《深入理解JavaScript》第 15/20 章

手动转换为布尔值：Boolean(value) ; value ? true : false; !! value

假值：undefined null 0 NaN

Boolean函数的两种调用方式：Boolean(value) new Boolean(bool)

JS把所有的数字都作为浮点数。JS数字是双精度（64位），基于浮点算术的IEEEbi标准。

一个数字字面量可以是整形、浮点型、或十六进制整型。

指数e^x是10^x缩写。 5e^2 500。

对于数字字面量，用来访问一个属性的点必须要和小数点的点区别开来。例如：123..toString()。

手动转换为数字： Number(value) ;+value

检查一个值是否是NaN: isNaN(value)&& value='number'

Infinity是一个错误值，指出了两种错误:一个大到无法表示的数字，或者除以了0。

Infinity使用普通运算符会得到错误的结果。

JS数字的数值和符号是分开存储的，因此每一个非负数都有一个负值，包括0。

如何区分两种零：Math.pow(-0,-1) -Infinity Math.pow(+0,-1) Infinity。

JS只能安全地表示-2^53<i<2^53范围内地整型。

通过定制函数ToInteger()得到整数

function ToInteger(x){x=Number(x); return x<0?Math.ceil(x) :Math.floor(x);}

控制字符：\b空格符 \f换页符 \n换行符 \r回车符 \t水平制表符 \v垂直制表符 NUL字符 \0

手动转换为字符串：String(value) ;''+value ;value.toString() 第一种描述性强

转换并不总是不可逆的。

可以通过两种方法调用String()方法：String(value) new String(value)

length属性表示JS字符串中字符的个数，且它是不可枚举的。

提取子字符串：String.prototype.charAt(pos):返回给定位置pos的字符；String.protorype.charCodeAt(pos):返回给定位置posdeJS字符对应的无符号整数；

String.prototype.slice(start,end):返回一个字符串，原字符串从start（包括）到end（不包括）的字符串，两个参数都可以为负数，此时会将参数与字符串的length属性相加。

String.prototype.substring(start,end)与slice()类似，但是处理参数为负数时且更好地跨浏览器兼容。

函数的调用形式：普通函数 构造器 方法。定义函数：通过函数表达式 通过函数声明 通过Funtion()构造器。

函数表达式会产生一个值：函数对象。var add=function(x,y){return x+y};具名的函数表达式使得函数表达式可以引用它自己，这对递归很有用。

通过Funtion()构造器例：var add=new Function('x','y','return x+y');相对较慢，不建议使用。

函数声明是完全提升的，变量声明是部分（声明有效，赋值无效）提升的。

函数声明相比函数表达式的两个优势：函数声明会做代码提升 它们具有名字。

控制函数调用：func.apply(thisValue,argArray):argArray参数作为函数的arguments传递给函数，thisValue这个参数可以指定执行func时的this的值。

func.bind(thisValue,arg1，……，argN):会执行部分的函数功能。它会创建一个新的函数，这个函数会调用func，并将thisValue指定为this,同时应用以下参数：arg1~argN，

紧随其后的是新函数的实际参数。

arguments变量非常特别，是一个类数组对象，包含了当前函数调用的所有实参。

4中方法调用Date构造函数：new Date(year,month,date,hours,minutes,seconds,millseconds)

new Date(dateTimeStr): new Date('2004-08-29') Date{Sum Aug 29 2004 02:00:00 GMT+0200(CEST)}

new Date(timeValue):new Date(0) Date{Thu Jan 01 1970 01:00:00 GMT+0100(CET)}:用1970年1月1日00：00：00UTC以来的毫秒数创建一个日期。逆运算是getTime()函数：new Date(123).getTime() 123

new Date():用当前的日期和时间创建一个日期对象，效果相当于 new Date(Date,now())。

Date的构造函数：Date.now()作用相当于new Date().getTime()

new.parse(dateTimeString):该函数将dateTimeString转换为毫秒（从1970年1月1日00:00:00UTC开始计算）。

Date的原型方法：Date.prototype.get<<Unit>>():根据当地时间返回单位时间。Date.prototype.set<<Unit>>(number):根据当地时间设置单位时间。

Date.prototype.getUTC<<Unit>>()：根据世界时间返回单位时间。Date.prototype.setUTC<<Unit>>(number):根据世界时间设置单位时间。

将日期转换为字符串：时间：Date.prototype.toTimeString();Date.prototype.toLocaleTimeString()

Date:Date.prototype.toDateString() Date.prototype.toLocalDateString()

日期和时间：Date.prototype.toString() Date.protorype.toLocalString() Date.prototype.toUTCString()