# html

## h1-h6:标题标签

## hr: 水平线标签

## p:段落标签

## <!--注释信息-->

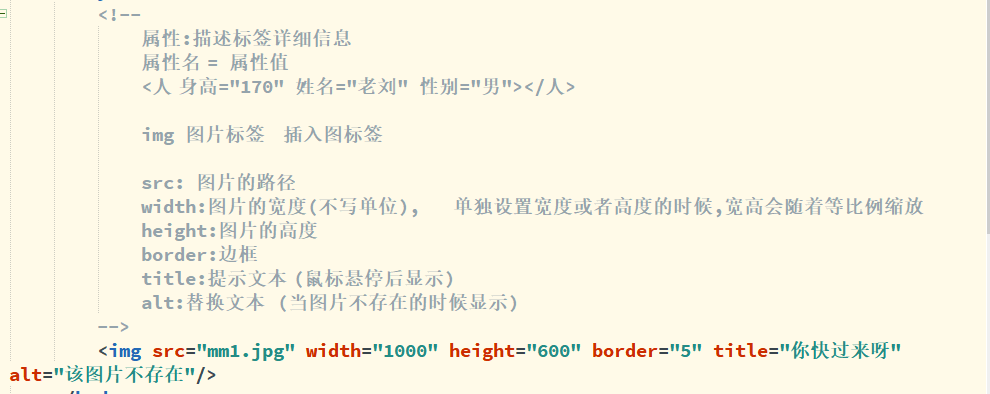
给代码进行注释,方便维护 快捷键 ctrl+/

## font:文本标签



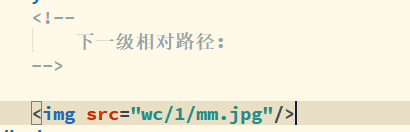
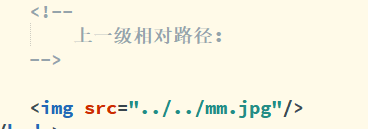
## br:换行标签

## img 图片标签



### 路径src=””



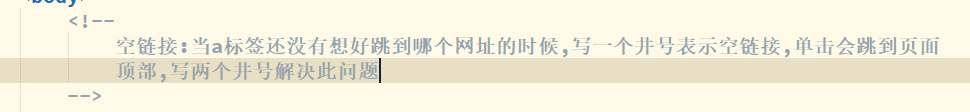
 

注意:文件夹和文件不要用中文,不要有空格和括号

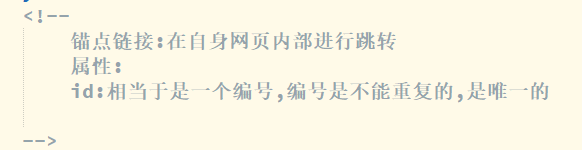
## a超链接

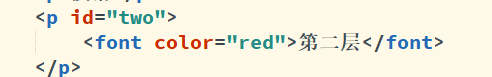


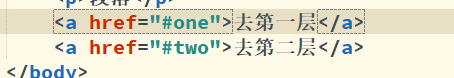


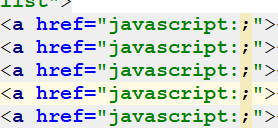




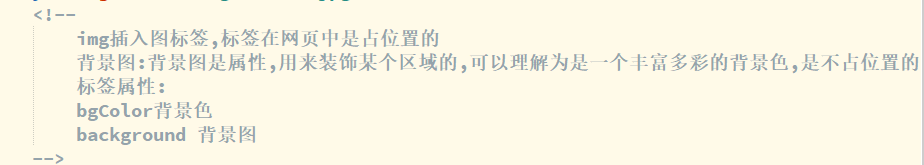






javascript:;表示空连接

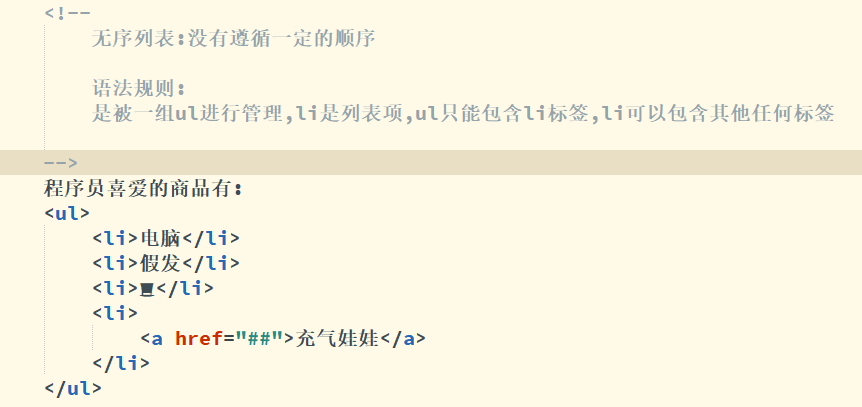
## 插入图和背景图



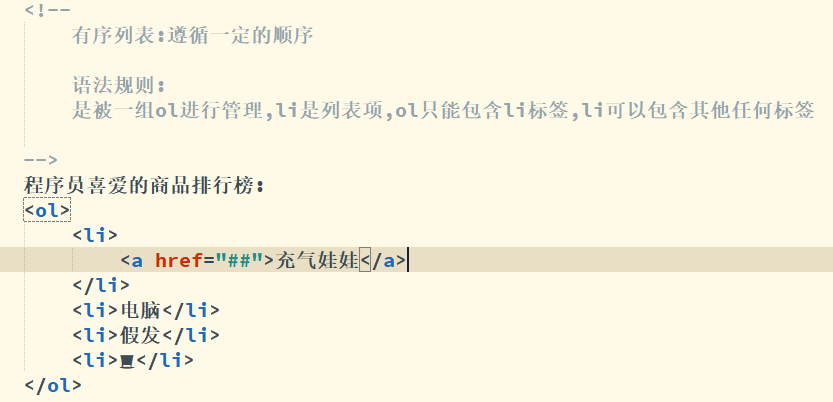
注：如果图片仅仅起一个装饰作用，则使用背景图

## 列表

### <ul>



### <ol>



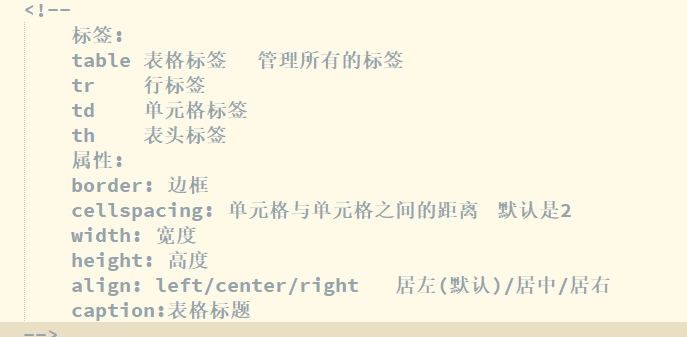
### <dl>

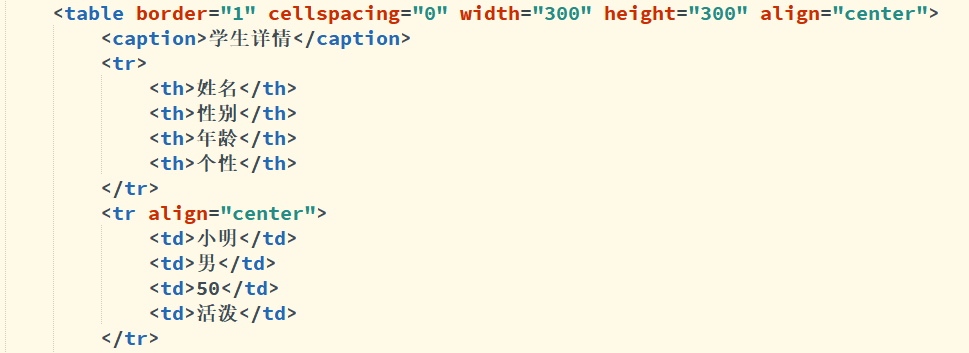


## 特殊字符

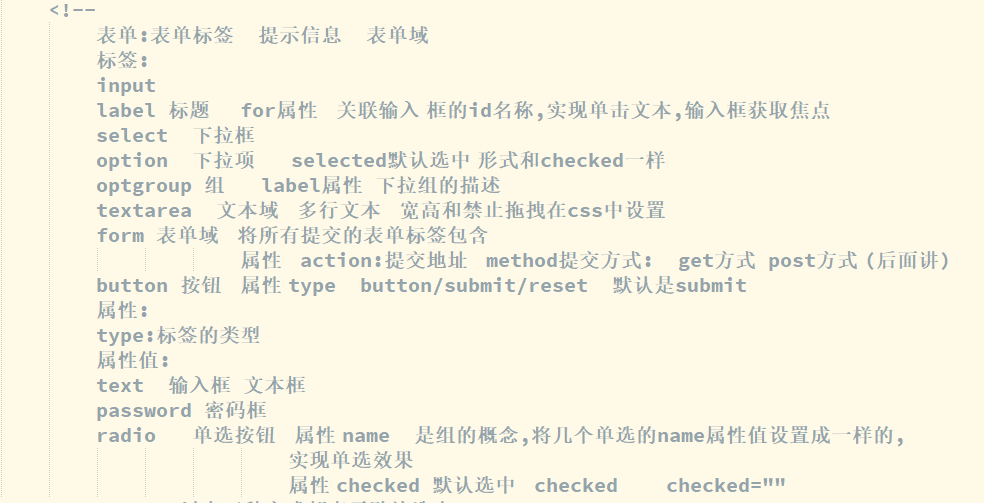


## 表格





## 表单

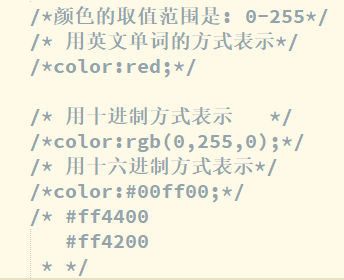






# css

## color



## 常用文本属性



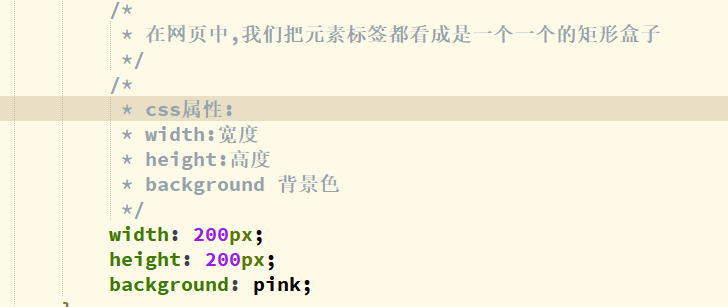
## 字体的表示方式



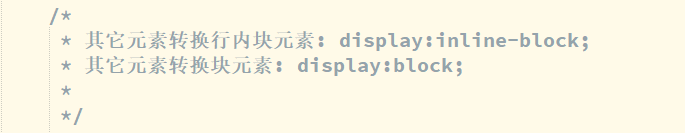
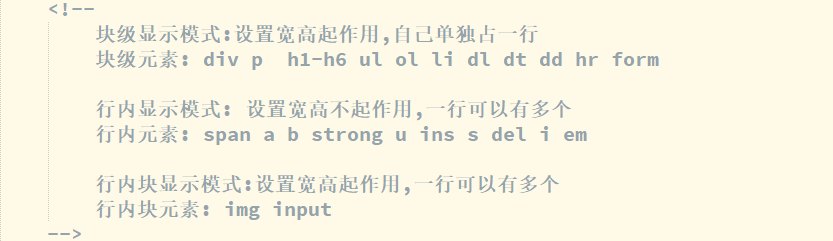
字体可以用以上三种形式表示,字体可以设置多个,用逗号隔开



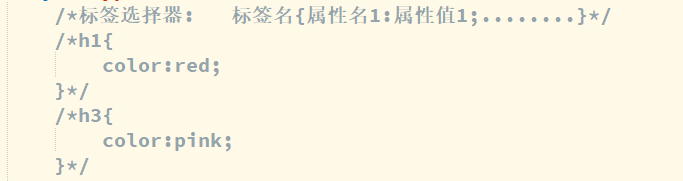
## 盒子实体化三属性



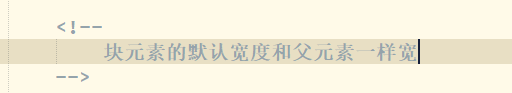
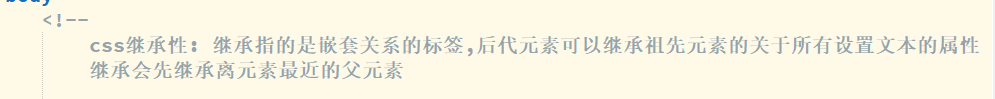
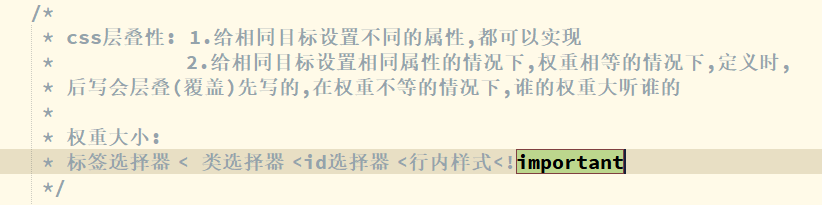
## 显示模式转换



