

WEB前端

jQuery基础2





鼠标键盘事件



标签的添加删除



属性的获取设置



Class的添加删除



jQuery编写图片轮播

该颜色字体为演示案例

01

鼠标键盘事件



鼠标事件



翡翠教育
EMERALD EDUCATION

事件名称	描述	举例说明
mousedown	当鼠标指针移动到元素上方，并按下鼠标按键时，会发生 mousedown 事件	<pre>\$("#button").mousedown(function(){ \$("#p").slideToggle(); })</pre>
mouseup	当在元素上放松鼠标按钮时，会发生 mouseup 事件	<pre>\$("#button").mouseup(function(){ \$("#p").slideToggle(); })</pre>
mouseenter	当鼠标指针穿过元素时，会发生 mouseenter 事件	<pre>\$("#p").mouseenter(function(){ \$("#p").css("background-color","yellow"); });</pre>

鼠标事件



鼠标事件

事件名称	描述	举例说明
mousemove	当鼠标指针在指定的元素中移动时，就会发生 mousemove 事件	<code>\$("#button").mousemove(function(){</code> <code>})</code>
<u>event.pageX</u>	pageX() 属性是鼠标指针的位置，相对于文档的左边缘	<code>\$(document).mousemove(function(e){ \$("#span").text("X: " + e.pageX);</code> <code>});</code>
event.pageY	pageY() 属性是鼠标指针的位置，相对于文档的上边缘	<code>\$(document).mousemove(function(e){ \$("#span").text("Y: " + e.pageY);</code> <code>});</code>

相似事件的区别



鼠标事件

事件名称	描述	举例说明
mouseover	当鼠标指针位于元素上方时，会发生mouseover 事件	<code>\$("button").mouseover(function(){</code> <code>});</code>
mouseout	当鼠标指针从元素上移开时，发生mouseout 事件	<code>\$("p").mouseout(function(){</code> <code>});</code>
mouseleave	当鼠标指针离开元素时，会发生mouseleave 事件	<code>\$("p").mouseleave(function(){</code> <code>});</code>
<code>event.preventDefault();</code>	阻止元素发生默认的行为	<code>\$("a").click(function(event){ event.preventDefault(); });</code>



鼠标事件的案例1

百叶窗效果：
鼠标经过图片变成鼠标经过样式

原始样式



鼠标经过样式





鼠标事件的案例2

鼠标跟随效果：
鼠标经过添加一层蒙层，并添加一张图片随蒙层移动而移动。

原始样式



鼠标经过样式





键盘事件

事件名称	描述	举例说明
keydown	当按钮被按下时，发生 keydown 事件	<pre>\$("#input").keydown(function(){ \$("#input").css("color","#FFFFCC"); });</pre>
keypress	当按钮被按下时，会发生该事件。它发生在当前获得焦点的元素上	<pre>\$("#input").keypress(function(){ \$("#span").text(i+=1); });</pre>
keyup	当按钮被松开时，发生 keyup 事件。它发生在当前获得焦点的元素上。	<pre>\$("#input").keyup(function(){ \$("#input").css("color","#D6D6FF"); });</pre>

02

标签的添加删除



标签的添加删除

添加元素：

- `append()` - 在被选元素的尾部插入内容
- `appendTo()` - 在被选元素的结尾（仍然在内部）插入指定内容

例：

```
$("p").append( " World." );
```

如p标签的原文本为: "Hello"
则现文本为: Hello World.

等同于

```
$("<b>World.</b>  
>").appendTo("p")
```



标签的添加删除

添加元素

- `prepend()` - 在被选元素的开头插入内容
- `prependTo()` - 在被选元素的开头（仍位于内部）插入指定内容

例：

```
$("p").prepend("你好，");
```

如p标签的原文本为：“李明”
则现文本为：你好，李明

等同于

```
$("你好  
").prependTo("p");
```




标签的添加删除

添加元素

append_prepend

append()和prepend()方法添加若干新元素
例：

```
function appendText() {  
  var txt1= "<p>Text.</p>" ; // 以 HTML 创建新元素  
  var txt2=$( "<p></p>" ).text( "Text." ); // 以 jQuery 创建新元素  
  var txt3=document.createElement( "p" ); // 以 DOM 创建新元素  
  txt3.innerHTML="Text."  
  $("p").append(txt1,txt2,txt3); // 追加新元素 }
```



标签的添加删除

after_before

添加元素

- after() – 在被选元素之后插入内容
- before – 在被选元素之前插入内容

用法与append()和prepend()相同

after()和before() 方法添加若干新元素



标签的添加删除

例：

```
function afterText() {  
    var txt1= "<b>I </b>" ; // 以 HTML 创建新元素  
    var txt2=$( "<i> </i>" ).text( "love " ); // 通过 jQuery 创建新元素  
    var txt3=document.createElement( "big" ); // 通过 DOM 创建新元素  
    txt3.innerHTML="jQuery!";  
    $("img").after(txt1,txt2,txt3); // 在 img 之后插入新元素 }
```



