1.兄弟节点解决兼容性方法：

*function* getNextElementSibling(*element*) {

*var* el = element;

        while (el = el.nextSibling) {

            if (el.nodeType === 1) {

                return el;

            }

        }

        return null;

    }

*var* li = document.querySelector('#top').querySelector('li');

*var* nextElementSibling = getNextElementSibling(li);

*console*.log(nextElementSibling);

Element是所获取的元素 查询的就是获取的元素的兄弟

2.处理事件监听的兼容性方法

*var* li = document.querySelector('#top').querySelector('li');

*function* addEventListener(*element*, *eventName*, *fn*) {

        if (element.addEventListener) {

            element.addEventListener(eventName, fn);

        } else if (element.attachEvent) {

            element.attachEvent('on' + eventName, fn);

        } else {

            element['on' + eventName] = fn;

        }

    }

    addEventListener(li, 'click', *function*() {

        this.style.background = 'red';

    })

3.事件监听删除的兼容性处理

// 事件监听删除兼容性处理方法

    // var li = document.querySelector('#top').querySelector('li');

*function* removeEventListener(*element*, *eventName*, *fn*) {

        if (element.removeEventListener) {

            element.removeEventListener(eventName, fn);

        } else if (element.datachEvent) {

            element.datachEvent('on' + eventName, fn);

        } else {

            element['on' + eventName] = null;

        }

    }

  // 事件监听兼容性处理方法

*var* li = document.querySelector('#top');

*function* addEventListener(*element*, *eventName*, *fn*) {

        if (element.addEventListener) {

            element.addEventListener(eventName, fn);

        } else if (element.attachEvent) {

            element.attachEvent('on' + eventName, fn);

        } else {

            element['on' + eventName] = fn;

        }

    }

    addEventListener(li, 'click', fn)

*function* fn() {

        alert(1);

        removeEventListener(li, 'click', fn);

    }

4.阻止冒泡的兼容性处理

*var* ul = document.querySelector('#top');

*var* li = ul.querySelector('li');

    ul.onclick = *function*() {

        alert(1);

    }

    li.onclick = *function*(*e*) {

        alert(1);

        //有兼容性

        e.stopPropagation();

        //无兼容性

        if (e && e.stopPropagation) {

            e.stopPropagation();

        } else {

            window.event.cancelBubble = true;

        }

    }

1. 处理滚动条的兼容性

function getScroll() {

return {

left: window.pageXOffset || document.documentElement.scrollLeft || document.body.scrollLeft||0,

top: window.pageYOffset || document.documentElement.scrollTop || document.body.scrollTop || 0

};

}

使用的时候 getScroll().left