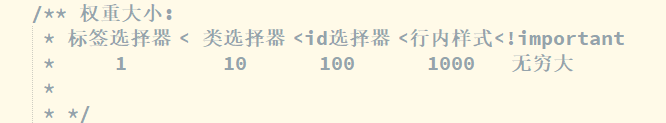
## 选择器



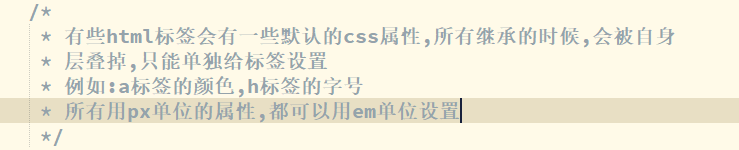
## css属性



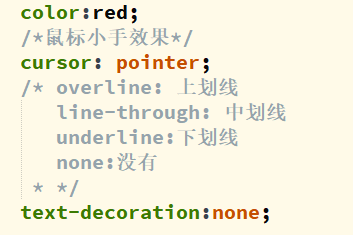
## 权重



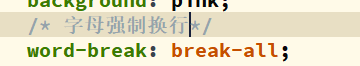
## 标签默认自带属性



## 常用属性







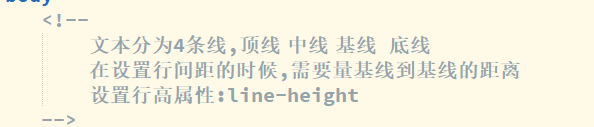


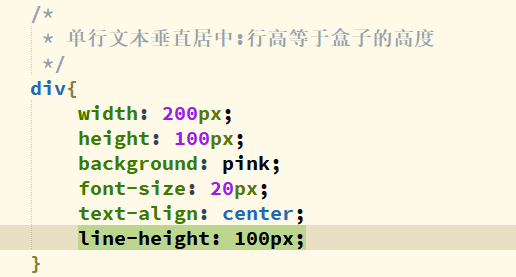
## 伪类



## 行高(行间距)







注：可以通过调整行间距来调整字体的位置，使用时给盒子设定一个固定的宽度，调整行高

## 复合属性

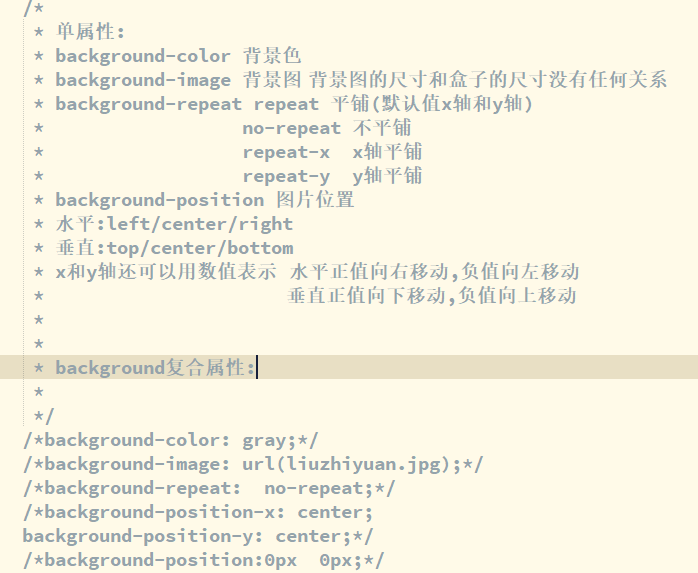
### font

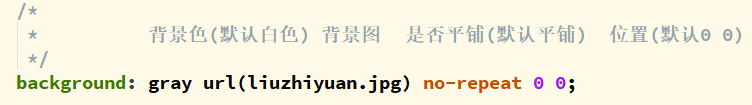


### border

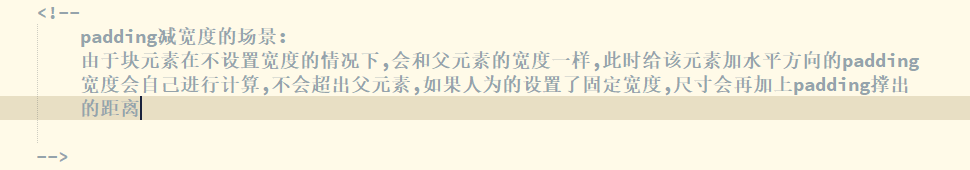
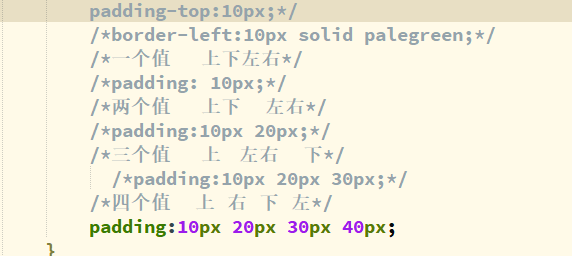
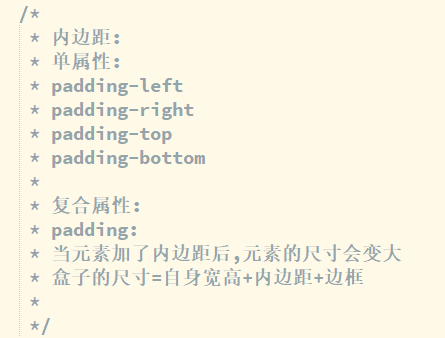


### background

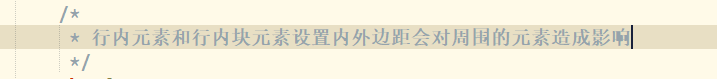
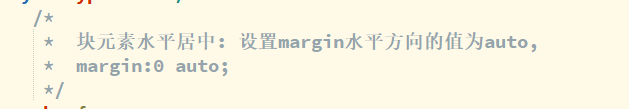
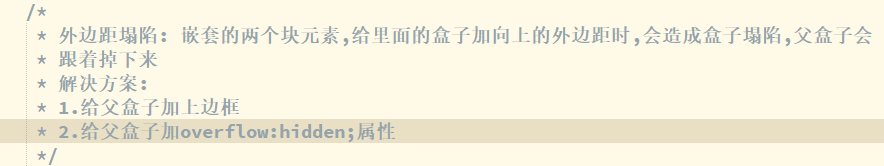
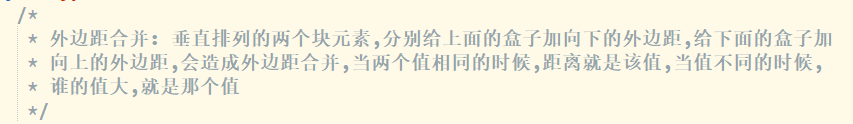




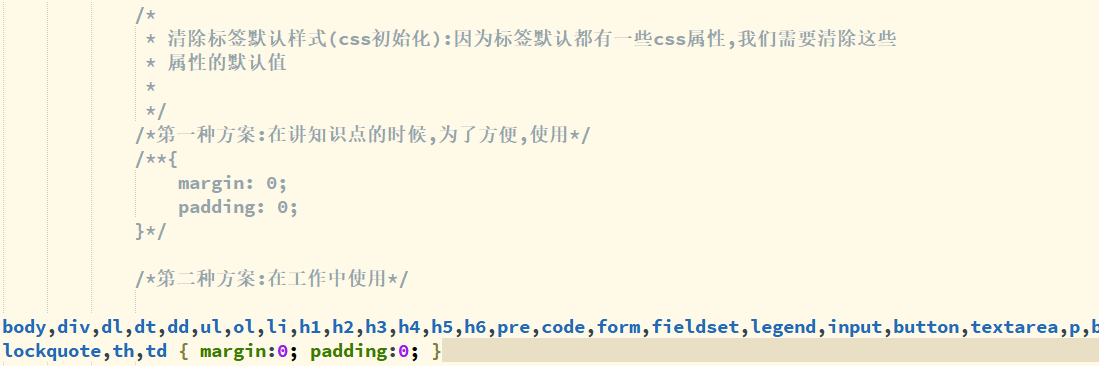
### 内边距



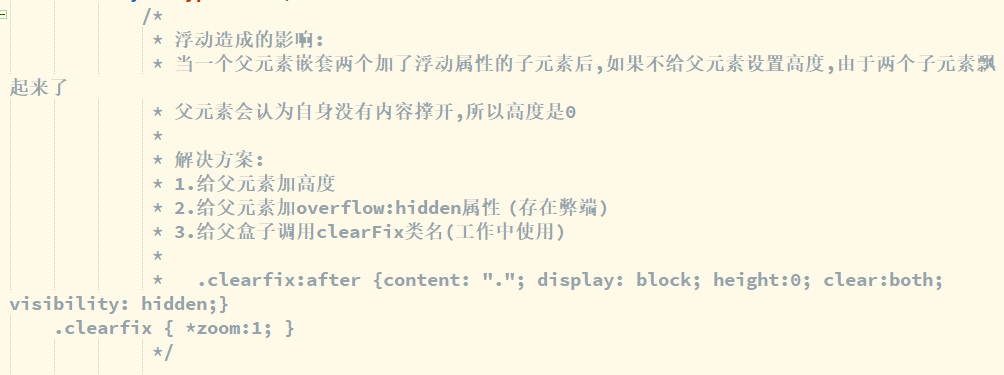
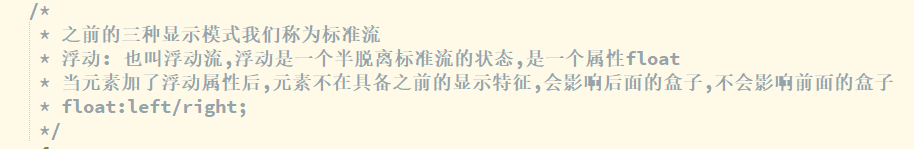
### 外边距



## 清除标签默认样式(css初始化)



## 浮动

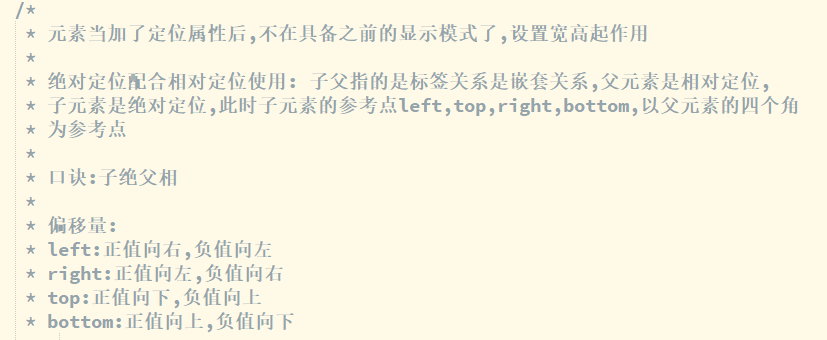
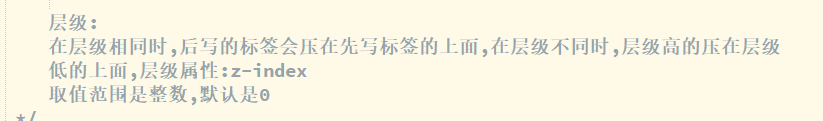
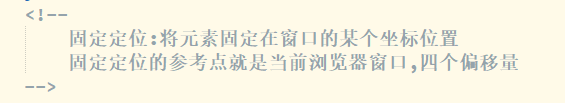
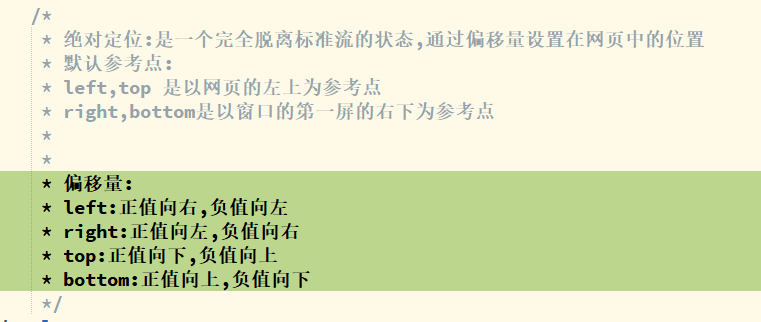
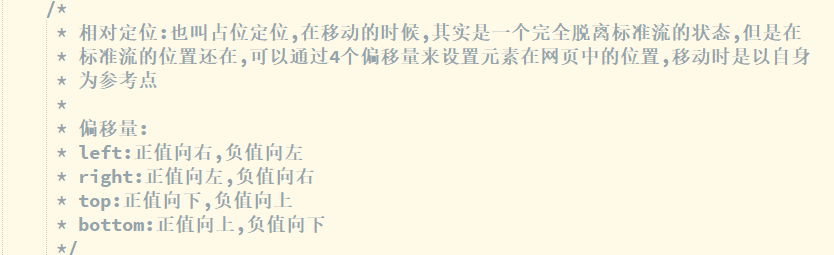
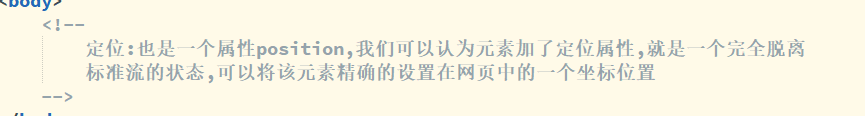


