DOM：文档对象类型（document object model）

HTML DOM：规定了访问和操作HTML元素的标准方法，通过它可以对HTML元素进行获取、修改、删除、添加等操作

1. 获取标签的方法
2. 通过id名获取

Var div（div为变量名称）= document.getElementById(‘ id名称’)

1. 通过标签名称获取

Var main（main变量名）= document.getElementsByTagName(‘标签名称’)[ 索引值（index）] 注意：由于标签名称重复，getElements是有s的，要想获取同名标签中的第几个

1. 通过类名获取

Var 变量名 = document.getElementsByClassName(‘类名’)[ 索引值 ] 获取类名为xx的第几个标签，不加索引，表示所有该类名的标签

1. Var 变量名 = document.querySelector(‘ 条件（可以是标签名称，也尅是id，类名，） ’) 表示通过css选择器获取符合选择的标签中的第一个
2. var 变量名 = document.querySelectorAll(可以是标签名称，也尅是id，类名，) ,表示通过css选择器获取所有符合的标签

6.获取文本节点，即标签内部的内容开始标签到结束标签之间 var 变量名1 = 变量名2.innerHTML 这里的 ‘标签’指的是标签（声明的变量名）

7.获取标签的属性：var 变量名1 = 变量名2.attributes

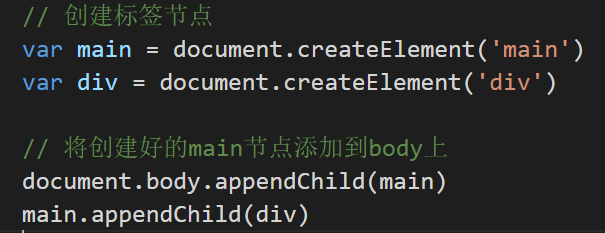
8.获取标签的父级元素和子元素：var 变量名 = 变量名.parentNode / 变量名.childNodes

9.添加新的子节点：div.appendChild(标签名),这里改变的是原标签的位置（选择html中的一个标签为子元素添加到父元素中）该方法没有在html中增加一个新的标签

10.创建新标签，在子元素的最后面添加

Var 变量名1 = document.createElement(‘标签名’)

变量名2（要添加子元素的标签）.appendChild( 变量名1)

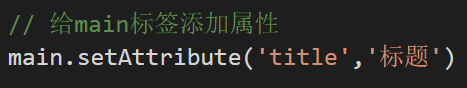


1. 创建文本节点，并添加

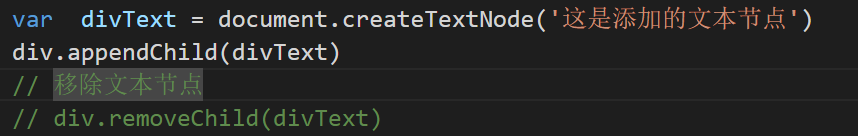
Var 变量名1 = document.createTextNode(‘ 文本内容’)

变量名1.appendChild(变量名1)

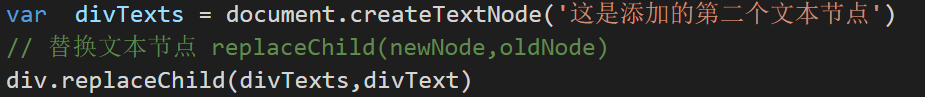
1. 创建新属性：变量名.setAttribute( name,value )



