

SWEN1005

MOBILE WEB PROGRAMMING

Session Eleven

LETS EXPLORE A FEW JQUERY FADING METHODS

jQuery Fading Methods

- With jQuery you can fade an element in and out of visibility.
- jQuery has the following fade methods:
 - `fadeIn()`
 - `fadeOut()`
 - `fadeToggle()`
 - `fadeTo()`

jQuery fadeIn(), fadeOut() and fadeToggle Methods

- The jQuery fadeIn(), fadeOut() and fadeToggle methods are used to fade elements.
- **Syntax:**
- `$(selector).fadeIn(speed,callback);`
- The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.
- The optional callback parameter is a function to be executed after the fading completes.

fadeIn()... fadeout()

```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0
/jquery.min.js"></script>
<script>
$(document).ready(function(){
    $("#button").click(function(){
        $("#div1").fadeIn();
        $("#div2").fadeIn("slow");
        $("#div3").fadeIn(3000);
    });
});
</script>
</head>
<body>

<p>Demonstrate fadeIn() with different parameters.</p>

<button>Click to fade in boxes</button><br><br>

<div id="div1" style="width:80px;height:80px;display:none;background-
color:red;"></div><br>
<div id="div2" style="width:80px;height:80px;display:none;background-
color:green;"></div><br>
<div id="div3" style="width:80px;height:80px;display:none;background-
color:blue;"></div>

</body>
</html>
```

Demonstrate fadeIn() with different parameters. Demonstrate fadeIn() with different parameters.

Click to fade in boxes

Click to fade in boxes

Demonstrate fadeIn() with different parameters.

Click to fade in boxes



fadeToggle()

```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0
/jquery.min.js"></script>
<script>
$(document).ready(function(){
    $("#button").click(function(){
        $("#div1").fadeToggle();
        $("#div2").fadeToggle("slow");
        $("#div3").fadeToggle(3000);
    });
});
</script>
</head>
<body>

<p>Demonstrate fadeToggle() with different speed parameters.</p>

<button>Click to fade in/out boxes</button><br><br>

<div id="div1" style="width:80px;height:80px;background-color:red;"></div>
<br>
<div id="div2" style="width:80px;height:80px;background-color:green;"></div>
<br>
<div id="div3" style="width:80px;height:80px;background-color:blue;"></div>

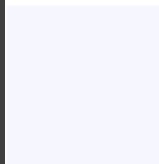
</body>
</html>
```

Demonstrate fadeToggle() with different speed parameters.

Click to fade in/out boxes

Demonstrate fadeToggle() with different speed parameters.

Click to fade in/out boxes



jQuery fadeTo() Method

- The jQuery fadeTo() method allows fading to a given opacity (value between 0 and 1).

Syntax:

- `$(selector).fadeTo(speed,opacity,callback);`
- The required speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

jQuery fadeTo() Method

Syntax:

- `$(selector).fadeTo(speed,opacity,callback);`
- The required speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.
- The required opacity parameter in the fadeTo() method specifies fading to a given opacity (value between 0 and 1).
- The optional callback parameter is a function to be executed after the function completes.

fadeTo() Method

```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0
/jquery.min.js"></script>
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("#div1").fadeTo("slow", 0.15);
        $("#div2").fadeTo("slow", 0.4);
        $("#div3").fadeTo("slow", 0.7);
    });
});
</script>
</head>
<body>

<p>Demonstrate fadeTo() with different parameters.</p>

<button>Click to fade boxes</button><br><br>

<div id="div1" style="width:80px;height:80px;background-color:red;">
</div><br>
<div id="div2" style="width:80px;height:80px;background-color:green;">
</div><br>
<div id="div3" style="width:80px;height:80px;background-color:blue;"></div>

</body>
</html>
```

Demonstrate fadeTo() with different parameters.

Click to fade boxes



Demonstrate fadeTo() with different parameters.

Click to fade boxes



jQuery Sliding Methods

- With jQuery you can create a sliding effect on elements.
- jQuery has the following slide methods:
 - `slideDown()`
 - `slideUp()`
 - `slideToggle()`

jQuery slideDown() Method

- The jQuery slideDown() method is used to slide down an element.
- **Syntax:**
 - `$(selector).slideDown(speed,callback);`
 - The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.
 - The optional callback parameter is a function to be executed after the sliding completes.

slideDown() Method

Click to slide down panel

Click to slide down panel

Hello world!

```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0
/jquery.min.js"></script>
<script>
$(document).ready(function(){
    $("#flip").click(function(){
        $("#panel").slideDown("slow");
    });
});
</script>

<style>
#panel, #flip {
    padding: 5px;
    text-align: center;
    background-color: #e5eccc;
    border: solid 1px #c3c3c3;
}

#panel {
    padding: 50px;
    display: none;
}
</style>
</head>
<body>

<div id="flip">Click to slide down panel</div>
<div id="panel">Hello world!</div>

</body>
</html>
```

jQuery slideToggle() Method

- The jQuery slideToggle() method toggles between the slideDown() and slideUp() methods.
- If the elements have been slid down, slideToggle() will slide them up.
- If the elements have been slid up, slideToggle() will slide them down.
 - `$(selector).slideToggle(speed,callback);`
 - The optional speed parameter can take the following values: "slow", "fast", milliseconds.
 - The optional callback parameter is a function to be executed after the sliding completes.

