



Web UI框架及应用

第1章 jQueryUI 基础

- 
- 课程目标及学习方法
 - jQuery UI简介
 - jQuery UI下载及使用
 - 选项卡控件
 - 折叠菜单控件



课程介绍

- 课程名称：Web UI框架及应用
- 课时：40学时



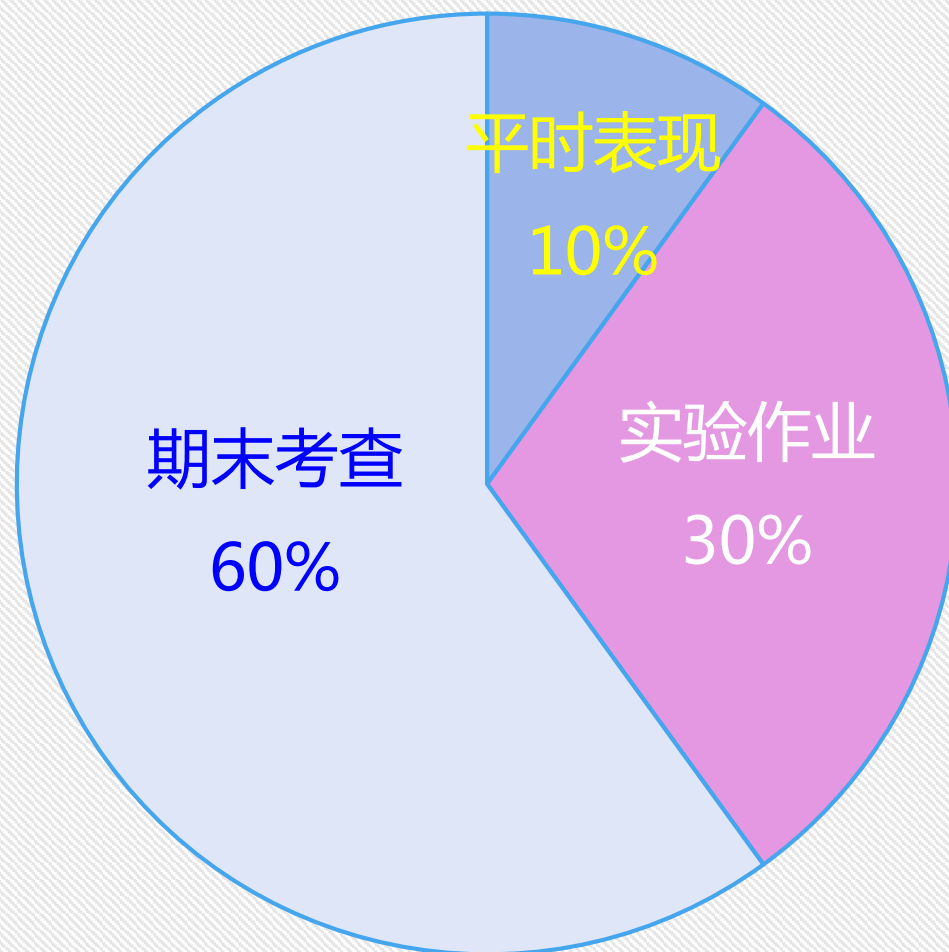
课程目标

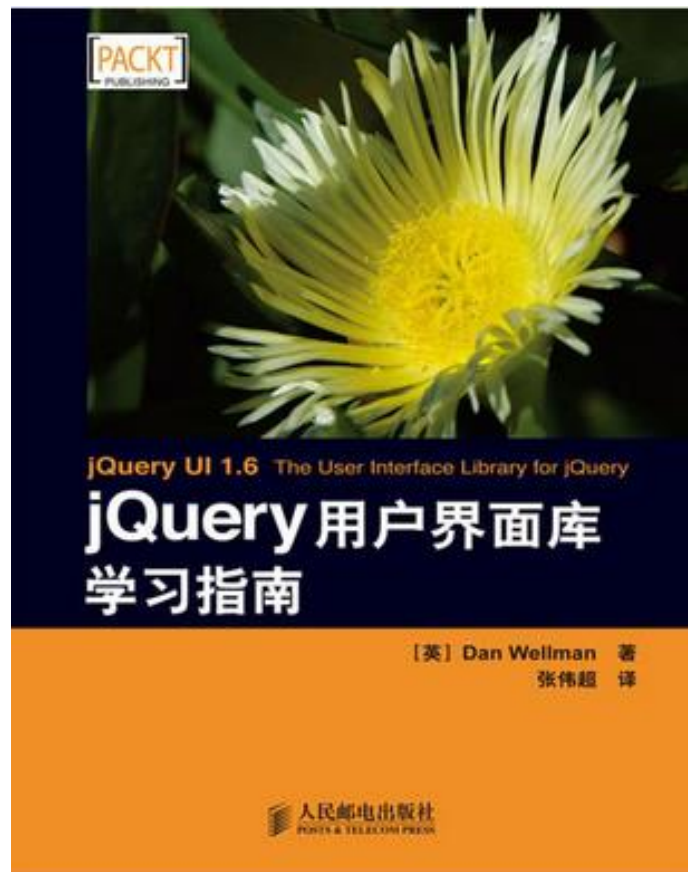
- 1. 掌握Web UI开发框架：jQuery UI。能够使用jQuery UI的小部件、用户界面交互、特效及主题创建高度交互的 Web 应用程序。
- 2. 了解网页UI设计基础。
- 3. 掌握网站基本图形元素设计。
- 4. 掌握网站布局与版式设计。
- 5. 能够使用Web UI开发框架构建完整的Web页面。



课程考核

成绩组成





- 
- 课程目标及学习方法
 - jQuery UI简介
 - jQuery UI下载及使用
 - 选项卡控件
 - 折叠菜单控件



jQuery 插件

jQuery插件：基于jQuery规范的应用程序接口编写的扩展程序。为了提供更多的功能，和更方便的使用方式。

jQuery UI(The User Interface Library for jQuery)，是基于jQuery 基础类库开发的开源的 JavaScript 网页UI代码组件库。包含底层用户交互、动画、特效和可更换主题的可视控件，可以直接使用它来构建具有很好交互性的web应用程序。

兼容IE 6.0+, Firefox 3+, Safari 3.1+, Opera 9.6+, 和 Google Chrome。

jQuery UI主要包含三大部分：

