。

5.对象操作语句：with，this，new。

with语句的语法如下：

    with (对象名称){

        执行语句

    }

作用是这样的：如果你想使用某个对象的许多属性或方法时，只要在with语句的（ ）中写出这个对象的名称，然后在下面的执行语句中直接写这个对象的属性名或方法名就可以了。

new语句是一种对象构造器，可以用new语句来定义一个新对象。

语法是这样的：新对象名称 = new 真正的对象名

譬如说，我们可以这样定义一个新的日期对象： var curr = new Date()，然后，变量curr就具有了Date对象的属性。

this运算符总是指向当前的对象。

6.注释语句：//，/\*...\*/。

//这是单行注释

/\*这可以多行注释.... \*/

三.js对象的属性及方法.

在JavaScript中是基于对象的编程，而不是完全的面向对象的编程。

那么什么是对象呢？如果你学过一些VB的编程，对这个名词一定不会陌生。通俗的说，对象是变量的集合体，对象提供对于数据的一致性的组织手段，描述了一类事物的共同属性。

在JavaScript中，可以使用以下几种对象：

- 1.由浏览器根据web页面的内容自动提供的对象。
- 2.JavaScript的内置对象，如Date,Math等。
- 3.服务器上的固有对象。
- 4.用户自定义的对象。

JavaScript中的对象是由属性和方法两个基本的元素构成的。对象的属性是指对象的背景色，长度，名称等。对象的方法是指对属性所进行的操作，就是一个对象自己所属的

函数，如对对象取整，使对象获得焦点，使对象获得个随机数等等一系列操作。

举个例子来说，将汽车看成是一个对象，汽车的颜色，大小，品牌等叫做属性，而发动，刹车，拐弯等就叫做方法。

可以采用这样的方法来访问对象的属性：对象名称.属性名称，例：

mycomputer.year=1996，mycomputer.owner = “me”。

可以采用这样的方法，将对象的方法同函数联系起来：对象.方法名字=函数名字或对象.属性.方法名，例：this.display=display，document.writeln（“this is method”）。

多看或多写一些程序，就会理解对象的方法和属性的含义了！

#### 四.js事件的处理

事件是浏览器响应用户交互操作的一种机制，JavaScript的事件处理机制可以改变浏览器响应用户操作的方式，这样就开发出具有交互性，并易于使用的网页。

浏览器为了响应某个事件而进行的处理过程，叫做事件处理。

事件定义了用户与页面交互时产生的各种操作，例如单击超级连接或按钮时，就产生一个单击（click）操作事件。浏览器在程序运行的大部分时间都等待交互事件的发生，并在事件发生时，自动调用事件处理函数，完成事件处理过程。

事件不仅可以在用户交互过程中产生，而且浏览器自己的一些动作也可以产生事件，例：当载入一个页面时，就会发生load事件，卸载一个页面时，就会发生unload事件等。

归纳起来，必需使用的事件有三大类：

- 1.引起页面之间跳转的事件，主要是超连接事件。
- 2.事件浏览器自己引起的事件。
- 3.事件在表单内部同界面对象的交互。

另：

#### Javascript 基础

##### 一、 变量

```
var myBook;
```

```
myBook=5;
```

变量名要求以字母或\_打头，不能含有空格

常见的类型有：字符串，数值，布尔和对象类型。

```
var num=6
```

```
b=(3>5)
```

```
false true
```

##### 二、 表达式与操作符

###### 1、 比较操作符

```
== != > < >= <=
```

###### 2、 运算操作符

```
+ - * / % ++ --
```

###### 3、 逻辑操作符

与&&，或||，非!

###### 4、 位操作符

& | ^ ( 异或 ) ~ << >> >>> (填0右移操作符)

## 5、赋值操作符

=

+= -= \*= /=

&= |= ^=

<<= >>= >>>=

## 6、其它操作符

条件操作符 : (条件) ? 值1 : 值2    a=5   b=6   c=(a>b)?a-b:a+b

new操作符   var

com=new   Array("Zhang","Li","wang","Chen")

com[2]

delete 操作符   delete com[2]

## 7、

### 三、语句

#### 1、条件语句

( 1 ) if.....else

if (mark>60)

    s="pass"

else

    s="fail"

( 2 ) tch   case   标签1 : 代码块1 ; break;

case   标签2 : 代码块2 ; break;

.....

case   标签n : 代码块n ; break;

default: 缺省代码块 ;

}

( 3 )

#### 2、循环语句

( 1 ) for 语句

for ( 初始表达式;循环条件;递增表达式 )

{ 代码块   }

( 2 ) while 语句

while ( 循环条件 )

{代码块}

( 3 ) do.....while语句

do{

    代码块

    } while ( 循环条件 )

( 4 ) label语句

label : 代码块

( 5 ) break语句

跳出循环语句或tch

break label 跳出label标识的代码块

( 6 )

