

# jQuery UI Part 3: Panels

Slides © 2016 Marty Hall, hall@coreservlets.com

For additional materials, please see <http://www.coreservlets.com/>. The JavaScript tutorial section contains complete source code for all examples in the entire tutorial series, plus exercises and exercise solutions for each topic.



For customized training related to JavaScript or Java, email [hall@coreservlets.com](mailto:hall@coreservlets.com)  
Marty is also available for consulting and development support

Taught by lead author of *Core Servlets & JSP*,  
co-author of *Core JSF* (4<sup>th</sup> Ed), and this tutorial.

Available at public venues, or  
custom versions can be held on-site at your organization.

- **Courses developed and taught by Marty Hall**
  - JavaScript, jQuery, Ext JS, JSF 2.3, PrimeFaces, Java 8 programming, Spring Framework, Spring MVC, Android, GWT, custom mix of topics
  - Courses available in any state or country.
  - Maryland/DC companies can also choose afternoon/evening courses.
- **Courses developed and taught by coreservlets.com experts (edited by Marty)**
  - Hadoop, Hibernate/JPA, HTML5, RESTful Web Services

Contact [hall@coreservlets.com](mailto:hall@coreservlets.com) for details



## Topics in This Section

- **Tabbed panels with static content**
- **Tabbed panels with dynamic (Ajax) content**
- **Animated tabbed panels**
- **Accordion panels**

4

coreservlets.com – custom onsite training



# Tabbed Panels with Static Content

Slides © 2016 Marty Hall, hall@coreservlets.com

For additional materials, please see <http://www.coreservlets.com/>. The JavaScript tutorial section contains complete source code for all examples in the entire tutorial series, plus exercises and exercise solutions for each topic.

## Overview

- **Idea**

- Tabbed panel where clicking tabs changes content inside

- **HTML**

- Make a div container to hold the tabs

```
<div id="main-tabbed-panel-div">...</div>
```

- Include a ul list containing internal hyperlinks

```
<ul>  
  <li><a href="#panel1">Show Panel 1</a></li>  
  <li><a href="#panel2">Show Panel 2</a></li>  
</ul>
```

- Include divs whose ids match the href values (minus “#”)

```
<div id="panel1">Content for panel 1</div>  
<div id="panel2">Content for panel 2</div>
```

- **Basic JavaScript**

- Call tabs() on the div container

```
$("#main-tabbed-panel-div").tabs();
```

6

## Main Options for tabs({...}) with Static Content

- **selected (default: 0)**

- Tab to be selected initially

- **collapsible (default: false)**

- Can you click selected tab to hide it (and thus *all* panels)?

```
$("#main-id").tabs({ collapsible: true });
```

- **disabled (default: empty array)**

- Array of tab nums (0-based) that should be disabled on startup

- **event (default: "click")**

- The type of event that should initiate tab selection

```
$("#main-id").tabs({ event: "mouseover" });
```

- **show, hide**

- Animation options. See upcoming section for details.