jQuery slideToggle() Method

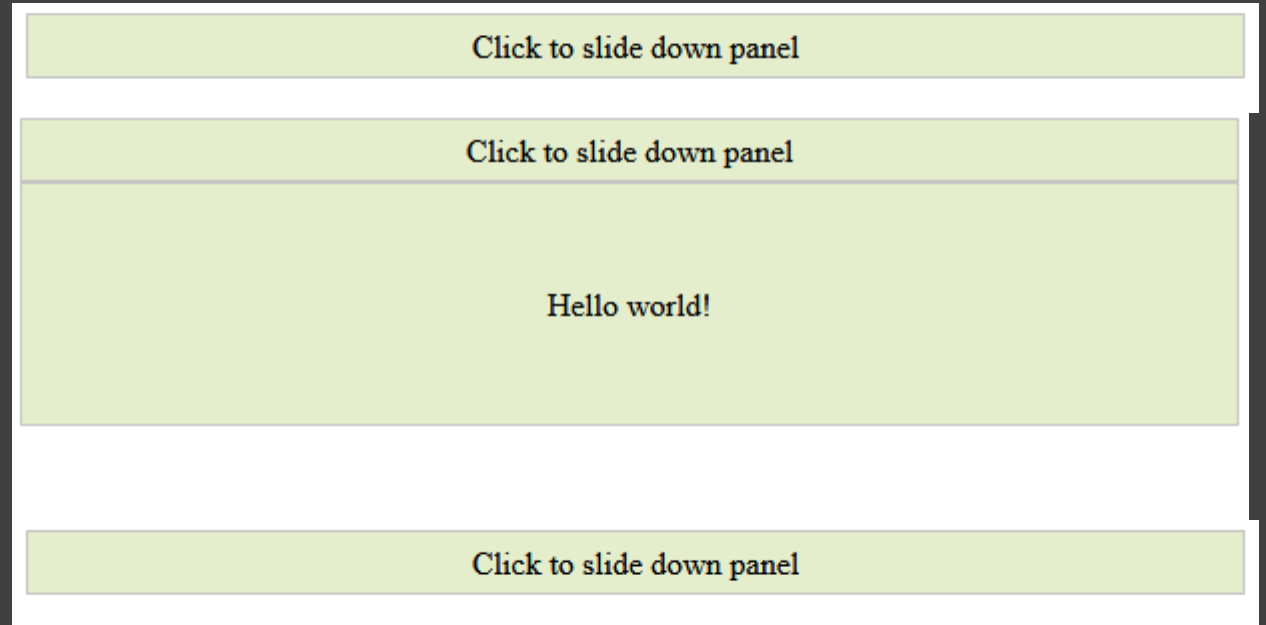
```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0/
jquery.min.js"></script>
<script>
$(document).ready(function(){
    $("#flip").click(function(){
        $("#panel").slideToggle("slow");
    });
});
</script>

<style>
#panel, #flip {
    padding: 5px;
    text-align: center;
    background-color: #e5eccc;
    border: solid 1px #c3c3c3;
}

#panel {
    padding: 50px;
    display: none;
}
</style>
</head>
<body>

<div id="flip">Click to slide the panel down or up</div>
<div id="panel">Hello world!</div>

</body>
</html>
```



jQuery Animations - The animate() Method

- The jQuery animate() method is used to create custom animations.

Syntax:

- `$(selector).animate({params},speed,callback);`
- The required params parameter defines the CSS properties to be animated.

```

<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0
/jquery.min.js"></script>
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("div").animate({
            left: '250px',
            opacity: '0.5',
            height: '150px',
            width: '150px'
        });
    });
});
</script>
</head>
<body>

<button>Start Animation</button>

<p>By default, all HTML elements have a static position, and cannot be
moved. To manipulate the position, remember to first set the CSS position
property of the element to relative, fixed, or absolute!</p>

<div style="background:#98bf21;height:100px;width:100px;position:absolute;">
</div>

</body>
</html>

```

Start Animation

By default, all HTML elements have a static position, and cannot be moved. To manipulate the position, remember to first set the CSS position property of the element to relative, fixed, or absolute!



Start Animation

By default, all HTML elements have a static position, and cannot be moved. To manipulate the position, remember to first set the CSS position property of the element to relative, fixed, or absolute!



jQuery Animations - The animate() Method

- Is it possible to manipulate ALL CSS properties with the animate() method?
- Yes, almost! However, there is one important thing to remember: all property names must be camel-cased when used with the animate() method: You will need to write paddingLeft instead of padding-left, marginRight instead of margin-right, and so on.
- Also, color animation is not included in the core jQuery library. If you want to animate color, you need to download the [Color Animations plugin](#) from jQuery.com.

Session Eleven

JQUERY MOBILE

What is jQuery Mobile?

- jQuery Mobile is a touch-optimized web framework for creating mobile web applications.
- jQuery Mobile is built on top of the jQuery library, which makes it easy to learn if you already know jQuery.
- It uses HTML5, CSS3, JavaScript and AJAX to accomplish its work for laying out pages with minimal scripting.



How is jQuery mobile able to do this

- Instead of writing one application for each mobile device or OS:
 - Android and Blackberry is written in Java
 - iOS is written in Objective C
 - Windows Phone is written in C# and .net, etc.
- jQuery Mobile solves this problem, as it only uses HTML, CSS and JavaScript, which is standard for all mobile web browsers!

What You Should Already Know

- Before you start studying jQuery Mobile, you should have a basic knowledge of:
 - HTML
 - CSS
 - jQuery (where a basic knowledge of JavaScript is required)

Best Learning Experience

- Even though jQuery Mobile works on all mobile devices, it may have some compatibility issues on desktop computers (due to limited CSS3 support).
- It is recommended that you use the Google Chrome Browser for a better learning experience.

jQuery Mobile – Where do we begin

- Just like jQuery there are two ways to add jQuery Mobile to your web site. You can:
 - Link to a jQuery Mobile library stored at a CDN
 - Link to a jQuery Mobile library stored at your computer

Link to jQuery Mobile From a CDN

- A CDN (Content Delivery Network) is used to distribute often-used files across the web.
- This makes download speed much faster for the user.

