目录

[第一节 jQuery概述 1](#_Toc10045)

[1.JavaScript库 1](#_Toc26398)

[2.常见的JavaScript库 1](#_Toc17616)

[3.jQuery概述 1](#_Toc30571)

[4.jQuery的优点 2](#_Toc20856)

[5.jQuery的基本使用 2](#_Toc12004)

[1.1jQuery的下载 2](#_Toc20565)

[1.2 jQuery的使用步骤 2](#_Toc2456)

[1.3 jQuery的入口函数 3](#_Toc25851)

[1.4 jQuery的顶级对象$ 4](#_Toc23938)

[1.5 jQuery对象和DOM对象 4](#_Toc13130)

[第二节 jQuery常用API 6](#_Toc5516)

[1.jQuery选择器 6](#_Toc7910)

[1.1 jQuery基础选择器 6](#_Toc8610)

[1.2 jQuery层级选择器 6](#_Toc28545)

[1.3 隐式迭代 6](#_Toc11356)

[1.4jQuery筛选选择器 7](#_Toc18514)

[1.5 jQuery筛选方法 7](#_Toc27406)

[1.6 jQuery里面的排他思想 8](#_Toc5232)

[1.7 jQuery链式编程 9](#_Toc17211)

[2.jQuery样式操作 10](#_Toc22256)

[1.1操作CSS方法 10](#_Toc6338)

[1.2 设置类样式方法 11](#_Toc1261)

[1.3 jQuery类操作与className的区别 12](#_Toc29238)

[3.jQuery效果 12](#_Toc236)

[1.1 显示隐藏效果 13](#_Toc19782)

[1.2 滑动效果 14](#_Toc18998)

[1.3 事件切换 16](#_Toc2850)

[1.4 动画队列及停止排队方法 16](#_Toc18456)

[1.5 淡入淡出效果 17](#_Toc23866)

[1.6 自定义动画 animate 19](#_Toc6196)

[4.jQuery属性操作 20](#_Toc7378)

[1.1设置或获取元素固有属性值prop（ ） 20](#_Toc16454)

[1.2设置或获取元素自定义属性值attr（ ） 20](#_Toc3122)

[1.3 数据缓存 data（ ） 21](#_Toc26369)

[5.jQuery内容文本值 22](#_Toc13965)

[6. jQuery元素操作 23](#_Toc5365)

[1.1遍历元素 23](#_Toc3013)

[1.2创建元素 24](#_Toc17831)

[1.3添加元素 24](#_Toc32702)

[1.4删除元素 25](#_Toc4657)

[7.jQuery尺寸、位置操作 26](#_Toc28897)

[1.1jQuery尺寸 26](#_Toc18987)

[1.2jQuery位置 26](#_Toc30702)

[第三节 jQuery事件 28](#_Toc13492)

[1.jQuery事件注册 28](#_Toc4961)

[2.jQuery事件处理 28](#_Toc17098)

[1.1事件处理on( )绑定事件 28](#_Toc29424)

[1.2 事件处理off( )解绑事件 30](#_Toc1361)

[1.3 自动触发事件trigger( ) 31](#_Toc3042)

[3.jQuery事件对象 32](#_Toc3202)

[第四节 jQuery其他方法 33](#_Toc9563)

[1.jQuery拷贝对象 33](#_Toc17299)

[2.多库共存 34](#_Toc17362)

[3.jQuery插件 35](#_Toc13012)

**第一节 jQuery概述**

**1.JavaScript库**

·仓库:可以把很多东西放到这个仓库里面。找东西只需要到仓库里面查找到就可以了。

·JavaScript库:即 library，是一个封装好的特定的集合(方法和函数)。从封装一大堆函数的角度理解库，就是在这个库中，封装了很多预先定义好的函数在里面，比如动画animate、hide、show，比如获取元素等。