## img底部留白问题

行内块元素的一些缺陷



## 定位



## 隐藏



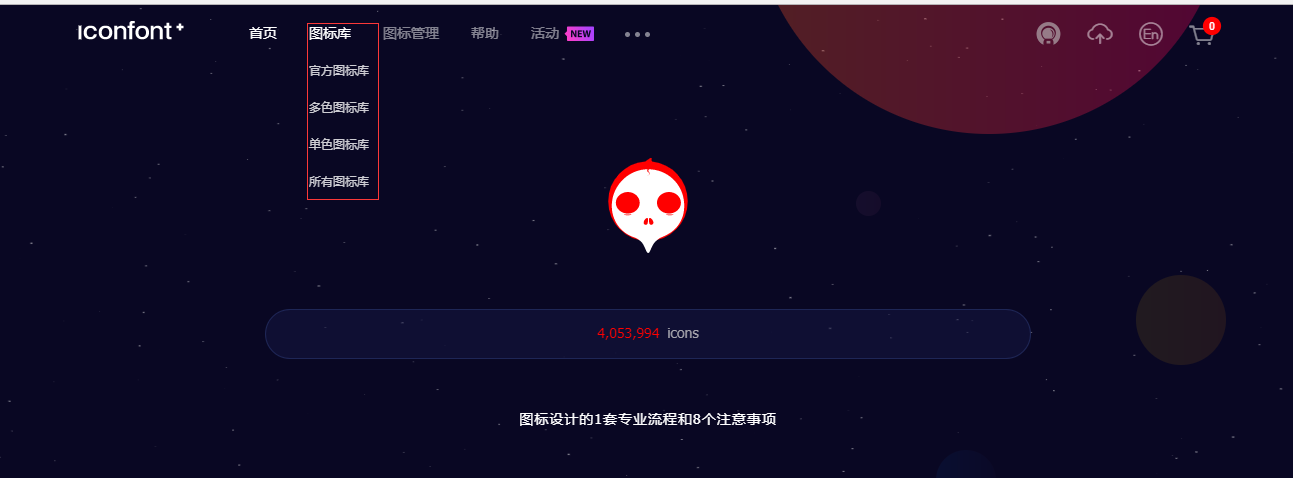
## 透明度

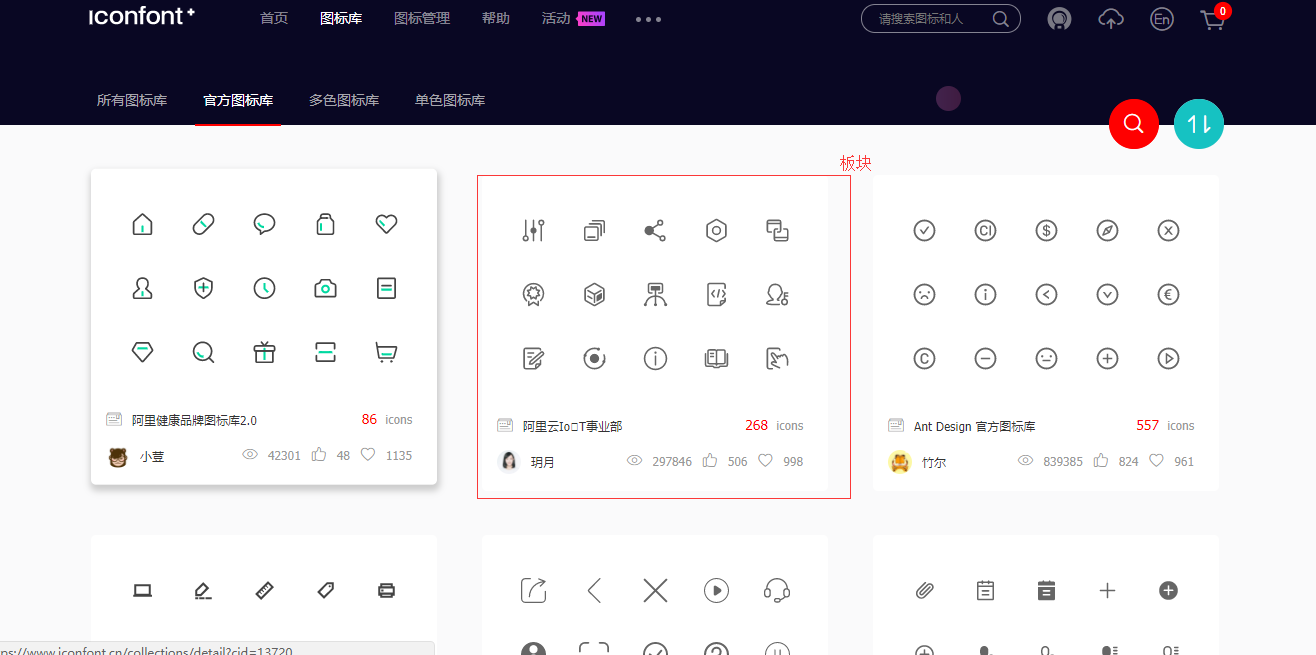
rgba(255,255,255,0.5)

