



做真实的自己，用良心做教育

H5教学部

DOM下

目录

做真实的自己，用良心做教育

1

DOM属性节点

2

DOM节点间关系

3

DOM操作

4

滚动事件onscroll

DOM属性节点

1, 属性节点的属性

attributes属性: 获取某元素节点的所有属性节点(返回一个数组)。

```
var box = document.getElementById( 'box' );
```

```
box.attributes; //[object NamedNodeMap]
```

```
box.attributes.length; //返回属性节点个数
```

```
box.attributes[0]; //[object Attr], 返回第一个属性节点
```

```
box.attributes[0].nodeType; //2 , 节点类型
```

```
box.attributes[0].nodeName; //属性名称
```

```
box.attributes[0].nodeValue; //属性值
```

```
box.attributes[ 'id' ]; //返回属性名称为id的节点
```

```
box.attributes.getItem( 'id' ); //返回属性名称为id的节点
```

DOM属性节点

2, 属性节点操作的三个函数

我们通过三个函数可以操作元素的属性:

`getAttribute()`: 通过属性名获取对应的属性值

`setAttribute()`: 设置一个key-value形式的属性键值对

`removeAttribute()`: 移除指定的属性

DOM属性节点

getAttribute()

getAttribute()方法将获取元素中某个属性的值。它和直接使用点语法(.属性)获取属性值的方法有一定区别.

例如:

`document.getElementById('box').getAttribute('id');` 获取元素的id属性值

`document.getElementById('box').id;` 获取元素的id属性值(点语法)

`document.getElementById('box').getAttribute('class');` 获取元素的class值

`document.getElementById('box').className;` 获取元素的class值(点语法)

//获取元素的自定义属性值(不可以使用点语法)

`document.getElementById('box').getAttribute(bbb);`

`document.getElementById('box').bbb;` 无法获取元素的自定义属性值

DOM属性节点

setAttribute()

setAttribute()方法可以给元素添加属性; 如果属性已经存在, 则会覆盖原来的属性; 需要传入两个参数: 属性名和属性值.

例如:

//设置属性和值

```
document.getElementById( 'box' ).setAttribute( 'align' , 'center' );
```

//设置自定义的属性和值

```
document.getElementById( 'box' ).setAttribute( 'bbb' , 'ccc' );
```

removeAttribute()

removeAttribute()方法可以移除元素的属性。

//移除属性

```
document.getElementById('box').removeAttribute('style');
```

DOM节点间关系

元素节点间关系的相关属性

`childNodes`: 子节点集合

`firstChild`: 用于获取当前元素节点的第一个子节点, 相当于`childNodes[0]`;

`lastChild`: 用于获取当前元素节点的最后一个子节点, 相当于
`childNodes[box.childNodes.length-1]`.

例如:

//获取第一个子节点的文本内容

```
console.log(box.firstChild.nodeValue);
```

//获取最后一个子节点的文本内容

```
console.log(box.lastChild.nodeValue);
```

DOM节点间关系

元素节点间关系的相关属性

parentNode、previousSibling、nextSibling

parentNode: 属性返回该节点的父节点，

previousSibling: 属性返回该节点的前一个同级节点，

nextSibling: 属性返回该节点的后一个同级节点。

例如:

```
console.log(box.parentNode.nodeName); //获取父节点的标签名  
console.log(box.lastChild.previousSibling); //获取前一个同级节点  
console.log(box.firstChild.nextSibling); //获取后一个同级节点
```


DOM操作

DOM节点操作

DOM节点操作指的是: 创建节点、复制节点、插入节点、删除节点和替换节点等.

DOM操作所涉及的方法有:

`createElement()`; 创建一个元素节点

`appendChild()`; 添加一个新子节点到子节点的末尾

`createTextNode()`; 创建一个文本节点

`insertBefore()`; 将新节点插入到某节点前面

`replaceChild()`; 将新节点替换旧节点

`cloneNode()`; 复制节点

`removeChild()`; 移除节点

DOM操作

1, createElement()

createElement()方法可以创建一个元素节点。

例如:

```
var newEle = document.createElement( 'div' );
```

2, appendChild()

appendChild()方法可以将一个新节点添加到某个节点的子节点列表的末尾上。

例如:

```
var box = document.getElementById( 'box' );
```

```
var pNode = document.createElement( 'p' ); //创建一个新元素节点<p>
```

```
box.appendChild(pNode); //把新元素节点<p>添加box节点的子节点末尾
```