·简单理解︰就是一个JS文件，里面对我们原生js代码进行了封装，存放到里面。这样我们可以快速高效的使用这些封装好的功能了。

·比如jQuery ，就是为了快速方便的操作DOM，里面基本都是函数(方法)。

**2.常见的JavaScript库**

·jQuery

·Prototype

·YUI

·Dojo

·Ext JS

·移动端的zepto

这些库都是对原生JavaScript的封装，内部都是用JavaScript实现的，我们主要学习的是jQuery。

**3.jQuery概述**

·jQuery是一个快速、简洁的JavaScript库，其设计的宗旨是“write Less , Do More”，即倡导写更少的代码，做更多的事情。

·j就是JavaScript ; Query查询;意思就是查询js，把js中的DOM操作做了封装，我们可以快速的查询使用里面的功能。

·jQuery封装了JavaScript常用的功能代码，优化了DOM操作、事件处理、动画设计和Ajax交互。学习jQuery本质:就是学习调用这些函数（方法)。

·jQuery出现的目的是加快前端人员的开发速度，我们可以非常方便的调用和使用它，从而提高开发效率。

**4.jQuery的优点**

·轻量级。核心文件才几十kb，不会影响页面加载速度

·跨浏览器兼容。基本兼容了现在主流的浏览器

·链式编程、隐式迭代

·对事件、样式、动画支持，大大简化了DOM操作

·支持插件扩展开发。有着丰富的第三方的插件，例如∶树形菜单、日期控件、轮播图等

·免费、开源

**5.jQuery的基本使用**

**1.1jQuery的下载**

官网地址: https://jquery.com/

版本︰

·1x:兼容IE 678等低版本浏览器，官网不再更新

·2x :不兼容IE 678等低版本浏览器，官网不再更新

·3x:不兼容IE 678等低版本浏览器，是官方主要更新维护的版本

各个版本的下载: https://code.jquery.com/

**1.2 jQuery的使用步骤**

1.引入jQuery文件

2.使用即可

**1.3 jQuery的入口函数**

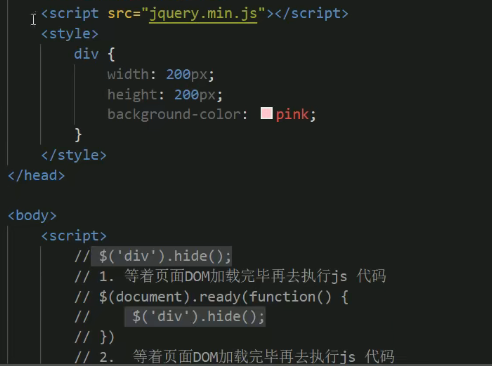


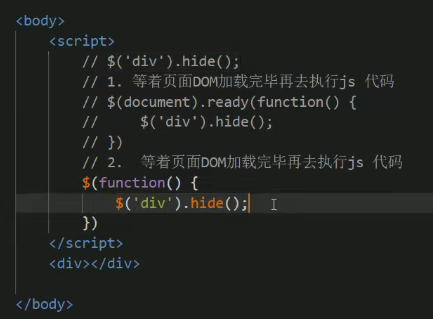
1. 等看DOM结构渲染完毕即可执行内部代码，不必等到所有外部资源加载完成，jQuery帮我们完成了封装。