Link to jQuery Mobile From a CDN

- As with jQuery core, there is nothing to install on your computer; you just include the following stylesheet (.css) and JavaScript libraries (.js) directly into your HTML page, to get jQuery Mobile to work:

```
<head>

<!-- Include meta tag to ensure proper rendering and touch zooming -->
<meta name="viewport" content="width=device-width, initial-scale=1">

<!-- Include jQuery Mobile stylesheets -->
<link rel="stylesheet" href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">

<!-- Include the jQuery library -->
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>

<!-- Include the jQuery Mobile library -->
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>

</head>
```

jQuery Mobile Pages: Creating a Page

- The HTML5 data-* attributes are used throughout jQuery Mobile to create a "touch-friendly" and attractive look for mobile devices.
- The data-role="page" is the page displayed in the browser
- The data-role="header" creates a toolbar at the top of the page (often used for title or search buttons)
- The data-role="main" defines the content of the page, like text, images, buttons, forms, etc.

jQuery Mobile Pages: Creating a Page

- The "ui-content" class adds extra padding and margin inside the page content
- The data-role="footer" creates a toolbar at the bottom of the page
- Inside these containers, you can add any HTML elements - paragraphs, images, headings, lists, etc.

```
<body>
<div data-role="page">

  <div data-role="header">
    <h1>Welcome To My Homepage</h1>
  </div>

  <div data-role="main" class="ui-content">
    <p>I Am Now A Mobile Developer!!</p>
  </div>

  <div data-role="footer">
    <h1>Footer Text</h1>
  </div>

</div>
</body>
```

jQuery Mobile Pages: Creating a Page

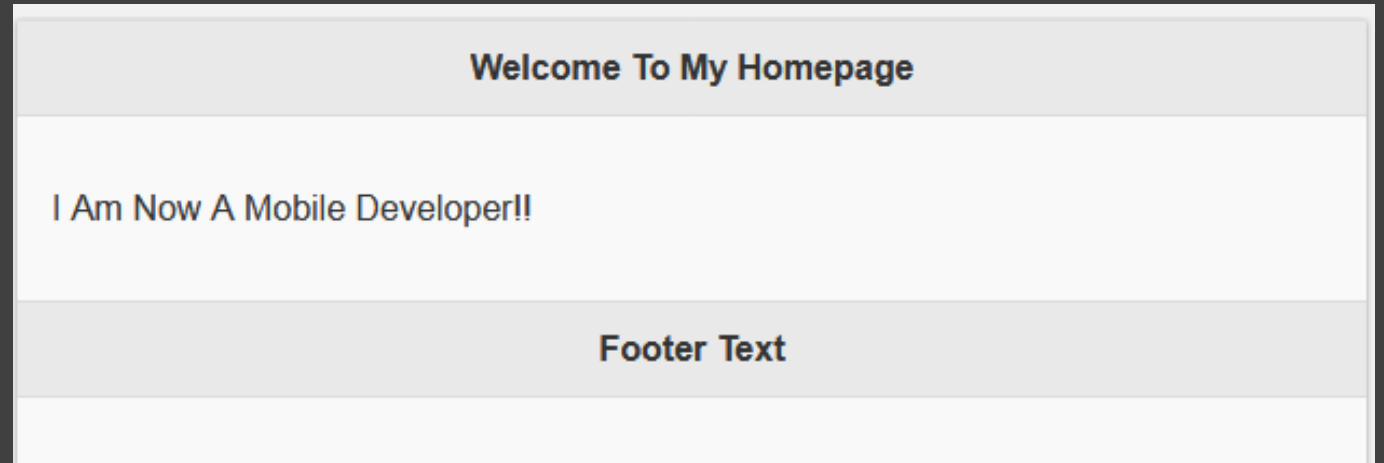
```
<body>
<div data-role="page">

  <div data-role="header">
    <h1>Welcome To My Homepage</h1>
  </div>

  <div data-role="main" class="ui-content">
    <p>I Am Now A Mobile Developer!!</p>
  </div>

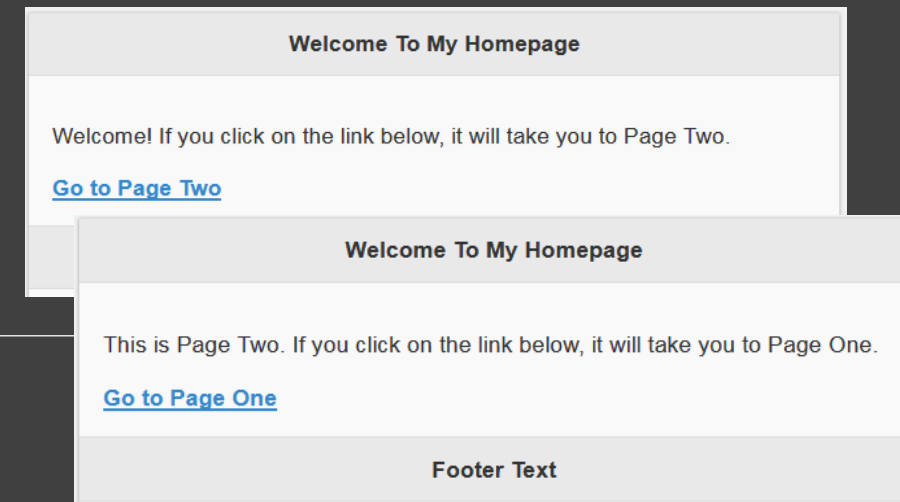
  <div data-role="footer">
    <h1>Footer Text</h1>
  </div>

</div>
</body>
```



jQuery Mobile Pages – Adding Pages

- In jQuery Mobile, you can create multiple pages in a single HTML file.
- Separate each page with a unique id and use the href attribute to link between them
- Of course you can also link pages externally as well



```
<div data-role="page" id="pageone">
  <div data-role="main" class="ui-content">
    <a href="#pagetwo">Go to Page Two</a>
  </div>
</div>
```

