Using the JQuery Cookie plugin to emulate server side forms

Posted At : 5 July 2010 12:27 | Posted By : Shaun McCran
Related Categories: JQuery, CSS, Javascript

A project request recently came up where we needed to track some users responses to a series of questions, then build an account 'home' page based on those responses. Pretty straight forward you'll agree. The problem here arose from the clients using a bespoke online solution that blocked any kind of server side interaction.

I've played with JavaScript cookies in the past to remember display states in JQuery functions, so I thought they would be a great solution to this. This article deals with how to use JavaScript cookies to emulate a normal form submission.

There is a [demo of using the JQuery Cookie plugin to emulate forms here](#) .

Because we can't use any server side requests we are going to store all the users' choices in their cookie scope. For the cookie handling I'm using a JQuery plugin, http://plugins.jquery.com/project/cookie . It is a great little plugin that allows you to read, write and delete cookies. Firstly I've built a simple form, so that you can choose some options.

```
<form action="results.cfm" method="post">

                        <div>
        <div class="label">What is your favourite colour?</div>
    <div class="field"><select name="colour" style="width: 100px;">
                        <option value="">Select</option>
                        <option value="red">Red</option>
                    <option value="green">Green</option>
                    <option value="blue">Blue</option>
                    <option value="yellow">Yellow</option>
                        </select></div>
                        </div>
                    <p><br></p>
                        <div>
        <div class="label">Choose one of these three foods</div>
                    <p><br></p>
```

```
                                    <div>
                                <h4>Cheese</h4>
"cheese.jpg" alt="cheese" height="100" width="100"><input type="radio" name="foods

                                <h4>Ham</h4>
 src="ham.jpg" alt="ham" height="100" width="100"><input type="radio" name="foods"

                                <h4>Pickles</h4>
pickles.jpg" alt="pickles" height="100" width="100"><input type="radio" name="foods
                                    </div>
                                </div>

                <input type="submit" name="Action" value="Submit">

                                </form>
```

As you can see it is very simple. It only contains a Select field, and a set of Radio values. I've chosen different field types to demonstrate that this works across different form field types.

Next we will create a JavaScript function that has the form fields listed as selectors.

```
            <s/cript language="javascript">
                $(document).ready(function() {
                    // cookie handling function
                $('select, input').change(function() {

                    var cookieName = $(this).attr('name');
                    var cookieValue = $(this).attr('value');

                            // set cookie
                    $.cookie(cookieName, cookieValue);

                            })
                        });
                </s/cript>
```

This function will watch for a change event against any 'select' and 'input' fields, and set a cookie values based on the name of the field, and the

value of the field. The syntax for the $.cookie() plugin is incredibly simple. Here we are creating cookies, passing in the name of the cookie, and the value we want the cookie to have.

If it already exists it will simply overwrite it. It works in a very similar way to the cfcookie tag.

Next we are passed to a 'results.cfm' page, where we read the value of the cookies and change a few display options accordingly.

```javascript
<s/cript language="javascript">

$(document).ready(function() {

        // colour cookie
var colourValue = $.cookie('colour');

$("#cookieColour").text(colourValue);
$("#cookieColour").css({'color' : colourValue, 'font-size': '20px'});

        // food cookie
var foodValue = $.cookie('foods');
$("#food").attr({src: foodValue + '.jpg'});

        });
</s/cript>

favourite colour is: <span id="cookieColour" style="border: 1px #ff0000 solid"></s

Your favourite food is:
<img id="food" width="50" height="50">
```

The script above is setting a JavaScript variable called 'colourValue' as the value of the cookie we just created, then using the JQuery .css function we are dynamically writing out text in the right colour.

The second value 'foodValue' is simply assigning a src value to an html image tag.

There is a demo of using the JQuery Cookie plugin to emulate forms here .