Setattribute一般还是用来设置属性吧，

因为ie不支持setarrtibute设置onclick，还有css样式之类

因此setattribute一般用来设置class，id之类的

而关于js动态修改css样式，还有onclick

可以如a.onclick=function(){};

a.style.color=”#00f”;

但设置css最好还是

阿三  ***a.style.cssText=””（双引号里面按照css格式写，也不用考虑大小写了）***

**1. 直接设置style的属性**  某些情况用这个设置 !important值无效

如果属性有'-'号，就写成驼峰的形式（如textAlign）  如果想保留 - 号，就中括号的形式  element.style['text-align'] = '100px';

element.style.height = '100px';

**2. 直接设置属性**（只能用于某些属性，相关样式会自动识别）

element.setAttribute('height', 100);

element.setAttribute('height', '100px');

**3. 设置style的属性**

element.setAttribute('style', 'height: 100px !important');

**4. 使用setProperty**  如果要设置!important，推荐用这种方法设置第三个参数

element.style.setProperty('height', '300px', 'important');

**5. 改变class**   比如JQ的更改class相关方法

因JS获取不到css的伪元素，所以可以通过改变伪元素父级的class来动态更改伪元素的样式

element.className = 'blue';

element.className += 'blue fb';

**6. 设置cssText**

element.style.cssText = 'height: 100px !important';

element.style.cssText += 'height: 100px !important';

**7. 创建引入新的css样式文件**

[复制代码](javascript:void(0);)

复制代码

function addNewStyle(newStyle) {

var styleElement = document.getElementById('styles\_js');

if (!styleElement) {

styleElement = document.createElement('style');

styleElement.type = 'text/css';

styleElement.id = 'styles\_js';

document.getElementsByTagName('head')[0].appendChild(styleElement);

}

styleElement.appendChild(document.createTextNode(newStyle));

}

addNewStyle('.box {height: 100px !important;}');

复制代码

[复制代码](javascript:void(0);)

**8. 使用addRule、insertRule**

复制代码

// 在原有样式操作

document.styleSheets[0].addRule('.box', 'height: 100px');

document.styleSheets[0].insertRule('.box {height: 100px}', 0);

// 或者插入新样式时操作

var styleEl = document.createElement('style'),

styleSheet = styleEl.sheet;

styleSheet.addRule('.box', 'height: 100px');

styleSheet.insertRule('.box {height: 100px}', 0);

document.head.appendChild(styleEl);

用JS来动态设置CSS样式，常见的有以下几种

**1. 直接设置style的属性**  某些情况用这个设置 !important值无效

如果属性有'-'号，就写成驼峰的形式（如textAlign）  如果想保留 - 号，就中括号的形式  element.style['text-align'] = '100px';

element.style.height = '100px';

**2. 直接设置属性**（只能用于某些属性，相关样式会自动识别）

element.setAttribute('height', 100);

element.setAttribute('height', '100px');

**3. 设置style的属性**

element.setAttribute('style', 'height: 100px !important');

**4. 使用setProperty**  如果要设置!important，推荐用这种方法设置第三个参数

element.style.setProperty('height', '300px', 'important');

**5. 改变class**   比如JQ的更改class相关方法

因JS获取不到css的伪元素，所以可以通过改变伪元素父级的class来动态更改伪元素的样式

element.className = 'blue';

element.className += 'blue fb';

**6. 设置cssText**

element.style.cssText = 'height: 100px !important';

element.style.cssText += 'height: 100px !important';

**7. 创建引入新的css样式文件**

[复制代码](javascript:void(0);)

复制代码

function addNewStyle(newStyle) {

var styleElement = document.getElementById('styles\_js');

if (!styleElement) {

styleElement = document.createElement('style');

styleElement.type = 'text/css';

styleElement.id = 'styles\_js';

document.getElementsByTagName('head')[0].appendChild(styleElement);

}

styleElement.appendChild(document.createTextNode(newStyle));

}

addNewStyle('.box {height: 100px !important;}');

复制代码

[复制代码](javascript:void(0);)

**8. 使用addRule、insertRule**

复制代码

// 在原有样式操作

document.styleSheets[0].addRule('.box', 'height: 100px');

document.styleSheets[0].insertRule('.box {height: 100px}', 0);

// 或者插入新样式时操作

var styleEl = document.createElement('style'),

styleSheet = styleEl.sheet;

styleSheet.addRule('.box', 'height: 100px');

styleSheet.insertRule('.box {height: 100px}', 0);

document.head.appendChild(styleEl);