* **getElementById(“ID”)**

getElementById()可以访问Documnent中的某一特定元素，顾名思义，就是通过ID来取得元素，所以只能访问设置了ID的元素。

* **document.getElementById("id").value**

直接给元素（文本框）赋值

* **clear 属性规定元素的哪一侧不允许其他浮动元素。**

clear有四个属性值，分别是left（在左侧不允许浮动元素）、right（在右侧不允许浮动元素）、both（在左右两侧均不允许浮动元素）、none（默认值。允许浮动元素出现在两侧）、inherit（规定应该从父元素继承 clear 属性的值）。

* **元素视图属性elem.属性/方法**

\* **offsetWidth** 水平方向 width + 左右padding + 左右border-width

\* **offsetHeight** 垂直方向 height + 上下padding + 上下border-width

\* **clientWidth** 水平方向 width + 左右padding

\* **clientHeight** 垂直方向 height + 上下padding

\*

\* **offsetTop** 获取当前元素到 定位父节点 的top方向的距离

\* **offsetLeft** 获取当前元素到 定位父节点 的left方向的距离

\*

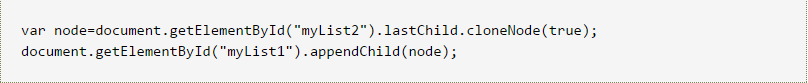
\* **scrollWidth** 元素内容真实的宽度，内容不超出盒子高度时为盒子的clientWidth

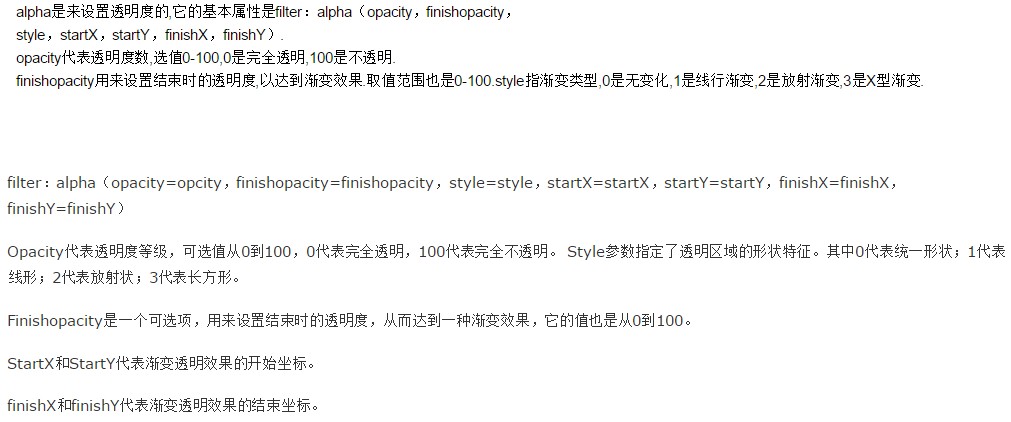
\* **scrollHeight** 元素内容真实的高度，内容不超出盒子高度时为盒子的clientHeight

* **children() 方法返回返回被选元素的所有直接子元素。**
* **appendChild() 方法可向节点的子节点列表的末尾添加新的子节点。**

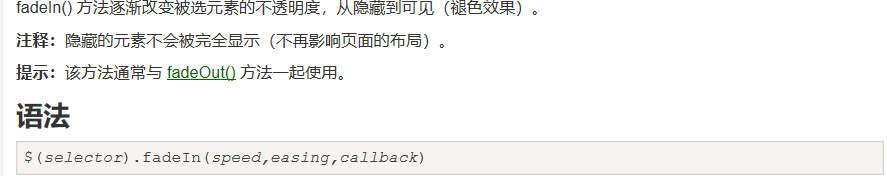
提示：如果文档树中已经存在了 newchild，它将从文档树中删除，然后重新插入它的新位置。如果 newchild 是 DocumentFragment 节点，则不会直接插入它，而是把它的子节点按序插入当前节点的 childNodes[] 数组的末尾。

* **className 属性设置或返回元素的 class 属性。**
* **cloneNode 拷贝一个列表项到另外一个列表**





* **fadeln（）方法**





**each() 方法规定为每个匹配元素规定运行的函数。each();**$(*selector*).each(*function(index,element)*)

**提示：**返回 false 可用于及早停止循环。

* **Absolution：元素会脱离文档流，定位是相对于离它最近的且不是static定位的父元素而言，若该元素没有设置宽度，则宽度由元素里面的内容决定，且宽度不会影响父元素，定位为absolution后，原来的位置相当于是空的，下面的的元素会来占据。**

**Relative：元素仍处于文档流中，定位是相对于原本自身的位置，若没有设置宽度，则宽度为父元素的宽度，该元素的大小会影响父元素的大小。**