2．相当于原生JS中的 DOMContentLoaded。

3．不同于原生JS中的 load事件是等页面文档、外部的JS文件、CSS文件、图片加载完毕才执行内部代码。

4．更推荐使用第一种方式。





**1.4 jQuery的顶级对象$**

1.$是jQuery的别称，在代码中可以使用jQuery代替$，但一般为了方便，通常都直接使用$。

2.$是jQuery的顶级对象，相当于原生JavaScript中的window。把元素利用$包装成jQuery对象，就可以调用jQuery的方法。

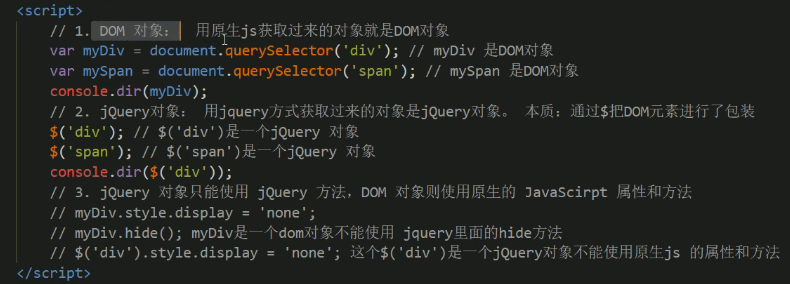


**1.5 jQuery对象和DOM对象**

1.用原生JS获取来的对象就是 DOM对象

2.jQuery方法获取的元素就是jQuery对象。

3.jQuery对象本质是∶利用$对DOM对象包装后产生的对象（伪数组形式存储〕。



DOM对象与jQuery对象之间是可以相互转换的。

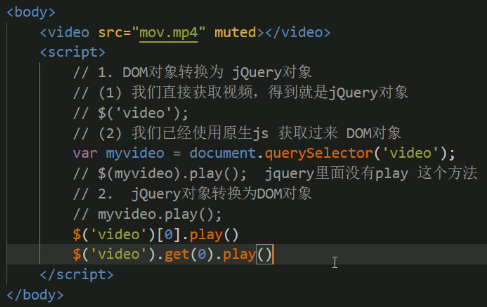
因为原生JS比jQuery更大，原生的一些属性和方法jQuery没有给我们封装.要想使用这些属性和方法需要把jQuery对象转换为DOM对象才能使用。

1.DOM对象转换为jQuery对象︰$(DOM对象)

4RC_M}V[`FKNY)8E~U)T]CI

2.jQuery对象转换为DOM对象（两种方式)

![IQXL)%B](1FT]`Q{_{F{`[0](data:image/png;base64,)



1. **jQuery常用API**

**1.jQuery选择器**

**1.1 jQuery基础选择器**

原生JS获取元素方式很多，很杂，而且兼容性情况不一致，因此jQuery给我们做了封装，使获取元素统一标准。

S}FW95PYA96AO{ZA[2][MVY



**1.2** **jQuery层级选择器**



**1.3** **隐式迭代**

遍历内部 DOM元素（伪数组形式存储）的过程就叫做隐式迭代。

简单理解∶给匹配到的所有元素进行循环遍历，执行相应的方法，而不用我们再进行循环，简化我们的操作，方便我们调用。

·jQuery设置样式

H~PA{}LTAWZ%TO6~M~RV({C

**1.4jQuery筛选选择器**





**1.5 jQuery筛选方法**



重点记住: parent( )、children( )、 find( )、 siblings( )、 eq( )