- **交互组件**：鼠标交互（ droppable - 拖拽组件； resizable – 可调整大小组件； selectable – 选择组件； sortable – 排序组件。 ）
- **小部件**：页面UI工具（ accordion – 折叠显示； datepicker – 日历操作； dialog – 对话框； slider – 滑块； progressbar – 进度条； Tabs – 选项卡。 ）
- **效果库**：在jQuery核心库之上增强的动画效果（ effect – 效果操作 ）

jQuery UI的特点：

- 继承 jQuery 简易使用特性，封装了大量的常用操作，提供更为丰富的UI效果和组件，短期改善网站易用性。
- 兼容各主流桌面浏览器。
- 组件间相对独立，可按需加载，避免浪费带宽拖慢网页打开速度。
- jQuery官方提供的插件，有丰富的在线文档，易于学习。
- 开发包内置包含中文在内的 40 多种语言包。

缺乏全面的测试用例，组件间 API 缺乏协调。

- 
- 课程目标及学习方法
 - jQuery UI简介
 - **jQuery UI下载及使用**
 - 选项卡控件
 - 折叠菜单控件



jQuery UI的下载

- 使用jQuery UI之前，要从其官网<http://jqueryui.com>下载最新的jQuery UI库（目前最新版本为1.12.1）。

步骤 1：选择 jQuery UI 的版本

在下载生成器（Download Builder）中，选择一个版本号。这个步骤很重要，因为 jQuery UI 的版本是配合特定的 jQuery 版本设计的。

步骤 2：选择需要的组件

下载生成器页面的第一栏列出了 jQuery UI 所有的 JavaScript 组件分类：核心（UI Core）、交互部件（Interactions）、小部件（Widgets）和效果库（Effects）。

