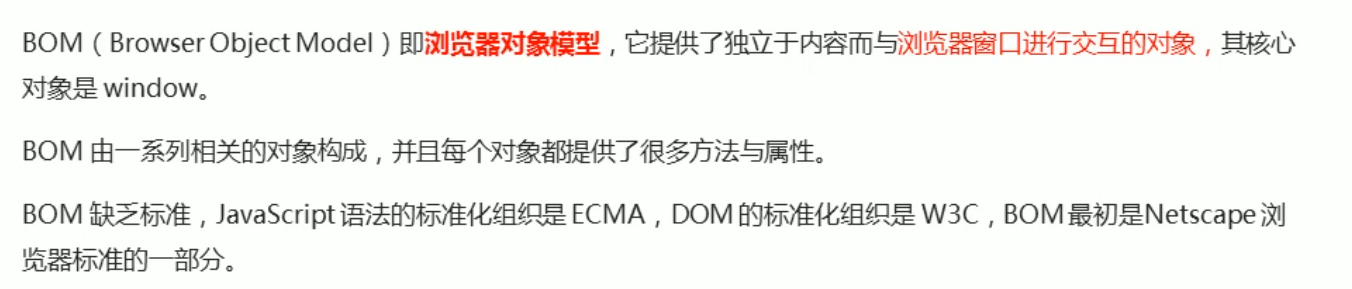
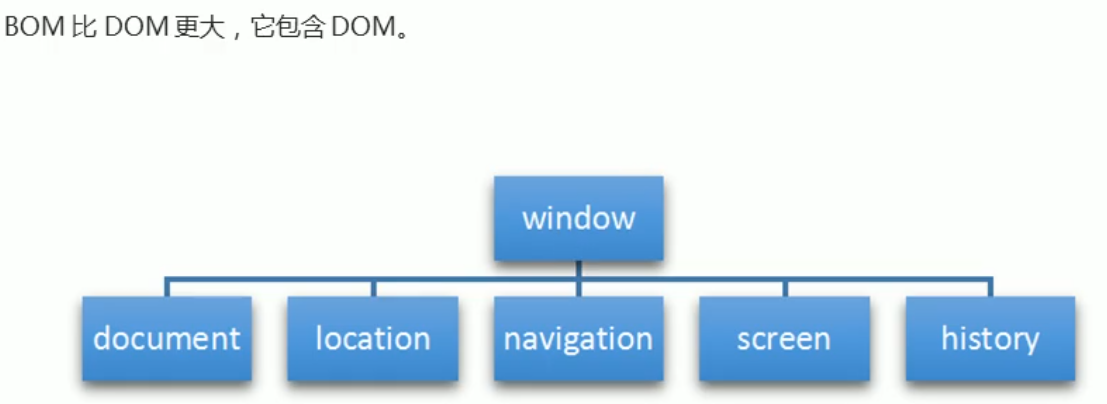
### BOM概述