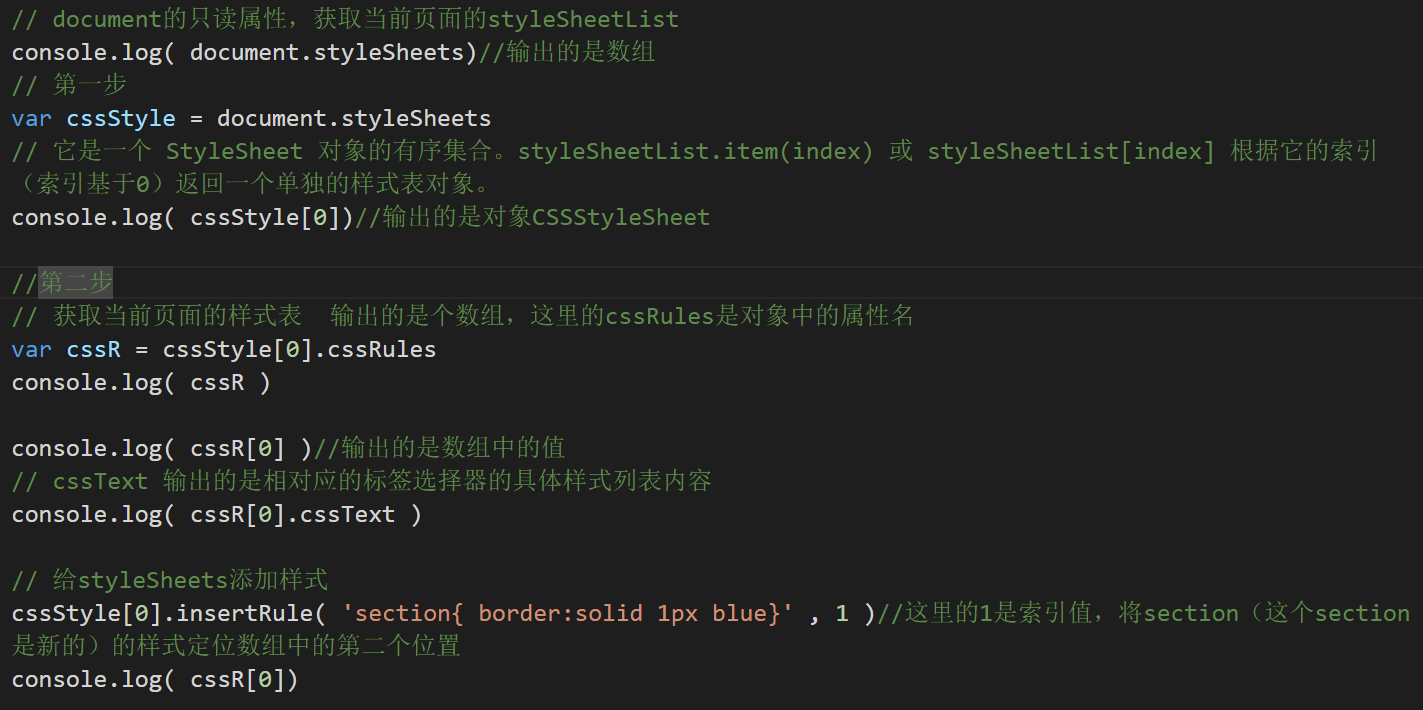
1. 移除节点：removeChild()

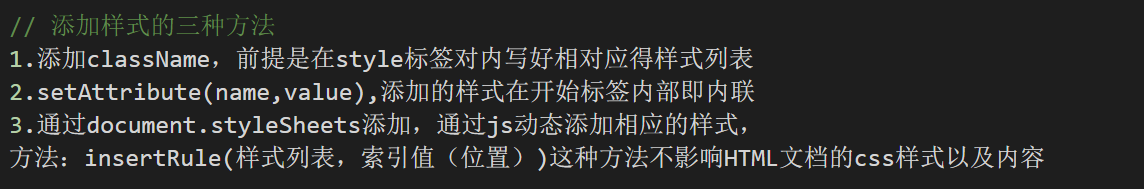


1. 替换节点：replaceChild()

