### NaN概念和一元运算转换

NaN和任何其他计算都为NaN

NaN不等于NaN

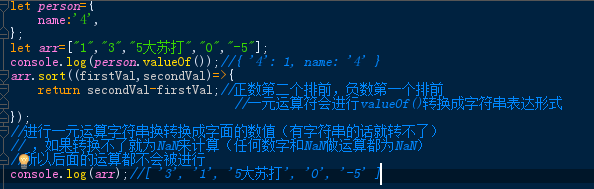
NaN为不可转换成数字的东西

True转1 false转0

String做运算会转换成字面数字不能转换则以NaN计算

Object会使用valueof转换成字符串表达自己内容

然后字符串如果不能转换数字则为NaN来计算



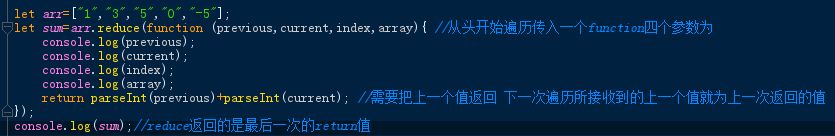
### Reduce和reduceRight

Reduce意思是使..分解

它需要一个函数 该函数有四个参数，见图

从左往右遍历 每一次遍历都会返回一个值给下一次遍历

最终返回最后一次return的东西



reduceRight一样不过是从右往左

### Filter和map和some和every

Let Arr=[1,2,3]

Arr=Arr.filter((item,index,arr)=>{

If(xxxx)

Return true;

})

会过滤掉不需要的项 返回一个判断为true的项组成的数组（注意 是重新返回一个）

Arr=Arr.map((item,index,arr)=>{

Do someThing..

Return someThing

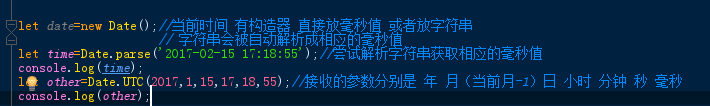
})

会返回一个以自己return元素组成的数组，会遍历每一个项 如果某个项没有return元素则为undefined;

Some和上面一样的用法 用于判断返回值有一个为true则返回整个数组

Every 返回值要全部为true才返回数组

### Date API方法



Es5添加了一些新方法

Date.now()

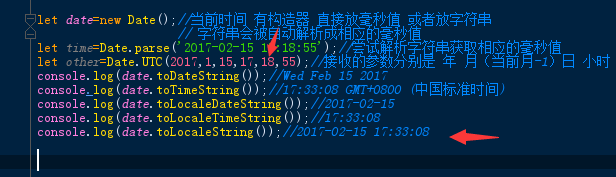
Let start=Date.now();//返回当前时间的毫秒值

doSomeSthing;

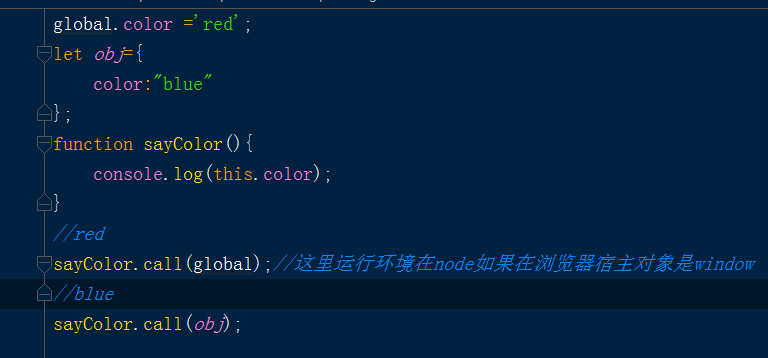
Let end=Date.now();

Let result=end-start 可以获取间隔时间

日期格式化方法：



### Apply()和call()



Call()和apply()效果一样

只不过前者第二个参数要显示的声明

Call(当前域(该方法的this对应的域),xxx变量,xxx变量)

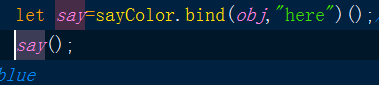
Apply(当前域(该方法的this对应的域),此处可以直接给一个数组[a,b,c])

可以用来显示控制方法所生效的域 很有用

### Bind()

Es5还定义了bind方法 和上面的效果是一样的

不过bind是重新创建了一个绑定了指定域的方法



把obj绑定到这个方法生成的实例的this上 后面参数也要对应的传和call一样

然后显示的去调用重新生成的实例

### toFixed()四舍五入

可以用来对浮点型取位 还可以四舍五入

Var num=10.005

Num.toFixed(2) 10.01

表示四舍五入到两位小数

### String相关操作

stringValue.length//返回字符串的长度

stringValue.chartAt(index) //返回指定角标的字符

Es5中还定义了

stringValue[index]用方括号去访问

stringValue.slice(3)表示从角标为3的位置开始 返回一个新的字符串

stringValue.slice(3,7)表示从角标为3的位置到角标为7的位置

stringValue.substring(3,5)表示从角标为3的位置开始 返回3之后的5个元素组成的数组

Indexof(xxx,number) 从前往后找xxx 第二个参数指定从第几个角标开始找

Lastindexof(xxx,number) 从后往前找xxx 第二个参数指定从第几个角标开始找

stringValue=stringValue.trim()会重新返回一个删除掉了所有前缀的字符串