jQuery

jQuery - Generating a Table of Contents

- The Table of Contents jQuery Code
- Creating a Table of Contents jQuery Plugin

Jakob Jenkov
Last update: 2015-01-26

This tutorial shows how to use jQuery to generate a table of contents inside an HTML page, based on the headlines in the HTML page (the h1 to h3 elements).

Imagine you have an HTML page with headlines using the header HTML tags <h1>, <h2>, and <h3>. Imagine too, that you need to generate an internal table of contents for this page.

The <h1> element contains the page title, so you don't really want that as part of the internal table of contents. You just want the table of contents to contain the <h2>, <h3> elements in the page.

Here is an example of an HTML page like the one mentioned above:

```
<html>

<head>
<script type="text/javascript"
    src="//ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js">
</script>
</head>

<body>

<h1>Title</h1>

<div id="tocDiv">
<ul id="tocList">

</ul>
</div>

<h2>Chapter 1</h2>
<h3>Section 1.1</h3>
<h2>Chapter 2</h2>
<h3>Section 2.1</h3>
<h2>Chapter 3</h2>
<h3>Section 3.1</h3>
<h3>Section 3.2</h3>


<script>
$(document).ready(function() {
```

```
});
</script>

</body>

</html>
```

Notice the div with the id `tocDiv`, and the `ul` inside with the id `tocList`. It is inside the inner `ul` element that you want the table of contents to go, as `li` elements with nested `ul` and `li` elements inside.

Also notice the empty function passed to the `$(document).ready()` function. Inside this function is where you will put the code that generates the table of contents.

The list of h2 and h3 elements are what you want to list in the table of contents. The body contents of the h2 and h3 elements, that is, plus a link down to where the h2 or h3 element is located.

The Table of Contents jQuery Code

Here is the code that generates the table of contents:

```
$(document).ready(function() {
    $(tocList).empty();

    var prevH2Item = null;
    var prevH2List = null;

    var index = 0;
    $("h2, h3").each(function() {

        //insert an anchor to jump to, from the TOC link.
        var anchor = "<a name='" + index + "'></a>";
        $(this).before(anchor);

        var li     = "<li><a href='#" + index + "'>" + $(this).text() + "</a></li>";

        if( $(this).is("h2") ){
            prevH2List = $("<ul></ul>");
            prevH2Item = $(li);
            prevH2Item.append(prevH2List);
            prevH2Item.appendTo("#tocList");
        } else {
            prevH2List.append(li);
        }
        index++;
    });
```

```
});
```

This code iterates through all h2 and h3 elements using the jQuery call `$("h2, h3").each(...)`.

For each h2 or h3 element, an is inserted in the HTML document just before it. That is necessary so the links in the table of contents can jump down to the corresponding h2 or h3 element. The index variable inserted is incremented from 0 and up.

Both h2 and h3 elements have an li element created to hold the link in the table of contents. h2 elements also have an inner ul element created, which will hold the h3 elements found between this and the next h2 element. This is how h3 elements become nested underneath the latest h2 element before them.

Inserting the above code into the HTML page shown earlier, will generate the following table of contents at the top of the page:

- **Chapter 1**
  - **Section 1.1**
- **Chapter 2**
  - **Section 2.1**
- **Chapter 3**
  - **Section 3.1**
  - **Section 3.2**

Creating a Table of Contents jQuery Plugin

Let us try to turn the table of contents code into a jQuery plugin. Then you can see how to turn a real example into a jQuery plugin. How to create a jQuery plugin is covered in my **jQuery Plugins Tutorial**, so I will not repeat that here.

First the code inside the $(document).ready(...) function call is moved to its own function. Now the code inside the <script> element looks like this:

```
$(document).ready(function() {
    tableOfContents("#tocList");
});

function tableOfContents(tocList) {
    $(tocList).empty();
    var prevH2Item = null;
    var prevH2List = null;
    var index = 0;
```

```
    $("h2, h3").each(function() {
        //insert an anchor to jump to, from the TOC link.
        var anchor = "<a name='" + index + "'></a>";
        $(this).before(anchor);

        var li      = "<li><a href='#" + index + "'>" +
                        $(this).text() + "</a></li>";

        if( $(this).is("h2") ){
            prevH2List = $("<ul></ul>");
            prevH2Item = $(li);
            prevH2Item.append(prevH2List);
            prevH2Item.appendTo(tocList);
        } else {
            prevH2List.append(li);
        }
        index++;
    });
}
```

Since all the code fits inside a single function it is a natural fit for a single function jQuery plugin. Here is how the jQuery plugin code looks:

```
jQuery.tableOfContents =
function (tocList) {
    jQuery(tocList).empty();
    var prevH2Item = null;
    var prevH2List = null;

    var index = 0;
    jQuery("h2, h3").each(function() {

        var anchor = "<a name='" + index + "'></a>";
        jQuery(this).before(anchor);

        var li      = "<li><a href='#" + index + "'>" +
                        jQuery(this).text() + "</a></li>";

        if( jQuery(this).is("h2") ){
            prevH2List = jQuery("<ul></ul>");
            prevH2Item = jQuery(li);
            prevH2Item.append(prevH2List);
            prevH2Item.appendTo(tocList);
        } else {
            prevH2List.append(li);
        }
```

```
        index++;
    });
}
```

This code should be put in a file called `jQuery.tocPlugin.js`.

Notice how all uses of the `$()` function has now been replaced with the `jQuery()` function instead. You should always use the `jQuery()` function, when creating a jQuery plugin.

The jQuery plugin can now be used like this:

```
<html>

<head>
<script type="text/javascript"
        src="//ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js">
</script>

<script type="text/javascript"
        src="jquery.tocPlugin.js">
</script>

</head>

<body>

<div id="tocDiv">
<ul id="tocList">

</ul>
</div>

<h2>Chapter 1</h2>
<h3>Section 1.1</h3>
<h2>Chapter 2</h2>
<h3>Section 2.1</h3>
<h2>Chapter 3</h2>
<h3>Section 3.1</h3>
<h3>Section 3.2</h3>


<script>
$(document).ready(function() {
    jQuery.tableOfContents("#tocList");
});
</script>
```

```
    </body>

</html>
```

The code marked in bold is what is changed in the HTML page in order to use the table of contents plugin.

That is all there is to it. Now you can spray table of contents all over your HTML pages.

Next: jQuery - Creating an Expandable Tree

Tweet

Jakob Jenkov