## 图标字体

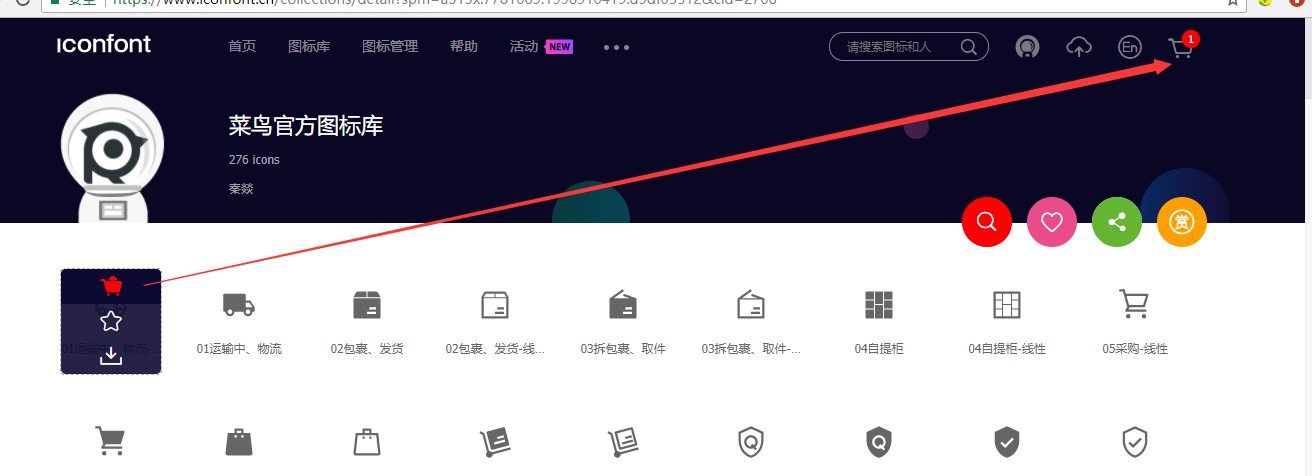
<https://www.iconfont.cn/> 下载

下载方式

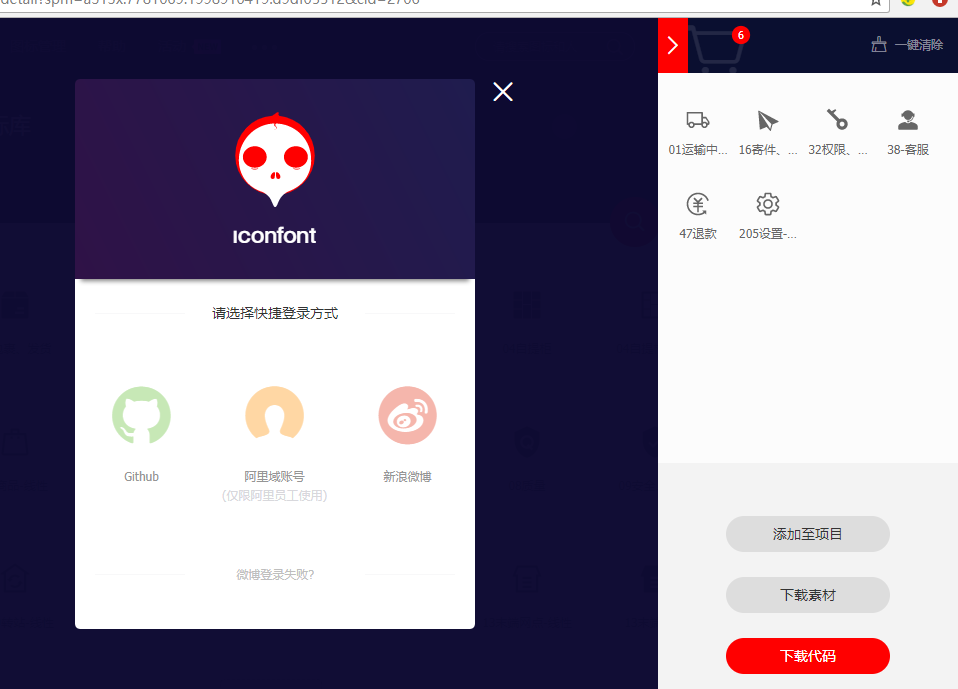




放进购物车

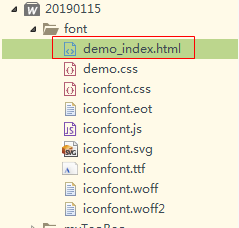


下载前需要进行登录,注册方式有三种



下载之后把font文件夹放在根目录下

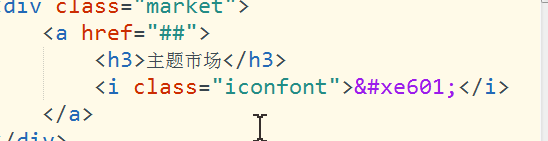




打开框内网页 使用步骤如下

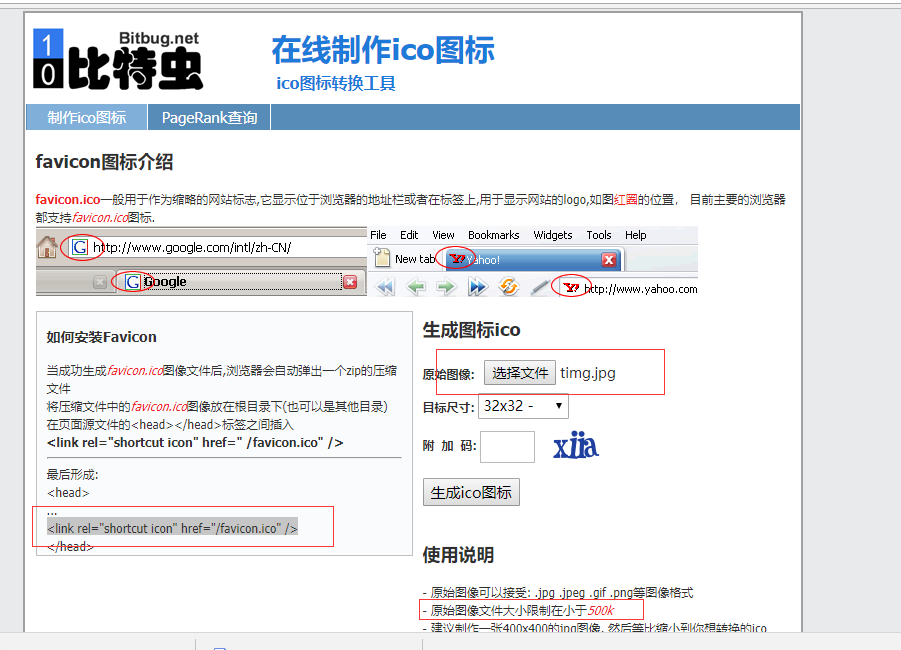






## ico图标

<http://www.bitbug.net/> 转换网址



# 网页布局

## 网页的版心

每个网页都有一个固定宽度的版心,通常是960 980 1000 1180 1190 1200 1300,用来显示网页的主要区域.版心之外的,称为通栏

## Logo的布局

因为logo是网页中最重要的,所以用h1,h1只能出现一次,由于logo是可以单击跳转的,所有要在h1里嵌套一个a标签,在a标签中可以选择插入图和背景图两种方式

## 3.网格布局

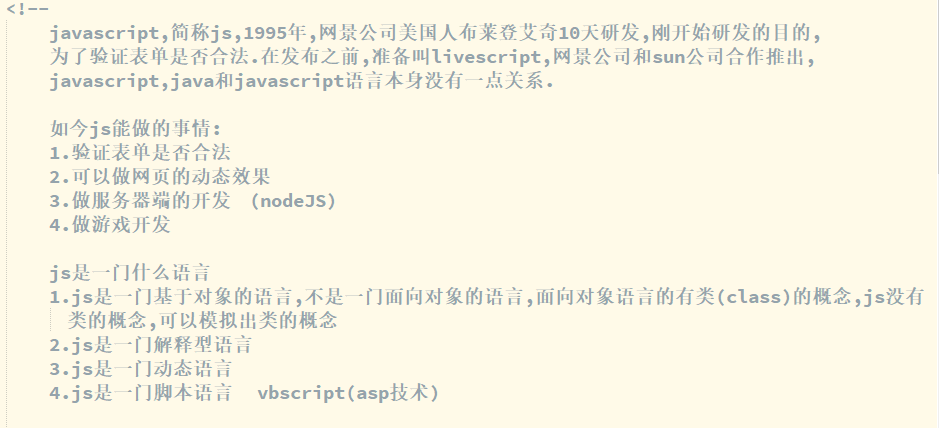
1.父元素用ul，子元素用li；

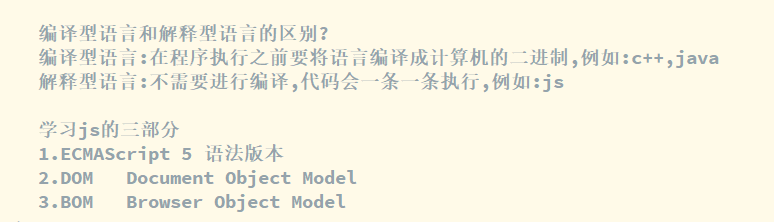
2.给li加上margin-right、bottom，宽度为1px；

3.给ul加上padding-top、left，宽度也为1px；

# JavaScript

## js简介





## 体验js

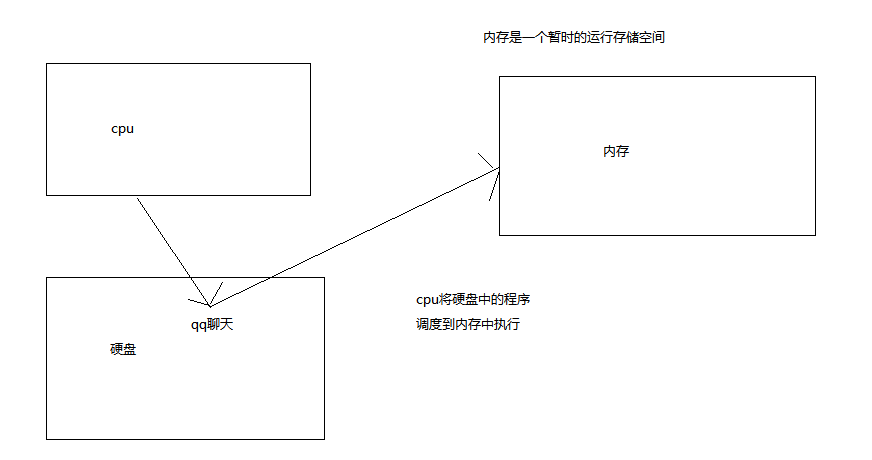


## 三种书写位置

## 变量



### 内存

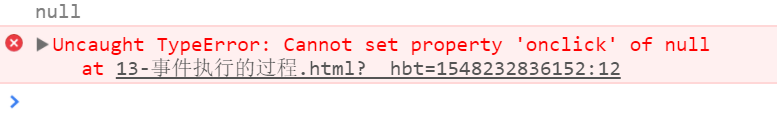


## js报错信息

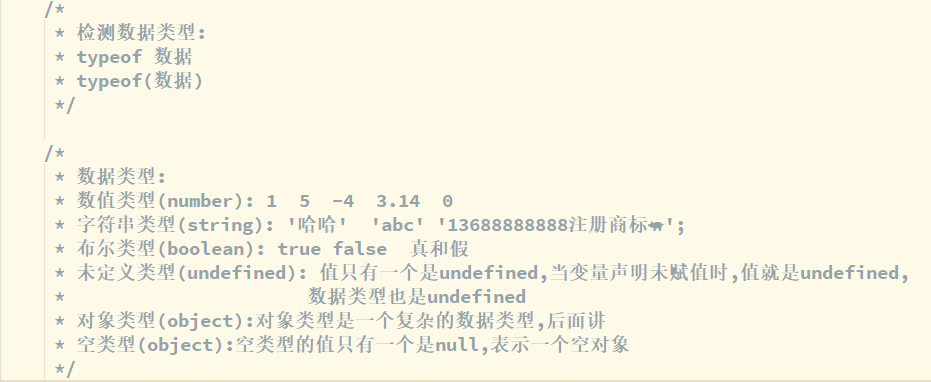
### 5.1 变量未定义



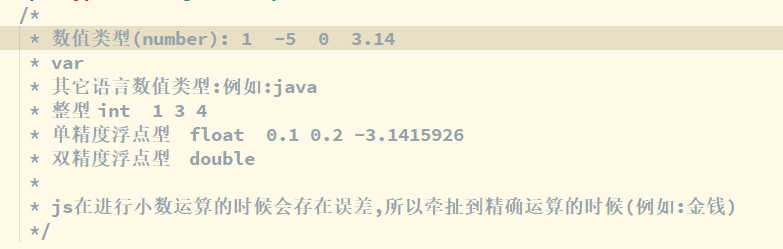
### 5.2 空对象注册事件



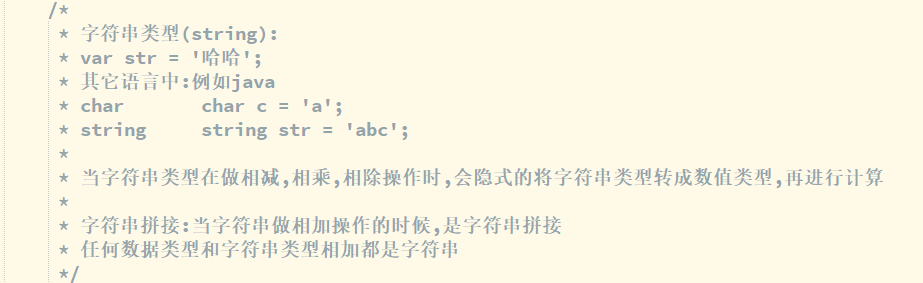
## 6.数据类型

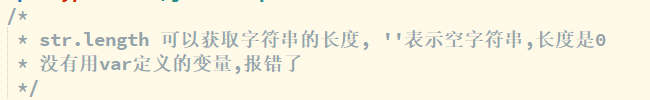


### 6.1 数值类型



### 6.2 字符串类型





### 6.3 数据类型转换



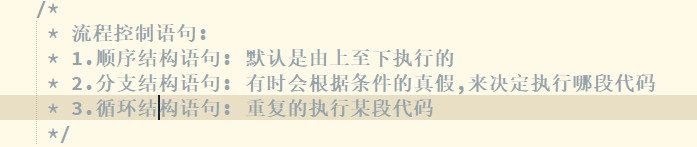