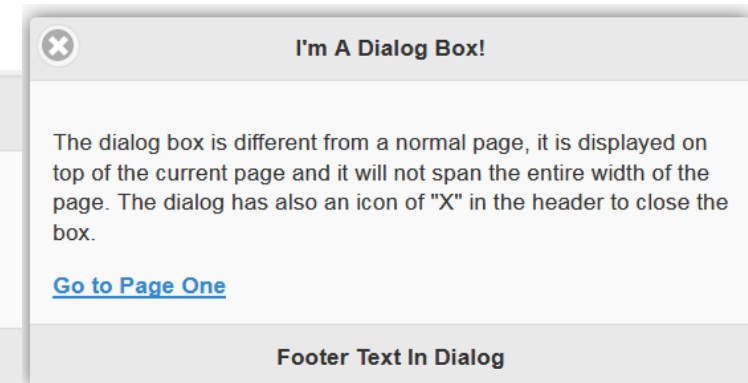
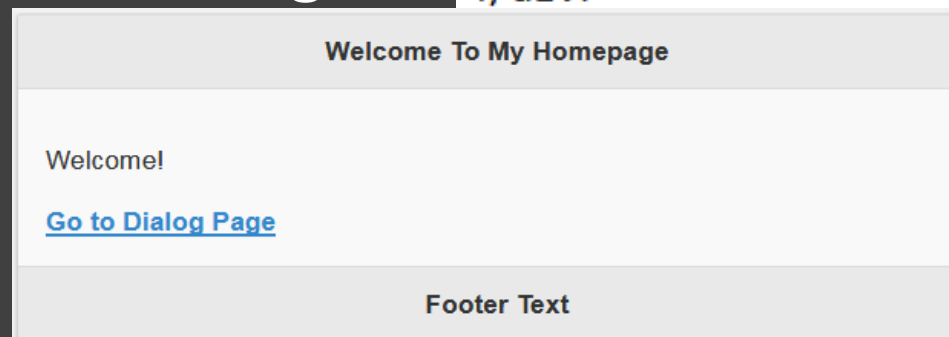
```
<div data-role="page" id="pagetwo">
  <div data-role="main" class="ui-content">
    <a href="#pageone">Go to Page One</a>
  </div>
</div>
```

jQuery Mobile Pages: Pages as Dialogs

- A dialog box is a type of window used to show special information or request input.
- To create a dialog box that opens when a user taps on a link, add `data-dialog="true"` to the page you want displayed as a dialog

```
<div data-role="page" id="pageone">  
  <div data-role="main" class="ui-content">  
    <a href="#pagetwo">Go to Page Two</a>  
  </div>  
</div>
```

```
<div data-role="page" data-dialog="true" id="pagetwo">  
  <div data-role="main" class="ui-content">  
    <a href="#pageone">Go to Page One</a>  
  </div>  
</div>
```



jQuery Mobile Transitions

- jQuery Mobile includes CSS3 effects that lets you choose the way a page should open.
- jQuery Mobile has a variety of effects for how to transition from one page to the next.
- To achieve the transition effect, the browser must support CSS3 3D Transforms

jQuery Mobile Transitions

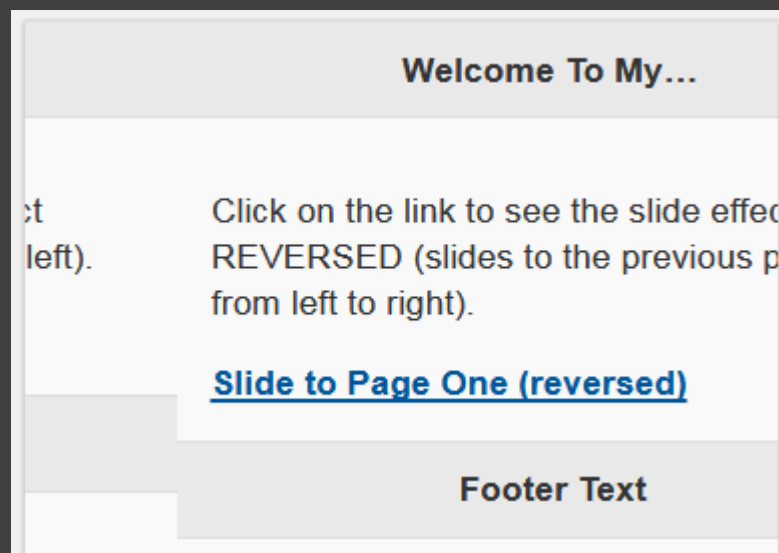
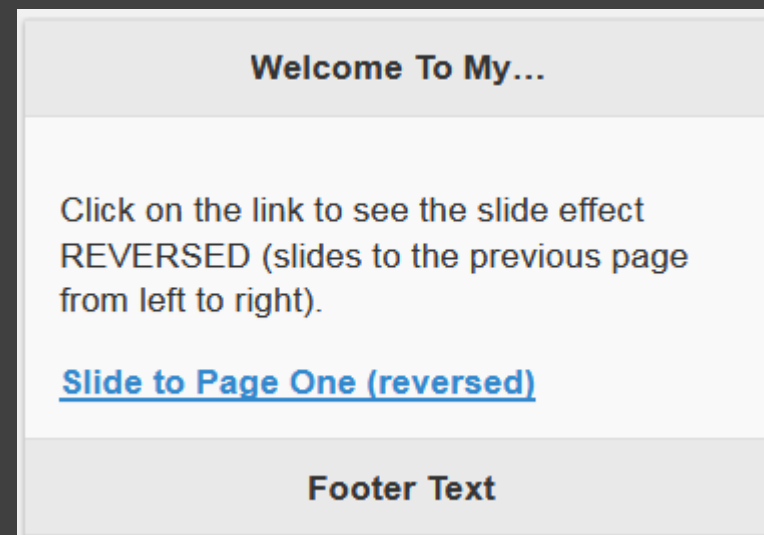
- The transition effect can be applied to any link or form submission by using the data-transition attribute:
 - `Slide to Page Two`
- The fading effect is default on all links in jQuery Mobile (if the browser support it).
- All effects also support reverse/backward actions, e.g. if you want the page to slide from left to right, instead of right to left, use the data-direction attribute with value "reverse".


```

<body>
<div data-role="page" id="pageone">
  <div data-role="header">
    <h1>Welcome To My Homepage</h1>
  </div>
  <div data-role="main" class="ui-content">
    <p>Click on the link to see the slide effect (slides to the next
page from right to left).</p>
    <a href="#pagetwo" data-transition="slide">Slide to Page Two</a>
  </div>

  <div data-role="footer">
    <h1>Footer Text</h1>
  </div>
</div>
<div data-role="page" id="pagetwo">
  <div data-role="header">
    <h1>Welcome To My Homepage</h1>
  </div>
  <div data-role="main" class="ui-content">
    <p>Click on the link to see the slide effect REVERSED (slides to
the previous page from left to right).</p>
    <a href="#pageone" data-transition="slide"
data-direction="reverse">Slide to Page One (reversed)</a>
  </div>
  <div data-role="footer">
    <h1>Footer Text</h1>
  </div>
</div>
</body>

