Display a Hidden Div When a Page Loads and Based on Scroll-bar Position

- **http://technerdia.com/1301_jquery-fade-in.html**
- **Demo: http://demo.technerdia.com/jquery-fade-in/**
- **Download The Demo**

Loading a hidden div when the page loads or when the scroll-bar moves pulls a visitors eye towards the faded-in feature. For areas like advertisements or opt-in boxes this effect can make a feature stand out from the rest of the Website.

I use this nifty trick to load my code examples once the page is loaded, and when the scroll-bar reaches 600px from the bottom, the opt-in box at the bottom of my posts loads. Split tests thus far have not improved my opt-in rates, but it has not reduced them either, thus the feature will stay.

Setting Up

Before we get to the jQuery that does the magic, a few features need to be setup first.

Loading jQuery into your Wordpress Headers

Add the function below to your themes functions.php file. The first part loads the jQuery library from Google's API and the second part calls a jquery.js script, which will be placed in your theme directory.

```
function myjQueryScript() {
```

```
        if( !is_admin() ) { /* only display if not in admin area */
                wp_deregister_script( 'jquery' );
                wp_register_script( 'jquery',
'http://ajax.googleapis.com/ajax/libs/jquery/1.8.0/jquery.js' );
                wp_enqueue_script( 'jquery' );

                /* jquery.js file in local theme folder */
                wp_deregister_script( 'default' );
                wp_register_script( 'default', get_bloginfo('stylesheet_directory').'/jquery.js' );
                wp_enqueue_script( 'default' );
        }
}
add_action( 'wp_print_scripts', 'myjQueryScript' );
```

Adding the jQuery for Non-Wordpress Websites

If you are not using Wordpress then you will have to manually add the javascript lines to your html documents (header files), above the close of the < / head > tag.

- • Change out the your-domain.com and path to the jquery.js file within the 2nd script line.

```
<script type='text/javascript'
src='http://ajax.googleapis.com/ajax/libs/jquery/1.8.0/jquery.js'></script>
<script type='text/javascript' src='http://your-domain.com/jquery.js'></script>
```

HTML and CSS Requirements

Next, it is time to add the needed HTML and CSS calls.

For this example the DIV class names are named div1, div2, div3, and div4. If you change the class names, you will need to also update the CSS and jquery.js file to match the new class names.

The HTML

The divs are rather straight forward, simply add in whatever text or images you would like to display.

```
<div class="div1">Content of Div Goes Here</div>
<div class="div2">Content of Div Goes Here</div>
<div class="div3">Content of Div Goes Here</div>
<div class="div4">Content of Div Goes Here</div>
```

The CSS

The CSS contains 2 opacity rules per class, which hides the divs from view. Older browsers

do not know what opacity is, so they will ignore opacity call and simply display the div.

```css
.div1 {
        opacity:0;
        filter:alpha(opacity=0);
}

.div2 {
        opacity:0;
        filter:alpha(opacity=0);
}

.div3 {
        opacity:0;
        filter:alpha(opacity=0);
}

.div4 {
        opacity:0;
        filter:alpha(opacity=0);
}
```

jQuery Fade In Examples

Below are 4 examples of displaying hidden divs with jQuery. The first example loads the moment that jQuery is loaded, and the rest load depending on the position of the scroll-bar.

Place the examples below in a jquery.js file and upload it. For Wordpress users, the jquery.js file is placed within your active theme directory. For standalone Websites, upload the jquery.js file and make sure the <script> header line (above) points to the proper location.

Load Hidden DIV When Page Loads

Line 3 contains the class name .div1 – This example fades in the div container when the page and jQuery are first loaded.

```javascript
/* Load DIV When Page Loads */
        $(window).load( function () {
                $('.div1').fadeIn('slow', function() {
                        $(this).fadeTo("slow", 1);
                });
        });
```

Load Hidden DIV After Scrollbar is Moved 100px

The first part of this jQuery example determines where the scroll-bar location is, once the location is reached it calls the function loadDiv2 that fades in the class named .div2.

- Line 3 tells how far down from the top you must scroll before the fade-in function is called. Line 3 is currently set to 100, meaning once the scroll-bar reaches 100px from the top, the div will fade-in.

- Line 4 calls the function loadDiv2. The loadDiv2 function is located on line 8.

- Line 9 contains the class name .div2.

```
/* Load DIV After Scrollbar Is Moved 100px */
      $(window).scroll( function() {
    if ( $(window).scrollTop() > 100 ) {
      loadDiv2(); /* calls loadDiv2 Function below */
    }
      });
/* the loadDiv2 function */
      function loadDiv2() {
            $('.div2').fadeIn('slow', function() {
                  $(this).fadeTo("slow", 1);
            });
      }
```

Load Hidden DIV When Scrollbar is 200px from the Bottom of the Page

Again, the first part of this jQuery example determines where the scroll-bar location is, which then calls the function loadDiv3 that fades in the class named .div3.

- Line 3 tells how far down from the bottom the scroll-bar is before the fade-in function is called. Line 3 is currently set to 200, meaning once the scroll-bar reaches 200px from the bottom of the page, the div will fade-in.

- Line 4 calls the function loadDiv3. The loadDiv3 function is located on line 8.

- Line 9 contains the class name .div3.

```
/* Load DIV When Scrollbar is Near the Bottom of the Page */
      $(window).scroll( function() {
    if ( $(window).scrollTop() + $(window).height() >= $(document).height() - 200 ) {
      loadDiv3(); /* calls loadDiv3 Function below */
    }
      });
/* the loadDiv3 function */
      function loadDiv3() {
            $('.div3').fadeIn('slow', function() {
                  $(this).fadeTo("slow", 1);
            });
      }
```

Load Hidden Div Once Scroll-bar Reaches the Bottom of the Page

And finally, the first part of this jQuery example determines when the scroll-bar reaches the bottom of the page, which then calls the function loadDiv4 that fades in the class named .div4.

- Line 4 calls the function loadDiv4. The loadDiv4 function is located on line 8.

- Line 9 contains the class name .div4.

```
/* Load Div Once Scrollbar Reaches the Bottom of the Page */
    $(window).scroll( function() {
    if ( $(window).scrollTop() + $(window).height() >= $(document).height() ) {
    loadDiv4(); /* calls loadDiv4 Function below */
    }
    });
/* the loadDiv4 function */
    function loadDiv4() {
            $('.div4').fadeIn('slow', function() {
                    $(this).fadeTo("slow", 1);
            });
    }
```

Hidden divs Not Fading In?

If the hidden divs are not loading, then check the names of the classes (div1, div2, etc). The HTML, CSS and jquery.js file must all use the same class/div names. Next, check to make sure the function calls and function names within the jquery.js file are named the same. Finally, check to make sure the jquery.js file is uploaded to the proper location. (The first 2 examples within this article.)

No matter what, if they are not loading, something is named incorrectly or the jQuery file is in the wrong location.

Have a comment? Find an error? Would you like to improve on this article? Then shoot me some feedback! ~tribalNerd

---

Follow techNerdia on Google+ | Follow Chris on Google+
Subscribe to techNerdia's RSS Feed