## 7.运算符



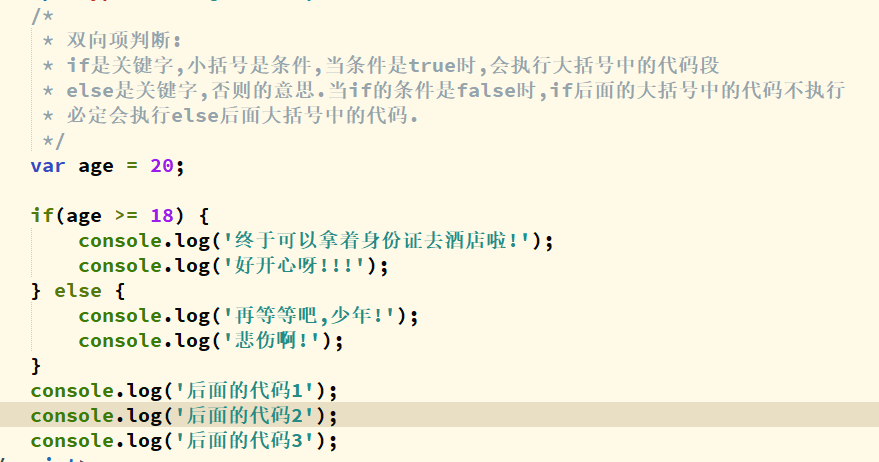
## 流程控制语句

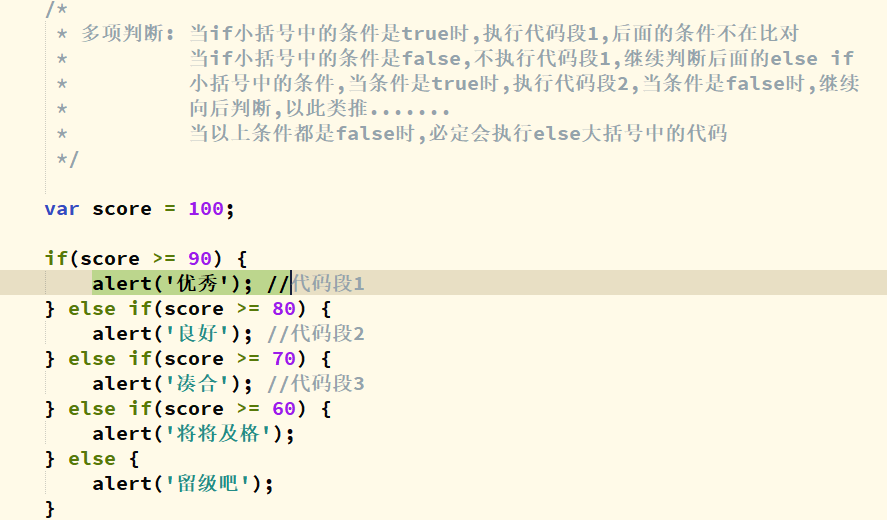


### 分支结构语句

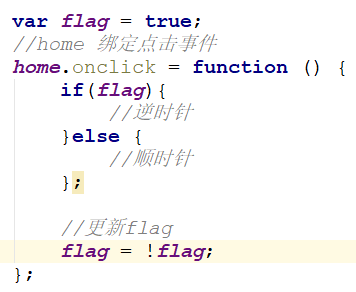
#### 8.1.1 if判断



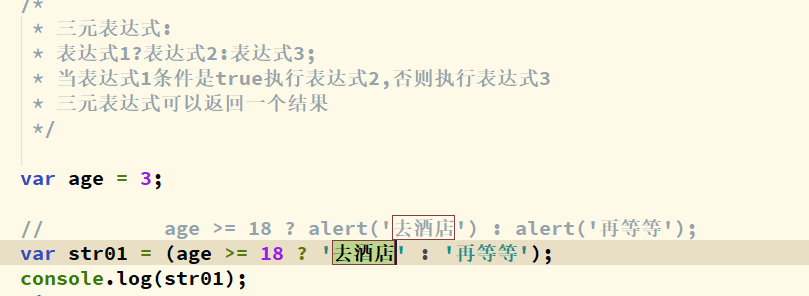




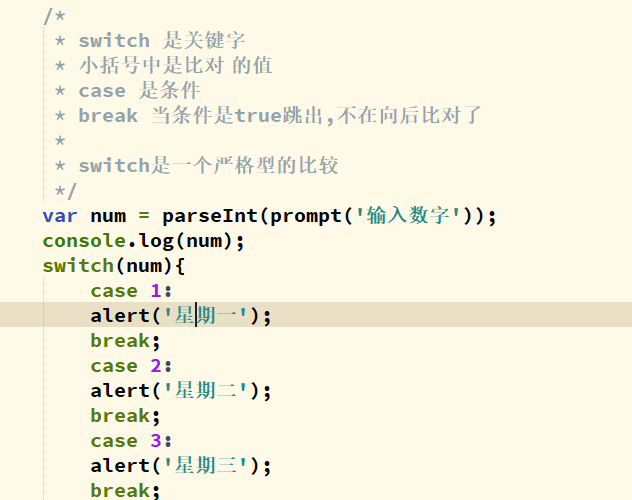
判断条件只有两个时，尽量使用布尔值



#### 8.1.2 三元表达式

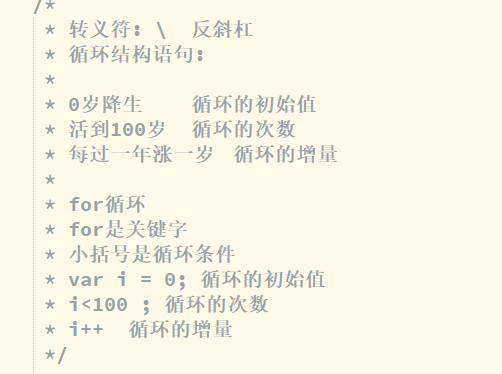


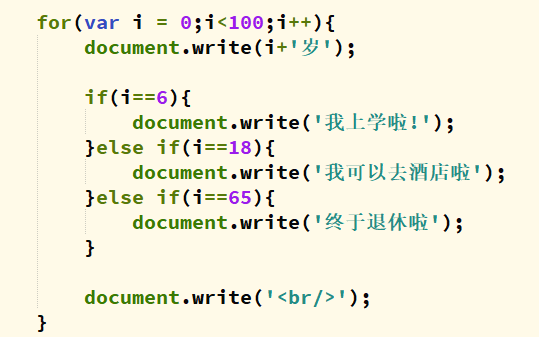
#### 8.1.3 switch-case



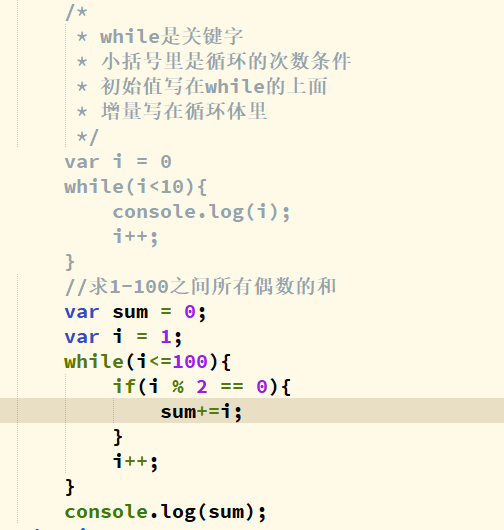
### 循环结构语句

#### for

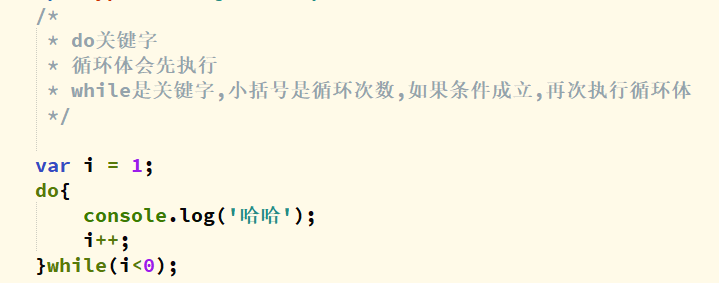




#### while



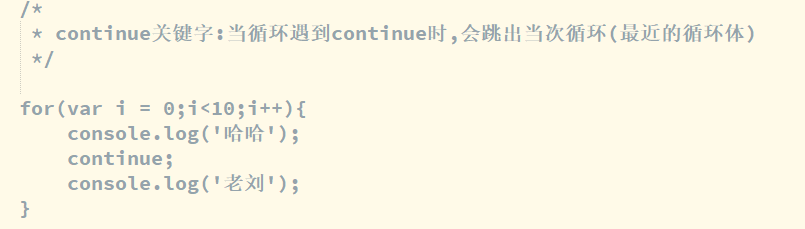
#### do while



#### break关键



#### continue关键字



#### 8.2.6 for ....in



## 输出方式

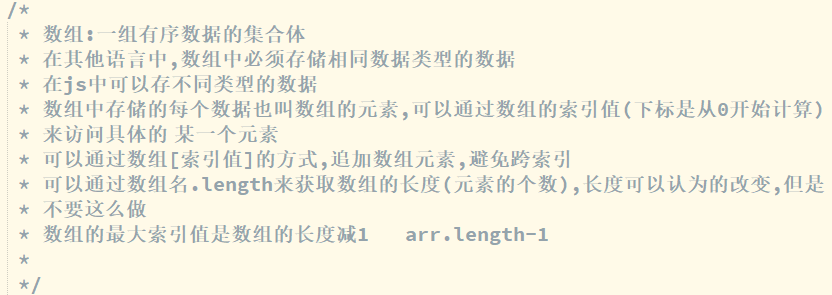
alert() 弹出对话框

prompt() 弹出带输入框的对话框

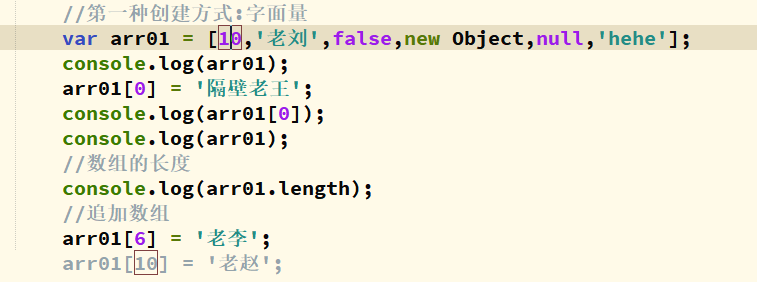
console.log()控制台日志

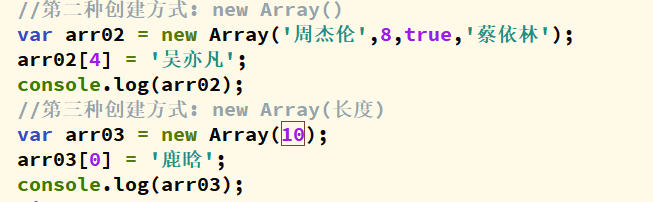
document.write()网页输出

## 数组

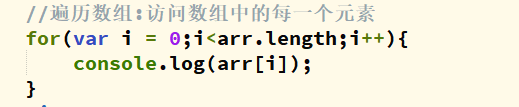


### 定义数组的三种方式

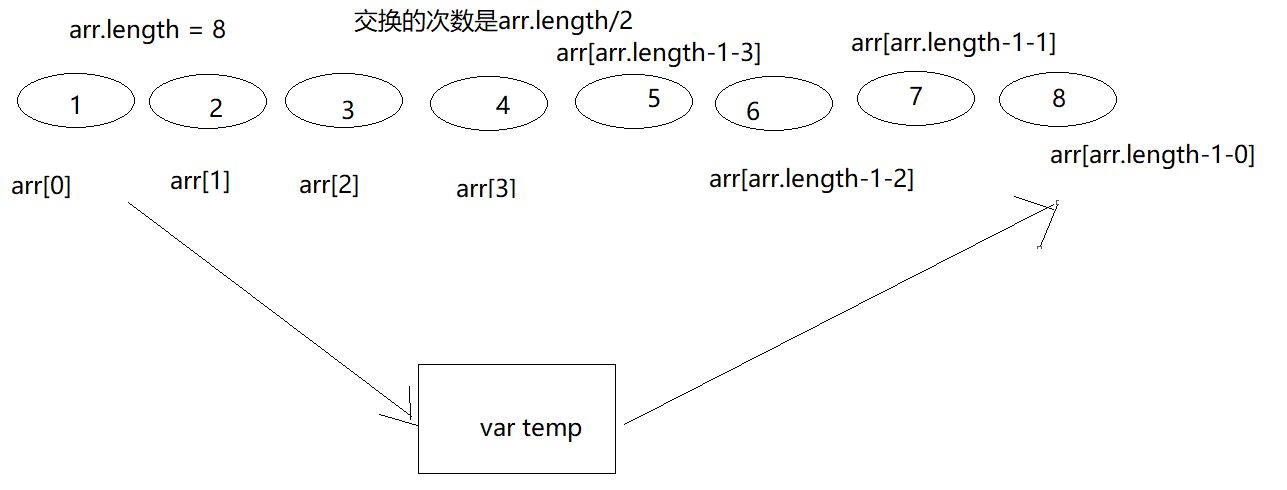


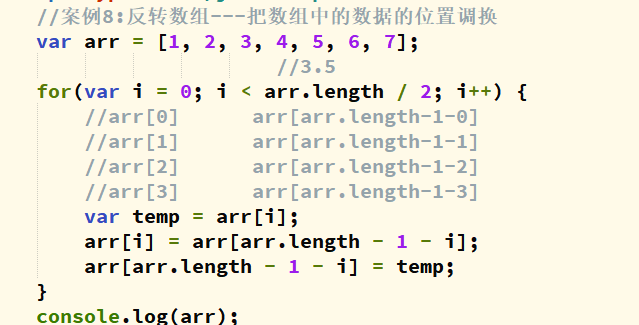


### 遍历数组

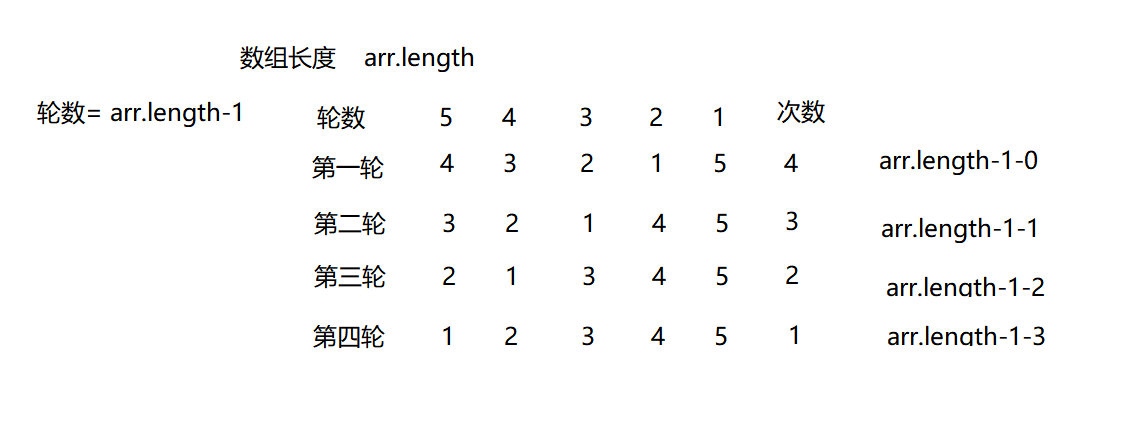


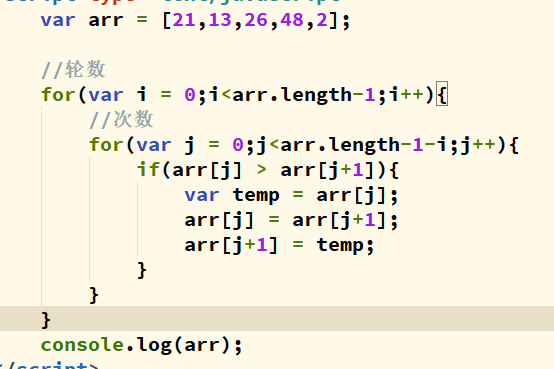
### 反转数组



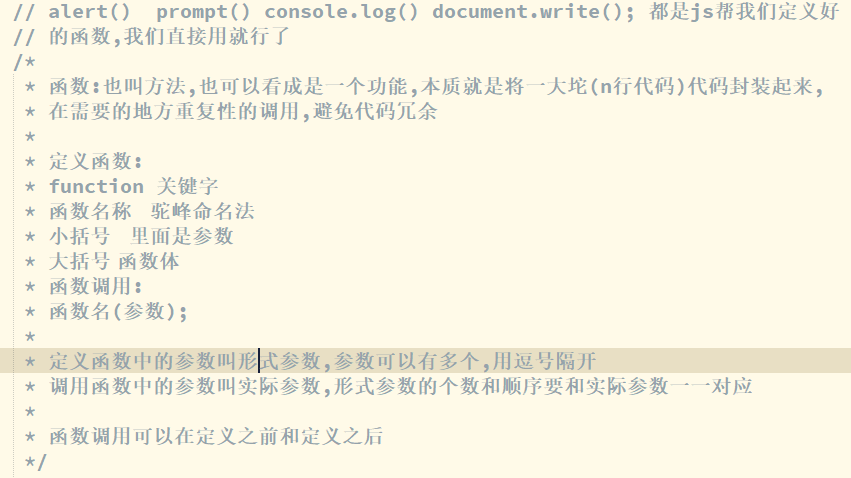


### 冒泡排序

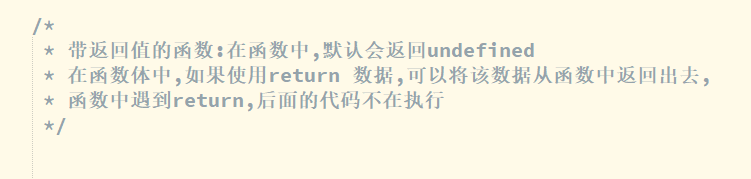


