1. 强制类型转换
2. 值类型转换

将值从一种类型转换为另一种类型称为类型转换

隐式类型转换也称为强制类型转换

1. toString

null “null”

undefined “undefined”

true “true”

12 “12” 极小和极大的数字是指数形式

普通对象 自定义toString()或内部属性[[Class]]，如”[object Object]”

[1,2,3] 1,2,3

JSON JSON.stringify()和toString()效果基本相同，结果总是字符串，JSON.stringify()在对象中遇到undefined function symbol时会自动忽略，在数组中返回null，对包含循环引用的对象会报错

console.log(JSON.stringify(42));//"42"

console.log(JSON.stringify("42"));//""42""

console.log(JSON.stringify(true));//"true"

console.log(JSON.stringify(null));//"true"

JSON.stringify()的第二个参数可以是数组或函数，如果是数组，其中包含要处理的属性的名称，如果是函数，对象本身调用一次，每个属性也都调用一次。第三个可选参数指定缩进的格式。

JSON.stringify()不是强制类型转换

将值转换为基本类型值，首先检查是否有valueOf()方法，然后检查是否有toString()

Object.create(null)创建的对象[[Prototype]]属性为null,没有valueOf()和toString(),无法进行强制类型转换

1. toBoolean

假值：undefined null false +0 -0 NaN “”

所有的对象都是真值

1. 显示强制转换

通过String() Number()进行字符串和数字之间的转换

+：将字符串转换为数字 将日期对象强制转换为数字

获取时间戳的方法：

var time1=+new Date();

var time2=new Date().getTime();

var time3=Date.now();//推荐

解析与转换：

解析允许字符串含有非数字字符，按从左到右，遇到非数字字符就停止，转换不允许出现非数字字符，返回NaN

从es5开始，默认解析为十进制，除非特意指定

Boolean()显示转换为布尔值 推荐Boolean(n)和!n

1. 字符串拼接

当至少一个操作数是字符串时，+执行的是字符串拼接操作

var a=[1,2];

var b=[3,4];

console.log(a+b);//"1,23,4"

console.log([]+{});//[object Object]

console.log({}+[]);//[object Object]

1. symbol符合类型

es6允许从符号到字符串的显式类型转换，隐式强制类型转换会报错

1. == ==

==允许在比较时进行强制类型转换，而===不允许

字符串比较按字母表顺序

a<=b被处理为b<a，然后将结果相反

1. 博客
2. 元素可见性

visibility:visible hidden collapse inherit collapse表示(单元格)不显示

display:none脱离了文档流，重新显示需要页面的重新绘制。visibility:hidden保留了元素的物理区域，但不能接收js效果了。

document.hidden表示当前页面是否可见，当浏览器最小化、切换tab、电脑锁屏时。

1. js延迟加载

async defer 动态创建dom 按需异步加载 其中defer只有ie支持

1. RegExp

test() 测试一个字符串是否匹配某个正则表达式

exec() 检索字符串与正则表达式匹配的值

compile() 编译正则表达式

match是支持正则表达式的String对象的方法

1. 属性

onj.hasOwnProperty(“val”)判断一个对象是否含有此属性(不包括原型链)

obj.isPrototypeOf()检查其原型链的对象是否存在于指定对象实例中