**一、Javascript Object**

Arrary Object

Boolean Object

Date Object

Math Object

Number Object

String Object

RegExp Object

Functions Object

Events Object

**二、browser Object**

Window Object

Navigator Object

Screen Object

History Object

Location Object

**三、HTML Object**

Dom Document Object

Dom Element Object

Dom Attribute Object

Dom Event Object

Arrary Object

1. 创建arrary有三种 方式：

new Array();

new Array(size);

new Array(element0, element1, ..., elementn);

1. 属性

Constructor： 返回对创建此对象的数组函数的引用。

Length: 设置或返回数组中元素的数目。

Prototype: 对arrary object 进行扩展。

1. 方法

(1). Concat(opts): 合并数组，opts需要合并的数组，并返回新的数组；

(2). Join(opts): 把数组的中的所有元素放入一个字符串，opts为分隔的字符，参数为空时，默认为”,”;

(3). Pop(): 删除数组中最后一个元素，并返回删除的元素，删除后数组的长度会变更；

(4). Push(opts, optsN,…): 向数组中添加一个或多个元素，并返回新的数组长度；

(5). Reverse(): 颠倒数组中元素的顺序，并返回颠倒后的数组；

(6). Shift(): 删除数级中第一个元素，并返回删除的元素，删除后会改变数组的长度；

(7).Slice(start, end): 根据下标选定（复制）数组中指定的元素， start: 从哪里开始选定，可用负数，负数为从后向前选定，end：从哪里停止选定，start为必需，end为可选 ，如果end没有传入，则从start的位置开始选定到数组的最后一位，选定后不会修改数组的长度，而是返回选定的数组；

(8). Sort(fn): 对数据的元素进行排序， fn: 规定排序顺序，必须是函数。如果没有传入参数，则按元素的字母进行排序；

(9). Splice(position, delNum/0, addParams): 向数组中添加/删除项目，然后返回添加/删除的的项目，此方法直接向数组修改；position： 规定添加/删除项目的位置，如果使用负数可从数组结尾处开始；delNum/0: 此参数如果为0则表示添加，如果不为0则删除项目；addParams: 向数组添加的新项目；

(10). ToSource(): 表示对象的源码，该原始值由Array对象派生的所有对像继承，目前只有FF浏览器支持，不常用；

(11). toString(): 把数组转换成字符串，并返回，跟没有参数的Join()返回的一致；

(12). toLocaleString(): 把数组转换成本地字符串，它首先调用每个元素的此方法，然后使用地区特定的分隔符把生成的字符串串连起来，开成一个字符串；

(13). unshift(): 向数组的开头添加一个或者多个元素，并返回新的长度；

(14). valueOf(): 返回数组对象的原始值，该原始值由Array对象派生的所有对象继承；

Boolean Object

对象表示两个值： true or false

语法：

New Boolean(value)

Boolean(value);

Value 由布尔对象存放的值或者要转换成布尔值的值

当做为一个构造函数(new)使用时,Boolean()将把它的参数转换成一个布尔值，并且返回该值的Boolean对象；

当做为一个函数使用时，Boolean()将把它的参数转换成一个原始布尔值，并且返回该值；

属性：

Constructor: 返回对创建此对象的Boolean函数引用；

Prototype: 对Boolean Object进行扩展；

方法：

ToSource(): 返回该对象的源代码；

ToString(): 把逻辑转换成字符串；

ValueOf(): 返回Boolean对象的原始值；

Date Object

用于处理日期和时间

语法：

Var date = new Date();

New Date().getTime();

属性：

Constructor: 返回创建对象的Date函数引用；

Prototype: 对Date Object进行扩展；

方法：

Date(): 返回当前的日期和时间；

GetDate(): 返回一个月中的某一天（1-31）；

GetDay(): 返回一周的某一天（0-6）；

GetMonth():返回月份（0-11）；

GetFullYear(): 返回4位数年份；

GetHours(): 返回小时数（0-23）；

GetMinutes(): 返回分钟数（0-59）；

GetSeconds(): 返回秒数（0-59）；

GetMilliseconds(): 返回毫秒数（0-999）；

GetTime(): 返回1970-01-01至今的毫秒数；

GetTimezoneOffset(): 返回本地时间与格林威治标准时间的分钟差；

GetUTCDate(): 根据世界时间从Date对象返回月中的一天（1-31）;