```



jQuery Mobile Transitions

```
<div data-role="main" class="ui-content">  
  <p>Click on the link to see the pop effect.</p>  
  <a href="#pagetwo" data-transition="pop">Pop to  
Page Two</a>  
</div>
```

Transition	Description
fade	Default. Fades to the next page
flip	Flips to the next page from back to front
flow	Throws the current page away and comes in with the next page
pop	Goes to the next page like a popup window
slide	Slides to the next page from right to left
slidefade	Slides from right to left and fades in the next page
slideup	Slides to the next page from bottom to top
slidedown	Slides to the next page from top to bottom
turn	Turns to the next page
none	No transition effect

Creating a Button in jQuery Mobile

- A button in jQuery Mobile can be created in three ways:
 - Using the `<input>` element
 - `<input type="button" value="Button">`
 - Using the `<button>` element with `class="ui-btn"`
 - `<button class="ui-btn">Button</button>`
 - Using the `<a>` element with `class="ui-btn"`
 - `Button`

Creating a Button in jQuery Mobile

- Buttons in jQuery Mobile are automatically styled, making them attractive and useable on both mobile devices and desktop computers.
- It's recommended that you use the `<a>` element with `class="ui-btn"` to link between pages, and `<input>` or `<button>` elements for form submission.

```

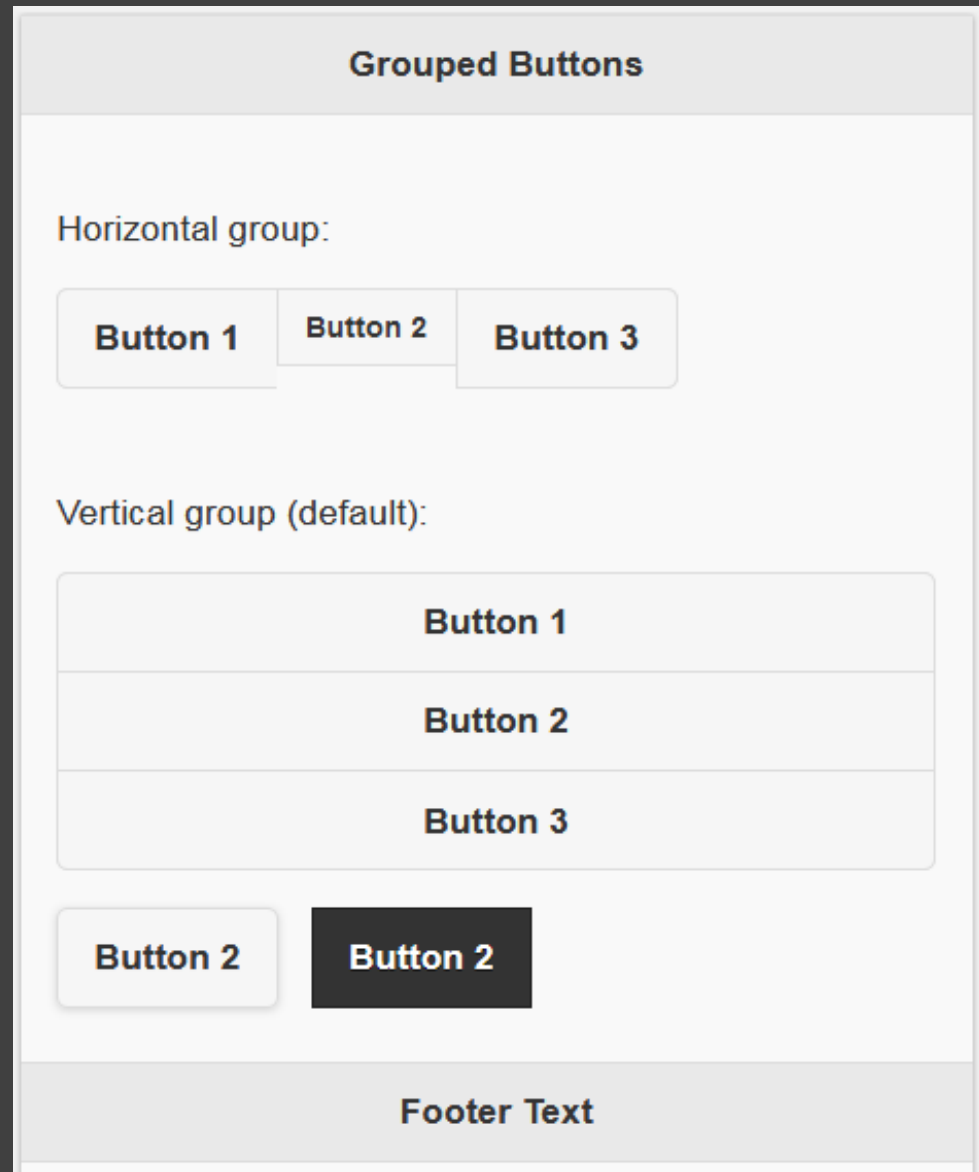
<div data-role="page" id="pageone">
  <div data-role="header">
    <h1>Grouped Buttons</h1>
  </div>

  <div data-role="main" class="ui-content">