步骤 3：选择一个主题或者自定义一个主题

Version

- ☒ 1.12.1 (Stable, for jQuery1.7+)
- ☐ 1.11.4 (Legacy, for jQuery1.6+)
- ☐ 1.10.4 (Legacy, for jQuery1.6+)
- ☐ 1.9.2 (Legacy, for jQuery1.6+)



jQuery UI的下载

- 解压缩下载的文件，把js文件夹下的jquery-ui.js文件，保存到相同的文件夹中；把css文件夹下的jquery-ui.css文件和images文件夹，复制到自己程序的css目录中。
- 注意：jQuery库（ jquery.js ）一定要配置在jQuery UI库（ jquery-ui.js ）的前面。



jQuery UI的使用

- 在页面中引用三个文件

```
<link rel="stylesheet" href="css/jquery-ui.css" />  
<script src="js/jquery.js"></script>  
<script src="js/jquery-ui.js"></script>
```

- 添加框架images文件夹

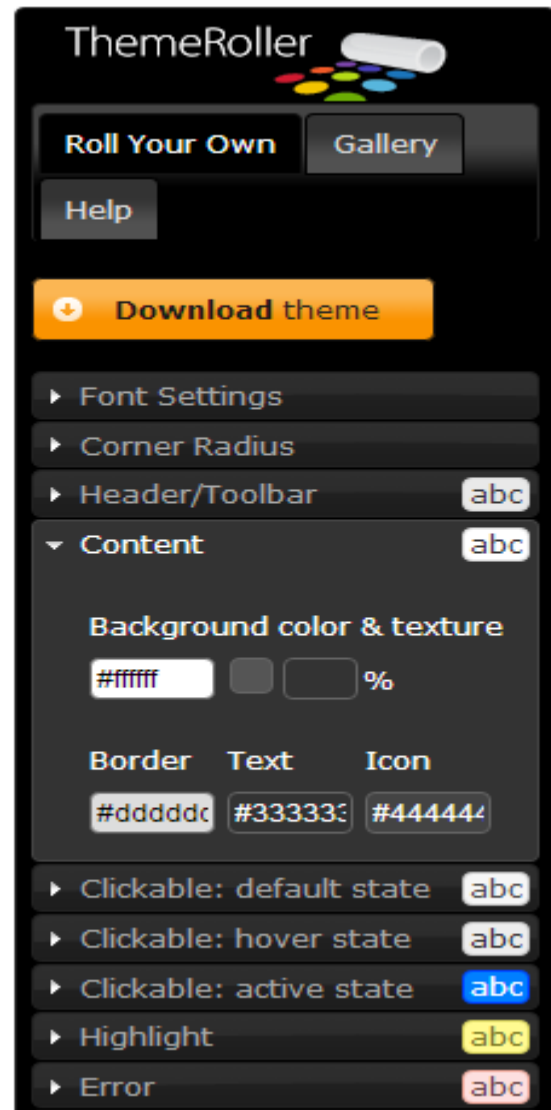


jQuery UI主题

- 三种主题化 jQuery UI 插件的一般方法：
- ①下载 ThemeRoller 主题

ThemeRoller 为 jQuery UI 设计和下载自定义主题提供了直观的界面。

步骤：设计、下载、在项目中添加





jQuery UI主题

- 三种主题化 jQuery UI 插件的一般方法：
- ②修改 CSS 文件

为了对外观和感观做进一步的控制，您可以选择从默认主题（Smoothness）开始，或者从一个由 ThemeRoller 生成的主题开始，然后调整 ui.theme.css 文件，或者任意一个独立插件的样式表。