**1.6 jQuery里面的排他思想**

想要多选一的效果，排他思想:当前元素设置样式，其余的兄弟元素清除样式。



**1.7 jQuery链式编程**

链式编程是为了节省代码量，看起来更优雅。

{~1Z5EYI5V6V)5D9NAZ@5BY

使用链式编程一定注意是哪个对象执行样式



**2.jQuery样式操作**

**1.1操作CSS方法**

jQuery可以使用CSS方法来修改简单元素样式;也可以操作类，修改多个样式。

1.参数只写属性名，则是返回属性值

Z9S@CAZC5V(CJH06602`$)X

2．参数是属性名，属性值，逗号分隔，是设置一组样式，属性必须加引号，值如果是数字可以不用跟单位和引号

WN(~2C%FKUOCCS([)MN(%SQ

3．参数可以是对象形式，方便设置多组样式。属性名和属性值用冒号隔开，属性可以不用加引号

39UPQ5IUW63}~G1`SQV179C



**1.2 设置类样式方法**

作用等同于以前的classList，可以操作类样式，注意操作类里面的参数不要加点。

1.添加类

7123SA6@D@UIMLYA]]9PBK2

2.移除类

ZGN_FIE[G{Z7VKX{ONV$F%3

3.切换类（有这个类就删除，没这个类就加上）

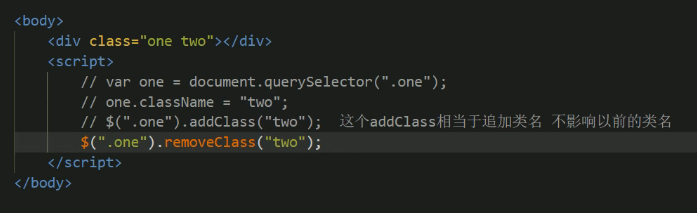
HOC9)2L0GI]62{CG2KO)L5Q



**1.3 jQuery类操作与className的区别**

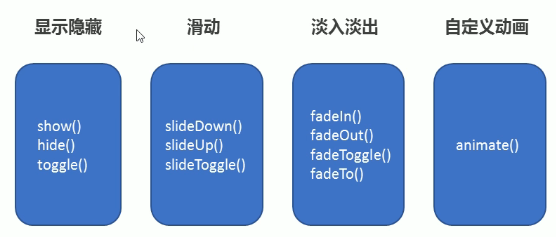
原生JS中className会覆盖元素原先里面的类名。

jQuery里面类操作只是对指定类进行操作，不影响原先的类名。



**3.jQuery效果**

jQuery给我们封装了很多动画效果，最为常见的如下∶



**1.1 显示隐藏效果**

①显示语法规范

}$}P2}5CNUVSPERN@SI3EXS

②显示参数

( 1）参数都可以省略，无动画直接显示。

( 2 ) speed :三种预定速度之一的字符串(“slow”,"normar”,or “fast”)或表示动画时长的毫秒数值(如∶1000)。

( 3 ) easing : (Optional)用来指定切换效果，默认是“swing”，可用参数“linear"。

( 4 ) fn:回调函数，在动画完成时执行的函数，每个元素执行一次.

①隐藏语法规范

L7V(RH0(83B90%I0YTB@RCM

②隐藏参数

( 1)参数都可以省略，无动画直接显示。

( 2 ) speed :三种预定速度之一的字符串(“slow”，"normal",or“fast”)或表示动画时长的毫秒数值(如∶1000)。

( 3 ) easing :(Optional)用来指定切换效果，默认是“swing”，可用参数“linear".

( 4 ) fn:回调函数，在动画完成时执行的函数，每个元素执行一次。

②切换参数

Y8X8ZCZ2F4BL%P0)PD8VN4F

( 1 )参数都可以省略，无动画直接显示。

( 2 ) speed :三种预定速度之一的字符串( “slow”，"normal”,or“fast”)或表示动画时长的毫秒数值(如∶1000)。

( 3 ) easing : (Optional)用来指定切换效果，默认是“swing”，可用参数“linear”。

( 4 ) fn:回调函数，在动画完成时执行的函数，每个元素执行一次.



**1.2 滑动效果**

①下滑效果语法规范

35Q8I@[F[MS{6)C0@2~VINX

②下滑效果参数

( 1)参数都可以省略。

( 2 ) speed:三种预定速度之一的字符串(“slow”, "normal",or “fast”)或表示动画时长的亳秒数值(如∶1000)。

( 3 ) easing:(Optional)用来指定切换效果，默认是“swing”，可用参数“linear” 。

( 4 ) fn: 回调函数，在动画完成时执行的函数，每个元素执行一次。

①上滑效果语法规范

A5UI2}8B$(NRE}~%%R%P%F7

②上滑效果参数

( 1)参数都可以省略。

( 2 ) speed:三种预定速度之一的字符串(“slow”, "normal",or “fast”)或表示动画时长的亳秒数值(如∶1000)。

( 3 ) easing:(Optional)用来指定切换效果，默认是“swing”，可用参数“linear” 。

( 4 ) fn: 回调函数，在动画完成时执行的函数，每个元素执行一次。

①滑动切换效果语法规范

TXRO8}812S7][VZFDAU@F]9

②滑动切换效果参数

( 1)参数都可以省略。

( 2 ) speed:三种预定速度之一的字符串(“slow”, "normal",or “fast”)或表示动画时长的亳秒数值(如∶1000)。

( 3 ) easing:(Optional)用来指定切换效果，默认是“swing”，可用参数“linear” 。

( 4 ) fn: 回调函数，在动画完成时执行的函数，每个元素执行一次。

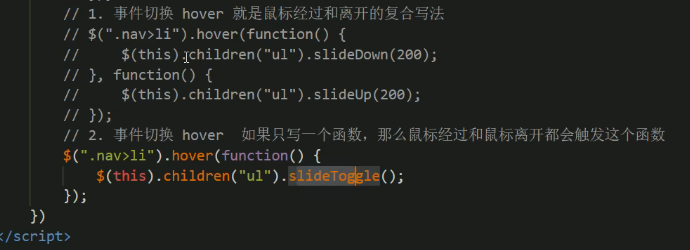


**1.3 事件切换**

~6CN25$0DK%CWKDJLK2ZC0N

( 1 ) over:鼠标移到元素上要触发的函数(相当于mouseenter )

( 2 ) out鼠标移出元紊要触发的函数(相当于mouseleave )



