1. input文本框为只读不可编辑的方法

方法1：onfocus=this.blur() 当鼠标放上就离开焦点

<input type="text" name="input1" value="中国" onfocus=this.blur()>

方法2：readonly

<input type="text" name="input1" value="中国" readonly>

<input type="text" name="input1" value="中国" readonly="true">

方法3：disabled 文字会变成灰色

 <input type="text" name="input1" value="中国" disabled="true">

1. eval(string) 计算其中的字符串，并执行其中的js代码
2. parseInt(num)转化为int parseFloat(num)转化为float
3. input文本框中的内容从右到左

方法1：text-align:right

<input style="text-align:right" />

方法2：dir=”rtl”

<input type="text" dir="rtl" />

1. 截取字符串

string.slice(start,end)

string.substring(start,stop)

stringObject.substr(start [, length ])

1. 字符串 替换

var str = 'abcadeacf';

var str1 = str.replace('a', 'o');//只替换第一个

alert(str1);

// 打印结果： obcadeacf

var str2 = str.replace(/a/g, 'o');//替换所有

alert(str2);

//打印结果： obcodeocf,

1. 数组添加删除

添加：

push(num)从尾部

unshift(num)从头部

删除：

pop()从尾部

shift()从头部

1. sort

//从小到大

arr.sort(function(num1,num2){

return num1-num2;

})

1. splice 先删除 后插入

删除 splice(开始位置，长度)

插入 splice(开始位置,0,元素…)

替换 splice(开始位置，删除的长度，要插入的元素…)

1. 创建和插入节点

var oLi=document.createElement('li');//创建

oLi.innerHTML=oTxt.value;

//父.appendChild(newNode);

oUl.appendChild(oLi);//插入

//父.insertBefore(newNode,某子节点)

oUl.insertBefore(oLi,aLi[0]);//在某节点前插入

如果原来没有子节点，不能insertBefore 可以判断一下有没有子节点

1. 删除

oUl.removeChild(this.parentNode);

1. 文档碎片

理论上提高dom性能 一般不使用 实际上不怎么提高

window.onload=function(){

var iStart=new Date().getTime();

var oBtn=document.getElementById('btn1');

var oUl=document.getElementById('ul1');

var oFrag=document.createDocumentFragment();

var i=0;

for(i=0;i<i<10000;i++){

var oLi=document.createElement('li');

oFrag.appendChild(oLi);

}

oUl.appendChild(oFrag);

console.log(new Date().getTime()-iStart);

}

1. DOM节点

childNodes 子节点 火狐把前后空格也算作子节点

parentNode 父节点

nodeType 节点类型 3-文本节点 1-元素节点(标签)

offsetParent 比如positon-absolute的元素的offsetParent是relative的元素

html标签是定位的根

childNodes和nodeType结合

if(nodex.childNodes[i].nodeType==1){…}

children是childNodes的兼容版

firstChild 第一个子节点 在火狐下不兼容 火狐下是firstElementChild

var oFirst=oUl.firstChild||oUl.firstElementChild

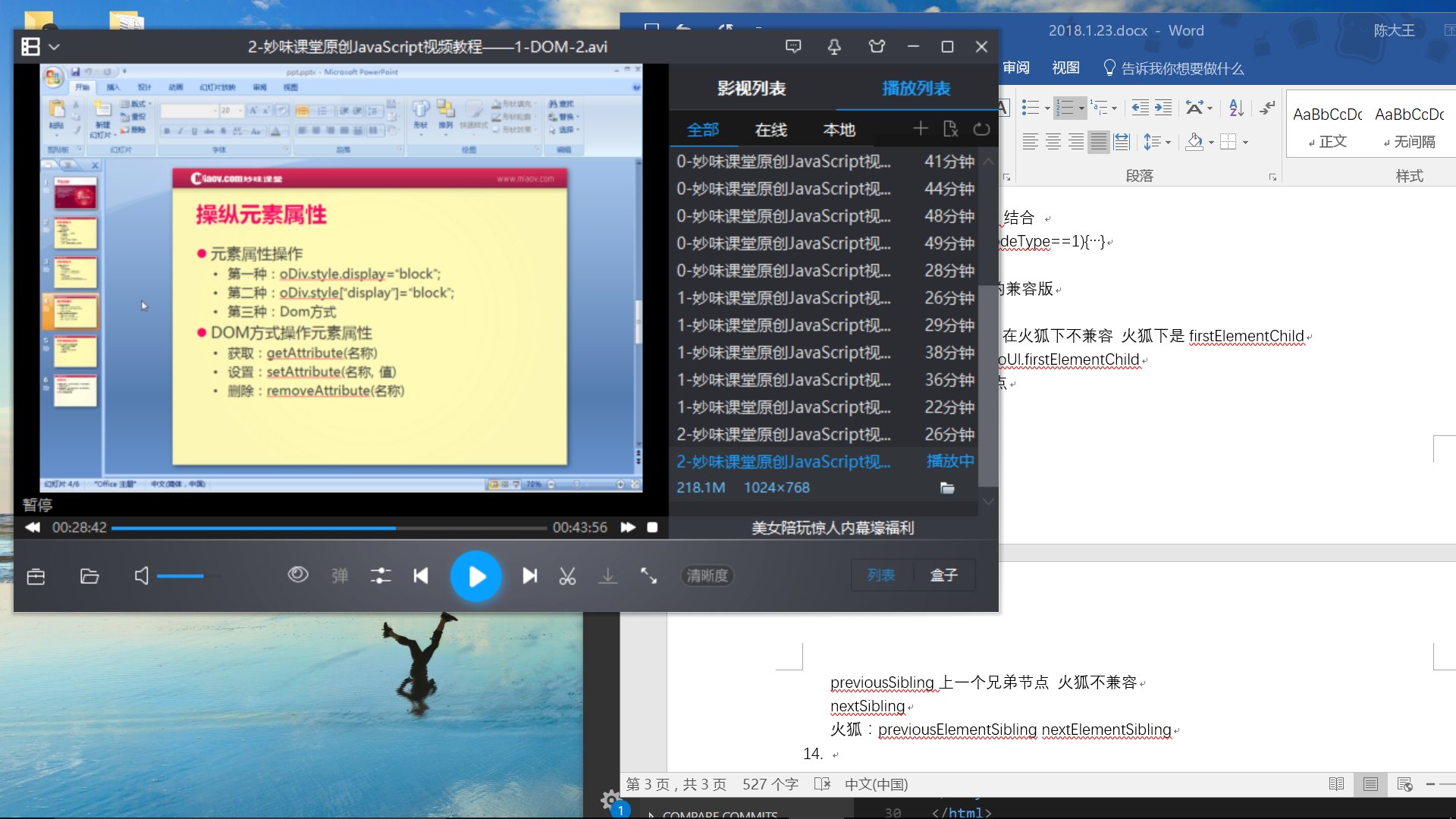
lastChild 最后一个子节点

previousSibling上一个兄弟节点 火狐不兼容

nextSibling

火狐：previousElementSibling nextElementSibling

1. 操纵元素属性



1. 用className选择元素

function getByClass(oParent,sClass)

{

var aEle=oParent.getElementByTagName(‘\*’);

var aResult=[];

var i=0;

for(i=0;i<aEle.length;i++)

{

if(aEle[i].className==sClass)

{

aResult.push(aEle[i]);

}

}

return aResult;

}