- ③重新编写自定义的 CSS

为了最大程度地控制外观和感观，可以重新开始编写每个插件的 CSS，而不使用框架类或者特定的插件样式表。

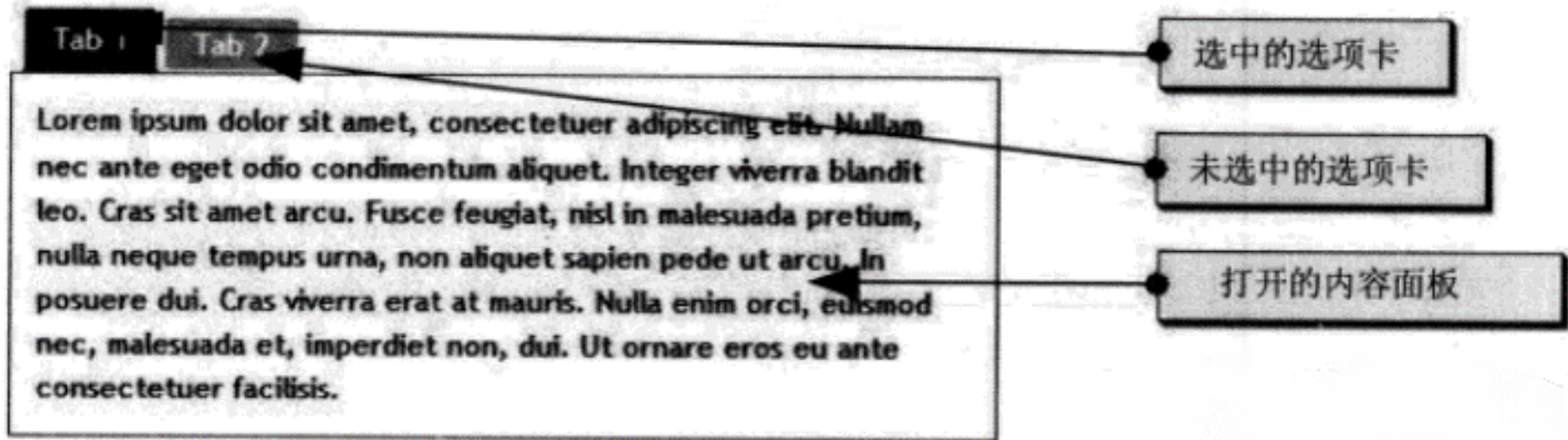
- 
- 课程目标及学习方法
 - jQuery UI简介
 - jQuery UI下载及使用
 - 选项卡控件
 - 折叠菜单控件



jQuery UI选项卡控件

选项卡控件 (tab) 用于在一组不同元素之间切换视角

可以通过单击每个元素的标题来访问该元素包含的内容，这些标题都作为独立的选项卡出现。每个内容面板都具有一个与之关联的选项卡，同一时刻只能打开其中一个面板。





jQuery UI选项卡控件

①必须使用列表元素创建，列表可以是有序或者无序的

③选项卡关联的内容可以是任意有效的元素，但是它必须带有一个 id，该 id 与相关选项卡标题的href锚点相对应。

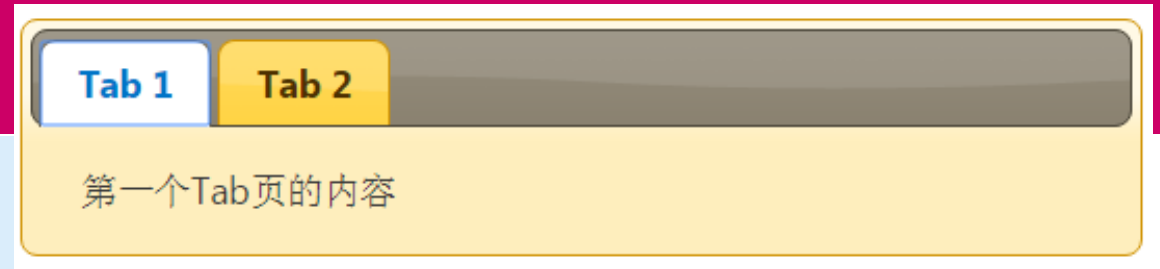


②每个列表项对应包含一个和一个<a>，用来设置选项卡标题

④在代表选项卡容器的对象（列表外侧的div）上调用tabs()构造方法



4 jQuery UI选项卡控件



```
<div id="myTabs">
  <ul>
    <li><a href="#tab0"><span>Tab 1</span></a></li>
    <li><a href="#tab1"><span>Tab 2</span></a></li>
  </ul>
  <div id="tab0">
    第一个Tab页的内容
  </div>
  <div id="tab1">
    第二个Tab页的内容，只有被选中时才会打开
  </div>
</div>
```

```
$(function(){
  $("#myTabs").tabs();
});
```

