**前言        为什么要优化性能对于前端工程师如此重要**

          在行业内有句话不知道大家有没有听说过，‘懂得性能优化并且研究过jquery源代码的人和不懂得性能优化写出来的代码对于性能的消耗会相差上百倍甚至上千倍'，现在的javascript属于从ECMAscript3到ECMAscript5以及ECMAscript6的一个过渡的过程。在javascript的编写不健全的时候编写代码方法不得当，引起的问题也是不容忽视的。

    性能优化

    下面将自己对于性能优化的一些见解与大家分享；

**1.精灵图**

最基本的是尽可能的将背景图片做成精灵图，减少图片的请求，所以一般web工程师的另一项基础本能就是精灵图的制作。

**2.css选择器优化**

        在css中要尽量的使用子代选择器>,少使用后代选择器，使用后代选择器时，搜素引擎会将所有的后代元素都进行搜索，如果我们使用子代选择器时，可以将搜索的范围缩小，从而减少搜索引擎的性能消耗。

**3.js改变样式直接操作类名**

       js操作元素样式时,不要用style去直接添加样式，一般属性少时不会影响多少性能，其实则不然，在每次添加样式时，页面都会重绘一次，重绘是不得不重视的，操作样式时，直接操作类名，只引起一次重绘，用style直接添加样式会引起多次的重绘。

**4.js直接操作dom节点**

      当操作节点时尽量将节点添加在元素的后面，如果插入到节点的前面时，会使插入节点之后的节点都引起回流，而插入到后面时只需要被插入的节点回流一次就可以了。

    可能有人不理解重绘与回流的概念

**5.正则匹配选择器**

在css3以及jQuery中的属性选择器，这些选择器中有的是用正则进行匹配的尽量不要去使用，当然如果对于性能优化不进行考虑的话，这些方法还是比较好用的，正则匹配选择器会使搜索引擎搜索所有的标签，很大的影响性能

**6.js获取元素优化**

在js中获得元素时，正常是使用document.getElementsById，搜索引擎会从Dom树的最底端，进行搜索，直到搜索到window中的document再进行返回搜索，所以在获得元素时最好是将document.body进行储存，当再次使用时，只需将这个变量取出使用，可以节省搜索引擎的性能

**7.内存溢出**

     一般在递归运行时，会产生内存溢出，造成在运行递归时性能大幅度下降，在运行结束后内存会被系统回收，所以在运行递归时需要用对象将值保存，在每次递归运算时检测，如果存在则直接返回，不存在则添加，这样就可以解决递归的很大性能。

**8.对Ajax用GET请求**

**POST请求是通过先发送HTTP请求头，再发送数据来实现的，GET而是没有请求头的，但是需要注意**：GET大小限制约4K，POST则没有限制。

**9.延迟加载图片**

在页面发起请求时，请求量过大，可以使图片进行懒加载，当页面滚到到图片的位置时，再进行加载图片。

**下面给大家推荐一个图片懒加载的插件**

[?](http://www.jb51.net/article/83676.htm)

|  |  |
| --- | --- |
| 1 | jquery.lazyload.js |

**10.避免图片src属性为空**

**src** 属性是空字符串的图片很常见，主要以两种形式出现：

    <img src=””>

    var img = new Image();img.src = “”;

    这两种形式都会引起相同的问题：浏览器会向服务器发送另一个请求。

    技巧  
**1.排他思想**

    先排除当前所有的，再进行下一次操作；一般在进行动画、添加样式等时,在二次操作时先清楚前面的样式再从新添加新的样式，避免同时进行动画的冲突。

**2.短路运算（||）**

    一般在一个值需要判断是否存在时，尽量不要使用if语句,可以使用短路运算符，可以省去代码占用的内存。比如：

[?](http://www.jb51.net/article/83676.htm)

|  |  |
| --- | --- |
| 1 | a=a||b; |

a存在则使用a，a不存在则使用b。

**3.运算（&&）**

    可以运用在条件匹配时，以便条件的多余查询，比如判断一个对象是否为数组时，

[?](http://www.jb51.net/article/83676.htm)

|  |  |
| --- | --- |
| 1 | a && a.length&& a.length>=0 |

**4.伪数组以及数组**

        当需要封装一个非数组元素变为一个数组时，最简单的办法就是给其添加一个[],如果需要是一个伪数组时，可以给其设置一个length属性。

**5.节流阀**

    一般运用在动画中，设定一个值，开始时设置为true，判断其值，在进入动画时给这个值赋值为false，在动画结束时使用回调函数再重新赋值为true。

**6.解除文本的被动选中状态（纯干货）**

    当在点击一些按钮时，有时会出现文本被选中的情况，让客户看见就犹如bug一般，以下是解除这一状态的代码，粘贴即可。

[?](http://www.jb51.net/article/83676.htm)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | if(document.all){     document.onselectstart= function(){return false}; //for ie   }else{     document.onmousedown= function(){return false};     document.onmouseup= function(){return true};   }   document.onselectstart = new Function('event.returnValue=false'); |

**7.巧妙使用三元运算符**

当需要判断一个值是否存在，也可以使用三元运算符使代码更加优化。比如

[?](http://www.jb51.net/article/83676.htm)

|  |  |
| --- | --- |
| 1 | obj=obj===undefined?Object:obj; |