7

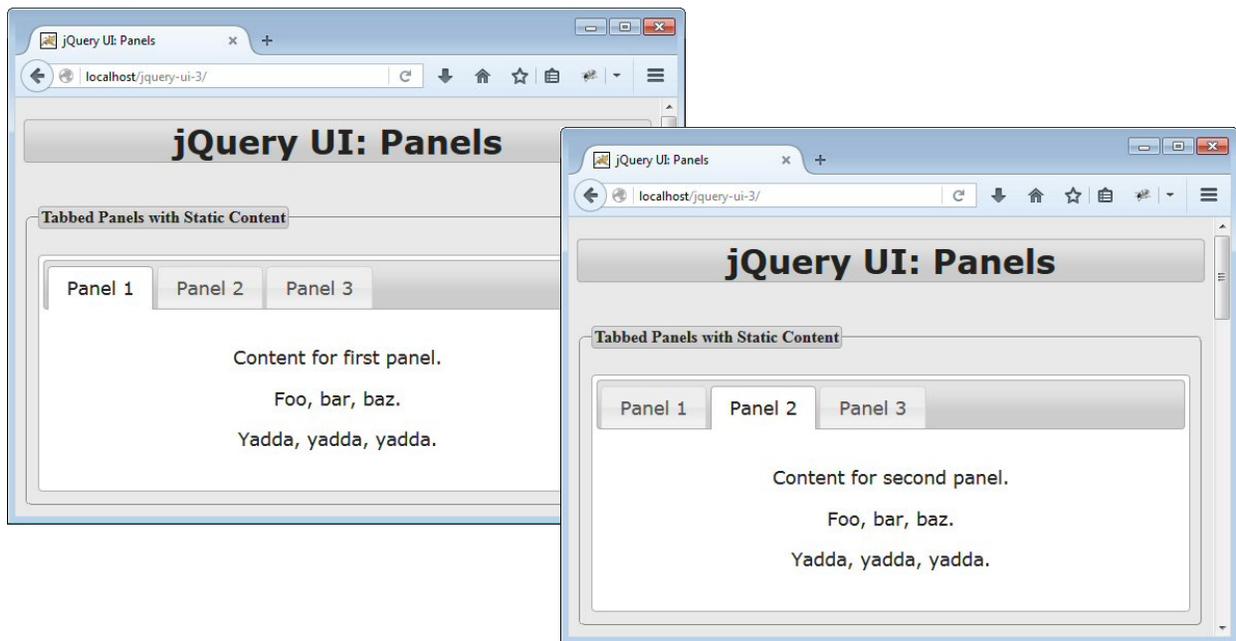
## Example: HTML

```
<div id="static-tabbed-panel-div">
  <ul>
    <li><a href="#panel1">Panel 1</a></li>
    <li><a href="#panel2">Panel 2</a></li>
    <li><a href="#panel3">Panel 3</a></li>
  </ul>
  <div id="panel1">
    <p>Content for first panel.</p>
    <p>Foo, bar, baz.</p><p>Yadda, yadda, yadda.</p>
  </div>
  <div id="panel2">
    <p>Content for second panel.</p>
    <p>Foo, bar, baz.</p><p>Yadda, yadda, yadda.</p>
  </div>
  <div id="panel3">
    <p>Content for third panel.</p>
    <p>Foo, bar, baz.</p><p>Yadda, yadda, yadda.</p>
  </div>
</div>
```

## Example: JavaScript

```
$(function() {
  $("#static-tabbed-panel-div").tabs();
  ...
});
```

## Example: Results



coreservlets.com – custom onsite training



# Tabbed Panels with Dynamic (Ajax) Content

Slides © 2016 Marty Hall, hall@coreservlets.com

For additional materials, please see <http://www.coreservlets.com/>. The JavaScript tutorial section contains complete source code for all examples in the entire tutorial series, plus exercises and exercise solutions for each topic.

## Overview

- **Idea**

- Tabbed panel where clicking tabs gets server content

- **HTML**

- Make a div container to hold the tabs

```
<div id="main-tabbed-panel-div">...</div>
```

- Include a ul list containing external relative hyperlinks

```
<ul>  
  <li><a href="url1">Get Content for Panel 1</a></li>  
  <li><a href="url2">Get Content for Panel 2</a></li>  
</ul>
```

- That's all!

- One of the simplest and best Ajax tabbed panel methods anywhere

- **Basic JavaScript**

- Call tabs() on the div container

```
$("#main-tabbed-panel-div").tabs();
```

12

## Example: HTML

```
<div id="ajax-tabbed-panel-div">  
<ul>  
  <li><a href="time-panel.jsp">Show Time</a></li>  
  <li><a href="list-panel.jsp?range=5">  
    Show Small Numbers</a></li>  
  <li><a href="list-panel.jsp?range=500">  
    Show Big Numbers</a></li>  
</ul>  
</div>
```

13

## Example: JavaScript

```
$(function() {  
    $("#ajax-tabbed-panel-div").tabs({ collapsible: true });  
    ...  
});
```

Collapsible means that clicking on the heading of an open section will close it, resulting in all sections being closed. Otherwise the only way to close one section is to open another section. Collapsible is useful for Ajax sections, so that you can close and reopen the same section to see the updated content.