**1.4 动画队列及停止排队方法**

1.动画或效果队列

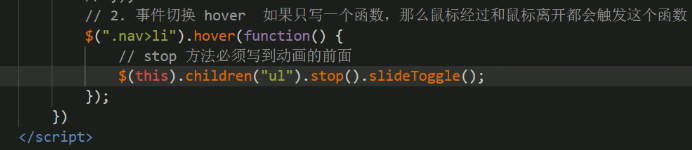
动画或者效果一旦触发就会执行，如果多次触发，就造成多个动画或者效果排队执行。

1. 停止排队

(D@WR(5)@(H)%~3[GN58W4Y

(1 ) stop( )方法用于停止动画或效果。

(2 ) 注意:stop( )写到动画或者效果的前面，相当于停止结束上一次的动画。



**1.5 淡入淡出效果**

①淡入效果语法规范

1{7}0LB0[RE67~7CF()157P

②淡入效果参数

( 1）参数都可以省略。

( 2 ) speed :三种预定速度之一的字符串(“slow","normal",or“fast”)或表示动画时长的亳秒数值(如:1000)。

( 3 ) easing : (Optional)用来指定切换效果，默认是“swing”，可用参数“linear”。

( 4 ) fn:回调函数，在动画完成时执行的函数，每个元素执行一次。

①淡出效果语法规范

9QH6W1JL_UVP}`DAD@0`ASO

②淡出效果参数

( 1）参数都可以省略。

( 2 ) speed :三种预定速度之一的字符串(“slow","normal",or“fast”)或表示动画时长的亳秒数值(如:1000)。

( 3 ) easing : (Optional)用来指定切换效果，默认是“swing”，可用参数“linear”。

( 4 ) fn:回调函数，在动画完成时执行的函数，每个元素执行一次。

①淡入淡出效果语法规范

(MCWZASO$[~X5DPH3@IXKAF

②淡入淡出效果参数

( 1）参数都可以省略。

( 2 ) speed :三种预定速度之一的字符串(“slow","normal",or“fast”)或表示动画时长的亳秒数值(如:1000)。

( 3 ) easing : (Optional)用来指定切换效果，默认是“swing”，可用参数“linear”。

( 4 ) fn:回调函数，在动画完成时执行的函数，每个元素执行一次。

①渐进方式调整到指定的不透明度

6RWFCP~4PWXU(6LRTDQR[12

②效果参数

( 1 ) opacity透明度必须写，取值0~1之间。

(2 ) speed :三种预定速度之一的字符串(“slow”, "normal" ,or“fast”)或表示动画时长的毫秒数值(如∶1000)。必须写。

( 3 ) easing :(Optional)用来指定切换效果，默认是“swing”，可用参数“linear"。

( 4 ) fn:回调函数，在动画完成时执行的函数，每个元素执行一次。



**1.6 自定义动画 animate**

①语法

FQSEUU84YRS{LC%$N8RCK_X

②参数

( 1 ) **params:想要更改的样式属性，以对象形式传递，必须写**。属性名可以不用带引号，如果是复合属性则需要采取驼峰命名法borderLeft。其余参数都可以省略。

( 2 ) speed :三种预定速度之一的字符串(“slow”, “normal”,or“fast”)或表示动画时长的亳秒数值(如:1000)。

( 3 ) easing :(Optional)用来指定切换效果，默认是“swing”，可用参数“linear”。

( 4 ) fn:回调函数，在动画完成时执行的函数，每个元素执行一次。



**4.jQuery属性操作**

**1.1设置或获取元素固有属性值prop（ ）**

所谓元素固有属性就是元素本身自带的属性，比如<a>元素里面的href，比如<input>元素里面的type。

①获取属性值语法

MX]3LIAZRBL5~S`$Z6U9$)C

②设置属性语法

LN}4JDT7C3K0YC3N0M[Z2HX

**1.2设置或获取元素自定义属性值attr（ ）**

用户自己给元素添加的属性，我们称为自定义属性。比如给div添加index = “1”。该方法也可以获取H5自定义属性。

①获取属性语法

%AT2%@1(V16R}2))2}_SGE8

②设置属性语法

_8K)W6I238J8DK0[I)O}X%D



**1.3 数据缓存 data（ ）**

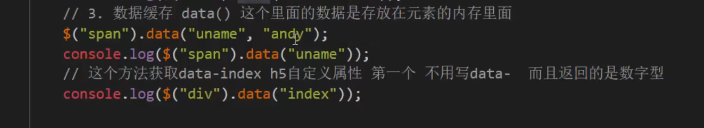
data( )方法可以在指定的元素上存取数据，并不会修改DOM元素结构。一旦页面刷新，之前存放的数据都将被移除。同时，还可以读取HTML5自定义属性data-index ，得到的是数字型。