标签的添加删除

append()和prepend()与after()和before()区别：

append() & prepend()是在元素内插入内容（该内容变成该元素的子元素或节点）

after() & before()是在元素的外面插入内容（其内容变成元素的兄弟节点）



标签的添加删除

删除元素

- remove() - 删除被选元素（及其子元素）

例：`$("#div1").remove();` //删除的是div1的

- remove() - 过滤被删除的元素

remove() 方法也可接受一个参数，允许对被删元素进行过滤。

该参数可以是任何 jQuery 选择器的语法。

例：`$("p").remove(".italic");`

//删除 class="italic" 的所有 <p> 元素



标签的添加删除

删除元素

- `removeAttr()`从被选元素中移除属性

例：`$("p").removeAttr("id");`

删除内容

- `Empty()` – 从被选元素中删除子元素

例：`$("#div1").empty();` //删除的是div1中的内容

删除元素及内容

03

属性的获取设置



属性的获取设置

设置内容 – text()、html()、以及val()

获取内容语法：\$(selector).text()

设置内容语法：\$(selector).text(“设置内容”)

其它两个方法语法一样

text()、html() 以及 val()的回调函数

回调函数由两个参数：被选元素列表中当前元素的下标，以及原始（旧的）值。然后以函数新值返回您希望使用的字符串。

获取 设置



属性的获取设置

例：

```
$("#btn1").click(function(){  
    $("#test1").text(function(i,origText){  
        return "Old text: " + origText + " New text: Hello " +  
        " world! (index: " + i + ")";  
    });  
});
```

回调函数



属性的获取设置

获取属性或设置属性 – attr()

获取属性语法：\$(selector).attr(“属性名”)

设置属性语法：\$(selector).attr(“属性名” , “属性值”)

attr()的回调函数

回调函数有两个参数：被选元素列表中当前元素的下标，以及原始（旧的）值。然后以函数新值返回您希望使用的字符串。

获取 设置



属性的获取设置

例：

```
$("#button").click(function(){  
    $("#taobao").attr("href", function(i,origValue){  
        return origValue + "/market/nvzhuang";  
    });  
});
```

attr回调函数



属性的获取设置

css()方法设置或返回被选元素的一个或多个样式属性。

➤返回CSS属性：

语法：`css("属性名")`

例：`$("p").css("color")`



属性的获取设置

➤设置CSS属性：

语法：`css("属性名" , "属性值")`

例：`$("p").css("color" , " #f00")`

➤设置多个CSS属性：

语法：`css({ "属性名" : "属性值" ,属性名" : "属性值" ,...})`

例：`$("p").css({ "color" : " #f00" , " background" : " #00f" })`

04

Class的添加删除



Class的添加删除

addClass() – 向被选元素中添加一个或多个class类

例1：

```
$("#button").click(function(){  
    $("#h1,h2,p").addClass("blue");  
});
```

添加类时可选取多个元素



Class的添加删除

例2：

```
$("#button").click(function(){  
    $("#div").addClass( "important" );  
    $("#div").addClass( "blue");  
});
```

```
$("#div").addClass("important");
```

```
$("#div").addClass("blue");
```

等同于：

```
$("#div").addClass("important blue");
```




Class的添加删除

- `removeClass()` – 从被选元素删除一个或多个class类
与`addClass()`用法相同
- `toggleClass()` – 对被选元素进行添加/删除class类的切换操作

该方法是对`addClass()`和`removeClass()`方法的切换

CSS类

05

jQuery编写图片轮播



图片轮播

计时器复习：

setInterval()

clearInterval();

指定时间提示

2秒切换图片

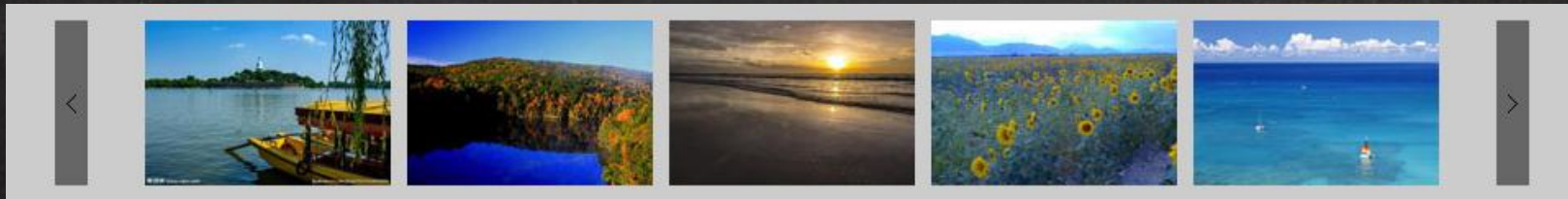
```
var t = setInterval(al, 3000);
    function al(){
        alert("计时开始！")
    }
$(function(){
    $("input:eq(0)").click(function()
    {
        clearInterval(t);
    })
    $("input:eq(1)").click(function()
    {
        t=setInterval(al,2000);
    })
})
```