#### 什么是BOM

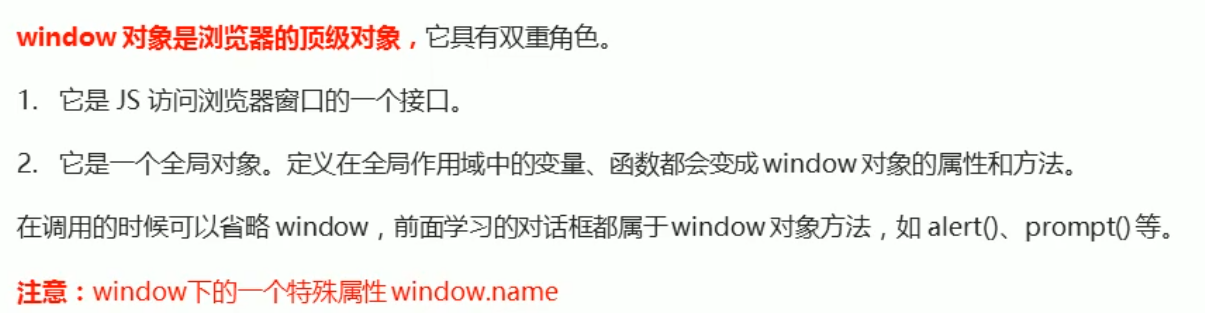




#### BOM的构成

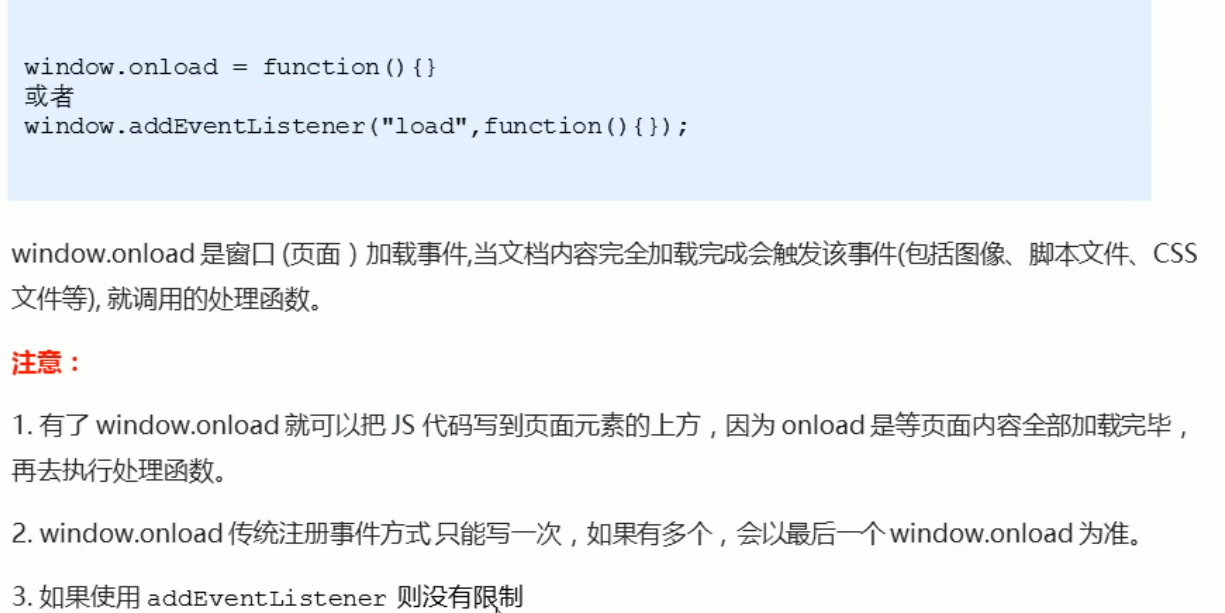


DOM对象



### WINDOW对象的常见事件

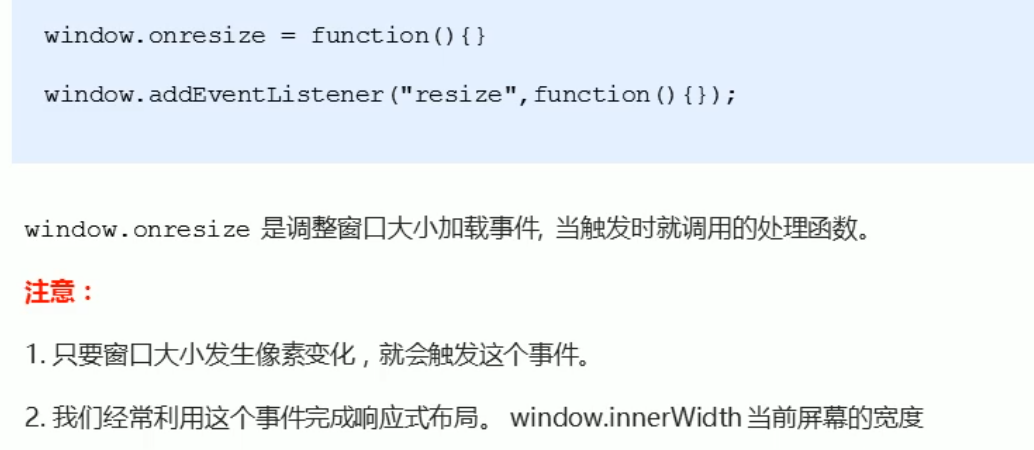
#### 窗口加载事件



以前书写代码都是将JS代码写在HTML标签下方，如果写在前方，由于文档执行由从上往下的顺序，JS比HTML标签更早生成，这样JS就获取不到文档对象模型。解决这个方法就是将JS代码写在窗口加载事件中。