14

## Example: JSP 1 (time-panel.jsp)

```
<h3>It is now <%= new java.util.Date() %></h3>  
Do you know what your closures are doing?
```

As seen earlier, printing out the "date" also prints the time down to the second. So, by reloading a section that uses this content, you can easily see whether the server-side data was updated.

15

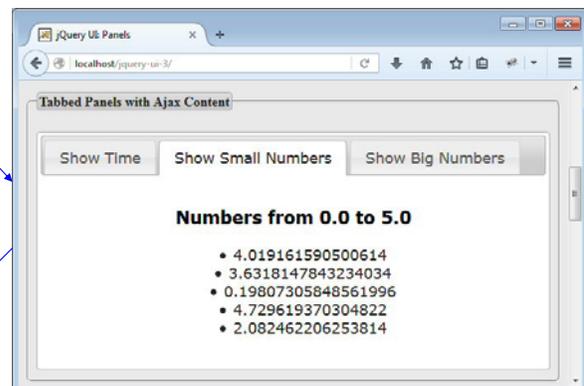
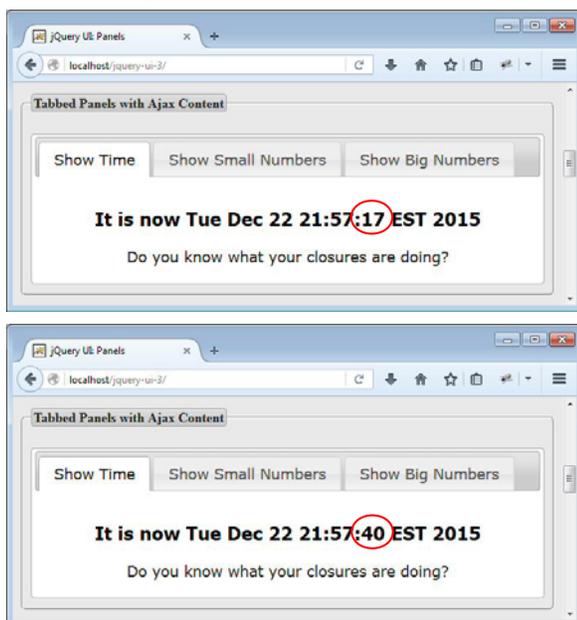
## Example: JSP 2 (list-panel.jsp)

```
<% double range = 10.0;
   try {
       String r = request.getParameter("range");
       range = Integer.parseInt(r);
   } catch(Exception e) {} %>
<h3>Numbers from 0 to <%= range %></h3>
<ul>
  <li><%= Math.random() * range %></li>
  <li><%= Math.random() * range %></li>
</ul>
```

If you have never seen servlets before, this syntax will be unfamiliar. The `request.getParameter` method reads the incoming parameter "range" - this parameter was supplied in the URL shown in the HTML on the previous slide. It parses it as an int, catching an exception in case the request parameter was missing or malformed. The exact details are not important; what matters is that since it uses `Math.random`, the output is different each time.

16

## Example: Results



Every time that you switch to a new tab or close and reopen the existing tab, the data is re-fetched from the server.

## Controlling the Ajax Requests

- **beforeLoad option**

- This takes a function that receives the request object and a UI object containing the object that was passed to \$.ajax. Although it gives you very fine-grained control, it has two problems:
  - The documentation under the Tabs widget is so terse as to be completely incomprehensible. However, the documentation in the change guide when this feature was first introduced is useful. See:
    - <http://jqueryui.com/upgrade-guide/1.9/#deprecated-ajaxoptions-and-cache-options-added-beforeload-event>
  - It replaced far-simpler (although less customizable) cache, spinner, and ajaxOptions features. In particular, cache: true (save data locally after first remote request) previously let you implement lazy loading in one line, and spinner: with some HTML let you super-easily give “Working...” messages for slow responses.
    - It now takes many lines of code now to reproduce those 1-liners

18

## Passing Options to \$.ajax

- **Idea**

- Any option you could supply to the \$.ajax object, you can now set in the ajaxOptions field of the second argument to beforeLoad.

