Utility Tips and Tricks

pluralsight
hardcore developer training

Using the $.map Method

Using the $.grep Method

Using the $.type Method

Feature Detect Not $.browser Detect

Using the $.Callbacks Object

When and How to use jQuery.noConflict

Using the $.extend Method

Utility

Using the $.map Method

pluralsight
hardcore developer training

Using the $.map() Method

```
var people = [
    { fn: "John", ln: "Doe",
    { fn: "Jane", ln: "Doe",
    { fn: "Joe",  ln: "Doe",
];

people = $.map(people, funct
    return {
        firstName: person.fn
        lastName: person.ln,
        age: moment().diff(m
    };
});
```

```
[
    {
        "firstName": "John",
        "lastName": "Doe",
        "age": 35
    },
    {
        "firstName": "Jane",
        "lastName": "Doe",
        "age": 33
    },
    {
        "firstName": "Joe",
        "lastName": "Doe",
        "age": 3
    }
]
```

Use the $().map() Method

```
<input id="name" value="Pluralsight" />
<input id="url"  value="http://pluralsight.com/" />

var output = $("input").map(function() {
    return $(this).val();
}).get().join(" ");

console.log(output);
```

```
Pluralsight http://pluralsight.com/
```

DEMO: Using the $.map Method

Utility

Use the $.grep Method

pluralsight
hardcore developer training

Use the $.grep() Method

```
var people = [
    { "firstName": "John", '
    { "firstName": "Jane", '
    { "firstName": "Joe",  '
];

people = $.grep(people, func
    return person.age > 18;
});
```

```
[
    {
        "firstName": "John",
        "lastName": "Doe",
        "age": 35
    },
    {
        "firstName": "Jane",
        "lastName": "Doe",
        "age": 33
    }
]
```

```
},
},
```

Utility

DEMO: Use the $.grep Method

pluralsight
hardcore developer training

Using the $.type Method

Using the typeof Operator

```
typeof true                            // boolean
typeof 10                              // number
typeof "Elijah"                        // string
typeof function() {}                   // function
typeof undefined                       // undefined
typeof { name: "Elijah" }              // object
typeof null                            // object
typeof new Error()                     // object
typeof [{ name: "Elijah" }]            // object
typeof new Date()                      // object
typeof /^\w+$/                         // object
```

Using the $.type Method

```
$.type(true)                            // boolean
$.type(10)                              // number
$.type("Elijah")                        // string
$.type(function() {})                   // function
$.type(undefined)                       // undefined
$.type({ name: "Elijah"})               // object
$.type(null)                            // null
$.type(new Error())                     // error
$.type([{ name: "Elijah"}])             // array
$.type(new Date())                      // date
$.type(/^\w+$/)                         // regexp
```

Utility

DEMO: Using the $.type Method

pluralsight
hardcore developer training

Utility

Feature Detect Not $.browser Detect

pluralsight
hardcore developer training

$.browser Is No Longer Supported

```
<script src="jquery-1.9.0.min.js"></script>
<script src="jquery-migrate-1.2.1.js"></script>
<script>
function doAwesomeStuff() {
    if ($.browser.msie && $.browser.version === "6.0") {
          return; // Browser Not supported
    }
    // ... more code ...
}
</script>
```

- jQuery 1.8 deprecated the $.browser object and 1.9 removed it
- However, there is a jQuery Migration plugin that restores $.browser

Detect Features Instead of Browsers

- Feature Detection is preferred over Browser Sniffing

```
if (!Modernizr.input.placeholder) {
    jQuery.getScript("jquery.placeholder.min.js",
        function() { $("input,textarea").placeholder() })
}

Modernizr.load({
    test: Modernizr.input.placeholder,
    nope: ["jquery.placeholder.min.js"],
    complete:
        function() { $("input,textarea").placeholder() }
});
```

DEMO: Feature Detect Not $.browser Detect

pluralsight
hardcore developer training

Utility

Using the $.Callbacks Object

pluralsight
hardcore developer training

Using the $.Callbacks Object

```
var calculator = {
    add: function (operand1, operand2) {
        console.log(operand1 + operand2);
    },
    multiply: function (operand1, operand2) {
        console.log(operand1 * operand2);
    }
};

var callbacks = $.Callbacks();
callbacks.add(calculator.add);
callbacks.add(calculator.multiply);
callbacks.fire(3, 3);
```

Utility

DEMO: Using the $.Callbacks Object

pluralsight
hardcore developer training

Utility

When and How to use jQuery.noConflict

pluralsight
hardcore developer training

Utility

Using jQuery Alongside Another Library

```
<script src="prototype.js"></script>
<script src="jquery.js"></script>
<script>
    $.noConflict();
    // `$` is now back to Prototype

    jQuery(document).ready(function ($) {
        // `$` is jQuery in this scope
    });

    (function ($) {
        // `$` is jQuery in this scope
    }(jQuery));
</script>
```

Utility

Using jQuery Alongside Another jQuery

```
<script src="jquery-1.10.1.js"></script>
<script src="jquery-1.6.2.js"></script>
<script>
// Remove jQuery 1.6.2 ($ and jQuery) from global scope
and restore previous version (1.10.1)
var jq162 = jQuery.noConflict(true);

console.log("1st jQuery: " + $.fn.jquery);     // 1.10.1
console.log("2nd jQuery: " + jq162.fn.jquery); // 1.6.2
</script>
```

DEMO: When and How to use jQuery.noConflict

pluralsight
hardcore developer training

Utility

Using the $.extend Method

pluralsight
hardcore developer training

Utility

Using the $.extend Method

```
jQuery.fn.valentines = function (options) {
    var settings = $.extend(
        {},
        { color: "red", fontSize: "16px" },
        options
    );

    return this.css(settings);
};

$("a").valentines({ color: "#A00000" });
```

Utility

DEMO: Using the $.extend Method

pluralsight
hardcore developer training

Summary

- **jQuery has a set of helpful lesser known Utility methods**
  - $.map manipulates and messages
  - $.grep filters
  - $.type provides richer type information
  - $.browser migration plugin
  - $.Callbacks Object
  - $.noConflict Mode
  - $.extend merges