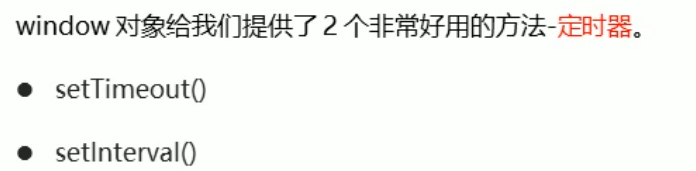


#### 调整窗口大小事件



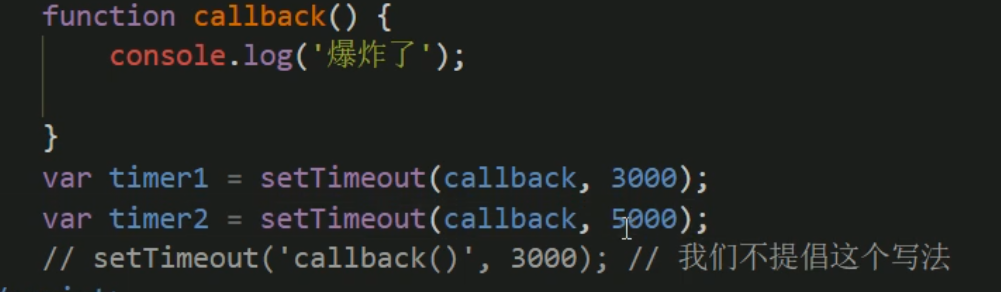
### WINDOW对象的常见方法

#### 定时器

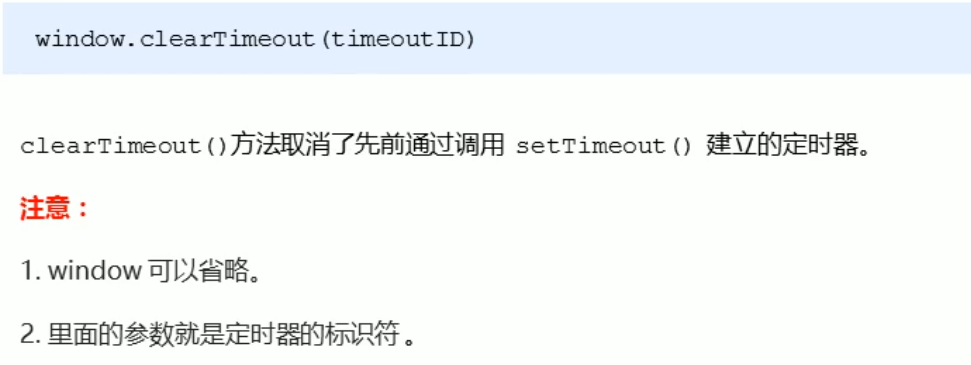


###### setTimeout()



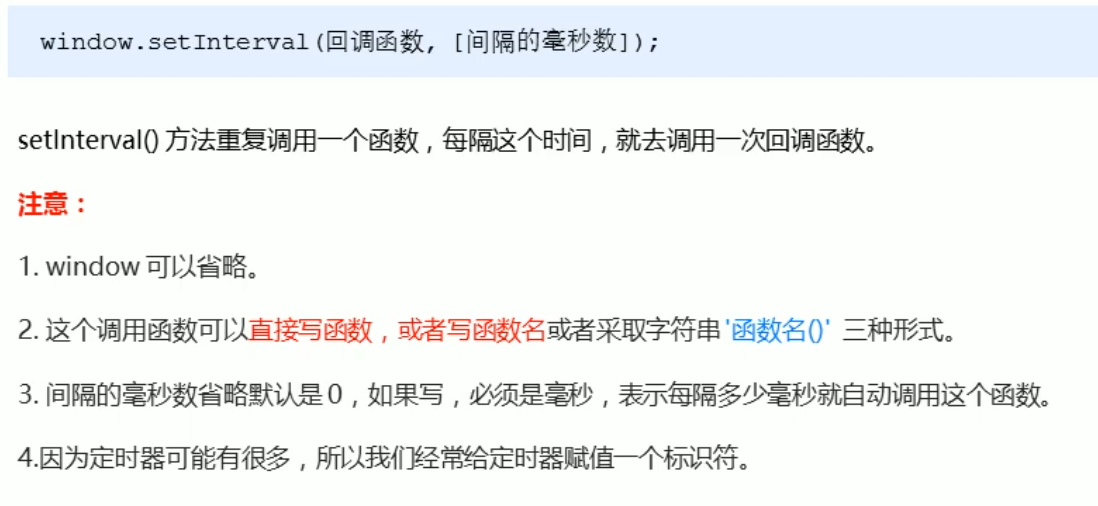


###### setTimeout()停止方法



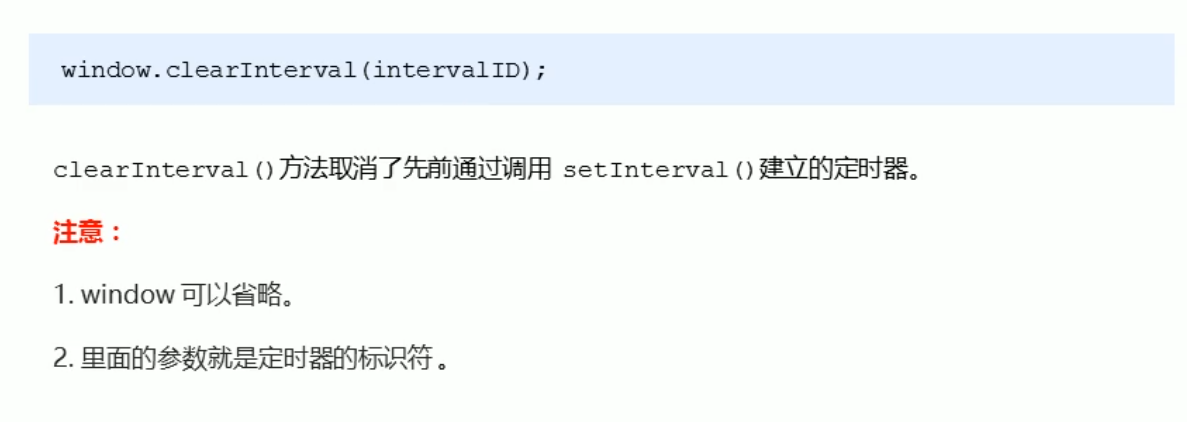


###### setInterval()



与setTimeout不同的就是，setTimeout只能调用一次，setInterval能重复调用。也就是说，setInerval每隔间隔的时间都会调用回调函数。