3、 其他语句

( 1 ) for.....in语句 [forin.htm]

for (变量 in 对象) {

代码块 }

( 2 ) with ( 对象 ) {

代码块 }

( 3 ) 注释

// 注释一行 /\* \*/

( 4 ) return

4、

四、 函数

1、 函数的定义

function 函数名 ( 参数列表 )

{ 代码块

}

2、 函数的调用

函数名 ( 参数列表 )

3、 javascript 的全局函数

( 1 ) eval ( 字符串 )

执行该字符串

( 2 ) parseInt ( 字符串 , [基数] ) parseFloat ( 字符串 )

var s="3.14"

var j=parseInt(s)

var k=parseFloat(s)

parseInt("2B",16)=

( 3 ) isNaN ( 表达式 ) : 不是数字

( 4 ) Number ( 对象 ) 和 String ( 对象 )

Var n=new Number(20)

document.write(n.toString(16))

( 5 ) Escape ( 字符串 ) 和unescape ( 字符串 ) 将消息串格式转换为ASC码格式

4、 方法

( 1 ) 滚动窗口scroll()

scroll(x,y) : 移至窗口某一点 , 左上角为0,0 [winscroll.htm]

( 2 ) 设置延迟setTimeout ( "表达式", 时间 ) 时间 : 以毫秒为单位

( 3 ) 清除延迟clearTimeout ( "延迟号" )

如 : id=setTimeout("disp()",1000)

clearTimeout(id)

( 4 )

## 五、 Javascript 中的对象

### 1、 建立自定义对象

方法1 : 对象={属性1 : 属性值1 , 属性2 : 属性值2.....属性n : 属性值n}

方法2 : 先定义构造函数 , 再new创建对象实例。

如 : function car(thecolor,thenumber,thewheels)

{ this.color=thecolor;

this.number=thenumber;

this.wheels=thewheels; }

var mycar=new car("RED","13245",4);

### 2、 定义对象的方法 [oop.htm]

function ReportInfo( )

{ var information=new string;

information="color:"+this.color+"<BR>";

information+="Number:"+this.Number+"<BR>";

information+="Wheels"+this.wheels;

window.document.write(information);

}

### 3、 javascript核心语言对象

#### ( 1 ) 数组对象 ( Array )

建立数组 : var st=new Array("zhang","wang","li","chen");

var st1=new Array(4)

访问数组元素 : st[2]

数组对象的属性 length (长度) [forin.htm]

方法 sort( ) 按ASCII码排序 sort([比较函数名]) [sort.htm]

比较函数返回值(a与b比较) <0 b排在a 的前面

=0 保持原来次序

>0 a排在b的前面

reverse( ) 元素颠倒顺序

join(分隔符) 转换成字符串

#### ( 2 ) 字符串对象 ( String )

属性 : length

方法 : toUpperCase() 转换为大写字母

toLowerCase() 转换为小写字母

indexOf(字符串, 起始位置) 返回子字符串在字符串中的位置, 若没有, 则为-1

LastIndexOf(字符串, 起始位置) 返回子字符串在字符串中最后的位置

charAt(位置) 返回字符串中下标位置的字母

substring(位置1, 位置2) 返回位置1, 位置2间字符串

split(分界符) 按分界符的分解成数组元素

以下的为格式化字符串方法 [str.htm]

big() blink() bold() fixed() fontcolor() fontsize() italics() small()

strike() sub() sup()

### ( 3 ) 日期对象 ( Date )

创建日期对象

```
var 对象名称=new Date (参数)
```

```
var theDate=new Date( )
```

```
var theDate=new Date( 1999,1,1)
```

方法 : getYear( )

getMonth( )

getDate( )

getHours( )

getMinutes( )

getSeconds( )

setYear (年份)

setMonth(月份)

setDate(日期)

setHours(小时数)

setMinutes(分钟数)

setSeconds(秒数)

getTime(毫秒数) 获得1970年1月1日0时0分0秒开始的毫秒

setTime(毫秒数)

### ( 4 ) 数学对象 ( Math )

属性 : PI 圆周率 3.14159265

SQRT2 2的平方根 1.414

LN2 2的自然对数 0.693147

E 2.718281828459

LN10 10的自然对数 2.302585

LOG2E 以2为底E的对数 1.442695

LOG10E 以10为底E的对数 0.4342944819

SQRT1-2 0.5的平方根 0.7071

方法 : min(值1, 值2)

max(值1, 值2)

round(数值) 四舍五入



ceil (数值) 返回 $\geq$ 参数的最小整数 负值取0 ( 向上取整 )  
floor (数值) 截尾取整 ( 向下取整 )  
random() 0-1的随机数  
sqrt(数值) 返回数值的平方根  
abs(数值) 取绝对值