/pages/feature/evaluation/evaluation.html

# 字符串求值

## eval求值

字符串求值，返回最后最后一个表达式的执行结果（运行）

返回一个对象

eval(“({name:’tom’})”)

返回一个函数

eval( “ false || function() {}” )

副作用：

会在eval执行的当前作用域增加变量

注意：

如果要返回对象，需要用括号包起来，否则如 {name:’tom’} 会返回’tom’

## 函数构造求值

new Function(“a”, “b”, “return a + b”);

好处：

不会创建闭包，当不想承担闭包的开销时，可以一用。

## 定时器求值

定时器可以接受字符串参数

var id = setTimeout(‘alert(‘go’)’, 100)

## 创建script 实现代码求值

将字符串加 到 script.text上即可执行。

# 值转化

## 真假转化

只有在非布尔值强制转化为布尔值时才会遵循这个规则。

假：

“” , 0, -0, NaN, null, undefined, false

真：

非假的都为真

## 是否相等

== 表示允许类型转换下的值的相等性

===表示检测不允许类型转化情况下的值相等性（严格相等）

经验：

①

如果两个值任意一个可能为true或false，要避免使用 ==。用 ===。

②

如果要比较任意一个可能为特定值（0 , [], “”）避免使用==

③

其他情况下使用 == 都是安全的。

注意：

对于两个非原生值之间的比较，== 和 === 比较只能简单的检查这些引用是否匹配。

原生值与非原生值比较（==），会将非原生值转化为原生值。

## 不等关系

a< b

如果都是字符串

都转化为字符位置

有一个为数字

转化为数字

# 位运算

https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Bitwise\_Operators

按位逻辑操作遵循的规则：

操作数会被转换为32位整数，用比特列（0,1）表示，超过32位的数字会被丢弃。

转换前: 11100110111110100000000000000110000000000001

转换后: 10100000000000000110000000000001

数字-2147483648 和 2147483647 是32位有符号数字所能表示的最小和最大整数。

按位非

var iNum1 = 25; //25 等于 00000000000000000000000000011001

var iNum2 = ~iNum1; //转换为 11111111111111111111111111100110

alert(iNum2); //输出 "-26"

实际上求的反码（数字求负，在减去1）

用法：

1. 取整

~~x

~~-1.100 -1

~~1.2 1

按位与

a & b， 只有当a,b这两个操作数对应的比特位都为1 时，才输出1

如 25和9

25 = 0000 0000 0000 0000 0000 0000 0001 1001

9 = 0000 0000 0000 0000 0000 0000 0000 1011

---------------------------------------------

AND = 0000 0000 0000 0000 0000 0000 0000 1001

25 & 9 // 得到为9

按位或

a | b， 只有当a,b这两个操作数对应的比特位有一个为1 时，输出1

按位异或

a ^ b, 只有当a,b这两个操作数对应的比特位不相同时，输出1

左移

a << num a向左移动指定的位数num。向左移出的位被丢弃，右侧用0补充。

有符号右移

a << num a向右移动指定的位数num。向右移出的位被丢弃，拷贝最左侧的的位以填补左侧。

因而符号位保留

无符号右移

a <<< num a向右移动指定的位数num。向右移出的位被丢弃，左侧用0补充。

因而总是正数

注意：

任意数值x按位非操作，结果都是-（x+1）

任何数值x与0按位与操作，结果都是0,。任数值k与-1 按位与操作，结果都是k，任意数值m 与 1按位与操作，结果是1,0（取决于m是奇数或偶数）

任意数值x与0 按位或操作，结果都是x。任意数值k与-1按或操作，结果都是-1.

任何数值x与0异或操作，结果都是x。任意数值k与-1异或操作，结果都是-x。

数字x左移y，则得到 x\* 2y