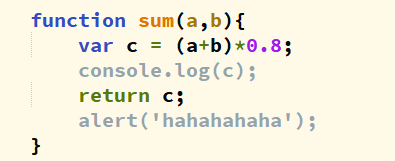


## 函数

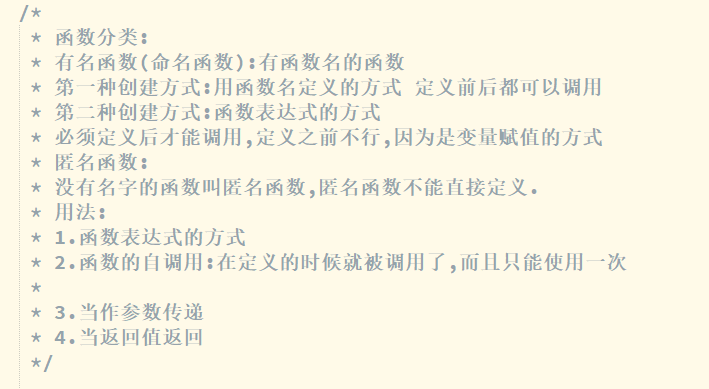


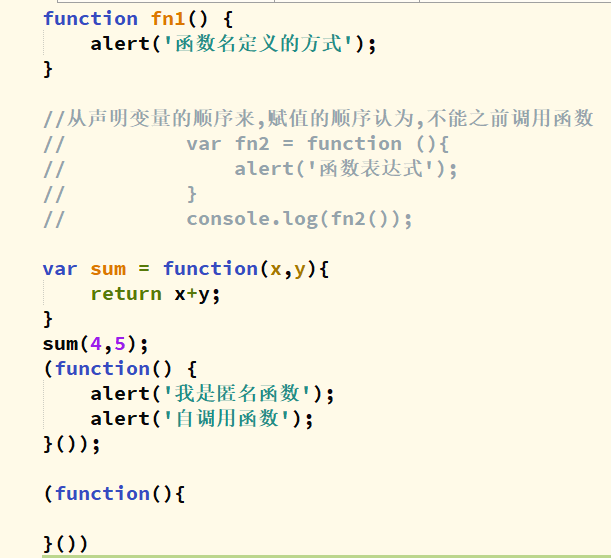
### 带返回值的函数



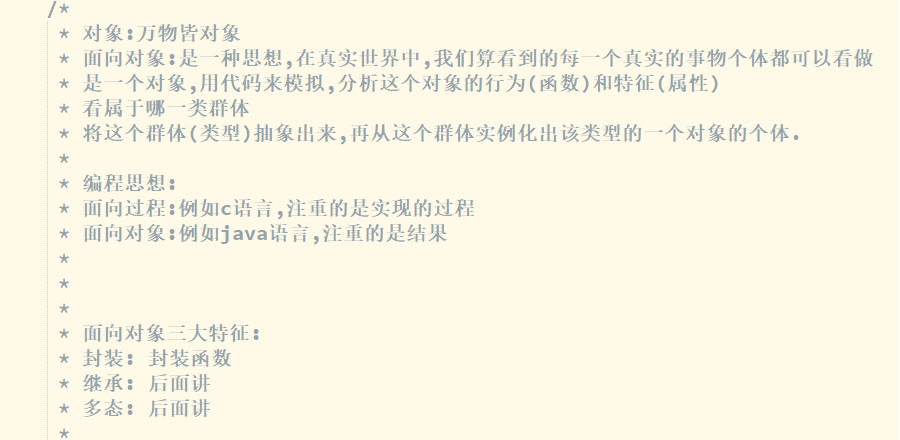


### 函数的几种定义方式





## 对象和面向对象



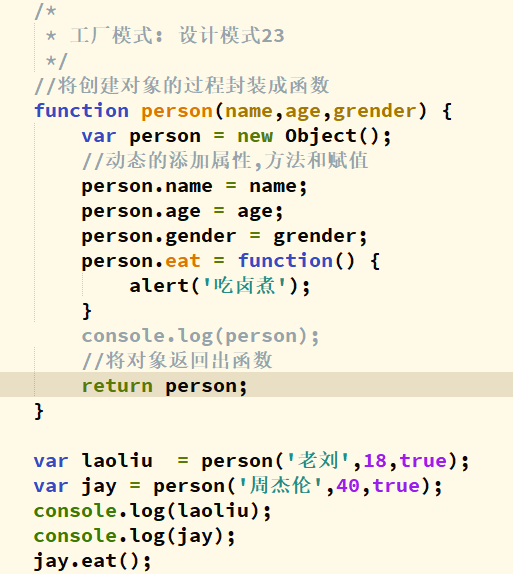


### 创建对象的几种方式

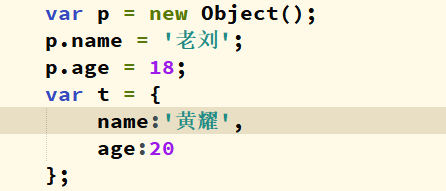
#### new Object



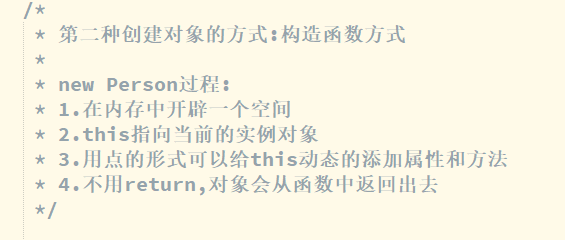
#### 工厂方式

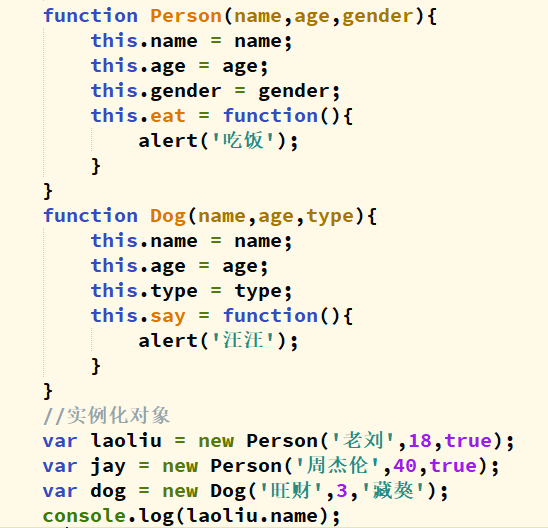


#### 字面量

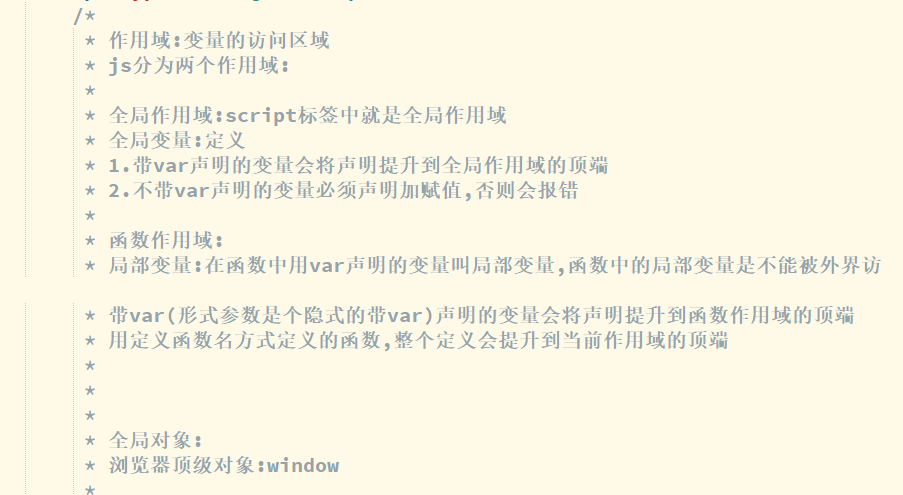


#### 构造函数

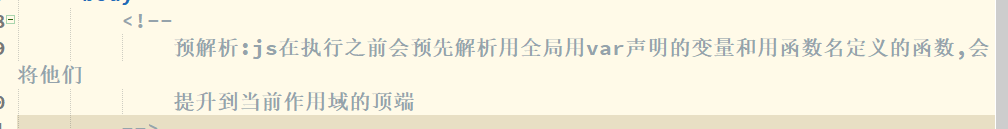




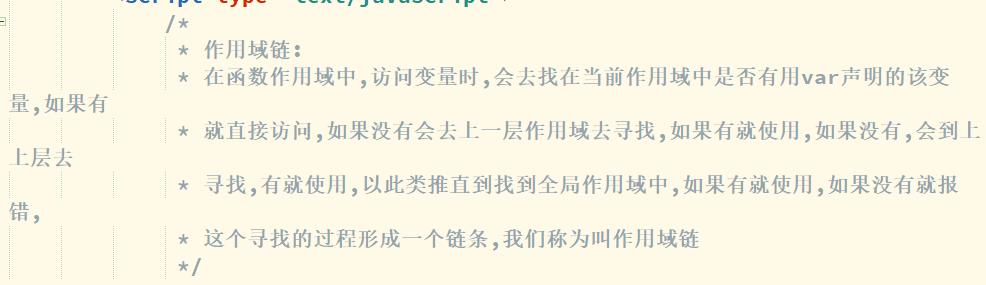
## 作用域

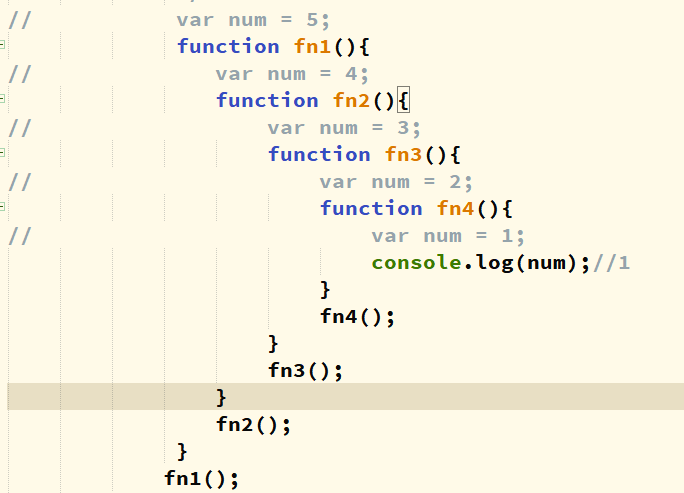


## js预解析

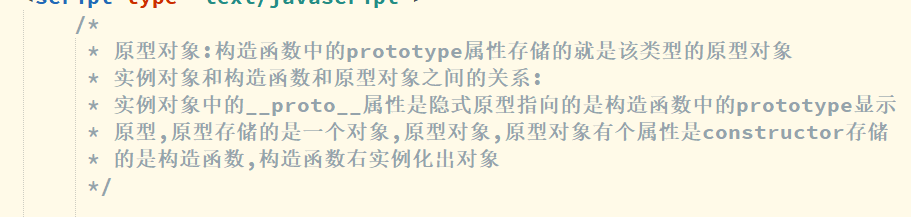


## 作用域链

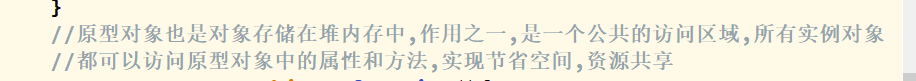




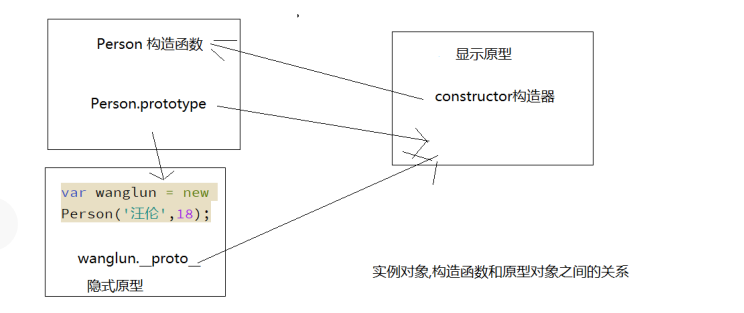
## 原型和原型对象



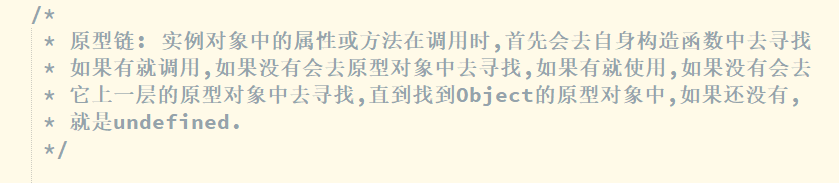
### 原型对象的作用之一



### 16.2原型对象和构造函数和实例对象的关系



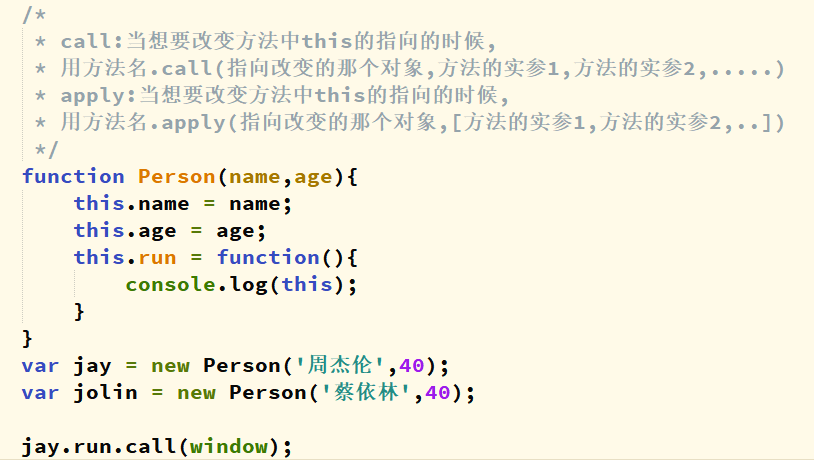
### 16.3原型链



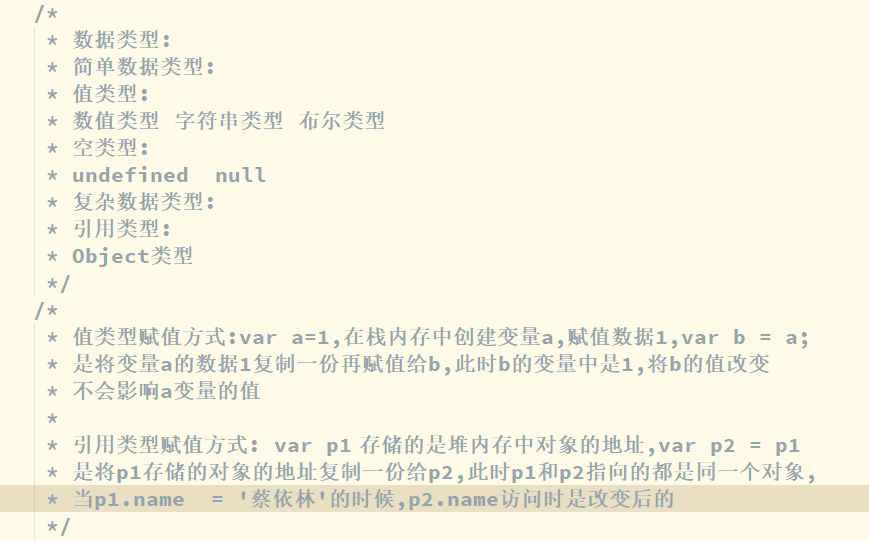
## this的指向

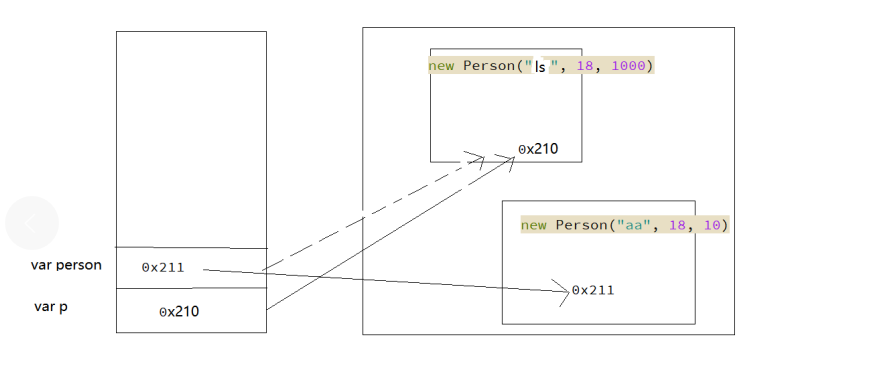


### call和apply函数

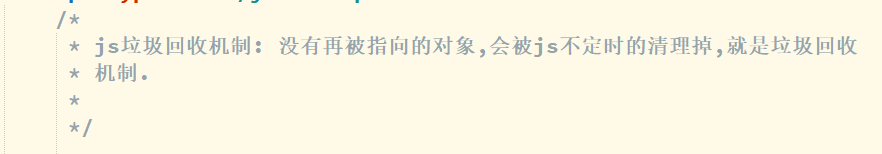


## 值类型和引用类型的传递

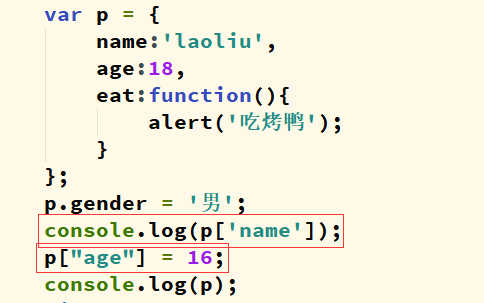




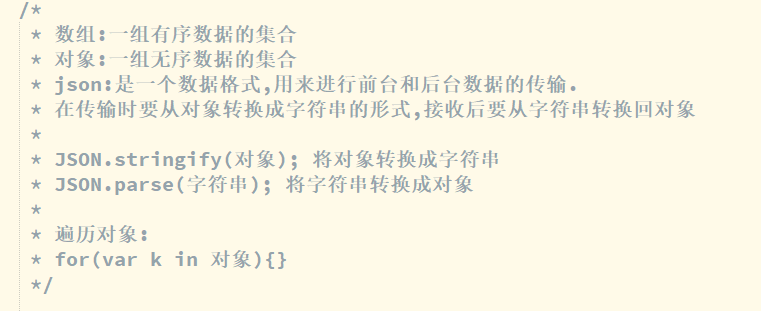