###### setIneterval()停止方法

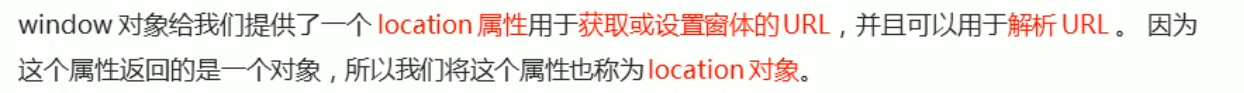


#### THIS指向问题

指向调用该函数/方法的对象。要注意定时器内部的this都是指向window。

### location对象

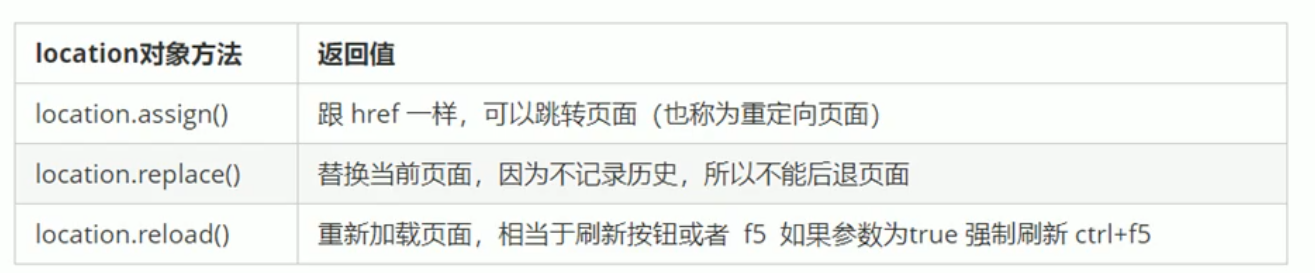
#### 什么是location对象



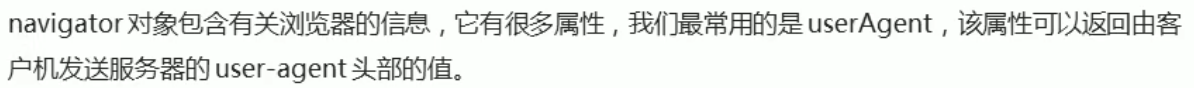