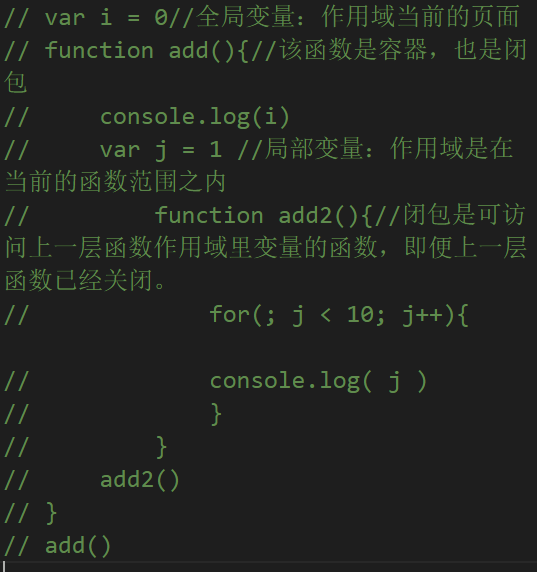


1. 函数的自我调用：(函数内容)()，可以通过这个方法将全局变量变为局部变量
2. styleSheets:样式列表及添加样式的方法





1. 闭包：首先得有两个函数嵌套



1. 类名classList:表示类名表



1. 获取窗口的大小

window.innerHeight:可视窗口的高度

window.innerWidth:可视窗口的宽度 --包含滚动条的宽度

window.outerHeight:整个浏览器的高度

Screen.width:屏幕的大小

document.documentElement.clientWidth:页面（）可以看到的内容的宽度

document.documentElement.offsetWidth：获取的是html的宽度和高度（当设置了html宽高的时候，即为设置的宽高；没有的话就是被内容撑开的宽高）

1. 获取标签的大小

标签的变量名.clientWidth :标签内容的宽度

标签的变量名.clientWidth :标签整个大小

标签的变量名.scrollWidth ：整个宽度（包含带滚动条的隐蔽的地方）

1. 事件的触发执行



1. 事件的捕获和冒泡

当一个文档中有多个事件同时发生，默认的执行顺序是：先小后大（即最小的子元素先执行，依次进行，称为冒泡）

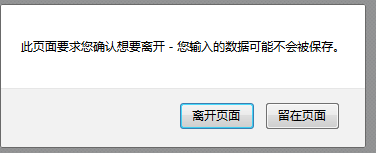
，执行顺序为先大后小的即称为捕获。

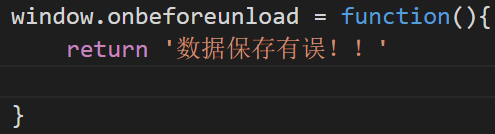
1. 鼠标点击事件
2. onclick：鼠标点击事触发事件
3. onmouseover：鼠标滑过触发事件
4. ondblclick：鼠标双击是触发事件
5. onmousedown:鼠标按钮被按下的时候（左右键和滚动）
6. onmouseleave:鼠标离开元素是触发事件：注意不支持冒泡；onmouseout支持冒泡
7. onmouseenter：鼠标指针移到元素上被触发，与onmouseover类似，但不支持冒泡
8. onmouseup：当鼠标按键被松开的时候触发事件
9. onmousemove：当鼠标被移动时触发事件，
10. 键盘事件
11. onkeydown:键盘按键被按下
12. onkeypress：键盘按键被按下并松开（并不适合所有按键，例如：ALT,CTRL,SHIFT,ESC）
13. onkeyup：键盘按键被松开
14. 表单事件
15. onfocus：获取焦点（即鼠标选中）触发事件
16. onblur：失去焦点（鼠标没有选中元素标签）触发事件
17. oninput:用户输入时触发事件
18. onchange：元素在失去焦点时候检查内容，如果输入框内容改变则触发事件
19. onreset:当重置的时候弹出提示框，这里在函数中可以使用confirm（message）-----会弹出一个带有确定和取消按钮的对话框，弹出的文字是message内容
20. onsubmit:当提交表单时触发事件