## js垃圾回收机制



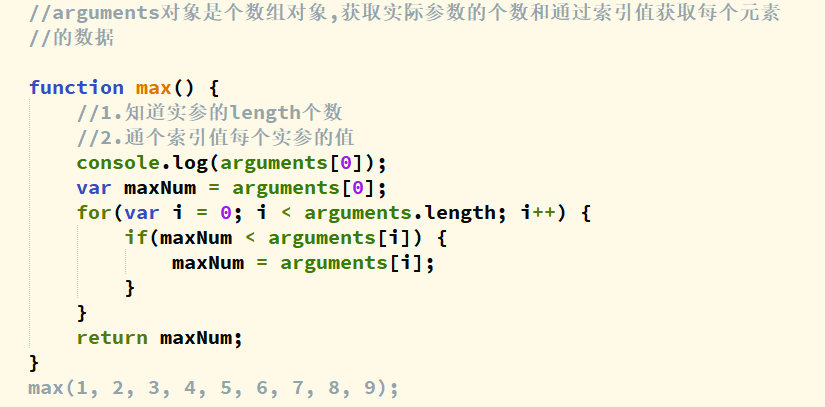
## 另一种访问对象的方式



## json对象



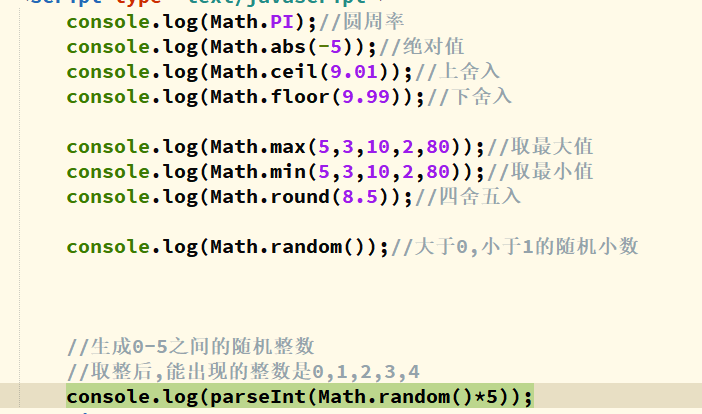
## arguments对象



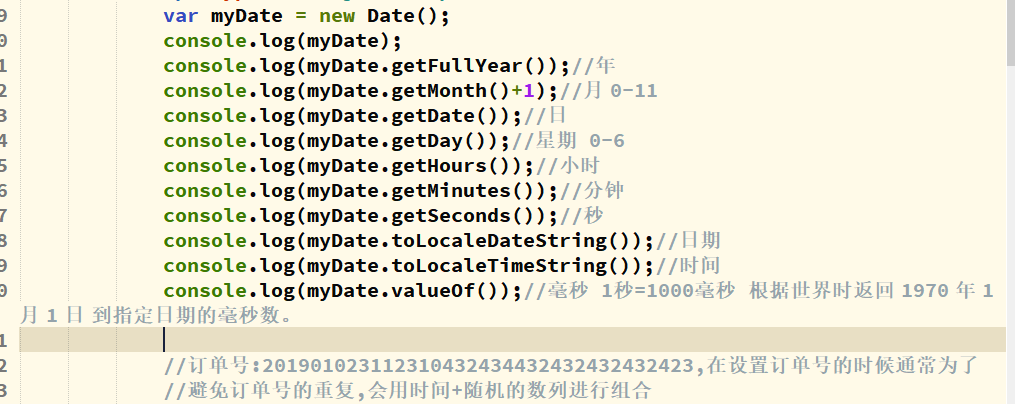
## js内置对象

### Math对象

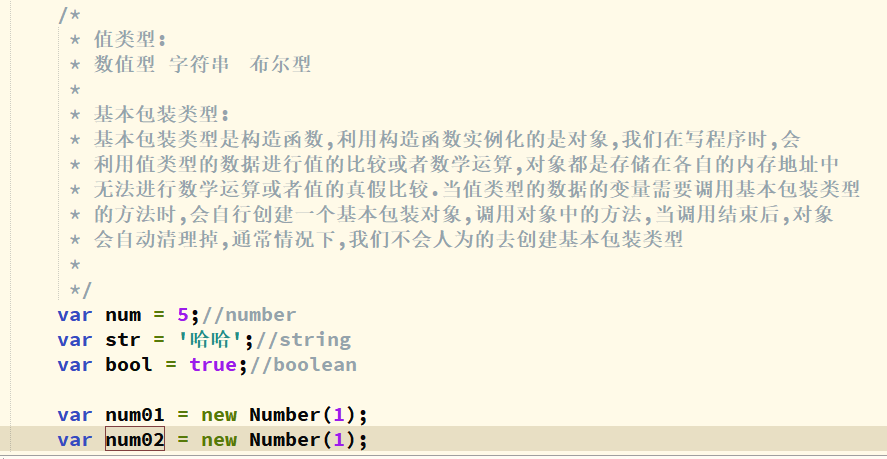
一个数字n的余数范围是0到n-1，在项目中求范围往取余上靠



### Date对象

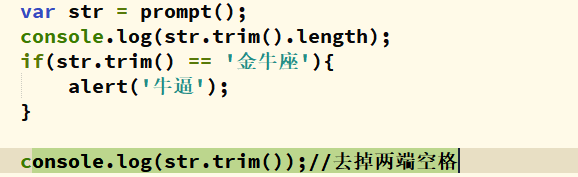


### 23.3基本包装类型

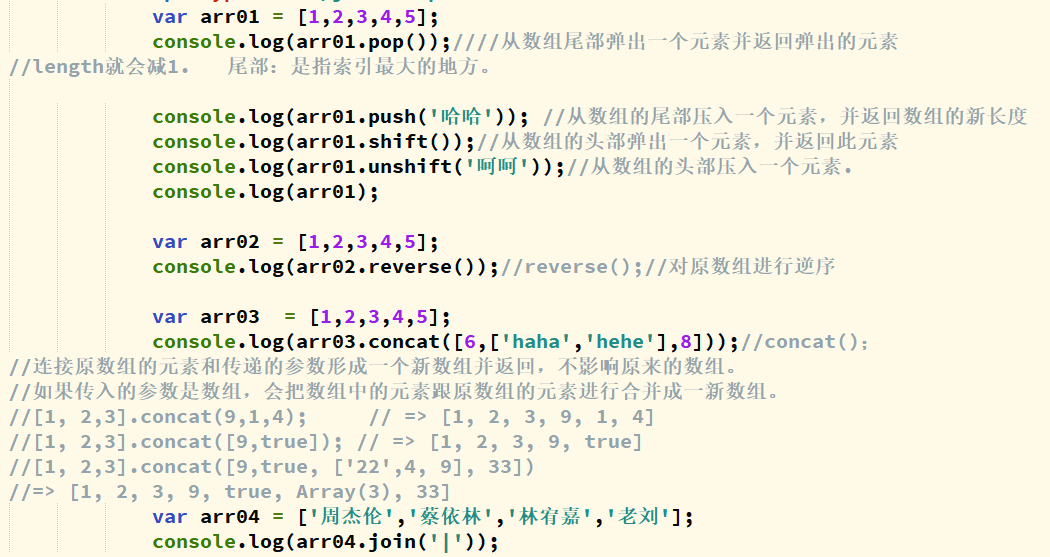


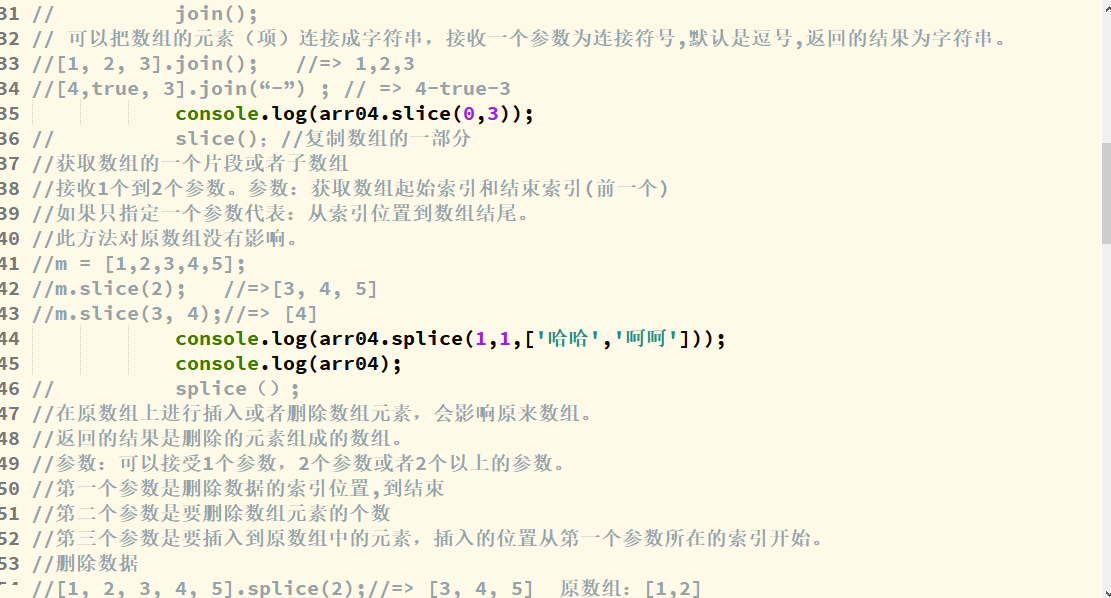
### 23.4 字符串常用方法

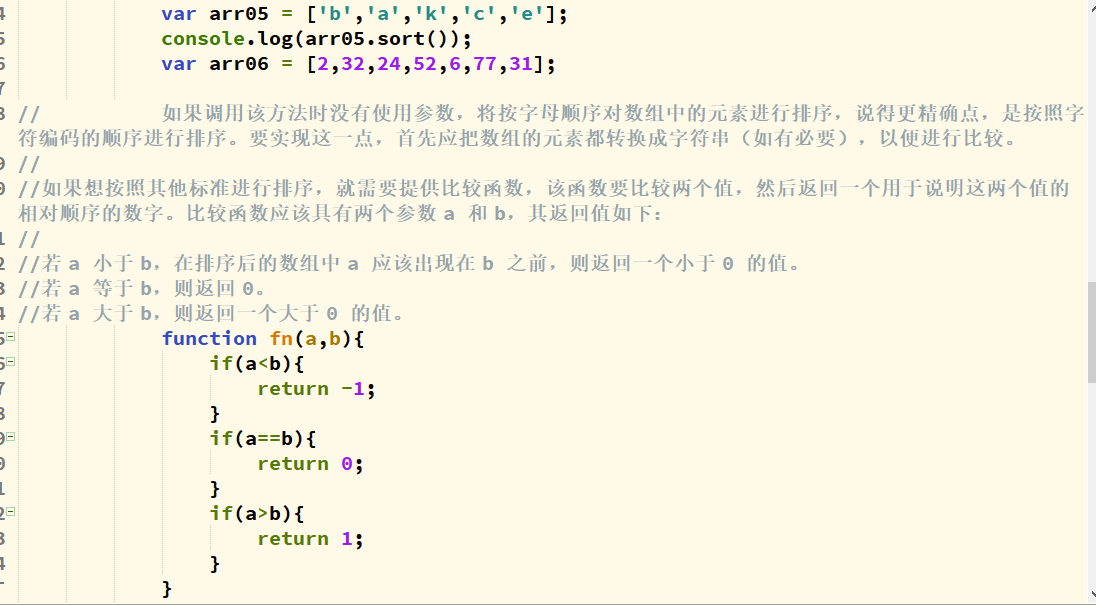


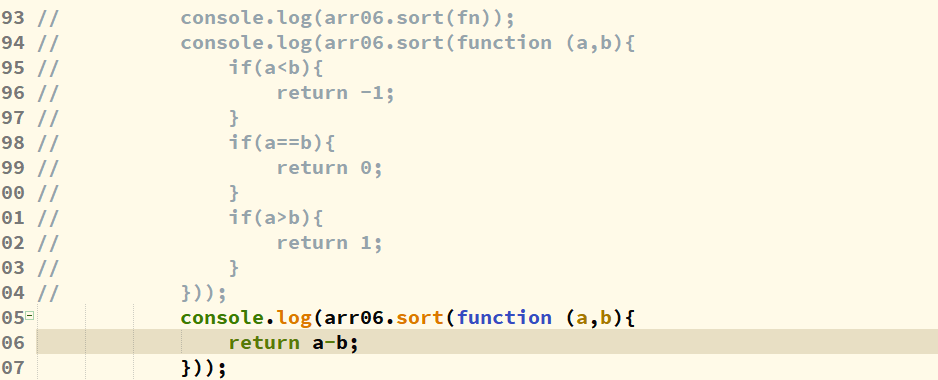


### 23.5 数组常用方法

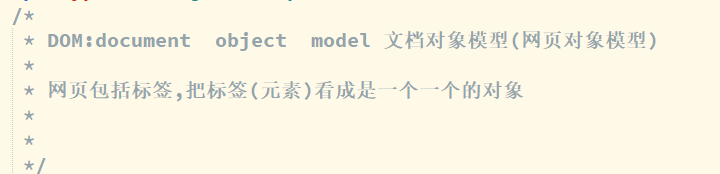






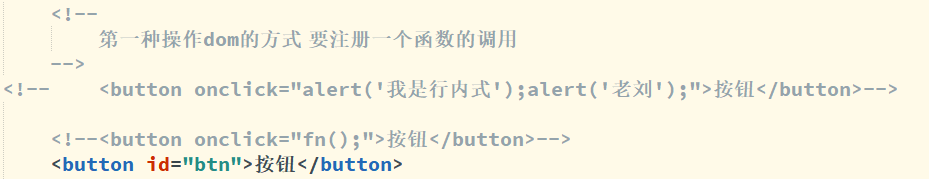


## Dom简介



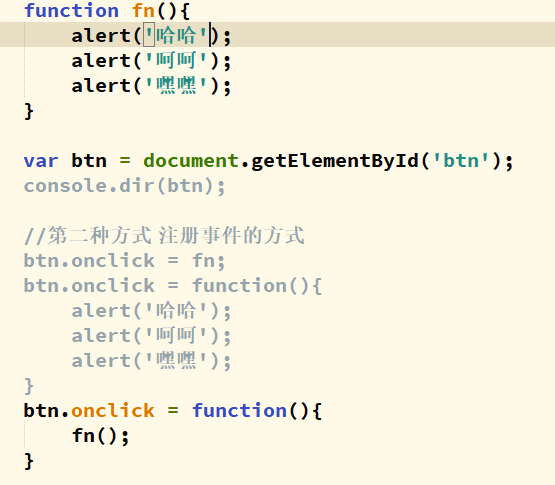
## 事件

### 25.1 第一种方式



### 25.2 第二种方式

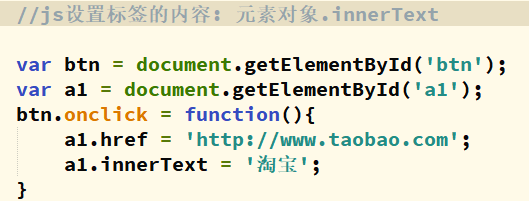




## js设置html属性



## js设置html内容



## Dom获取方式

### 通过id名获取

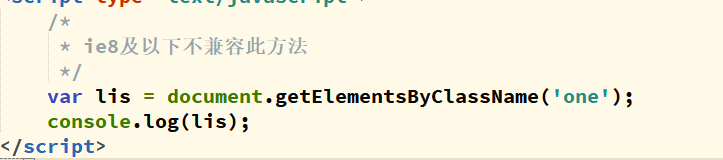
document.getElementById(‘id名’) 返回值是对象

### 通过标签名获取

docment.getElementsByTagName 返回对象数组

document是网页对象,可以是一个局部的对象,可以某个元素的后代

### 通过类名获取



### 28.4通过css选择器获取

document.querySelector('#cokeBg')

括号里面是css选择器，获取一个dom选择器，返回值是一个数组

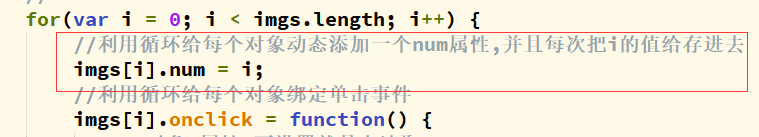
document.querySelectorAll('#cokeBg img')