①附加数据语法

%3P]69(ILQ{MI`C0ZWA_`VY

②获取数据语法

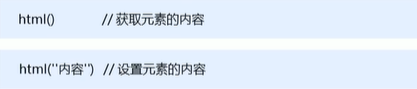
S21~70QTZNO6YB(WRW]%@)G



**5.jQuery内容文本值**

主要针对元素的内容还有表单的值操作。

①普通元素内容html（ ）(相当于原生innerHTML)



②普通元素文本内容text（ ）(相当与原生innerText)



③表单的值val（ ）(相当于原生value)

val( ) //获取表单的内容

val(“内容”) //设置表单的内容



1. **jQuery元素操作**

主要是遍历、创建、添加、删除元素操作。

**1.1遍历元素**

jQuery隐式迭代是对同一类元素做了同样的操作。如果想要给同一类元素做不同操作，就需要用到遍历。

语法1：

XK1%31X}_JZ`G6%XS%@~0C0

1.each( )方法遍历匹配的每一个元素。主要用DOM处理。each每一个

⒉.里面的回调函数有2个参数:index是每个元素的索引号; demEle是每个DOM元素对象，不是jquery对象

3.所以要想使用jquery方法，需要给这个dom元素转换为jquery对象$(domEle)



语法2：

5C)TICN}H(P_%OCQL{HE87E

1. $.each( )方法可用于遍历任何对象。主要用于数据处理，比如数组，对象

2.里面的函数有2个参数:index是每个元素的索引号;element遍历内容



**1.2创建元素**

语法：

DI9)}9MZ(W`QCW`IV@@)BLQ

动态的创建了一个<li>

**1.3添加元素**

①内部添加

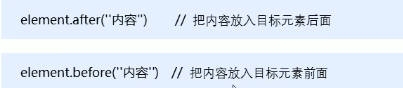
NRN[$[0BAT@9SXU`5O0S$04

把内容放入匹配元素内部最后面，类似原生appendChild

KR`UUNSB9(82EBSCO7D1IBH

把内容放入匹配元素内部最前面

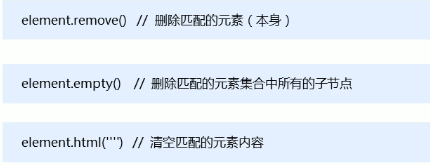
②外部添加

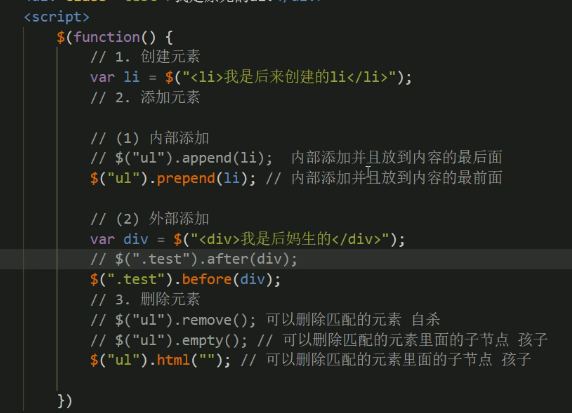


**内部添加元素，生成之后，它们是父子关系。**

**外部添加元素，生成之后，他们是兄弟关系。**

**1.4删除元素**





**7.jQuery尺寸、位置操作**

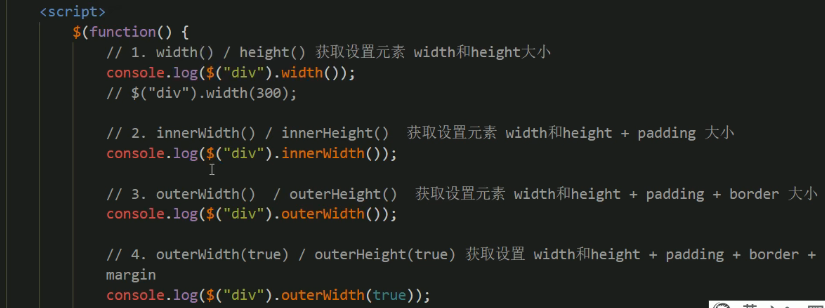
**1.1jQuery尺寸**



·以上参数为空，则是获取相应值，返回的是数字型。

·如果参数为数字，则是修改相应值。

·参数可以不必写单位。



**1.2jQuery位置**

位置主要有三个:offset（）、position（）、 scrollTop（）/scrollLeft（）

① offset( )设置或获取元素偏移

·offset( )方法设置或返回被选元素相对于文档的偏移坐标，跟父级没有关系。

·该方法有2个属性left、top。offset().top用于获取距离文档顶部的距离，offset( ).left用于获取距离文档左侧的距离。

② position( )获取元素偏移

position( )方法用于返回被选元素相对于带有定位的父级偏移坐标，如果父级都没有定位，则以文档为准。这个方法只能获取不能设置偏移。



③scrollTop( )/scrollLeft( )设置或获取元素被卷去的头部和左侧

scrollTop( )方法设置或返回被选元素被卷去的头部。

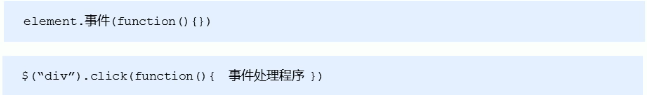


**第三节 jQuery事件**

**1.jQuery事件注册**

①单个事件注册

语法：



其他事件和原生基本一致。

比如mouseover、mouseout、blur、focus、change、keydown、keyup、resize、scroll等。

**2.jQuery事件处理**

**1.1事件处理on( )绑定事件**

·on( )方法在匹配元素上绑定一个或多个事件的事件处理函数

·语法：

(OYP_RGZ1}4FD}GO]4RF6(1

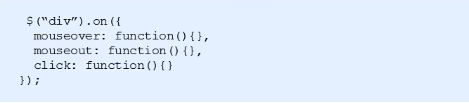
1.events:一个或多个用空格分隔的事件类型，如"click"或"keydown"。

