**jQuery's Ajax-Related Methods: Part 1**

While jQuery does offer many Ajax-related convenience methods, the core $.ajax method is at the heart of all of them, and understanding it is imperative. We'll review it first, and then touch briefly on the convenience methods.

*I generally use the $.ajax method and do not use convenience methods. As you'll see, it offers features that the convenience methods do not, and its syntax is more easily understandable, in my opinion.*

**$.ajax**

jQuery’s core $.ajax method is a powerful and straightforward way of creating Ajax requests. It takes a configuration object that contains all the instructions jQuery requires to complete the request. The $.ajax method is particularly valuable because it offers the ability to specify both success and failure callbacks. Also, its ability to take a configuration object that can be defined separately makes it easier to write reusable code. For complete documentation of the configuration options, visit <http://api.jquery.com/jQuery.ajax/>.

**Example 7.1. Using the core $.ajax method**

$.ajax({

// the URL for the request

url : 'post.php',

// the data to send

// (will be converted to a query string)

data : { id : 123 },

// whether this is a POST or GET request

type : 'GET',

// the type of data we expect back

dataType : 'json',

// code to run if the request succeeds;

// the response is passed to the function

success : function(json) {

$('<h1/>').text(json.title).appendTo('body');

$('<div class="content"/>')

.html(json.html).appendTo('body');

},

// code to run if the request fails;

// the raw request and status codes are

// passed to the function

error : function(xhr, status) {

alert('Sorry, there was a problem!');

},

// code to run regardless of success or failure

complete : function(xhr, status) {

alert('The request is complete!');

}

});