**DOM**

1. **DOM树：**

DOM树体现着html页面的层级结构,而DOM树有DOM文档树和DOM元素树两种,DOM元素树包含的只有元素节点,而DOM文档树则包括DOM文档中的所有内容.

**2.获取元素**

1.getElementById() //用id获取元素名

2.document.getElementsByTagName(); //用标签名获取元素

3.document.getElementsByClassName(); //用类名获取元素

4.document.getEelmentsByName();//用name属性值获取元素

5.querySelector()和querySelectorAll() //用css用选择器获取元素

1. **获取和设置元素信息**
2. TagName(); //获取元素名
3. .innerHTML; //获取元素节点的内容

3.:.innerText; //获取元素节点的文本

4..className;//获取元素的类名

5.style属性可以获取元素内联样式的所有属性//获取元素样式

6.getAttribute(“属性“);//获取元素属性：

7.setAttribute(“属性”,”属性值”);//设置元素属性：

8.removeAttribute(“需要删除的属性”)；//删除元素属性

1. **增加元素**

1**.**document.creatElement()//创建元素节点

2.document.createTextNode('一段文字'）//创建文本节点

3.style.cssText // css样式赋予

4.insertBefore() appendChild() //在某元素前插入创建新的元素

1. **删除和替换元素**
2. replaceChild() //元素的替换
3. removeChild()//元素的删除
4. **查找节点(node)**

1：node节点的属性

console.log(mydiv.nodeName);

console.log(mydiv.noValue);

console.log(mydiv.nodeType);

2：获取所有的子节点：childNodes; 所有的子节点不止包含元素.

3： 获取第一个子节点：firstChild;

获取最后一个子节点：lastChild;

4：获取父元素：parentNode

5.：获取兄弟节点：

获取第一个兄弟节点：proviousSibling

获取最后一个兄弟节点:nextSibling

**6.元素的宽高属性**

1：获取元素的宽：

元素.style：只能获取内联样式。

元素.offsetWidth ：获取元素的宽，也包含了内边距和边框

元素.clientWidth：获取元素的宽，不包含边框

1. 获取元素的高:

元素.offsetHeight ：获取元素的高，也包含了内边距和边框

元素.clientHeight：获取元素的高，不包含边框

1. 子元素与父元素的距离

元素.offsetLeft：获取当前元素距离body左边界的距离

元素.offsetTop：获取当前元素距离body上边界的距离

注意：如果该元素使用了定位属性，这时就是距离它最近的父元素的距离

# 事件

**1.鼠标键盘事件**

鼠标事件

1..onclick//点击事件

2..ondblclick//双击事件

3.onmousedown;//鼠标按下事件

4.Onmouseup//鼠标抬起事件

5.onmouseover//鼠标移入事件

6.Onmouseout//鼠标移出事件

7.Onmouseenter//鼠标移入事件

8.Onmouseleava//鼠标移出事件

键盘事件

Onkeydown//按下事件

.onkeyup//抬起事件

**2.Event事件对象**

1.event//对象的常见属性

2.clientX和clientY//鼠标的坐标值

3.keyCode //键盘的值

Event的兼容性

* 1. 非IE：event
  2. IE：window.event

**3.对象事件**

1.Window.onload//页面加载完毕之后才执行函数的内容

2.Window.onresize//监听页面窗口大小 ,只要页面变化就触发该事件

**4.表单事件**

1. 元素.onfocus 获取焦点
2. 元素.onblur 失去焦点
3. 元素.onchange 用户输入时触发
4. 元素.oninput 内容改变时触发

**5.事件绑定 事件监听**

1.onclick=function(){}//事件绑定

2..onclick=null; //事件解绑

3.addEventListener(监听事件,方法,冒泡),//监听器