tabs1.html



思考

为什么使用了tabs()方法会彻底改变已渲染页面中的HTML元素的外观

该方法遍历了HTML代码并给选项卡元素自动增加了特定的CSS类，赋予了它们适当的样式。





jQuery UI选项卡的风格

自定义选项卡的样式。如果需要使用标签页指定的样式，则使用下面的 CSS class 名称：

ui-tabs：选项卡的外层容器。

ui-tabs-nav：选项卡列表。

激活的选项卡会带有 ui-tabs-active class。内容通过 Ajax 调用加载的选项卡会带有一个 ui-tabs-loading class。

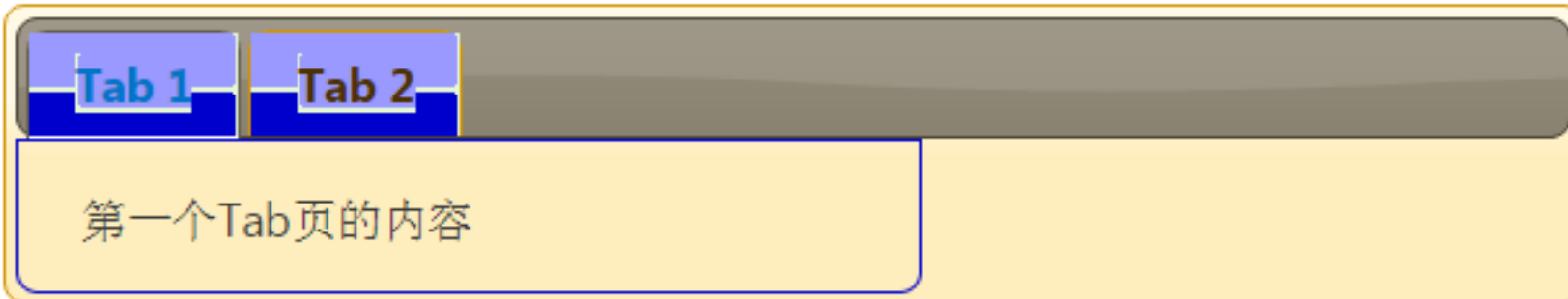
ui-tabs-anchor：用于切换面板的锚。

ui-tabs-panel：选项卡的内容面板。



4 jQuery UI选项卡的风格

```
.ui-tabs-nav a, .ui-tabs-nav a span {  
    background:url(../img/tab-sprite.gif) no-repeat;  
}  
.ui-tabs-nav a {  
    background-position:100% 0%;  
}  
.ui-tabs-panel {  
    width:300px; border:1px solid #0000cc;  
}
```



tabs2.html



4 tabs(options)方法

tabs(options)方法声明把一个HTML元素当作选项卡来管理。options参数是一个对象，用来指定选项卡的外观及行为。可以将对象传递给tabs()构造方法以配置选项卡的不同属性。

```
$(selector, content). tabs(options)
```




4 tabs(options)方法

管理选项卡的外观及行为的选项。

属性	默认值	用途
collapsible	false	如果设置为true，即允许用户将已经选中的选项卡内容折叠起来。即：点击一次选项卡，选项卡内容显示出来，再次点击选项卡，选项卡的内容区收起。
disabled	[]	设置哪些选项卡不可用，是一个数组例[0,1,2]，也就是第一个、第二个、第三个选项卡。
selected	0	初始化时，哪个选项卡被选中，默认为0，即第一个选项卡被选中。
fx	null	指定选择选项卡时的切换效果



jQuery UI选项卡可配置属性

可以将对象传递给tabs()构造方法以配置选项卡的不同属性。

```
$(function(){  
    var tabOpts = {  
        selected: 1,  
        disabled:[0]  
    };  
    $("#myTabs").tabs(tabOpts);  
});
```

tabs3.html



jQuery UI选项卡的过渡效果

通过使用fx属性，可以为打开和关闭选项卡增加过渡效果。默认为：null，无动画效果。

动画属性：opacity

动画发生的速度属性：duration (slow、normal、fast)



选项卡控件的事件

选项卡相关事件	触发条件
select	选项卡被选中
load	一个远程（ ajax ）选项卡的内容被加载完成
show	选项卡内容被显示
add	界面上增加了一个选项卡
remove	界面上删除了一个选项卡
enable	选项卡被激活
disable	选项卡被禁用



jQuery UI选项卡事件

```
$(function(){  
    function handleSelect(event, tab) {  
        alert("The tab at index " + tab.index + " was selected");  
    }  
    var tabOpts = {  
        select : handleSelect    // 将select属性值设置成回调函数名  
    };  
    $("#myTabs").tabs(tabOpts);  
});
```



jQuery UI选项卡事件

使用jQuery库的`bind()`方法将事件处理器和选项卡发起的自定义事件绑定：

```
$('#example').bind('tabsselect', function(event, ui) {  
    ui.tab    // 被选中（点击后）的选项卡元素  
    ui.panel  // 这个元素包含被选中（点击后）的选项卡的内容  
    ui.index  // 返回一个被选中（或点击后）选项卡的索引值（从0开始）  
});
```