    <div data-role="controlgroup" data-type="horizontal">
      <p>Horizontal group:</p>
      <a href="#" class="ui-btn">Button 1</a>
      <a href="#" class="ui-btn ui-mini">Button 2</a>
      <a href="#" class="ui-btn">Button 3</a>
    </div><br>

    <div data-role="controlgroup" data-type="vertical">
      <p>Vertical group (default):</p>
      <a href="#" class="ui-btn">Button 1</a>
      <a href="#" class="ui-btn">Button 2</a>
      <a href="#" class="ui-btn">Button 3</a>
    </div>
    <div>
      <a href="#" class="ui-btn ui-btn-inline ui-corner-all ui-shadow">Button 2</a>
      <a href="#" class="ui-btn ui-btn-inline ui-btn-b">Button 2</a>
    </div>
  </div>

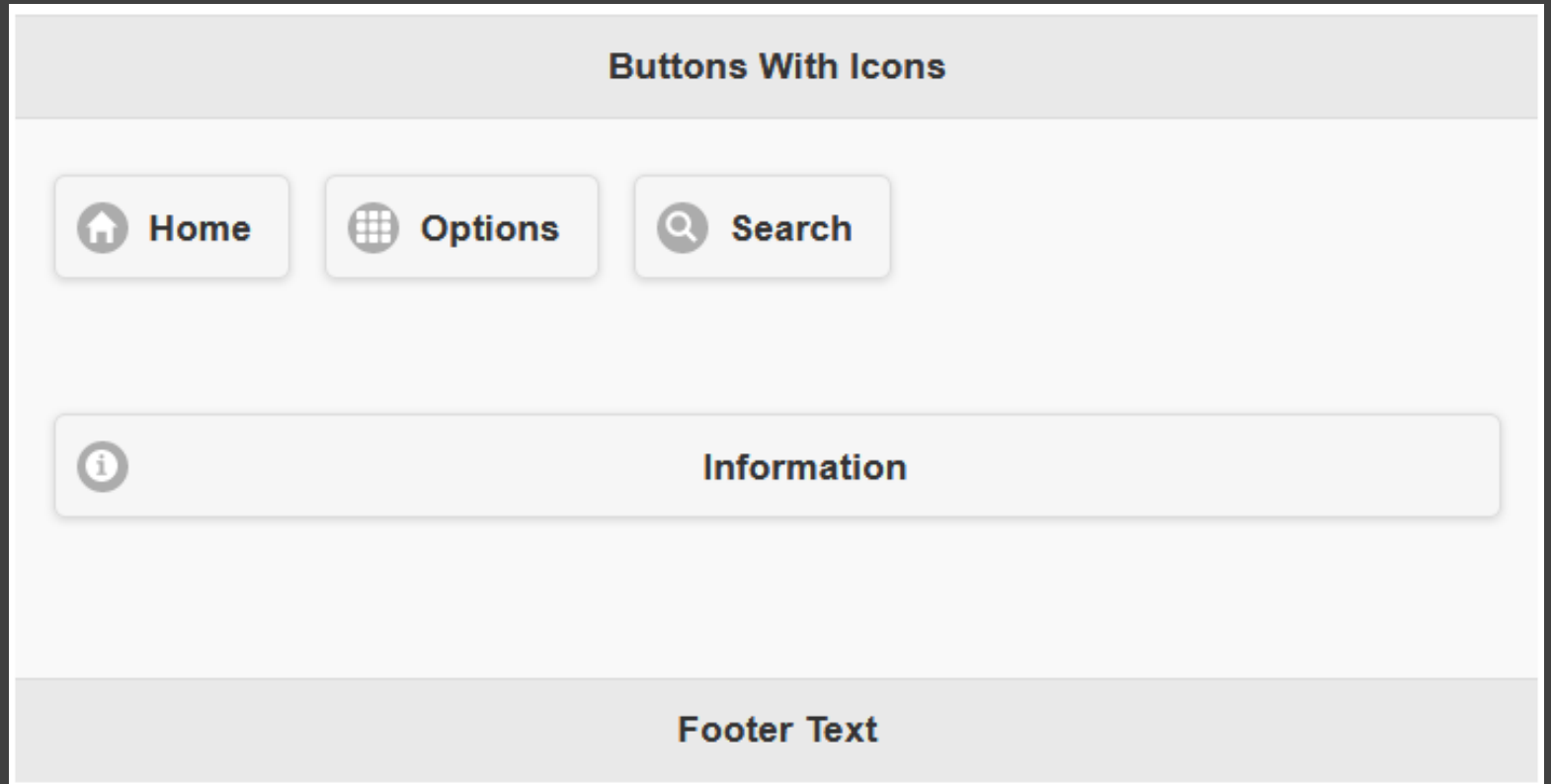
  <div data-role="footer">
    <h1>Footer Text</h1>
  </div>
</div>

```



jQuery Mobile Button Icons

- jQuery Mobile provides a set of icons that will make your buttons look more desirable.



```
<body>
```

```
<div data-role="page" id="pageone">
  <div data-role="header">
    <h1>Positioning Icons</h1>
  </div>

  <div data-role="main" class="ui-content">
    <p>Positioning Icons:</p>
    <a href="#" class="ui-btn ui-icon-search ui-btn-
icon-top">Top</a>
    <a href="#" class="ui-btn ui-icon-search ui-btn-
icon-right">Right</a>
    <a href="#" class="ui-btn ui-icon-search ui-btn-
icon-bottom">Bottom</a>
    <a href="#" class="ui-btn ui-icon-search ui-btn-
icon-left">Left</a>
  </div>
```

```

  <div data-role="footer">
    <h1>Footer Text</h1>
  </div>
</div>
```

```
</body>
```

