变量：存储数据或者操作数据

原始（基本）数据类型：number, string, boolean, null, undefined, object

Number:数字类型（整数和小数）

不要用小数验证小数，有BUG，历史遗留问题双精度不精确

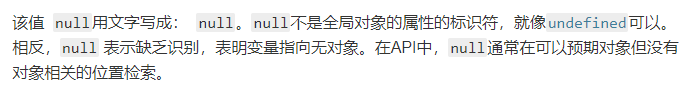
String:字符串类型（值一般用单引号或者双引号括起来）

Boolean：布尔类型（ture（1）和false（0））

Null：空类型，值只有一个：null，一个对象指向为空了，此时可以赋值为null

变量的值如果想为null，必须手动赋值





Undefined：未定义，值只有一个：undefined

变量声明了，没有赋值，结果就是undefined

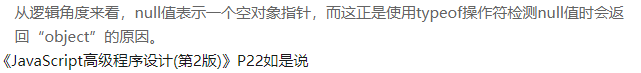
函数没有明确的返回值，如果调用了，结果也是undefined

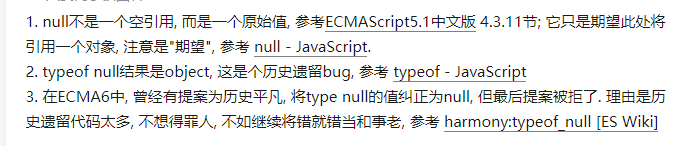
Undefined与数字相加和Number类型转换结果是：NaN

Obeject：对象

判断变量的数据类型：typeof（变量名）只能判断6个基本类型（原始类型）

用typeof判断null的数据类型结果为：object ==>





<https://juejin.im/post/5a439e30f265da4335630bf4>

NaN:不是数字

isNaN判断变量是不是数字，数字为false，不是数字为true

转义符：右斜杠\

字符串拼接：使用+可以把多个字符串放在一起形成一个字符串

只要一个是字符串的，其他是数字的相加也是拼接效果

一个字符串和一个数字相减会将字符串转变成数字，然后相减

其他类型转数字类型

parseInt（）；转整数，从左往右判断直到遇到不是数字的停止转换，开头就不是数字的输出结果为NaN

ParseFloot（）；转小数

Number（）；转数字，比较严格，有个不是数字的都是NaN

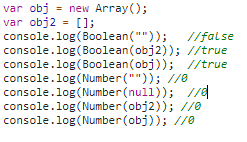
Number（null）：0

Number（“ ”）: 0

其他类型转字符串类型

toString（）变量有意义则使用tostring（）使用没有意义的变量转换则会报错

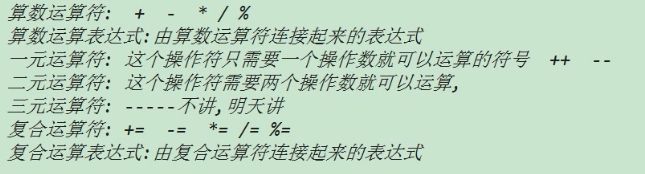
String（）变量没有意义则使用String（）

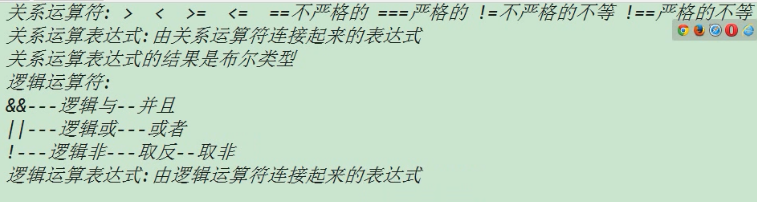
疑问？？？

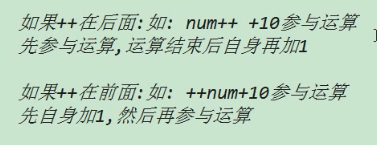
其他类型转换布尔类型



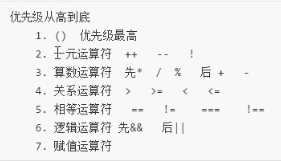
操作符







Switch-case中的判断是严格模式的比较（值和类型）



字面量：把一个值赋值给一个变量

Var n = 1; 字面量

Var n = y：不属于字面量,因为两个都是变量

数组：一组数据，有序的数据，一次性存储多个数据

通过构造函数创建数组

Var 数组名 = new Array();

通过字面量方式创建

Var 数组名 = [];

如果数组中没有数据，但是有长度，数组中的每一个值都是undefined

数组存储数据的类型可以不一样