2. selector:元素的子元素选择器。

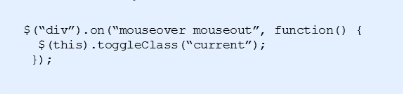
3.fn:回调函数，即绑定在元素身上的侦听函数。

**·On( )方法优势1：**

可以绑定多个事件，多个处理事件处理程序。



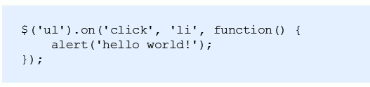
如果事件处理程序相同





**·On( )方法优势2：**

可以事件委派操作。事件委派的定义就是，把原来加给子元素身上的事件绑定在父元素身上，就是把事件委派给父元素。





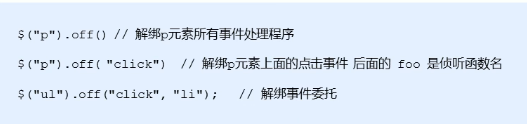
**·On( )方法优势3：**

动态创建的元素，click()没有办法绑定事件，on()可以给动态生成的元素绑定事件。



**1.2 事件处理off( )解绑事件**

off( )方法可以移除通过on( )方法添加的事件处理程序。



如果有的事件只想触发一次，可以使用one( )来绑定事件。



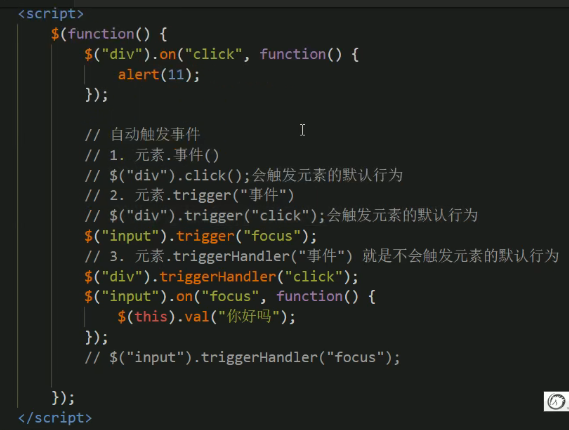
**1.3 自动触发事件trigger( )**

有些事件希望自动触发,比如轮播图自动播放功能跟点击右侧按钮一致。可以利用定时器自动触发右侧按钮点击事件，不必鼠标点击触发。

C)Q%OM6Y]`~_XMB~FS[~XSF

DB0Z2~U5H5}%{G44%726DP8

GR5BR6@}VV[SDFLMZJGO%IU



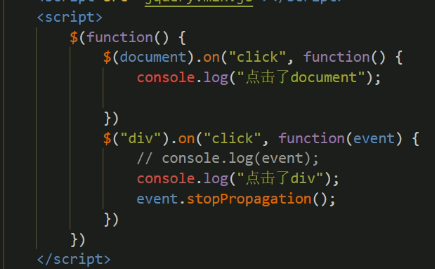
**3.jQuery事件对象**

事件被触发，就会有事件对象的产生。

SN)$R9X)X}J[~[{)%O9P])P

阻止默认行为: event.preventDefault( )或者return false

阻止冒泡: event.stopPropagation( )



**第四节 jQuery其他方法**

**1.jQuery拷贝对象**

如果想要把某个对象拷贝（合并）给另外一个对象使用，此时可以使用$.extend ( )方法。

语法：

$OTFRI(U1A50V~@3]E}3QI2

1.deep:如果设为true为深拷贝，默认为false 浅拷贝。

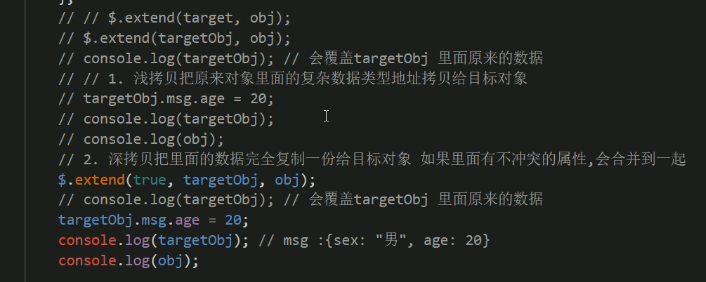
2.target:要拷贝的目标对象。