选项卡控件自定义绑定事件

选项卡相关事件	触发条件
tabsselect	选项卡被选中
tabsload	一个远程（ ajax ）选项卡的内容被加载完成
tabsshow	选项卡被显示
tabsadd	界面上增加了一个选项卡
tabsremove	界面上删除了一个选项卡
tabsenable	选项卡被激活时
tab:disable	选项卡被禁用时



jQuery UI激活和禁用选项卡

可以通过enable和disable通过索引号实现激活和禁用选项卡

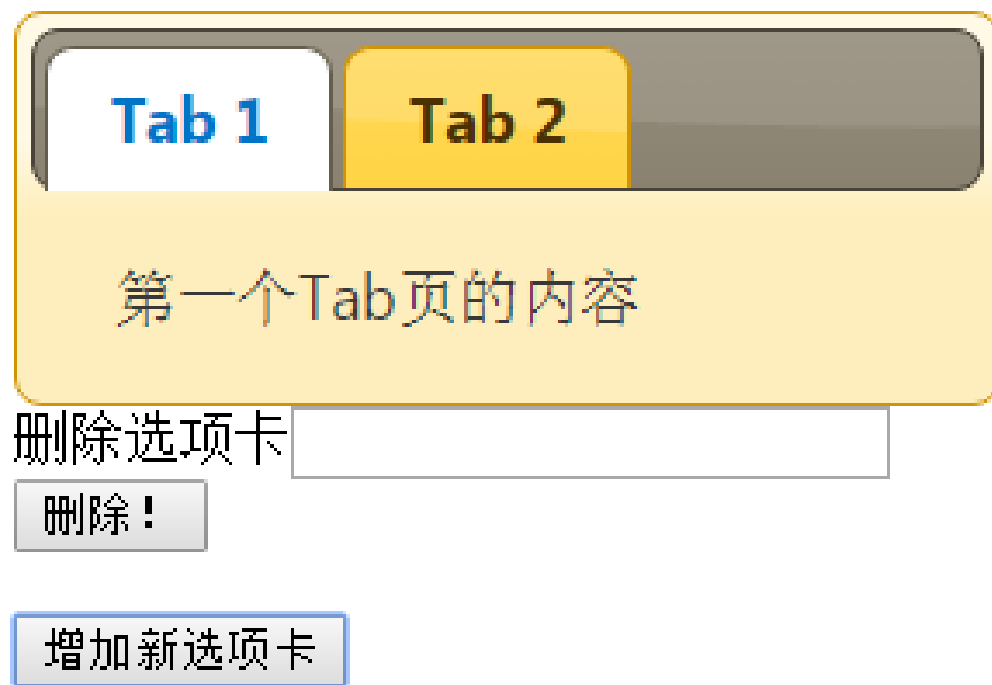
```
$(function(){  
    var tabOpts = {  
        selected:1,  
        disabled:[0]  
    };  
    $("#myTabs").tabs(tabOpts);  
    $("#enable").click(function() {  
        $("#myTabs").tabs("enable", 0);  
    });  
    $("#disable").click(function() {  
        $("#myTabs").tabs("disable", 1);  
    });  
});
```

tabs6.html



jQuery UI增加和删除选项卡

实例1：动态实现增加和除去选项卡



This 点击上面的按钮，这些内容将作为
tabs标签的一部分

tabs8.html



AJAX选项卡

可以创建AJAX选项卡从远程文件或URL中载入内容

```
$(function(){  
    $("#myTabs").tabs();  
  
    //add click handler for 'add' button  
    $("#add").click(function() {  
        var newLabel = "A New Tab!"  
        $("#myTabs").tabs("add", "tabContent.html", newLabel);  
    });  
});
```

