

Creating Dynamic Web Apps with jQuery

February 23, 2012

Outline

- Prerequisites
 - DOM events and event handlers
 - document.ready
 - JSON
- jQuery
 - jQuery object
 - jQuery selector
 - DOM manipulation
 - events
 - ajax
 - mini exercise: update lab 2 assignment with ajax calls
- jQM
 - data fields
 - touch events
 - PHP includes
 - theming
 - jQM theme roller
- Main exercise
 - multipage mobile app
 - dynamic updates
 - twitter api call
 - event binding

Prerequisites

- DOM events (i.e., *click*, *documentready*)
- Event listeners (i.e., *onclick*, *ondocumentready*)
- Event handlers (i.e., *someFunctionToDoSomethingOnClick*, *someFunctionToDoSomethingOnDocumentReady*)
- JSON – JavaScript Object Notation
- lambda functions

What is jQuery

- jQuery
 - a JS library
 - a convenient wrapper object
 - a function, a first-order object
 - <http://api.jquery.com/category/core/>
- note: \$ is the alias for the jQuery object

jQuery Selector

- `$(selector)` – selects set of objects based on the selector string and returns them wrapped in a jQuery object
 - extended from CSS rule-matching syntax
 - ex: `$("p")` ← select all paragraphs
 - ex: `$("input[type='button']")` ← select all buttons where attribute “type” equals “button”
 - ex: `$("p:first")` ← select first paragraph

DOM Manipulation with jQuery

- `$(html)` – build html object
 - ex: `$("<p></p>")` – create an empty paragraph
- `$(DOM object).append(html object or text)`
 - `$(document.body).append($("<p>Hi!</p>"))` – adds a paragraph containing the text “Hi!” to the end of the body element
 - <http://api.jquery.com/category/manipulation/>
- Note: CSS rules are only applied to objects that are part of the DOM. After creating an object with jQuery, it needs to be attached to some member of the DOM before it too becomes part of the DOM.

jQuery Event Handling

- jQuery objects come with a lot of convenience methods for attaching event listeners and handlers
 - <http://api.jquery.com/category/events/>
- `$(callback)` is the same as `$(document).ready(callback)`
- The onclick value of a button input can be replaced with `$("input[type='button']").click(someFunction);`
- To attach event listeners based on selectors (similarly to how CSS rules are applied), use `$(parent).bind(selector, event, handler)`

Ajax (simplified with jQuery)

- JSON is frequently used to pass arguments to jQuery functions and setup objects, such as event listeners.
- Specify an ajax call with `$.ajax(settings)`, where settings is a JS object with the various parameters needed to do the call
 - <http://api.jquery.com/jQuery.ajax/>

Exercise 1 - jQuery

- Upgrade Lab 2 to make updates dynamic (no page refreshing)
 1. link JS files: site.js and some version of jQuery
 2. replace calls to the backend with \$.ajax()
 3. prevent page refresh when posts are submitted
 4. add new posts to the top of the list
 5. **advanced: add a filter for posts

What is jQuery Mobile

- Introduces \$.mobile object
- Automatically applies mobile-friendly styles and classes to DOM elements
- new events, some of which are required
 - *mobileinit*
 - Hold similar importance to document.ready.
 - should bind initialization code to this
- emphasis on accessibility

jQM Page Structure

- <http://jquerymobile.com/demos/1.0.1/docs/pages/page-anatomy.html>
- Everything loaded into one page
- Use PHP includes to include each page into main file
 - allows team members to work separately and not worry about stepping on each other's work
- “data-role” attribute tells jQM how to treat the element
 - “page” (on a div) defines a page, a view
 - “header” (on a div inside a “page”) is page header content holder
 - “content” (on a div inside a “page”) is page content holder
 - “button” (on an a) defines a linked button
 - “list-view” (on a UL or OL) defines a list view

Dynamically Changing Content with jQM

- <http://jquerymobile.com/test/docs/api/data-attributes.html>
- jQM dynamically marks up widgets when pages are created; therefore, applying a class to an element in your static HTML file may not reflect the changes you would expect.
 - This does not affect dynamically added content. In order to jQM style a newly attached DOM element, you have to call the relevant method. For example, if you dynamically add a UL, you would then have to select it with jQuery and call `.listview()` to tell jQM to convert it.
 - Ex: After programmatically adding a LI to a list, you have to grab this list and call `.listview('refresh')` for the change to show in the view.
- Tip: Give IDs to any element you will have to frequently update dynamically (i.e., a dynamically updated list).

jQM Touch Events

- <http://jquerymobile.com/demos/1.0.1/docs/api/events.html>
- jQM events: *pageshow*, *pagebeforeshow*
- new user interaction events: *tap*, *swipe*
- You must read the documentation to learn the order in which these events are called and when they are called in order to determine which events your app needs to listen for

Theming

- With jQuery you can use themeroller or regular CSS
 - <http://jqueryui.com/themeroller/>
- With jQM you can use jQM themeroller or copy a default template and edit CSS
 - <http://jquerymobile.com/themeroller/>
 - <http://jquerymobile.com/test/docs/api/themes.html>
- note: jQuery UI does not integrate perfectly with jQM

Exercise 2 – jQuery Mobile

- Make a live tweet feed
 - Build asynchronous call to twitter API to grab tweets
 - dataType: 'jsonp'
 - JSONP, unlike JSON, is supported in cross-domain calls
 - add tweets to list items
- Advanced: Allow for customized search
 - make a new first page that allows you to change the query to the twitter api
 - switches to the updated feed page on submission

Downloads and References

- <http://isites.harvard.edu/fs/docs/icb.topic1048413.files/lab3.zip>
- <http://isites.harvard.edu/fs/docs/icb.topic1048413.files/lab3.pdf>