图片轮播

➤ 图片左右滑动

图片每隔3秒向左滑动一个图片的宽度，
单击左右箭头时图片向左向右切换。



图片左右滑动切换

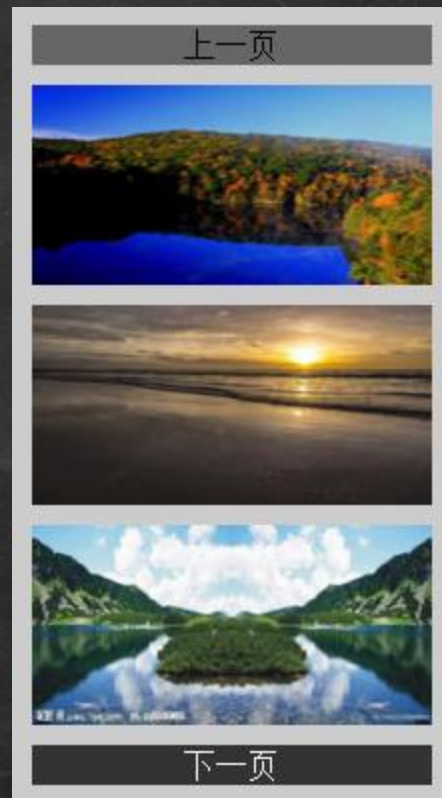


图片轮播

➤ 图片上下滑动切换

与左右滑动切换原理相同

图片上下滑动切换

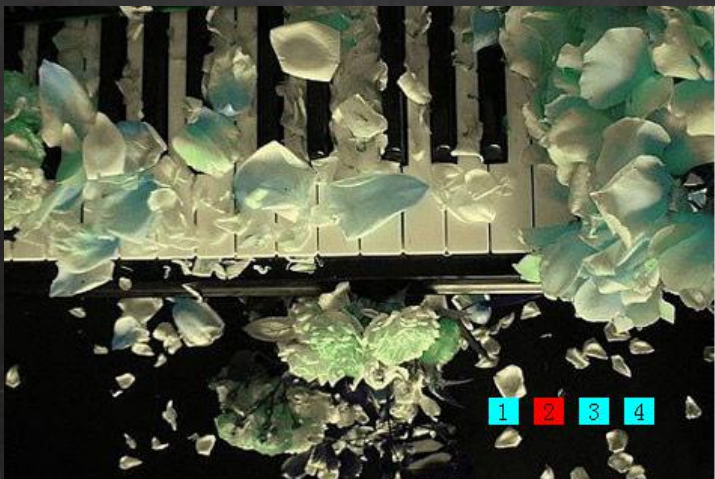




图片轮播

- 图片自动切换加单击按钮切换到指定图片

图片自动切换，当单击下面按钮时图片切换到指定图片



图片渐入渐出自动加手动



图片轮播

➤ 图片切换并改变内容切换

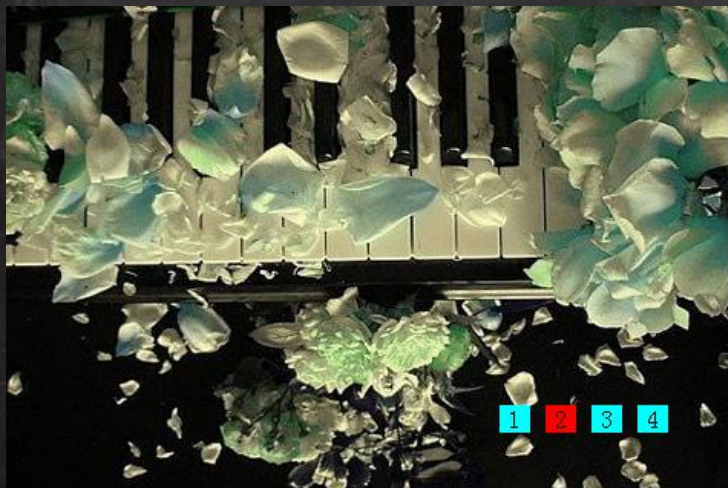


图片切换并改变相应内容



图片轮播练习

- Banner图切换（如图1）
- 图片切换相应内容也切换



谢 谢