Positioning Icons

Positioning Icons:



Top

Right














Bottom



Left

Footer Text

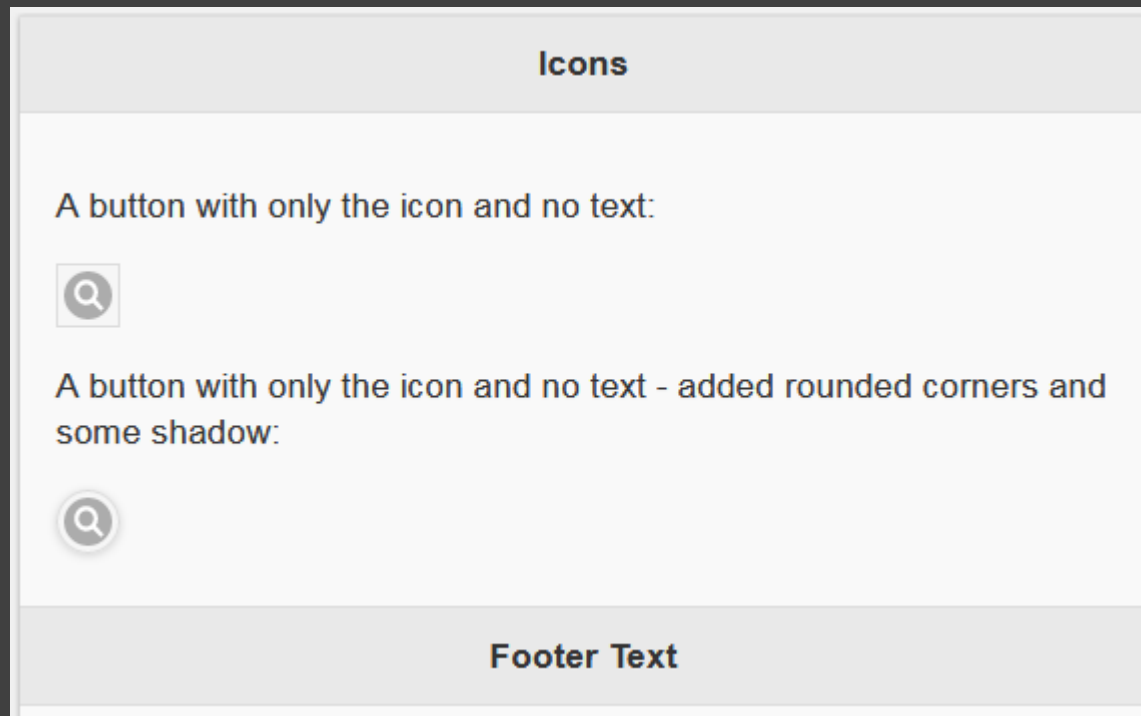
If you do not specify the icon position for link buttons, the icon will not be shown.

Icon class	Description	Icon
ui-icon-arrow-l	Left Arrow	
ui-icon-arrow-r	Right Arrow	
ui-icon-info	Information	
ui-icon-delete	Delete	
ui-icon-back	Back	
ui-icon-audio	Speakers	
ui-icon-lock	Padlock	
ui-icon-search	Search	
ui-icon-alert	Alert	
ui-icon-grid	Grid	
ui-icon-home	Home	

A full list of available button icons is available at:

https://www.w3schools.com/jquerymobile/jquerymobile_ref_icons.asp

jQuery Mobile Button Icons only



```
<body>

<div data-role="page" id="pageone">
  <div data-role="header">
    <h1>Icons</h1>
  </div>

  <div data-role="main" class="ui-content">
    <p>A button with only the icon and no text:</p>
    <a href="#" class="ui-btn ui-icon-search ui-btn-
icon-notext">Search</a>
    <p>A button with only the icon and no text - added
rounded corners and some shadow:</p>
    <a href="#" class="ui-btn ui-shadow ui-corner-all
ui-icon-search ui-btn-icon-notext">Search</a>
  </div>

  <div data-role="footer">
    <h1>Footer Text</h1>
  </div>
</div>

</body>
```

```

<div data-role="main" class="ui-content">
  <p>A white "search" icon (default):</p>
  <a href="#" class="ui-btn ui-btn-inline ui-icon-search
ui-btn-icon-left ui-corner-all ui-shadow">White icon
(default)</a>
  <a href="#" class="ui-btn ui-btn-inline ui-icon-search
ui-btn-icon-notext ui-corner-all ui-shadow">White icon
(default)</a>

  <p>A black "search" icon (class="ui-alt-icon"):</p>
  <a href="#" class="ui-btn ui-btn-inline ui-icon-search
ui-btn-icon-left ui-corner-all ui-shadow ui-alt-
icon">Black icon</a>
  <a href="#" class="ui-btn ui-btn-inline ui-icon-search
ui-btn-icon-notext ui-corner-all ui-shadow ui-alt-
icon">Black icon</a>


  <p>A black "search" icon without the gray circle
(combining "ui-nodisc-icon" and "ui-alt-icon"):</p>
  <a href="#" class="ui-btn ui-btn-inline ui-icon-search
ui-btn-icon-left ui-corner-all ui-shadow ui-nodisc-icon
ui-alt-icon">Black icon</a>
  <a href="#" class="ui-btn ui-btn-inline ui-icon-search
ui-btn-icon-notext ui-corner-all ui-shadow ui-nodisc-icon
ui-alt-icon">Black icon</a>
</div>

```

Icons


A white "search" icon (default):

White icon (default)




A black "search" icon (class="ui-alt-icon"):

Black icon



A black "search" icon without the gray circle (combining "ui-nodisc-icon" and "ui-alt-icon"):