#### Location对象的属性



#### Location对象的方法

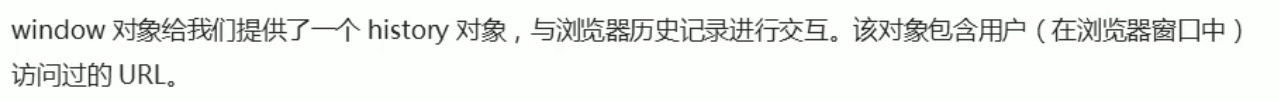


### Navigator对象





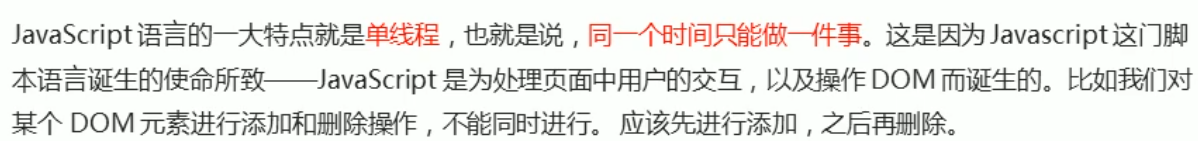
### History对象





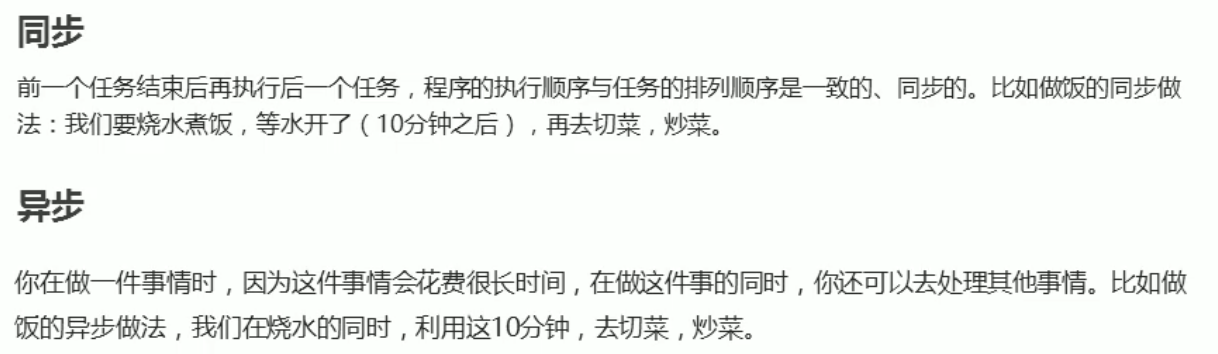
### JS执行机制

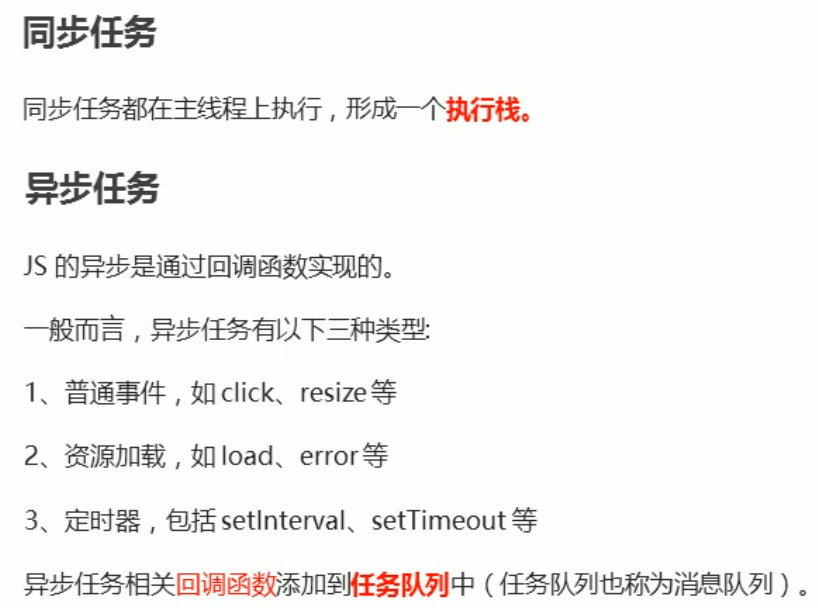
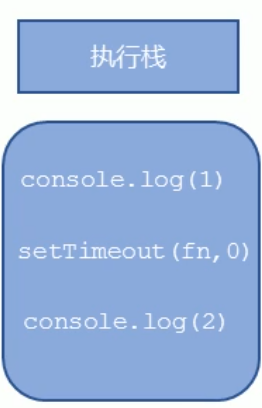
#### JS是单线程

