# 事件分类

## **鼠标事件的种类**

鼠标事件指与鼠标相关的事件，继承了MouseEvent接口。具体的事件主要有以下一些。

* click：按下鼠标（通常是按下主按钮）时触发。
* dblclick：在同一个元素上双击鼠标时触发。
* mousedown：按下鼠标键时触发。
* mouseup：释放按下的鼠标键时触发。
* mousemove：当鼠标在一个节点内部移动时触发。当鼠标持续移动时，该事件会连续触发。
* mouseenter：鼠标进入一个节点时触发，进入子节点不会触发这个事件。
* mouseover：鼠标进入一个节点时触发，进入子节点会再一次触发这个事件。
* mouseout：鼠标离开一个节点时触发，离开父节点也会触发这个事件。
* mouseleave：鼠标离开一个节点时触发，离开父节点不会触发这个事件。
* contextmenu：按下鼠标右键时（上下文菜单出现前）触发，或者按下“上下文菜单键”时触发。
* wheel：滚动鼠标的滚轮时触发，该事件继承的是WheelEvent接口。

click事件指的是，用户在同一个位置先完成mousedown动作，再完成mouseup动作。因此，触发顺序是，mousedown首先触发，mouseup接着触发，click最后触发。

dblclick事件则会在mousedown、mouseup、click之后触发。

**补充：**

1.客户区坐标位置  
鼠标事件都是在浏览器视口中的特定位置上发生的。这个位置信息保存在事件对象的 clientX 和clientY 属性中。所有浏览器都支持这两个属性，它们的值表示事件发生时鼠标指针在视口中的水平和垂直坐标。 注意，这些值中不包括页面滚动的距离，因此这个位置并不表示鼠标在页面上的位置。

2. 页面坐标位置  
通过客户区坐标能够知道鼠标是在视口中什么位置发生的，而页面坐标通过事件对象的 pageX 和pageY 属性，能告诉你事件是在页面中的什么位置发生的。换句话说，这两个属性表示鼠标光标在页面中的位置，因此坐标是从页面本身而非视口的左边和顶边计算的。

**在页面没有滚动的情况下， pageX 和 pageY 的值与 clientX 和 clientY 的值相等。**

## **键盘事件的种类**

键盘事件由用户击打键盘触发，主要有keydown、keypress、keyup三个事件，它们都继承了KeyboardEvent接口。

* keydown：按下键盘时触发。
* keypress：按下有值的键时触发，即按下 Ctrl、Alt、Shift、Meta 这样无值的键，这个事件不会触发。对于有值的键，按下时先触发keydown事件，再触发这个事件。
* keyup：松开键盘时触发该事件。

如果用户一直按键不松开，就会连续触发键盘事件，触发的顺序如下。

1. keydown
2. keypress
3. keydown
4. keypress
5. ...（重复以上过程）
6. keyup

## **表单事件的种类**

### **input 事件**

input事件当<input>、<select>、<textarea>的值发生变化时触发。对于复选框（<input type=checkbox>）或单选框（<input type=radio>），用户改变选项时，也会触发这个事件。

input事件的一个特点，就是会连续触发，比如用户每按下一次按键，就会触发一次input事件。

input事件对象继承了InputEvent接口。

### **change 事件**

Change事件当<input>、<select>、<textarea>的值发生变化时触发。它与input事件的最大不同，就是不会连续触发，只有当全部修改完成时才会触发，另一方面input事件必然伴随change事件。具体来说，分成以下几种情况。

* 激活单选框（radio）或复选框（checkbox）时触发。
* 用户提交时触发。比如，从下列列表（select）完成选择，在日期或文件输入框完成选择。
* 当文本框或<textarea>元素的值发生改变，并且丧失焦点时触发。

### **reset 事件，submit 事件**

这两个事件发生在表单对象<form>上，而不是发生在表单的成员上。

reset事件当表单重置（所有表单成员变回默认值）时触发。

submit事件当表单数据向服务器提交时触发。注意，submit事件的发生对象是<form>元素，而不是<button>元素，因为提交的是表单，而不是按钮。

## 其他事件

### **load 事件，error 事件**

load事件在页面或某个资源加载成功时触发。注意，页面或资源从浏览器缓存加载，并不会触发load事件。

window.addEventListener('load', function(event) {

console.log('所有资源都加载完成');

});

error事件是在页面或资源加载失败时触发。

### **scroll 事件**

scroll事件在文档或文档元素滚动时触发，主要出现在用户拖动滚动条。

window.addEventListener('scroll', callback);

该事件会连续地大量触发，所以它的监听函数之中不应该有非常耗费计算的操作。

### **resize 事件**

resize事件在改变浏览器窗口大小时触发，主要发生在window对象上面。

var resizeMethod = function () {

if (document.body.clientWidth < 768) {

console.log('移动设备的视口');

}

};

window.addEventListener('resize', resizeMethod, true);

### 焦点事件

焦点事件发生在元素节点和document对象上面，与获得或失去焦点相关。它主要包括以下四个事件。

* focus：元素节点获得焦点后触发，该事件不会冒泡。
* blur：元素节点失去焦点后触发，该事件不会冒泡。