DOM操作

3, createTextNode()

createTextNode() 方法可以创建一个文本节点。

例如:

```
var textNode = document.createTextNode( '段落' ); //创建一个文本节点  
p.appendChild(textNode); //将文本节点添加到子节点末尾
```

示例:

➤封装一个函数, 带两个参数(元素节点的名称, 该节点的文本内容), 返回一个带文本内容的元素节点

DOM操作

4, insertBefore()

insertBefore()方法可以把节点插入到指定节点的前面。

例如:

//通过父节点调用, 在box之前插入一个新节点p;

//第一个参数为新节点

```
box.parentNode.insertBefore(p, box);
```

5, replaceChild()

replaceChild()方法可以把节点替换成指定的节点。

例如:

//通过父节点调用, 新节点p替换了旧节点div

//第一个参数为新节点, 第二个参数为旧节点

```
box.parentNode.replaceChild(p, box);
```

DOM操作

6, cloneNode()

cloneNode()方法可以把子节点复制出来。

例如:

//获取第一个子节点, true表示复制标签和内容, false表示只复制标签

```
var box = document.getElementById( 'box' );
```

```
Var newNode = box.firstChild.cloneNode(true);
```

```
box.appendChild(newNode); //添加到子节点列表末尾
```

7, removeChild()

removeChild()方法可以删除指定子节点

例如:

//通过父节点调用, 来删除指定子节点

```
box.parentNode.removeChild(box);
```

事件中的this

在JS事件中, this表示触发事件的元素节点对象;

```
var box = document.getElementById('box');  
box.onclick = function() {  
    console.log(this.nodeName); //this表示box对象  
};
```

通过for循环添加事件, 使用this

```
var aInput = document.getElementsByTagName('input');  
for (var i=0; i<aInput.length; i++) {  
    aInput[i].onclick = function() {  
        console.log(this.value); //这里的this表示被点击的那个input元素节点对象  
    };  
}
```

练习

1, 动态表格的创建, 在输入框中输入内容后, 点击保存则在表格末尾添加一条记录, 点击删除则会删除对应行的记录

ID: 姓名: 电话:

Id	Name	Tel	操作
30	张三	135234234	删除
40	李四	14543423423	删除
222	王五	11112123	删除

提示: 也可以用以下方法删除table对于的行, rowIndex: 表中对应的行下标
`oTable.deleteRow(tr标签.rowIndex);`

滚动事件onscroll

onscroll滚动事件: 实时获取距离顶部/左边的距离(顶部不可见区域的高度/宽度)

例如:

```
window.onscroll = function() {  
    var scrollTop = document.documentElement.scrollTop  
                    || document.body.scrollTop;  
    var box = document.getElementById("box");  
    box.innerHTML = scrollTop;  
}
```

注意: 兼容问题

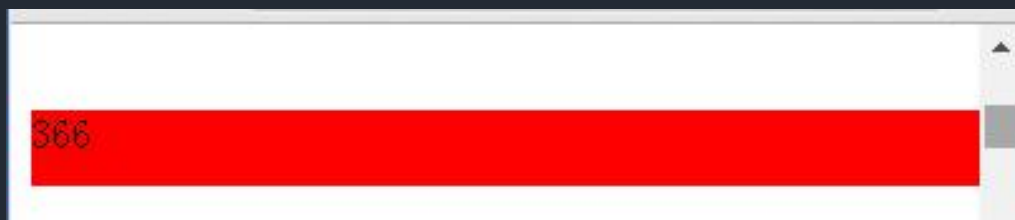
document.documentElement.scrollTop: 不支持谷歌