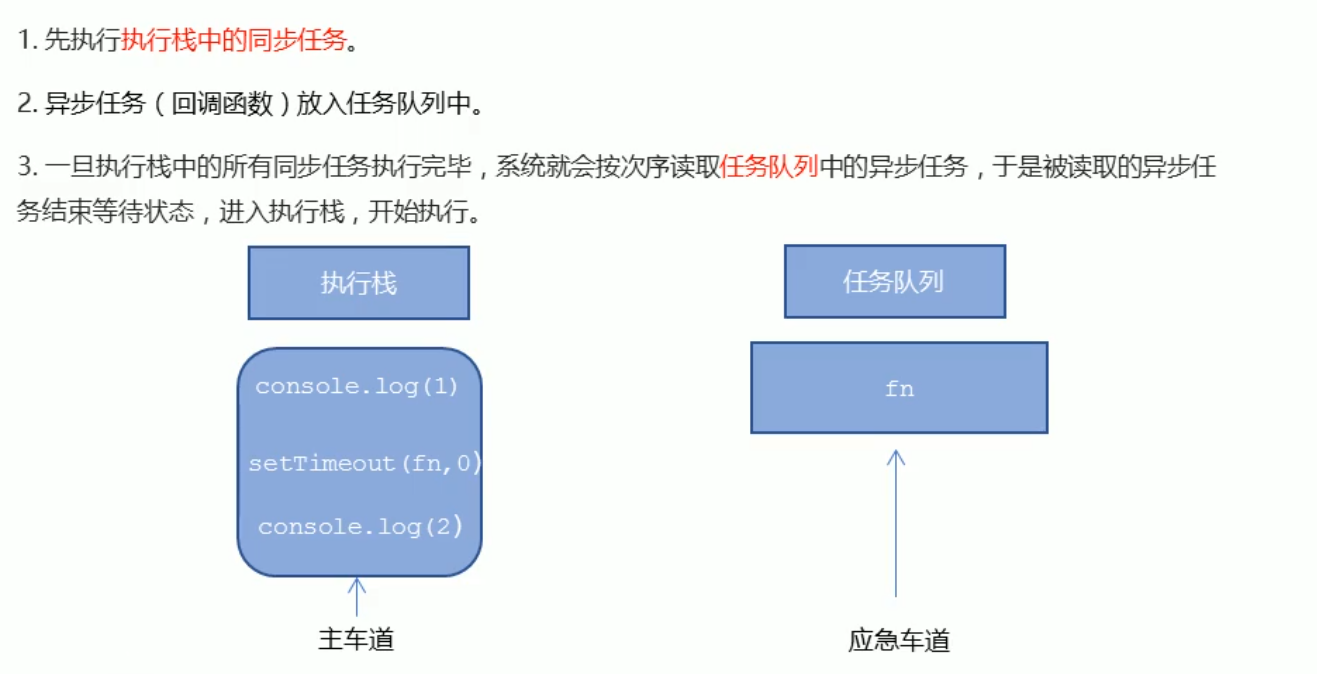


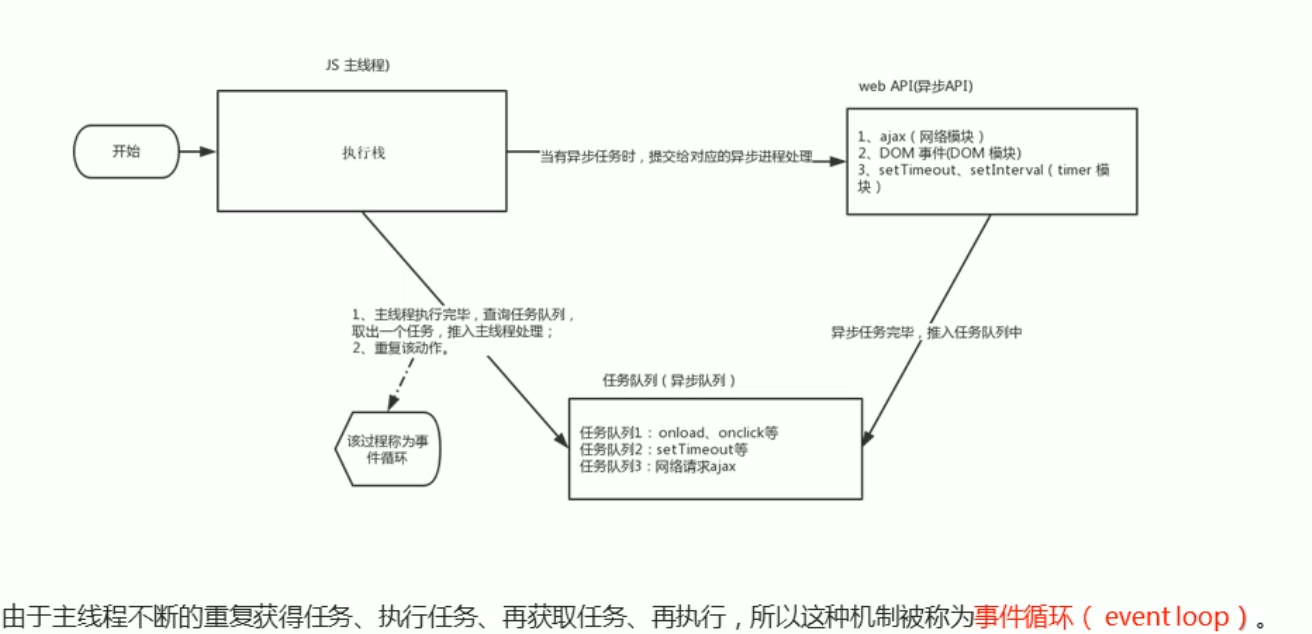
#### JS的同步异步

为了解决JS单线程造成的如果某线程堵塞，则后面的线程都会堵塞的情况，JS中出现了同步和异步。

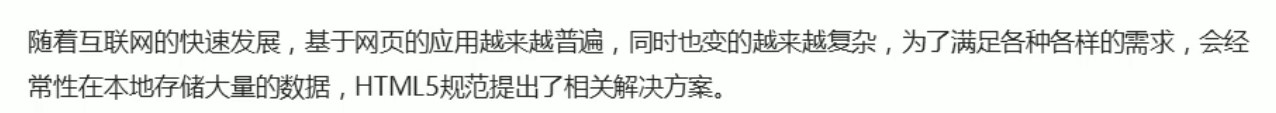




