1.流程控制语句

1.If语句中判断中的强制转换

如果if()里面传入的不是一个布尔值或者 表达式结果不是布尔值的

那么会强制转换成布尔值进行判断

转换标准: 0 null undefined “” false NaN都当做false

小括号里面是if语句对结果本身的强制转换,对原来的变量类型并没有转换

2.分支语句

switch 分支语句  
switch(判断条件){  
 //分支  
 case 固定值1(所匹配的标准):  
 语句体n多句  
 break;中止 中断 打断  
 //分支  
 case 固定值2(所匹配的标准):  
 语句体n多句  
 break;中止  
 .  
 .  
 .  
 default:  
 默认语句;  
 break;

*分支语句注意点:  
 switch()都可以放什么 只要有值即可  
 多个case 条件可以重复 但是不推荐 因为条件一样也只会匹配第一个出现的case  
 分支语句的结束条件是 default 么? 不是 是break  
 break 可以省略不写么? 绝对不能!  
 原因: 省略会引起case穿透  
 default 可以省略么? 可以 但是不推荐 因为程序要考虑完整性  
 除非接收值是单选  
  
 switch 比较值时 底层用的是全等 === switch 不会强制转换类型去比较*

var key=15;  
switch(key){  
 case 1:  
 console.log("拨打110");  
 break;  
 case 2:  
 console.log("拨打120");  
 break;  
 case 3:  
 console.log("拨打119");  
 break;  
 case 4:  
 console.log("拨打亲爱的");  
 break;  
 case 5:  
 console.log("拨打父母的");  
 break;  
 default:  
 console.log("按错了,瞎按啥!!");  
 break;

if结构和switch 的区别:

if结构可以做 几个固定值的判断 还可以做到范围值的判断

switch语句 一般用来做几个固定值的匹配

循环结构

*//循环的意义 反复执行大量重复的代码 需要封装到循环里面 :提高了代码的复用率*

3.循环结构: for循环

三个条件可选 但一般 一起使用

for(初始化变量; 循环判断条件; 更新循环变量){

循环语句体n多句

}

a).第一次加载会初始化变量 也只会加载一次

b)判断循环条件 看结果是true还是false 如果是true 去执行c false则循环终止

c)执行循环语句体

d)更新循环变量

继续走b~~~d的步骤

//for循环的标准格式  
// for(var i=0;i<10;i++){  
// console.log("Hello"+i);  
//  
// }

其他写法

//for循环的其他写法  
// var i=0;  
// for(;i<10;){  
// console.log("Hello"+i);  
// i++;  
//  
// }

死循环: 循环死了 出不来 无限循环

for(;;){  
 console.log("无限打印!!!非战斗人员紧急撤离!!!")  
}