GetUTCDay(): 根据世界时间从Date对象返回周的一天（0-6）；

GetUTCMonth(): 根据世界时间从Date对象返回月份（0-11）；

GetUTCFullYear(): 根据世界时间从Date对象返回4位数的年份；

GetUTCHours(): 根据世界时间从Date对象返回小时数（0-23）；

GetUTCMinutes(): 根据世界时间从Date对象返回分钟（0-59）；

GetUTCSeconds(): 根据世界时间从Date对象返回秒数（0-59）；

GetUTCMilliseconds(): 根据世界时间从Date对象返回好描述（0-999）；

Parse(): 返回1970-01-01午夜到指定日期的毫秒数；

SetDate(): 设置Date对象中月的某一天（1-31）；

SetMonth(): 设置Date对象中月份（0-11）；

SetFullYear(): 设置Date对象中的年份（4位数）；

SetHours(): 设置Date对象中的小时（0-23）；

SetMinutes(): 设置Date对象中的分钟（0-59）；

SetSeconds(): 设置Date对象中的秒钟（0-59）；

SetMilliSeconds(): 设置Date对象中的毫秒数（0-999）；

SetTime(): 以毫秒数设置Date对象；

SetUTCDate(): 根据世界时间设置Date对象中月份的一天（1-31）；

SetUTCMonth(): 根据世界时间设置Date对象中月份（0-11）；

SetUTCFullYear(): 根据世界时间设置Date对象中的年份（4位数）；

SetUTCHours(): 根据世界时间设置Date对象中的小时（0-23）；

SetUTCMinutes(): 根据世界时间设置Date对象中分钟（0-59）；

SetUTCSeconds(): 根据世界时间设置Date对象中的秒钟（0-59）；

SetUTCMilliSeconds(): 根据世界时间设置Date对象中的毫秒数（0-99）；

ToSource(): 返回该对象的源代码；

ToString(): 把Date对象转换成字符串；

ToTimeString(): 把Date对象的时间部分转换成字符串；

ToDateString(): 把Date对象的日期部份转换成字符串；

ToUTCString(): 根据世界时间，把Date对象转换成字符串；

ToLocaleString(): 根据本地时间格式，把Date对象转换成字符串；

ToLocaleTimeString(): 根据本地时间格式，把Date对象的时间转换成字符串；

ToLocaleDateString(): 根据本地时间格式，把Date对象的日期转换成字符串；

UTC(): 根据世界时间返回1970-01-01到指定的日期毫秒数；

ValueOf(): 返回Date对象的原始值；

Math Object

用于执行数学任务

语法：

Math.PI/abs() 不需要进行new

属性：

E: 返回算术常量E，即自然数的底数（约等于2.718）；

LN2： 返回2的自然对数（约等于0.693）；

LN10： 返回10的自然对数（约等于2.302）；

LOG2E： 返回以2为底的e对数（约等于1.414）；

LOG10E： 返回以10为底的e对数（约等于0.434）；

PI： 返加圆周率（约等于3.14159）；

SQRT1\_2: 返回2的平方根倒数（约等于0.707）；

SQRT2：返回2的平方根（约等于1.414）；

方法：

Abs(x): 返回数的绝对值；

Acos(x): 返回数的反余弦值；

Asin(x): 返加的正弦值；

Atan(x): 以介于-PI/2 与 PI/2 弧度之间的数值来返回x的反正切值；

Atan2(y, x): 返回从x轴点（x,y）的角度（介于 –PI/2 与 PI/2弧度之间）；

Ceil(x): 对数进行上舍入；

Cos(x): 返回数的余弦；

Exp(x): 返回E的指数；

Floor(x): 对数进行向下舍入；

Max(x,y): 返回x和y 中的最高值；

Min(x,y): 返回x和y中的最底值；

Pow(x, y): 返回x的y 次幂；

Random(): 返回0-1之间的随机数；

Round(x): 把数四舍五入为最接近整数；

Sin(x): 返回正弦数；

Sqrt(x): 返回数的平方根；

Tan(x): 返加确的正切；

ToSource(): 返回该对象的源代码；

ValueOf(): 返加Math对象的原始值；

Number Object

对原始值的包装对象

语法：

Var N = new Number(x);

Var B = Number(x);

当Number()和运算符new 一起作为构造函数使用时，它返回一个新创建的Number对象。如果不用new运算符，把Number()作为一个函数来调用，它将把自己的参数转换成一个原始值，并且返回这个值。如果转换失败，则返回NaN;

属性：

Constructor: 返回对创建此对象的Number函数的引用；

MAX\_VALUE： 可表示的最大数；

MIN\_VALUE： 可表示的最小数；

NaN: 非字值；

NEGATIVE\_INFINITY: 负无穷大，溢出时返回该值；

POSITIVE\_INFINITY: 正无穷大，溢出时返回该值；

Prototype: 对Number Object的扩展；

方法：

ToString(): 把数字转换成字符串，使用指定基数；

ToLocaleString(): 把数字转换成字符串，使用本地数字格式顺序；

ToFixed(x): 把数字转换成字符串，x:为返回的结果带的几位的小数字参数；

ToExponential(x): 把对象的值转换成指数计算法，x: 0-20 的参数，太大或太小会报错；

ToPrecision(x): 把数字格式化为指定的长度，x: 1-21，如果没有传参，内部则调用toString（）方法；

ValueOf(): 可以字符串返回数字；