**CSS模块**

style样式

css(styleName): 根据样式名得到对应的值

css(styleName, value): 设置一个样式

css({多个样式对}): 设置多个样式

位置坐标

offset(): 读/写当前元素始终到body的距离(原点是页面左上角)

获取时返回值一个对象

设置时传入一个对象 如果没定位属性则会则自动添加

position():

返回的是一个对象 读当前元素到带有定位的最近的父元素的坐标(原点是父元素左上角)

只能获取不能设置‘

相当于原生js的offsetLeft offsetTop

scrollTop()/scrollLeft(): 读/写元素/页面的滚动条坐标

尺寸

width()/height(): width/height 传参是设置 不传参是获取

innerWidth()/innerHeight(): width + padding

outerWidth()/outerHeight(): width + padding + border

**筛选模块**

过滤

在jQuery对象内部的元素中找出部分匹配的元素, 并封装成新的jQuery对象返回

first()

last()

eq(index)

filter(selector): 对当前元素提要求

not(selector): 对当前元素提要求, 并取反

has(selector): 对子孙元素提要求

查找

查找jQuery对象内部的元素的子孙/兄弟/父母元素, 并封装成新的jQuery对象返回

children(selector): 子元素

find(selector): 后代元素

preAll(selector): 前的所有兄弟

siblings(selector): 所有兄弟

parent(): 父元素

**文档处理(CUD)模块**

增加

append() / appendTo(): 插入后部

preppend() / preppendTo(): 插入前部

before(): 插到前面

after(): 插到后面

删除

text()=> innerText

html()=> innerHTML

remove(): 删除自己

empty(): 掏空自己(自己还在)

更新

replaceWith()

克隆

clone()

只有一个参数时是指示事件处理函数(包括子元素的)是否会被复制 1.5以上默认为false

两个参数的话第一个参数指示当前元素的事件处理函数是否会被复制

第二个参数指示事件处理程序和克隆的元素的所有子元素的数据是否被复制

事件模块

绑定事件

eventName(function(){}) 简单事件

bind事件 bind("事件1 事件2 事件3",fn)

推荐使用 on

$(祖先元素).on('eventName',子元素, function(){})

$(目标元素).on('eventName', function(){})

常用: click, mouseenter/mouseleave mouseover/mouseout focus/blur

hover(function(){}, function(){})

jQuery的事件对象特有属性 e.originalEvent 原生js的事件对象

解绑事件

off('eventName') 如果不传参则会全部解绑

事件委托

理解: 将子元素的事件委托给父辈元素处理

事件监听绑定在父元素上, 但事件发生在子元素上

事件会冒泡到父元素

但最终调用的事件回调函数的是子元素: event.target

好处

新增的元素没有事件监听

减少监听的数量(n==>1)

编码

delegate(selector, 'eventName1 eventName2', function(event){}) // 回调函数中的this是子元素

undelegate('eventName')

事件坐标

event.offsetX: 原点是当前元素左上角

event.clientX: 原点是窗口左上角

event.pageX: 原点是页面左上角

事件相关

停止事件冒泡: event.stopPropagation()

阻止事件的默认行为: event.preventDefault()