注意这里的ev代表的是重置这个事件，用来存储数据，preventDefault:阻止当前事件的发生，不用这个的话，它仍会重置数据

1. onselect：当文本被选中是触发事件
2. 页面框架、对象事件
3. onload：事件在页面或图像加载完成后立即完成以后立即触发
4. onbeforeUnload:如果使用window.onbeforeunload 要使用return'内容'来显示自定义提示内容，浏览器不同显示的也不同



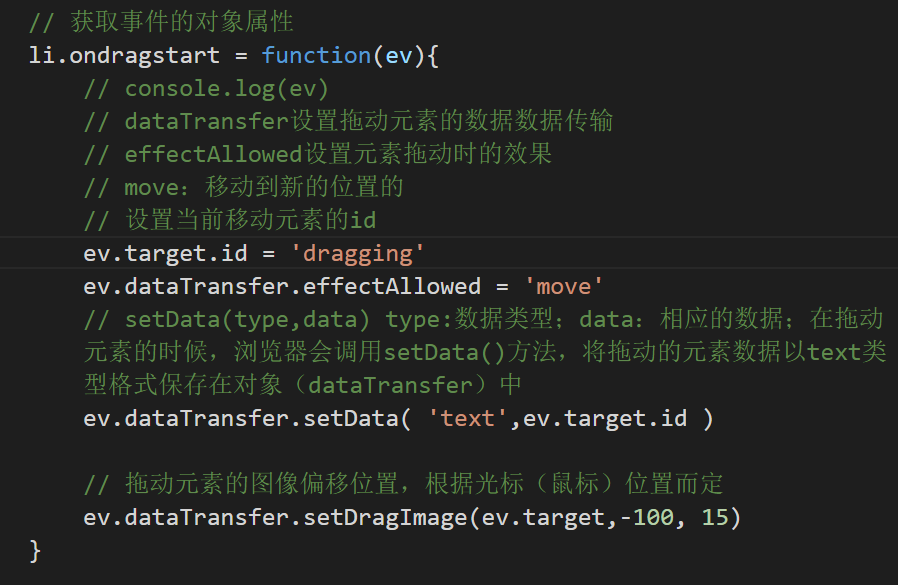


1. onresize:窗口大小改变触发的事件
2. onscroll：滚动条滚动的时候触发的事件
3. 拖放事件

执行拖放事件的前提条件：要在要执行拖放的标签内部写入：

draggable=” true”表示设置元素可以拖动；draggable=”false”表示设置元素不可拖动

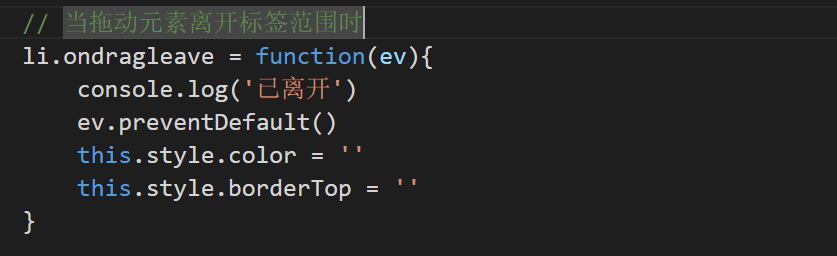
1. ondragstart：拖动开始，在函数内部的设置当前移动元素的id和拖动是的效果；dataTransfer设置拖动元素的数据数据传输；effectAllowed设置元素拖动时的效果；move：移动到新的位置的；setData(type,data) type:数据类型；data：相应的数据；在拖动元素的时候，浏览器会调用setData()方法，将拖动的元素数据以text类型格式保存在对象（dataTransfer）中