- **Example: setting username and password**

```
$("#your-id").tabs({
  beforeLoad: function(req, ui) {
    ui.ajaxSettings.username = "user1";
    ui.ajaxSettings.password = "secret";
  }
});
```

19

## Lazy Loaded Tabs

- **Idea**

- Load tab content by Ajax on first selection, then cache locally thereafter. Lets you speed up overall page load if content for non-active tabs is slow.

- **Code**

```
$("#your-id").tabs({
  beforeLoad: function(req, ui) {
    if (ui.tab.data("loaded") ) {
      req.preventDefault();
      return;
    }
    ui.jqXHR.success(function() {
      ui.tab.data( "loaded", true );
    });
  }
});
```

- Through jQuery UI version 1.8, this was just `$("#your-id").tabs({ cache: true });`

20

coreservlets.com – custom onsite training



# Animating Tabbed Panels

Slides © 2016 Marty Hall, hall@coreservlets.com

For additional materials, please see <http://www.coreservlets.com/>. The JavaScript tutorial section contains complete source code for all examples in the entire tutorial series, plus exercises and exercise solutions for each topic.

## Overview

- **Idea**

- Tabbed panels have the same show: and hide: options as dialog boxes, with the same behavior. This lets you specify any of the jQuery UI animation effects, and supply options such as duration if desired.

- **HTML**

- Same as in previous examples. Static or Ajax content.

- **Basic JavaScript**

```
$("#main-tabbed-panel-div").tabs({ show: { ... }, close: { ... } });
```

- The values for show and close contain the name of an effect and optionally effect options like duration.

```
show: { effect: "blind" }  
close: { effect: "explode", duration: 1000 }
```

- For names of all effects and details of options, see <http://api.jqueryui.com/category/effects/>
- Use animations lightly in most business apps

22

## Example: HTML

```
<div id="animated-tabbed-panel-div">  
<ul>  
  <li><a href="time-panel.jsp">Show Time</a></li>  
  <li><a href="list-panel.jsp?range=5">  
    Show Small Numbers</a></li>  
  <li><a href="list-panel.jsp?range=500">  
    Show Big Numbers</a></li>  
</ul>  
</div>
```

23

## Example: JavaScript

```
$(function() {  
  $("#animated-tabbed-panel-div")  
    .tabs({ show: { effect: "blind",  
                  duration: 2000 },  
           hide: { effect: "bounce",  
                  duration: 1000 } });  
  ...  
});
```

24

## Example: Results

When you click on tab 3, the content for tab 2 bounces up and down, then the content for tab 3 slowly slides down.

It is difficult to visualize the results from static screen shots. To see it in action, go to the online tutorial at <http://www.coreservlets.com/javascript-jquery-tutorial/>, download the app, deploy it, and try it yourself.

# Accordion Panels



Q: What is the difference between a crow and an accordion?  
A: One makes a loud and annoying sound. The other is a bird.

Q: What is the definition of a gentleman?  
A: Someone who knows how to play an accordion, but doesn't.

Q: What do you call a sunken ship full of accordions?  
A: A good start.

Slides © 2016 Marty Hall, hall@coreservlets.com

For additional materials, please see <http://www.coreservlets.com/>. The JavaScript tutorial section contains complete source code for all examples in the entire tutorial series, plus exercises and exercise solutions for each topic.