tabs9.html

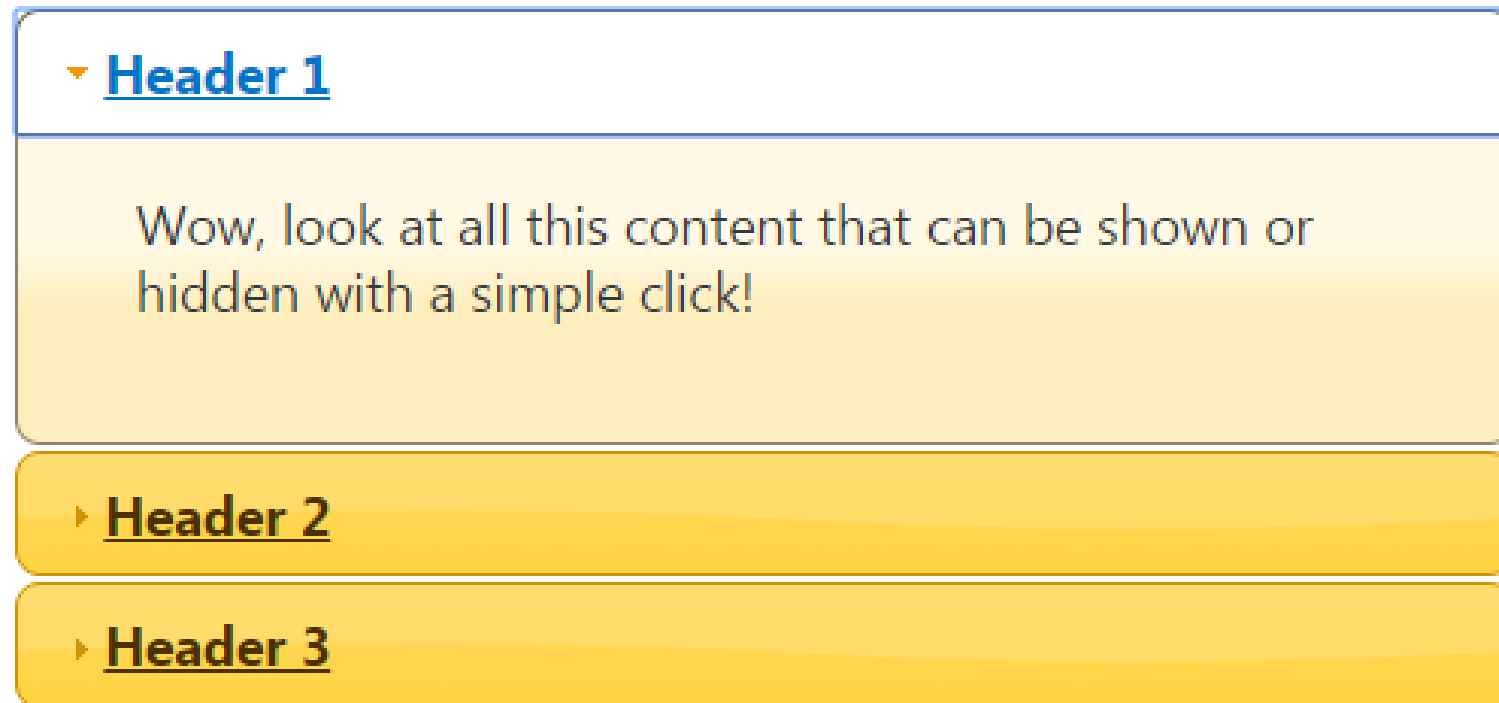
- 
- 课程目标及学习方法
 - jQuery UI简介
 - jQuery UI下载及使用
 - 选项卡控件
 - 折叠菜单控件



Accordion控件

Accordion控件用于创建**折叠菜单**。

和选项卡（ tabs ）一样，可以使用折叠菜单来组织HTML页面上的信息。通常与嵌套的列表、定义列表或嵌套的 div 一起使用。点击头部展开/折叠被分为各个逻辑部分的内容。



标题

处于选中状态的标题

打开的内容面板



5 折叠菜单的基本结构

- 一个包含整个折叠菜单的<div>元素；
- 一个作为菜单标题的元素，可以是<h1>~<h6>，包含一个<a>链接，是菜单的名称（但该链接不需要href属性，因为没有用到）
- 一个作为菜单内容的<div>元素。

```
<div id="myAccordion">
  <h3><a href="#">标题 1</a></h3>
  <div>哇,看看所有这些内容,可以用一个
  <h3><a href="#">标题 2</a></h3>
  <div>...</div>
  <h3><a href="#">标题 3</a></h3>
  <div>...</div>
</div>
```

```
$(function() {
    $("#myAccordion").accordion();
});
```