Black icon



Footer Text

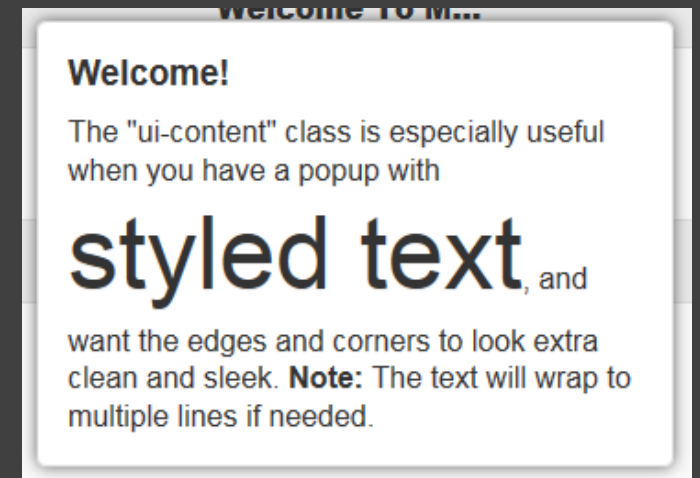
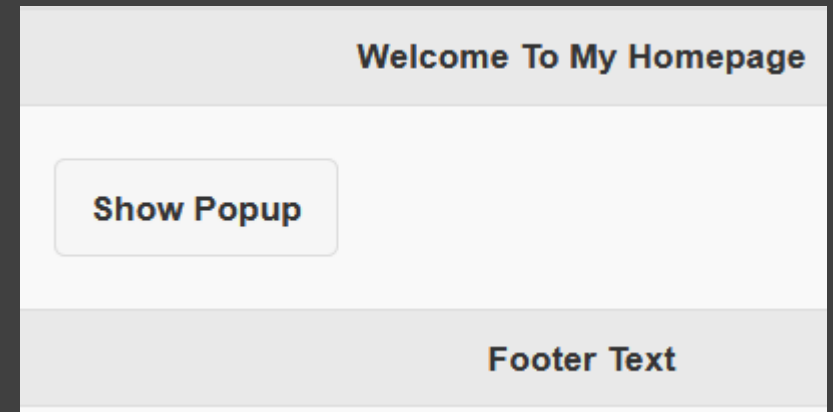
jQuery Mobile Popups

- Popups are similar to dialogs, in that they both overlay a part of a page. A popup box can be useful when you want to display small text, photos, maps or other content.
- To create a popup, start with an `<a>` element and a `<div>` element. Add the `data-rel="popup"` attribute to `<a>`, and the `data-role="popup"` attribute to `<div>`. Then specify an id for `<div>`, and set the href of `<a>` to match the specified id. The content inside `<div>` is the actual content that will pop up when a user clicks on the link.

jQuery Mobile Popups

```
<div data-role="main" class="ui-content">
  <a href="#myPopup" data-rel="popup" class="ui-btn
ui-btn-inline ui-corner-all">Show Popup</a>

  <div data-role="popup" id="myPopup" class="ui-
content">
    <h3>Welcome!</h3>
    <p>The "ui-content" class is especially useful when
you have a popup with <span style="font-size:55px;">styled
text</span>, and want the edges and corners to look extra
clean and sleek. <strong>Note:</strong> The text will wrap
to multiple lines if needed.</p>
  </div>
</div>
```



jQuery Mobile Touch Events

- Touch events are triggered when the user touches the screen (page).
- Touch events also work on desktop computers: tap and swipe with your mouse!

jQuery Mobile Tap

- The tap event is triggered when the user taps on an element.
- The following example says: When a tap event fires on a <p> element; hide the current <p> element:

Example

- ```
$("#p").on("tap",function(){
 $(this).hide();
});
```

# jQuery Mobile Taphold

---

- The taphold event is triggered when the user taps on an element and hold for one second:

## Example

- ```
$( "p" ).on( "taphold", function() {  
    $( this ).hide();  
});
```

jQuery Mobile Swipe

- The swipe event is triggered when the user swipes over an element horizontally by more than 30px:

Example

- ```
$("p").on("swipe", function() {
 $("span").text("Swipe detected!");
});
```



# jQuery Mobile Swipeleft

---

- The swipeleft event is triggered when the user swipes over an element in the left direction by more than 30px:

## Example

- ```
$( "p" ).on( "swipeleft", function() {  
    alert( "You swiped left!" );  
});
```

jQuery Mobile Swiperight

- The swiperight event is triggered when the user drags over an element in the right direction by more than 30px:

Example

- ```
$("#p").on("swiperight",function(){
 alert("You swiped right!");
});
```

# jQuery Mobile orientationchange Event

---

- The orientationchange event is triggered when the user rotates the mobile device vertically or horizontally.
- To use the orientationchange event, attach it to the window object:
  - ```
$(window).on("orientationchange",function(){  
    alert("The orientation has changed!");  
});
```

jQuery Mobile orientationchange Event

- The callback function can have one argument, the event object, which returns the orientation of the mobile device: "portrait" (the device is held in a vertical position) or "landscape" (the device is held in a horizontal position):

- **Example**

- `$(window).on("orientationchange",function(event){
 alert("Orientation is: " + event.orientation);
});`

jQuery Mobile orientationchange Event

- Because the orientationchange event is bound to the window object, we can use the window.orientation property to, for example, set different styles to distinguish between portrait and landscape views:

- **Example**

```
$(window).on("orientationchange",function(){  
    if(window.orientation == 0) // Portrait  
    {  
        $("p").css({"background-color":"yellow","font-size":"300%"});  
    }  
    else // Landscape  
    {  
        $("p").css({"background-color":"pink","font-size":"200%"});  
    }  
});
```

The window.orientation property returns 0 for portrait and 90 or -90 for landscape view.

Quick Review

- **JavaScript Frameworks**

- jQuery

- Syntax
 - Selectors
 - Events

Fin

FINALLY