## Overview

- **Idea**
  - Horizontally stacked panels with wide tabs
- **HTML**
  - Make a div container to hold the tabs  
`<div id="main-accordion-panel-div">...</div>`
  - Include *alternating* pairs of headings and content  
`<h2>Heading for Panel 1</h2>`  
`<div>Content for Panel 1</div>`  
`<h2>Heading for Panel 2</h2>`  
`<div>Content for Panel 2</div>`
- **Basic JavaScript**
  - Call `accordion()` on the div container  
`$("#main-accordion-panel-div").accordion();`

## Usage: Tab Panels vs. Accordion Panels

- **Tabbed panels**
  - When you have relatively few tabs
  - When the labels of the tabs are short
  - When you want Ajax content
    - Major flaw: accordion panels do not have builtin support for dynamic content
- **Accordion panels**
  - If you have many sections
  - When the section labels are long
- **Reasoning**
  - Users are expecting to scroll up and down, but not left to right

28

## Deficiencies

- **Inconsistent usage**
  - Tab panels use a list with hypertext links, followed by a set of block-level elements with those ids.
  - Accordion panels use alternating headers and content.
  - They should have used the same design for both, so merely changing `.tab()` to `.accordion()` would switch.
- **Lack of Ajax option for accordion panels**
  - You are just as likely to want dynamic content for accordion panels as for tabbed panels, and it would have been just as easy for them to supply this capability.
- **PrimeFaces note**
  - PrimeFaces wraps jQuery tab panels and accordion panels to make them easily available in JSF2 apps. It unifies the syntax using the approach with alternating headers and content. It allows Ajax for both panel types.

29

## Main Options for accordion({...})

- **active (default: 0)**
    - Index or selector for initially-selected tab.
      - Use false if you want no panel showing initially (needs collapsible: true).
  - **collapsible (default: false)**
    - Can you click to hide?
  - **event (default: "click")**
    - Event to trigger selection.
      - Same as with tabs()
  - **animated (default: true)**
    - Lets you control animation; see next slide
- ```
$("#main-id").accordion({ animated: 1000 });
```

30

## Animating Accordion Panels

- **animate option**
  - Lets you specify arbitrary animations, similarly to in the previous example with animated tabs

```
$("#your-id").accordion({ animate: "fadeIn" });
```

  - Arbitrary animations are rarely useful for business applications, but disabling the animation altogether (animate: false) and slowing it down (animate: 1000) are both reasonable options

```
$("#your-id").accordion({ animate: false });  
$("#your-id").accordion({ animate: 1000 });
```

31

## Example: HTML

```
<div id="accordion-panel-div">
<h2>Panel 1</h2>
<div>
  <p>Content for first panel.</p>
  <p>Foo, bar, baz.</p><p>Yadda, yadda, yadda.</p>
</div>
<h2>Panel 2</h2>
<div>
  <p>Content for second panel.</p>
  <p>Foo, bar, baz.</p><p>Yadda, yadda, yadda.</p>
</div>
<h2>Panel 3</h2>
<div>
  <p>Content for third panel.</p>
  <p>Foo, bar, baz.</p><p>Yadda, yadda, yadda.</p>
</div>
</div>
```