括号里面是css选择器，获取一组dom选择器，返回值是一个数组

## 事件处理函数中的this的指向

this指向事件源对象本身

## 利用循环给元素动态添加属性



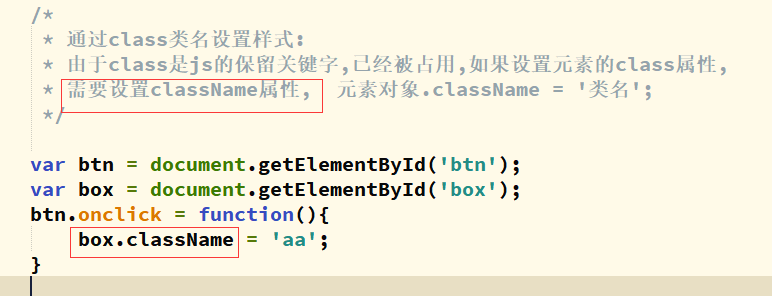
## 排它操作



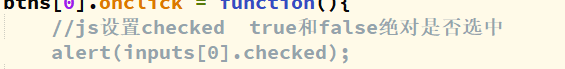
## 设置css样式



## class类名的设置

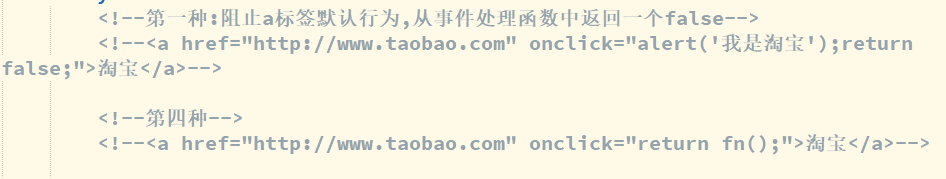


## 表单的选中和读取的js设置值



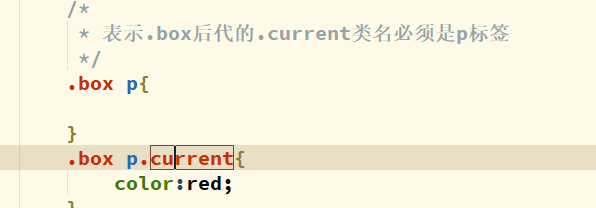
checked selected disable

## 阻止a链接跳转





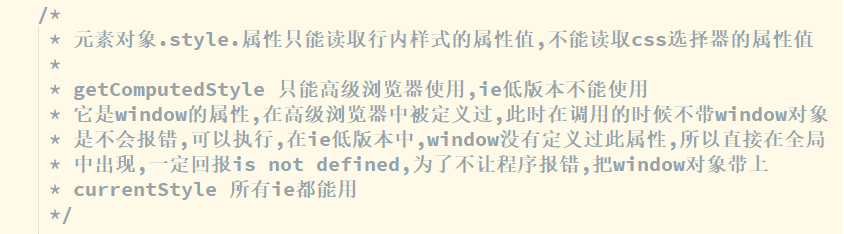
## 指定标签选择器



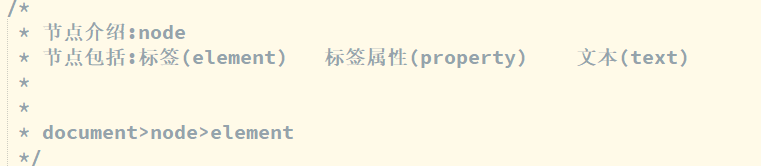
## innerText和textContent



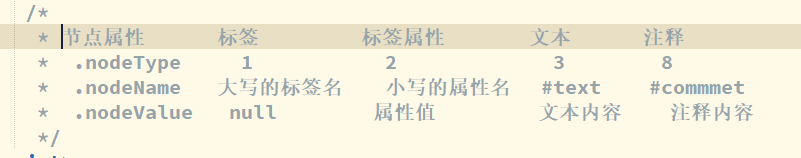
## GetComputedStyle和currentStyle 区别



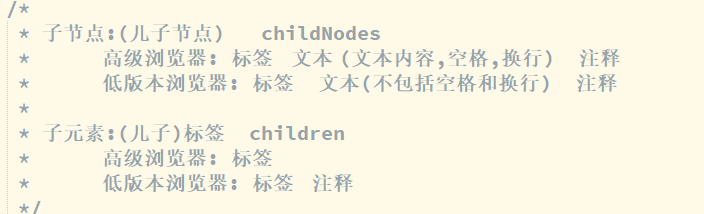
## 节点



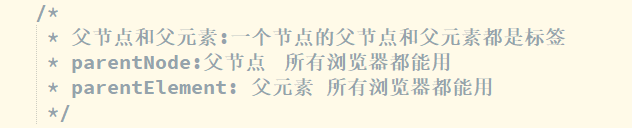
### 节点属性



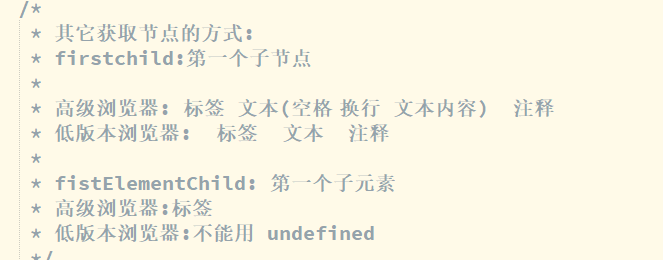
### 子节点和子元素

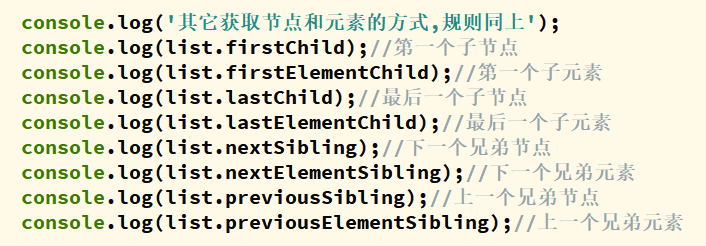


### 父节点和父元素



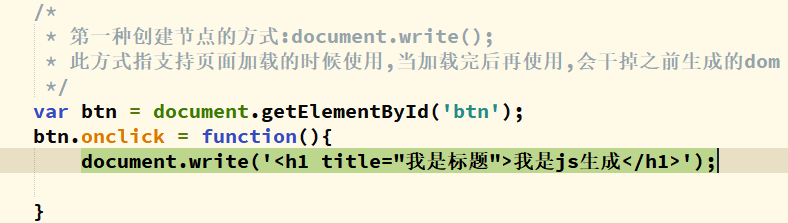
### 其它获取节点和元素的方式



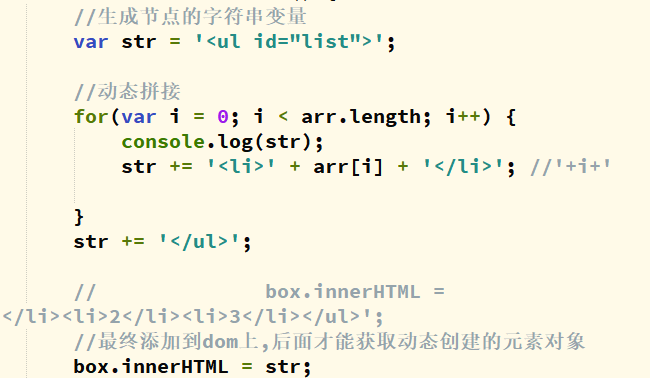


### 创建节点的三种方式

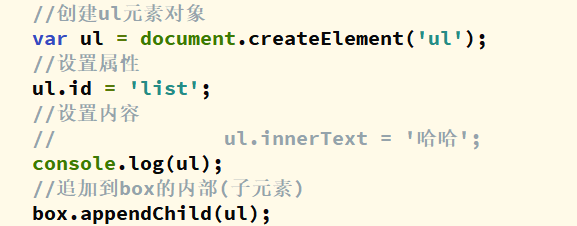
#### 第一种方式



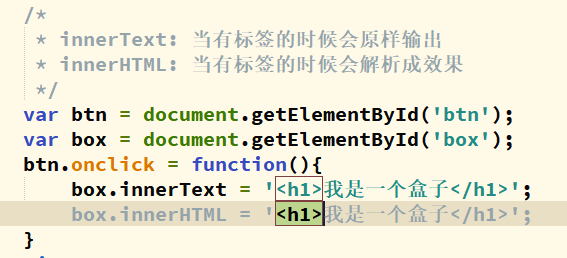
#### 第二种方式



#### 第三种方式



#### innerText和innerHTML区别



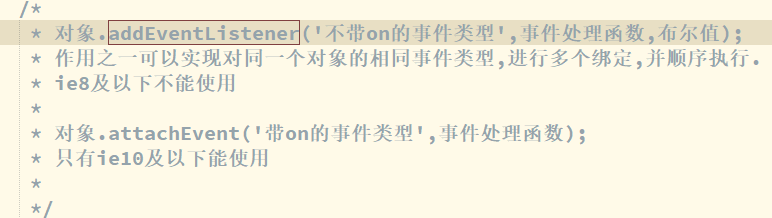
### 节点操作常用方法



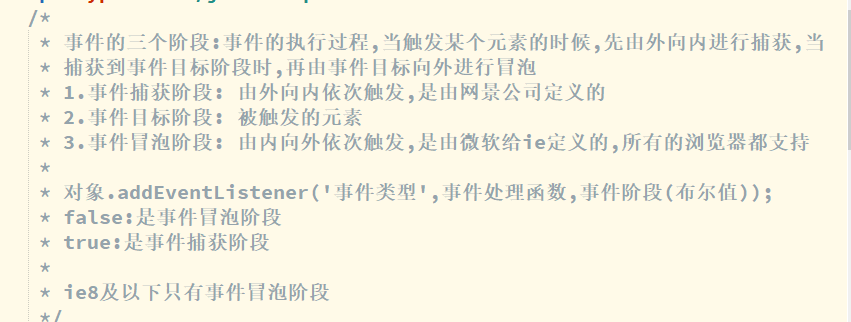
### 克隆节点



## 另一种绑定事件的方式



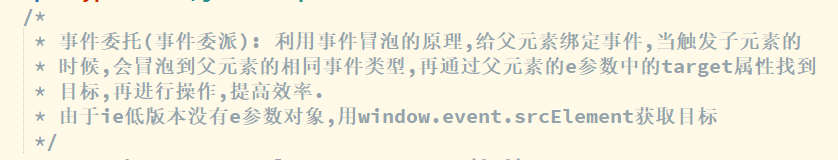
## 事件的三个阶段



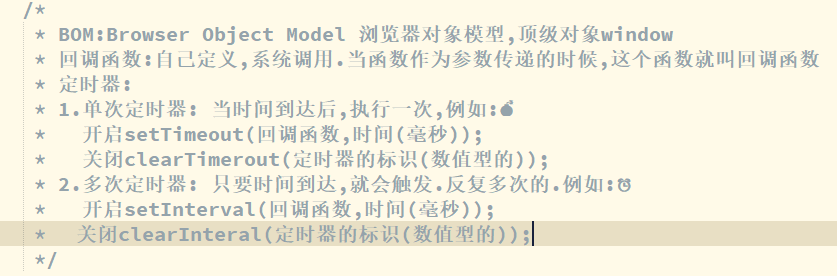
### 事件冒泡



### 事件委派

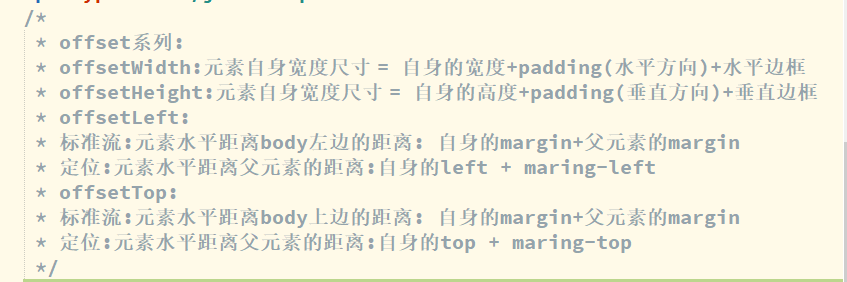


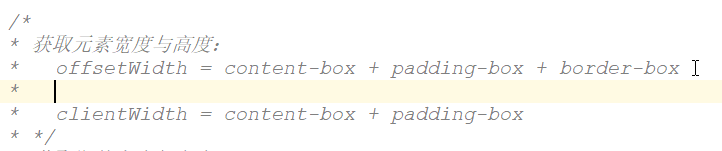
## 定时器





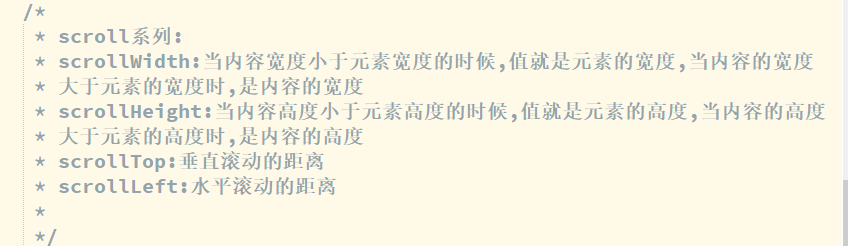
## offset系列



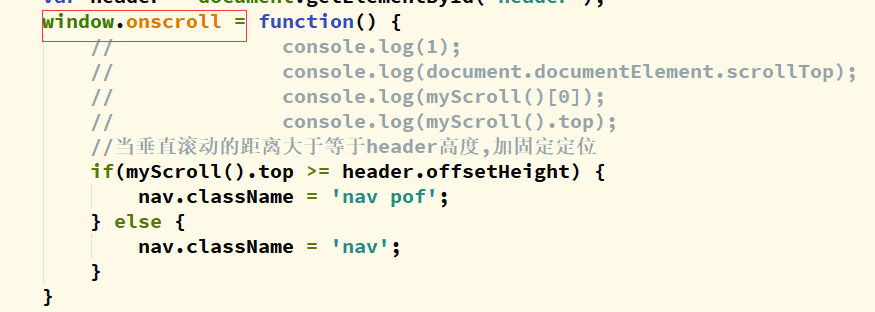


设置父元素宽高与子元素相同，通常在js内设置。

## scroll系列



### 滚动事件



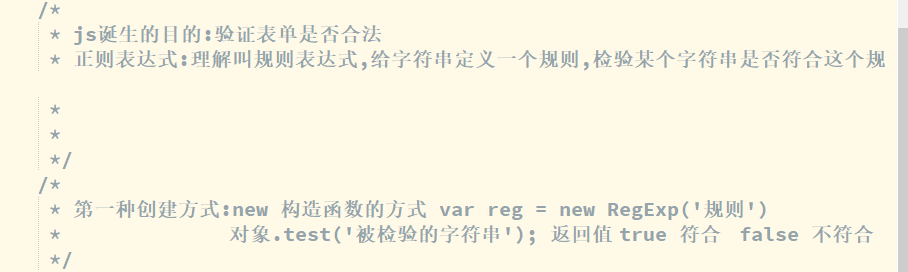


## 断点调试

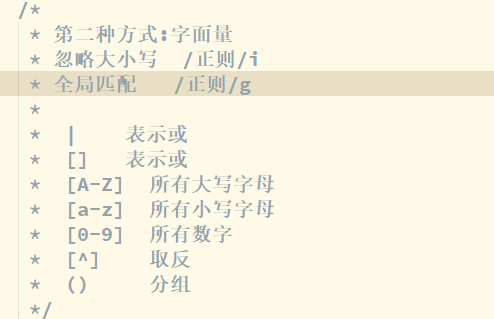


## 正则表达式

### 第一种

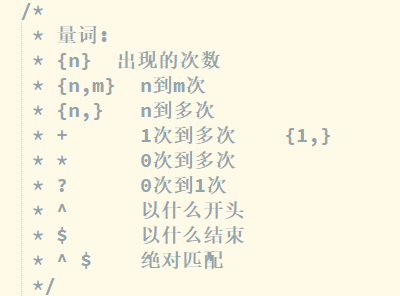


### 第二种

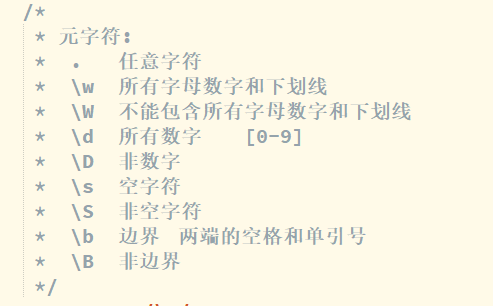


### 表达式

### 量词



### 元字符



### 常用方法

