**tipsy - Facebook-style tooltip plugin for jQuery**

**Overview**

Tipsy is a jQuery plugin for creating a Facebook-like tooltips effect based on an anchor tag's title attribute.

**Examples & Usage**

**Basic**

By default, tooltips will appear centred underneath their anchor:

[Hover over me](http://onehackoranother.com/projects/jquery/tipsy/)

Basic example:

$('#example-1').tipsy();

**Gravities**

Using the gravity parameter, it's possible to control the positioning of the tooltip relative to the pointee:

|  |  |  |
| --- | --- | --- |
| [Northwest](http://onehackoranother.com/projects/jquery/tipsy/) | [North](http://onehackoranother.com/projects/jquery/tipsy/) | [Northeast](http://onehackoranother.com/projects/jquery/tipsy/) |
| [West](http://onehackoranother.com/projects/jquery/tipsy/) |  | [East](http://onehackoranother.com/projects/jquery/tipsy/) |
| [Southwest](http://onehackoranother.com/projects/jquery/tipsy/) | [South](http://onehackoranother.com/projects/jquery/tipsy/) | [Southeast](http://onehackoranother.com/projects/jquery/tipsy/) |

Gravity example:

$('#foo').tipsy({gravity: 'n'}); // nw | n | ne | w | e | sw | s | se

As of version 0.1.3, it's possible to use a callback function to set the gravity dynamically at hover-time. Within the callback, this refers to the active element, and the function should return the calculated gravity as a string. Two demo callbacks are supplied - $.fn.tipsy.autoNS and $.fn.tipsy.autoWE - which select north/south and west/east gravity, respectively, based on the element's location in the viewport.

Here's an example (scroll the page to see the effect):

[Dynamic Gravity](http://onehackoranother.com/projects/jquery/tipsy/)

Dynamic gravity example:

$('#foo').tipsy({gravity: $.fn.tipsy.autoNS});

**Fading**

For full Wob2.0 compliance, you must fade these badboys in:

[Hover over me](http://onehackoranother.com/projects/jquery/tipsy/)

Fade example:

$('#example-fade').tipsy({fade: true});

**Bonus Feature**

You can EVEN COMBINE FADE AND GRAVITY! (exercise for reader)

**Slightly Advanced Usage**

Tooltip text can be set based on any attribute, not just title:

[Hover over me](http://onehackoranother.com/projects/jquery/tipsy/)

Custom attribute example:

$('#example-custom-attribute').tipsy({title: 'id'});

If any attribute isn't good enough, you may pass a callback function instead. It should return the tooltip text for thiselement:

[Hover over me](http://onehackoranother.com/projects/jquery/tipsy/)

Callback example:

$('#example-callback').tipsy({title: function() { return this.getAttribute('original-title').toUpperCase(); } });

Finally, it is possible to specify a fallback tooltip for any element which does not have any tooltip text:

[Hover over me](http://onehackoranother.com/projects/jquery/tipsy/)

Fallback example:

$('#example-fallback').tipsy({fallback: "Where's my tooltip yo'?" });

If your tooltip content contains HTML, set the html option to true (relies on invalid HTML, sorry):

[Hover over me](http://onehackoranother.com/projects/jquery/tipsy/)

HTML example:

$('#example-html').tipsy({html: true });

**Show/Hide Delay**

Delay example:

$('#example-delay').tipsy({delayIn: 500, delayOut: 1000});

[Hover and wait](http://onehackoranother.com/projects/jquery/tipsy/)

**Dynamically Updating Text**

Tipsy tooltips are 'live' - if the source attribute's value changes, the tooltip text will be updated the next time the tooltip is shown. There's one caveat - if you wish to remove the tooltip by setting the title attribute to an empty string, set the original-title attribute instead (this only applies if you're using the title attribute).

Dynamic updating example:

[Hover over me](http://onehackoranother.com/projects/jquery/tipsy/) | New tooltip text: 

**Using Tooltips on Form Inputs**

Tooltips can be bound to form inputs' focus/blur events using the option {trigger: 'focus'}:

Form input tooltips example:

  
  
  


Form input tooltips code:

<script type='text/javascript'>

$(function() {

$('#focus-example [title]').tipsy({trigger: 'focus', gravity: 'w'});

});

</script>

**Manually Triggering a Tooltip**

It's possible to disable hover events and instead trigger tooltips manually:

Manual triggering example:

My tooltip is manually triggered | [Show it](http://onehackoranother.com/projects/jquery/tipsy/) | [Hide it](http://onehackoranother.com/projects/jquery/tipsy/)

Code for manual triggering:

<p id='manual-example'>

<a rel='tipsy' title='Well hello there'>My tooltip is manually triggered</a>

<a href='#' onclick='$("#manual-example a[rel=tipsy]").tipsy("show"); return false;'>Show</a>

<a href='#' onclick='$("#manual-example a[rel=tipsy]").tipsy("hide"); return false;'>Hide</a>

</p>

<script type='text/javascript'>

$('#manual-example a[rel=tipsy]').tipsy({trigger: 'manual'});

</script>

**Support for Live Events**

Live events are supported using the option {live: true}. jQuery ≥ 1.4.1 is required and live tooltips do not support manual triggering.

Live events example:

Code for live events:

<script type='text/javascript'>

$('a.live-tipsy').tipsy({live: true});

</script>

**Summary of Configuration Options**

Here is the default options declaration:

$.fn.tipsy.defaults = {

delayIn: 0, // delay before showing tooltip (ms)

delayOut: 0, // delay before hiding tooltip (ms)

fade: false, // fade tooltips in/out?

fallback: '', // fallback text to use when no tooltip text

gravity: 'n', // gravity

html: false, // is tooltip content HTML?

live: false, // use live event support?

offset: 0, // pixel offset of tooltip from element

opacity: 0.8, // opacity of tooltip

title: 'title', // attribute/callback containing tooltip text

trigger: 'hover' // how tooltip is triggered - hover | focus | manual

};

**Notes**