32

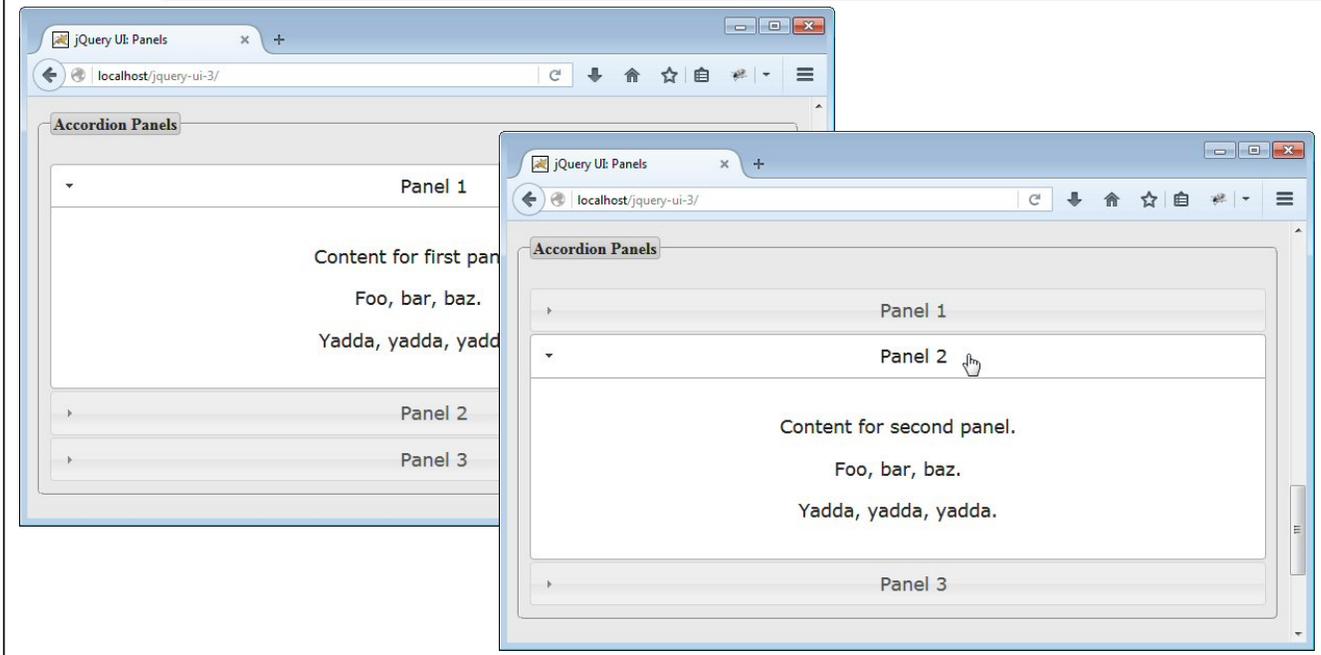
IMHO, this is a better design than for tabbed panels, and jQuery UI should have used this design for both tabs and accordion panels.

## Example: JavaScript

```
$(function() {
  $("#accordion-panel-div").accordion();
  ...
});
```

33

## Example: Results



[coreservlets.com](http://coreservlets.com) – custom onsite training



# Wrap-up

Slides © 2016 [Marty Hall](http://Marty Hall), [hall@coreservlets.com](mailto:hall@coreservlets.com)

For additional materials, please see <http://www.coreservlets.com/>. The JavaScript tutorial section contains complete source code for all examples in the entire tutorial series, plus exercises and exercise solutions for each topic.

## Summary

- **Static tabbed panels**
  - Make a container div, include a ul list with internal hyperlinks and divs whose ids match the href values
  - Call tabs() on the container div
- **Ajax tabbed panels**
  - Make a container div, include a ul list with external relative hyperlinks
  - Call tabs() on the container div.
- **Accordion panels**
  - Make a container div, include alternating pairs of headings and content
  - Call accordion() on the container div.

36

coreservlets.com – custom onsite training



# Questions?

More info:

<http://www.coreservlets.com/javascript-jquery-tutorial/> – Tutorial on JavaScript, jQuery, and jQuery UI

<http://courses.coreservlets.com/Course-Materials/java.html> – General Java programming tutorial

<http://www.coreservlets.com/java-8-tutorial/> – Java 8 tutorial

<http://courses.coreservlets.com/java-training.html> – Customized Java training courses, at public venues or onsite at your organization

<http://coreservlets.com/> – JSF 2, PrimeFaces, Java 8, JavaScript, jQuery, Ext JS, Hadoop, RESTful Web Services, Android, HTML5, Spring, Hibernate, Servlets, JSP, GWT, and other Java EE training

Many additional free tutorials at coreservlets.com (JSF, Android, Ajax, Hadoop, and lots more)

Slides © 2016 Marty Hall, hall@coreservlets.com

For additional materials, please see <http://www.coreservlets.com/>. The JavaScript tutorial section contains complete source code for all examples in the entire tutorial series, plus exercises and exercise solutions for each topic.