[**document的createDocumentFragment()方法**](http://www.cnblogs.com/yunfour/archive/2011/06/21/2085911.html)

在《javascript高级程序设计》一书的6.3.5:创建和操作节点一节中，介绍了几种动态创建html节点的方法，其中有以下几种常见方法：

· crateAttribute(name)：　　　　　 　　用指定名称name创建特性节点

· createComment(text)：　　　　　　　创建带文本text的注释节点

· createDocumentFragment()：　　　　创建文档碎片节点

· createElement(tagname)：　　　　　  创建标签名为tagname的节点

· createTextNode(text)：　　　　　　   创建包含文本text的文本节点

其中最感兴趣且以前没有接触过的一个方法是createComment(text)方法，书中介绍说：在更新少量节点的时候可以直接向document.body节点中添加，但是当要向document中添加大量数据时，如果直接添加这些新节点，这个过程非常缓慢，因为每添加一个节点都会调用父节点的appendChild()方法，为了解决这个问题，可以创建一个文档碎片，把所有的新节点附加其上，然后把文档碎片一次性添加到document中。

假如想创建十个段落，使用常规的方式可能会写出这样的代码：

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | for(var i = 0 ; i < 10; i ++) {      var p = document.createElement("p");      var oTxt = document.createTextNode("段落" + i);      p.appendChild(oTxt);      document.body.appendChild(p);  } |

当然，这段代码运行是没有问题，但是他调用了十次document.body.appendChild()，每次都要产生一次页面渲染。这时碎片就十分有用了：

[?](http://www.cnblogs.com/yunfour/archive/2011/06/21/2085911.html)

|  |  |
| --- | --- |
| 1 | var oFragment = document.createDocumentFragment(); |
| 1  2  3  4  5 | for(var i = 0 ; i < 10; i ++) {      var p = document.createElement("p");      var oTxt = document.createTextNode("段落" + i);      p.appendChild(oTxt);  oFragment.appendChild(p);  } |
| 1 | document.body.appendChild(oFragment); |

在这段代码中，每个新的<p />元素都被添加到文档碎片中，然后这个碎片被作为参数传递给appendChild()。这里对appendChild()的调用实际上并不是把文档碎片本追加到body元素中，而是仅仅追加碎片中的子节点，然后可以看到明显的性能提升，document.body.appenChild()一次替代十次，这意味着只需要进行一个内容渲染刷新。