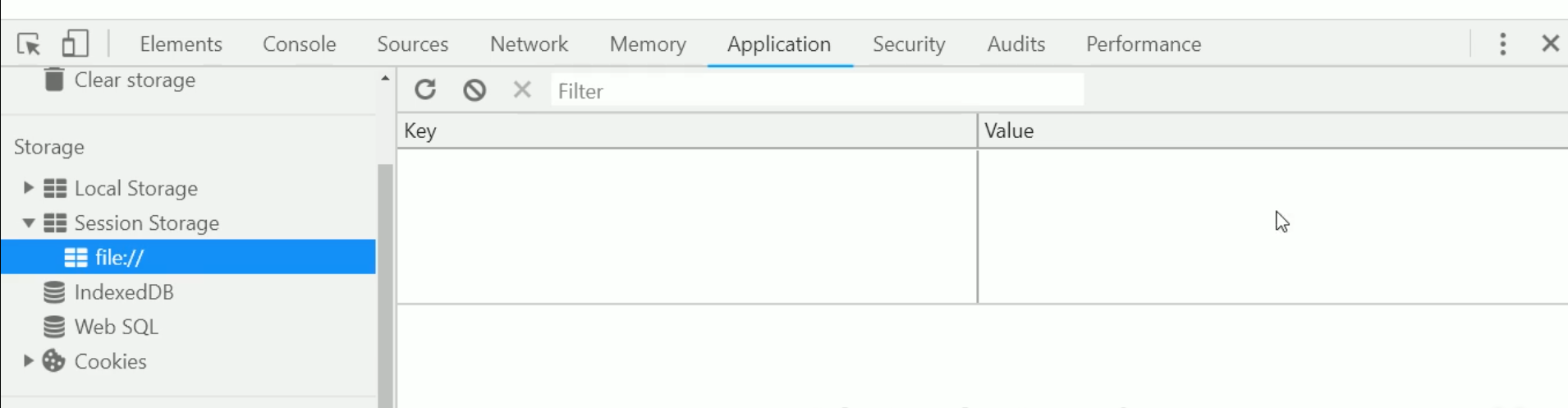


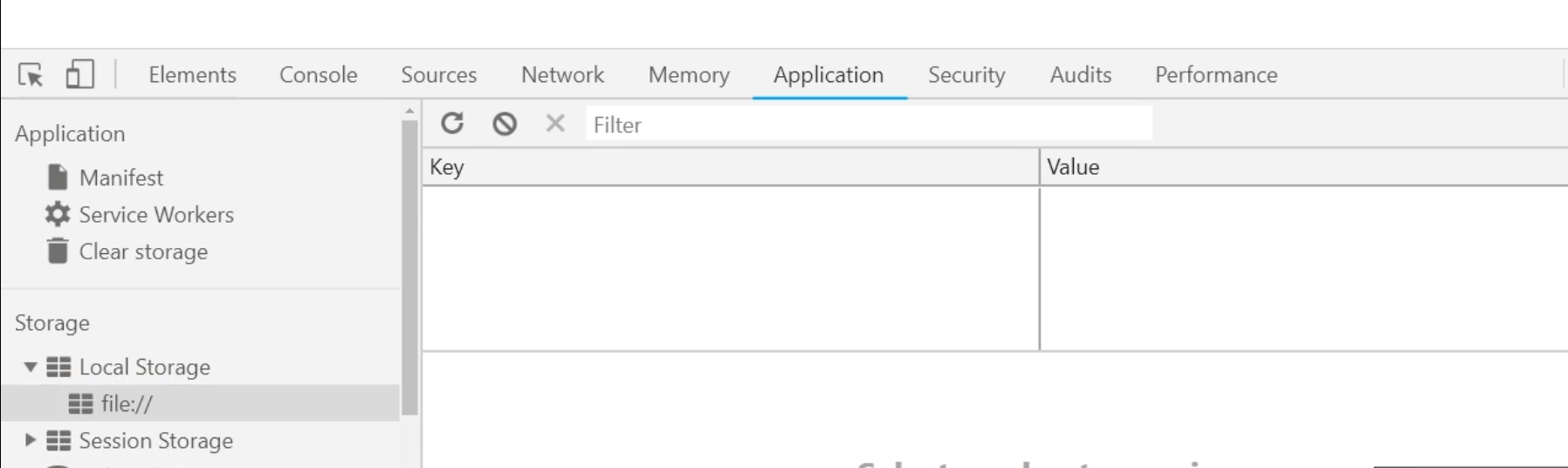
### JS的本地存储



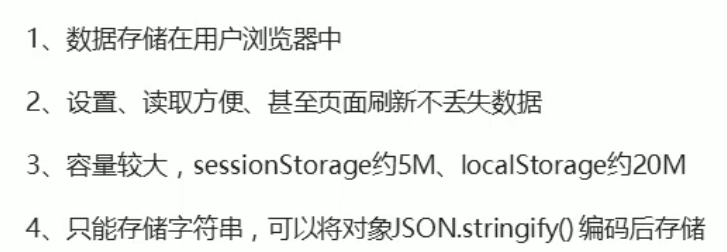
本地存储两种方法：window.sessionStorage与window.localStorage。