3.object1:待拷贝到第一个对象的对象。

4.objectN:待拷贝到第N个对象的对象。

5.浅拷贝是把被拷贝的对象复杂数据类型中的地址拷贝给目标对象，修改目标对象会影响被拷贝对象。

6.深拷贝，前面加true，完全克隆(拷贝的对象,而不是地址)，修改目标对象不会影响被拷贝对象。



**2.多库共存**

**·问题概述:**

jQuery使用$作为标示符，随着jQuery的流行,其他JS库也会用这$作为标识符，这样一起使用会引起冲突。

**·客观需求:**

需要一个解决方案，让jQuery和其他的JS库不存在冲突，可以同时存在，这就叫做多库共存。

**·jQuery解决方案∶**

1.把里面的$符号统一改为jQuery。比如jQuery("div")

2. jQuery变量规定新的名称:$.noConflict( ) var x = $.noConflict( );

![KHYXA~T4%1GC676S%](643N](data:image/png;base64,)

**3.jQuery插件**

jQuery功能比较有限，想要更复杂的特效效果，可以借助于jQuery插件完成。

注意:这些插件也是依赖于jQuery来完成的，所以必须要先引入jQuery文件，因此也称为jQuery插件。

**·jQuery插件常用的网站:**

1. jQuery插件库 <http://www.jq22.com/>
2. jQuery之家 <http://www.htmleaf.com/>

**·jQuery插件使用步骤:**

1.引入相关文件。( jQuery文件和插件文件)

2．复制相关html、css、js(调用插件)。

**·jQuery插件演示∶**

1.瀑布流

⒉.图片懒加载（图片使用延迟加载在可提高网页下载速度。它也能帮助减轻服务器负载)，当我们页面滑动到可视区域，再显示图片。我们使用jQuery插件库EasyLazyload。注意，此时的js引入文件和js调用必须写到DOM元素(图片）最后面。

3.全屏滚动( fullpage.js )

gitHub : https://github.com/alvarotrigo/fullPage.js

中文翻译网站: http://www.dowebok.com/demo/2014/77/

**·bootstrap JS插件:**

bootstrap框架也是依赖于jQuery开发的，因此里面的js插件使用，也必须引入jQuery文件。