

21.4.6学习记录

一、鼠标事件

--onmousedown

--onmouseup

--onmousewheel 鼠标滚轮滚动事件，会在滚轮滚动时触发

--但是火狐不支持该属性，在火狐中要使用DOMMouseScroll来绑定事件，而且该事件需要通过addEventListener () 函数来绑定

注意：当我们拖拽一个网页中的内容时，浏览器会默认去搜索引擎中搜索内容，此时会导致拖拽功能的异常，这个是浏览器提供的默认行为，可以通过return false来取消默认行为；但是对IE8无效

二、键盘事件

注意：键盘事件一般绑定给一些可以获取到焦点的对象或是document

--onkeydown：按键被按下

--如果一直按着某个按键不松手，则事件会一直被触发

--当onkeydown连续触发时，第一次和第二次之间会间隔稍长，其他飞快

--在文本框中输入内容时onkeydown的默认行为，如果在此事件中取消了默认行为，则文本框输入内容不会显示

--onkeyup：按键被松开

方法/属性

--setCapture () 方法：此方法会使调用元素把**下一次所有的鼠标按下**相关的事件捕获到自己身上；这个方法只有IE支持，火狐不报错谷歌会报错

--e.wheelDelta () 方法： 可以获取鼠标滚轮滚动的方向，向上滚120，向下滚-120；这个值我们不看大小只看正负；

--但是火狐不支持，火狐中使用e.detail，向上滚-3，向下滚3

--preventDefault () 方法：使用addEventListener () 方法绑定响应函数，取消默认行为时不能使用return false；则可以使用此方法，但是ie8

--keyCode：

--除了keyCode，还有altkey、ctrlkey、shiftkey用来判断三个特殊键位按下与否

三、BOM 浏览器对象模型

--BOM可以使我们通过JS来操作浏览器

--在BOM中为我们提供了一组对象，用来完成对浏览器的操作

--**BOM对象**

--**Window**

--代表的是整个浏览器的窗口，同时window也是网页中的全局对象

方法：

--setInterval () ： 定时调用，将一个函数每隔一段函数执行一次

--参数1： 回调函数 参数2： 间隔时间单位是毫秒

--返回number类型的数据，作为定时器的唯一标识

--clearInterval () ： 关闭定时器；

--参数是有效标识，则停止；若无效，也不报错

--setTimeout () ： 延时调用，只会执行一次

--clearTimeout () ： 关闭延时调用

注意：延时调用和定时调用是可以互相代替的

--**Navigator**

--代表当前浏览器的信息，通过该对象可以来识别不同的浏览器
--由于历史原因，Navigator对象中大部分属性以不能帮助我们识别浏览器了

--一般我们只会使用userAgent来判断浏览器信息

--属性：（！！俩叹号转布尔值）

--appName:

--userAgent: **用户代理等价于浏览器，userAgent是一个字符串，这个字符串中包含有用来描述浏览器信息的内容；**

--IE11中已经将微软和IE相关的标识都已经去除了，所以此方法不能识别是否为IE了；此时我们可以通过一些浏览器中特有的对象，来判断浏览器的信息，比如：ActiveXObject

--Location

--代表当前浏览器的地址栏信息，通过Location可以获取地址栏信息或操作浏览器跳转页面

--直接打印location可以获得取到地址栏的信息，即完整路径

--直接将location属性修改为一个完整路径或相对路径，则页面会自动跳转到该路径，并且会生成响应的历史记录。

方法：

--assign ()：跳转到其他页面，作用和直接修改location一样

--reload ()：重新加载当前文档，作用和刷新按钮一样，参数传true表示强制清空缓存并刷新

--replace ()：使用新的页面替换当前页面，会跳转页面但不会生成历史记录

--History

--代表浏览器的历史记录，可以通过该对象来操作浏览器的历史记录

由于隐私原因，该对象不能获取到具体的历史记录，只能操作浏览器向前或向后翻页；而且该操作正在当次访问时有效

属性：

--length：可以获取当次访问的链接数量

方法：

--back ()：可以用来回退到上一个页面

--forward ()：可以跳转到下一个页面

--go ()：可以用来跳转到指定页面，参数为整数，表示跳转几个页面，正前负后

--Screen

--代表用户的屏幕信息，通过该对象可以获取到用户的显示器的相关信息

--这些BOM对象在浏览器中都是作为window对象的属性保存的，可以通过window对象来使用，也可以直接使用；

--ie浏览器，用js获取css属性值，如果没有设置则会返回默认值