存储内容查看：

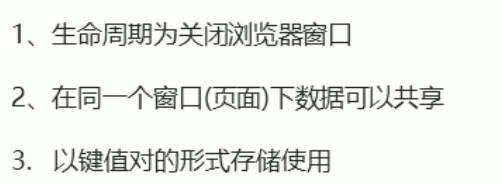


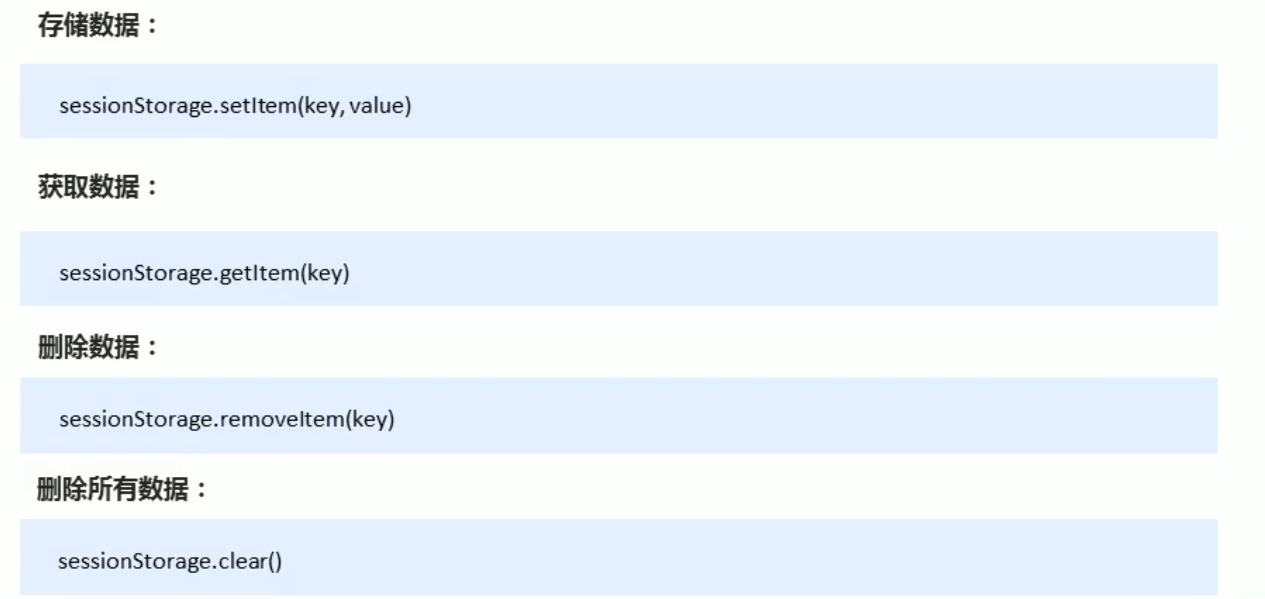


#### 本地存储的特性



#### window.sessionStorage









#### window.localStorage