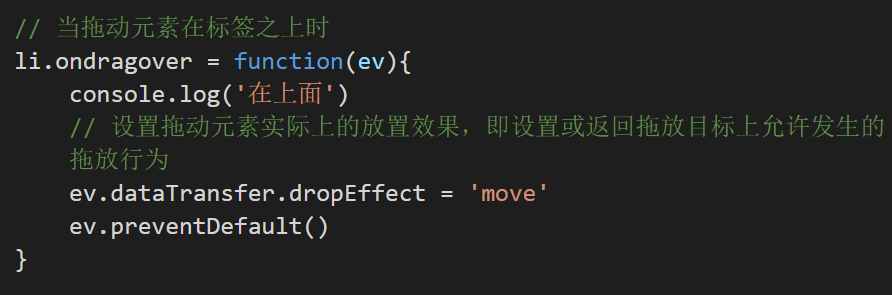


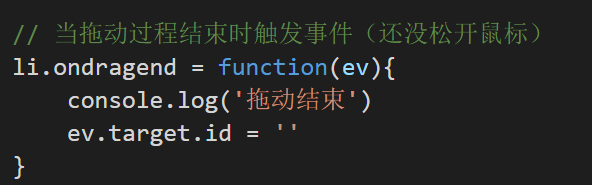
这是拖动开始必须进行的设置；

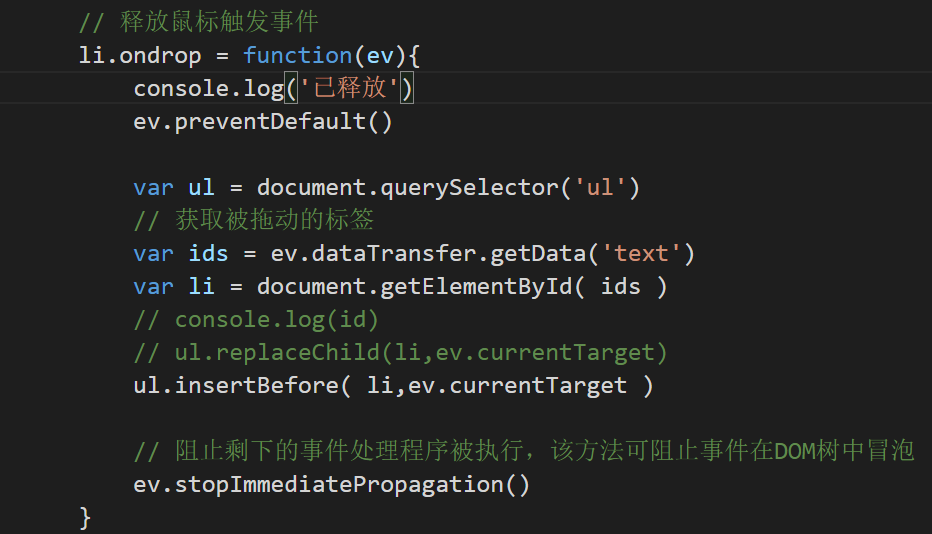
1. ondragenter:当拖动元素进入标签范围时触发事件
2. ondragleave：当拖动元素离开任意标签范围时触发事件
3. ondragover：当拖动元素在任意标签之上时触发事件
4. ondragend：当拖动过程结束时触发事件
5. ondrop：当鼠标松开元素被放下的时候











1. window对象
2. alert()弹出一个带有确定按钮的警告框
3. clearInterval（）清除由setInterval设置的timeout
4. clearTimeout（）清除setTimeout设置的timeout
5. setInterval（）每隔某段时间执行函数
6. setTimeout（）隔某段时间执行一次函数
7. Confirm（）弹出一个带有显示信息、确定和取消按钮的对话框
8. Prompt（‘提示信息’，‘输入框中的提示信息’）
9. Screen对象 返回的是浏览器客户端显示的屏幕的信息
10. availHeight：返回屏幕的高度（不包含windows的任务栏高度）
11. availWidth：返回屏幕的宽度（不包含windows的任务栏宽度）
12. Width、height：返回屏幕的总宽度和总高度
13. Location对象 location包含当前页面的url相关信息
14. assign（“地址”）载入一个新的文档
15. Reload（）重新载入当前文档
16. Replace（）用新的文档替换当前文档