accordion1.html



思考

为什么使用了accordion()方法会彻底改变已渲染页面中的HTML元素的外观

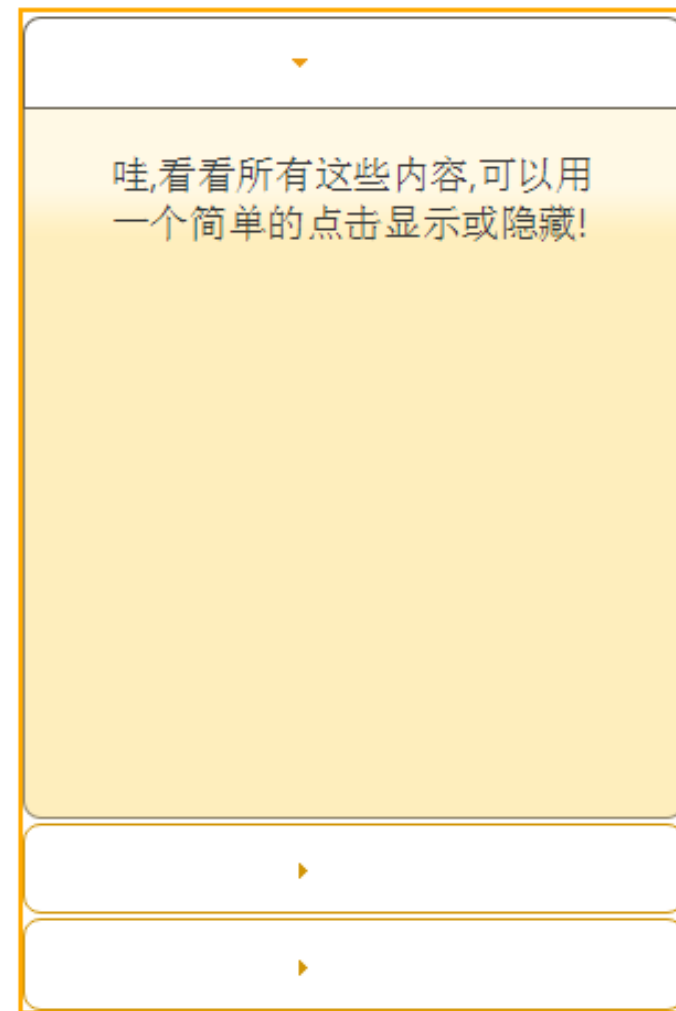
该方法遍历了HTML代码并给折叠菜单元素自动增加了特定的CSS类，赋予了它们适当的样式。





5 Accordion控件的样式

```
#myAccordion { width:200px; border:2px solid #000000;
position:relative; list-style-type:none; padding-left:0; }
.ui-accordion-header { text-decoration:none; font-
weight:bold; color:#000000; display:block; width:100%;
text-align:center; }
.ui-accordion div div { font-size:90%; }
.ui-accordion a { background:url(..img/accordion/header-
sprite.gif) repeat-x 0px 0px; color:#ffffff; }
.ui-accordion a.selected
{ background:url(..img/accordion/header-sprite.gif)
repeat-x 0px -22px; }
.ui-accordion a:hover
{ background:url(..img/accordion/header-sprite.gif)
repeat-x 0px -44px; }
```



accordion2.html



accordion(options)方法

accordion(options)方法指定使用折叠菜单来管理HTML元素。
options参数是一个对象，用来指定菜单的外观及行为。

这些选项与菜单的行为、菜单内容的高度或者菜单的事件相关。

```
$(selector, content). accordion(options)
```




5 accordion(options)方法

管理菜单行为的选项

属性	用途	默认值
active	设置第一次访问页面时被打开菜单的索引	0
animated	设置选择菜单时伴随的动画效果	slide
collapsible	是否允许用户通过单击来关闭菜单	false
event	如何展开菜单。	click



accordion(options)方法

管理菜单内容的高度

属性	用途	默认值
autoHeight	所有内容部分的高被设定为其中最高内容的高度。当设置为false时，每个菜单的高度等于其内容的实际高度。	true
fillSpace	当设置为true时，所有菜单内容拥有和全局<div>元素的父元素一样的高度和宽度。	false



谢谢