**1.**[Web Worker线程的限制是什么？](http://blog.jobbole.com/78346/" \l "q23" \t "http://blog.csdn.net/u013418331/article/details/_blank)

Web worker线程不能修改HTML元素，全局变量和Window.Location一类的窗口属性。你可以自由使用Javascript数据类型，XMLHttpRequest调用等。

**2.**[我们如何在JavaScript](http://blog.jobbole.com/78346/" \l "q24" \t "http://blog.csdn.net/u013418331/article/details/_blank)**中创建一个worker线程？**

创建一个worker线程，我们需要通过Javascript文件名创建worker对象  
varworker = newWorker("MyHeavyProcess.js");

我们需要使用“PostMessage”发送信息给worker对象，下面是相同的代码。

|  |  |
| --- | --- |
| 1 | worker.postMessage(); |

当worker线程发送数据的时候，我们在调用结束的时候，通过”onMessage”事件获取

**3.**[如何中止Web Worker?](http://blog.jobbole.com/78346/" \l "q25" \t "http://blog.csdn.net/u013418331/article/details/_blank)

w.terminate();

**4.**[为什么我们需要HTML5](http://blog.jobbole.com/78346/" \l "q26" \t "http://blog.csdn.net/u013418331/article/details/_blank)**的服务发送事件？**

网络世界的普遍需求是从服务器更新。以一个股票应用为例，浏览器必须定期从服务器更新最新的股票值。

现在实现这类需求开发者通常写一些PULL的代码，到服务器同时抓取某些区间数据。现在PULL的解决方案是很好的，但是这使得网络健谈有很多的调用，同时增加了服务器的负担。

因此相比于PULL，如果我们能采用某种PUSH的解决方案那会是很棒的。简而言之，当服务器更新的时候，将会发送更新到浏览器客户端，那可以被接受通过使用”SERVER SENT EVENT”

因此首要的是浏览器需要连接将会发送更新的服务器资源，比方说我们有一个”stock.aspx”页面会发送股票更新，因此连接该页面，我们需要使用附加时间来源对象，

**5.**[HTML5中的本地存储概念是什么？](http://blog.jobbole.com/78346/" \l "q27" \t "http://blog.csdn.net/u013418331/article/details/_blank)

很多时候我们会存储用户本地信息到电脑上，例如：比方说用户有一个填充了一半的长表格，然后突然网络连接断开了，这样用户希望你能存储这些信息到本地，当网络恢复的时候，他想获取这些信息然后发送到服务器进行存储  
现代浏览器拥有的存储被叫做“Local Storage”,你可以存储这些信息。

**6.**[我们如何从本地存储中添加和移除数据？](http://blog.jobbole.com/78346/" \l "q28" \t "http://blog.csdn.net/u013418331/article/details/_blank)

数据添加到本地存储采用键值对，以下示例显示了城市数据”India”添加了键”Key001”

**7.**[本地存储的生命周期是什么？](http://blog.jobbole.com/78346/" \l "q29" \t "http://blog.csdn.net/u013418331/article/details/_blank)

本地存储没有生命周期，它将保留知道用户从浏览器清除或者使用Javascript代码移除。

**8.**[应用缓存中的回退是什么？](http://blog.jobbole.com/78346/" \l "q39" \t "http://blog.csdn.net/u013418331/article/details/_blank)

应用缓存中的回退帮助你指定在服务器不可访问的时候，将会显示某文件。例如在下面的manifest文件中，我们说如果谁敲击了”/home”同时服务器不可到达的时候，”homeoffline.html”文件应送达

FALLBACK:

/home/ /homeoffline.html

**9. web storage和cookie的区别**

"与Cookie相比，Web Storage存在不少的优势，概括为以下几点：

1. 存储空间更大：能提供5MB的存储空间（不同[浏览器](https://www.baidu.com/s?wd=%E6%B5%8F%E8%A7%88%E5%99%A8&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3nvDLPhn1uW6vuy79mvNB0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHRYrjc1nWcd" \t "http://blog.csdn.net/u013418331/article/details/_blank)的提供的空间不同），Cookie仅4KB

2. 存储内容不会发送到服务器：当设置了Cookie后，Cookie的内容会随着请求一并发送的服务器，这对于本地存储的数据是一种带宽浪费。而Web Storage中的数据则仅仅是存在本地，不会与服务器发生任何交互。

3. 更多丰富易用的接口：Web Storage提供了一套更为丰富的接口，使得数据操作更为简便。

4. 独立的存储空间：每个域（包括子域）有独立的存储空间，各个存储空间是完全独立的，因此不会造成数据混乱。

**10. CSS3有哪些新特性？**

第 1 选择器  
第 2 RGBA和透明度  
第 3 多栏布局  
第 4 多背景图  
第 5 Word Wrap  
第 6 文字阴影  
第 7 @font-face属性  
第 8 圆角(边框半径)  
第 9 边框图片  
第 10 盒阴影  
第 11 盒子大小  
第 12 媒体查询  
第 13 语音