document.body.scrollTop: 支持谷歌

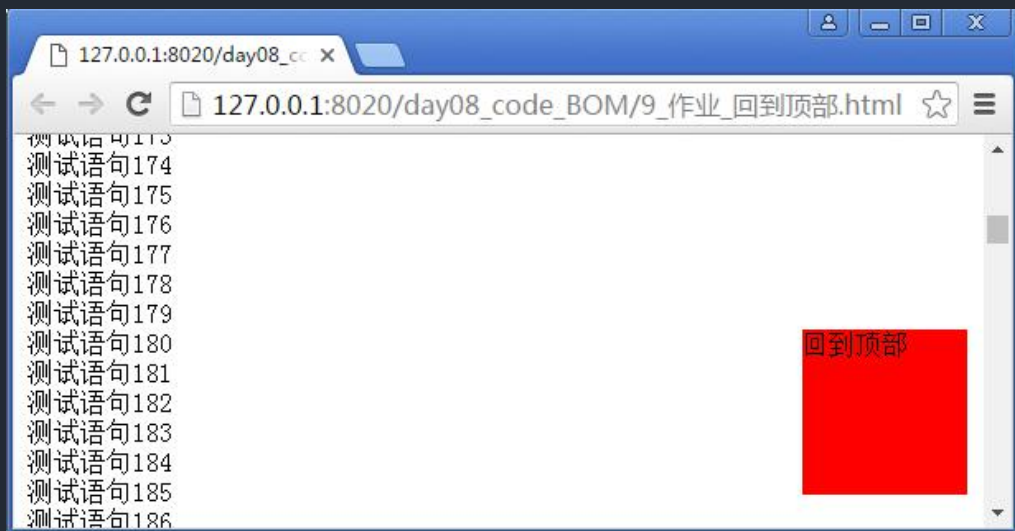
滚动事件onscroll

示例：

1, 滚动页面, 显示scrollTop的值



2, 回到顶部, 当滚动条滚动到距离顶部超过2000时, 显示回到顶部(红色部分), 并当点击回到顶部时, 让页面回到顶部



练习

1, 点击表格中的三张图片, 切换下面的图



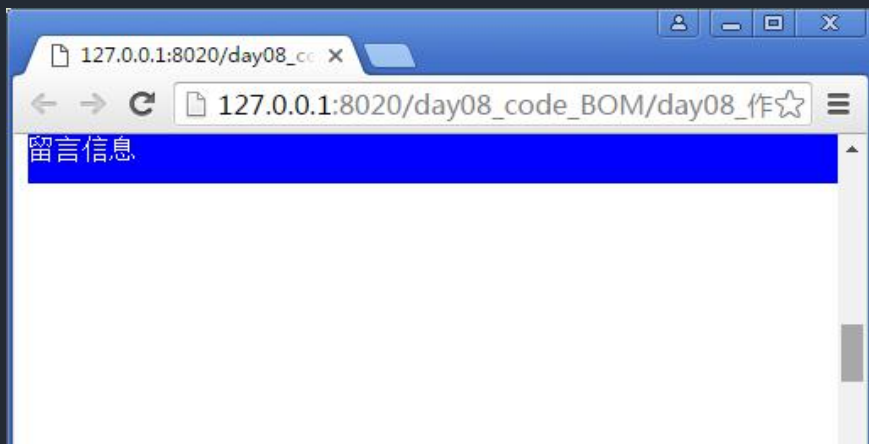
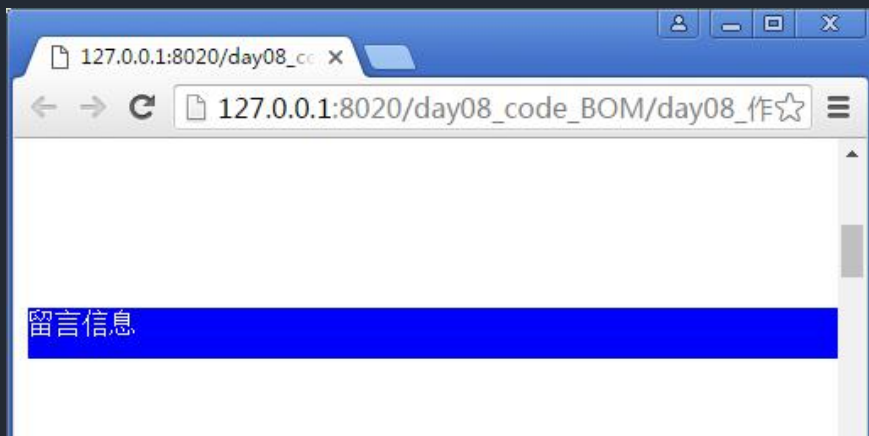
练习

2, 简易年历, 如右图, 可以点击对应的月份, 改变下面的文字内容



练习

3, 顶部悬浮(吸顶效果),当滚动条滚动到较下方时, 留言信息一直显示在顶部



练习

4, tab切换, 点击科技显示左图, 点击探索显示右图



THANK YOU



做真实的自己，用良心做教育