



search

[FAQs](#) [Awards](#) [Usage](#) [Terms](#) [Contact](#)

[Home](#) ▶ [User and System Preference](#) ▶ [Here](#)

Categories
Calendars
Date & Time
Document Effects
Dynamic Content
Form Effects
Games
Image Effects
Links & Tooltips
Menus & Navigation
Mouse and Cursor
Scrollers
Text Animations
User/System Preference
Window and Frames
Other
XML and RSS

Other Sections
Script Forums
Recommend Us
Usage Terms
Free JavaScripts

Sweet Ads

Compatibility
▶ IE5+: IE 5 and above
▶ FF1+: Firefox 1.0+. NS6+ and FF beta are assumed as well.
▶ Opr7+: Opera 7 and above.

Bookmark [Fluid Text Resizer](#) online:

Site24x7.com

Monitor Cloud & Web Applications

Get instant alerts when your website is down



:: Fluid Text Resizer

FF1+ IE6+ Opr7+

Author: Dynamic Drive

Description: Giving users the ability to adjust the text size of your webpages can only be a good thing, as different people have very different needs when it comes to font size and comfort. With that said, this is Fluid Text Resizer, a compact yet versatile jQuery based text resizer script. It can adjust the text size of a particular section of your page, such as just the primary content area, multiple sections, or the entire document in general. A fluid animation is used to morph into the new text size, and session cookies are used to remember the new setting. It works by first getting the default font size of the target section, whether the unit is pixels, em etc, then increasing or decreasing that accordingly. In other words, it's adaptive. Here are the main highlights of this script:

- Can be set to resize the text size of either a single element on the page (ie: DIV with ID="contentarea" or the BODY element) or multiple elements at once.
- Supports a fluid animation that's applied when the text is resized, which can be disabled.
- Support for persistence of selected text size within a browser session through the use of session cookies.
- Script works on both absolute (ie:px) and relative (% , em etc) font sizes.
- Specify the possible range in terms of levels that the text size can be enlarged or shrunk to.
- Easily define and style the controls used to toggle the text size, which can just be arbitrary HTML that call one of 3 public methods of the script.

Demo: Click the below links:

Increase/Decrease controls:

Font levels controls:

Arbitrary link control: [Set font level to default](#)

Directions:

Step 1: Add the following code to the <HEAD> section of your web page:

Select All

```
<style type="text/css">
.controlstyle a{ /*links inside DIV sizecontroldiv*/
outline:none;
}

.controlstyle a img{ /*image links inside DIV sizecontroldiv*/
border-width:0;
}

.controlstyle a.selectedcontrol img{ /*selected control's image*/
border-bottom:4px solid darkred;
}
```

The above script references an external .js file [fluidtextresizer.js](#), which you should download (right click file, and select "Save As").

Step 2: Now add the following sample resize controls to your page:

Select All

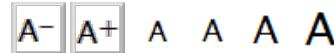
- [Bookmark](#) to del.icio.us
- [Bookmark](#) to Furl.net

[Bookmark](#)



```
<div id="sizecontroldiv" class="controlstyle">
Increase/Decrease controls: <a href="#smaller"></a> <a href="#bigger"></a><br /><br />
Font levels controls: <a href="#fontsize-1"></a> <a href="#fontsize0"></a> <a href="#fontsize1"></a> <a href="#fontsize2"></a>
</div>
<p>Arbitrary link control: <a href="javascript:mytextsizer.setFontLevel(0)">Set font level to default</a>
```

And as you may have noticed, the code above references a few images, made available below:



And that's it, for the quick and dirty installation at least! By default the script is set up to toggle the entire document's font size (BODY element), but surely you'll want to learn how to fine tune that. Read on for the customization info.

Customization

To set up the script for a page, the function to invoke in the HEAD section of your page is:

```
var uniquevar=new fluidtextresizer(options)
```

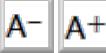
Where `uniquevar` is just an arbitrary but unique variable name, and `options` is a variable containing a list of options to pass into this instance of the script. In the code of Step 1, you see the following code, which creates a new instance of `fluidtextresizer()` and sets the script up for that page:

```
<script type="text/javascript">
var mytextsizer=new fluidtextresizer({
  controlsdiv: "sizecontroldiv", //id of special div containing your resize controls. Enter "" if not defined
  targets: ["body"], //target elements to resize text within: ["selector1", "selector2", etc]
  levels: 3, //number of levels users can magnify (or shrink) text
  persist: "session", //enter "session" or "none"
  animate: 200 //animation duration of text resize. Enter 0 to disable
})
</script>
```

This default set up looks for all the controls that will toggle the text size inside the DIV with ID="sizecontroldiv", toggles the text size of the entire BODY element, and lets the user increase and decrease the font size 3 levels in either direction. Persistence and animation are both enabled as well. The list of options and what they do is described below:

fluidtextresizer() options

option	Description
<code>controlsdiv: id</code>	<p>Set this to the ID of a DIV on your page where your resize control links are defined. Controls defined inside this DIV have the benefit of the current control being selected automatically as the user increases/ decreases the font size. Specifically, the currently selected control gets a CSS class of "<code>selectedcontrol</code>".</p> <p>If you don't need your controls to be automatically highlighted based on the current text size, you can simply set the option "<code>controlsdiv</code>" to a blank string (""), and define your controls anywhere on the page. See "Two ways to define your text size controls" for more info.</p>
<code>targets: ["selector1", "selector2", etc]</code>	<p>An array containing the elements on the page you wish to resize their text for. The actual syntax is a list of jQuery selectors that reference the desired elements. If you're not familiar with jQuery selectors, the good news is that they are almost identical to CSS selectors. Here are some examples:</p> <ul style="list-style-type: none"> • targets: ["body"] //resizes text for the BODY element (entire document) • targets: ["#contentarea"] //resizes text inside the container with <code>id="contentarea"</code> • targets: ["p.comments"] //resizes text inside paragraphs with <code>class="comments"</code> • targets: ["div#rightcolumn", "pre"] //resizes text inside the DIV with <code>id="rightcolumn"</code> and the PRE element in general. <p>While you can specify more than 1 selector (as in the last example), generally you</p>

	don't need to, and furthermore, the more selectors you specify, the more expensive the process of resizing the text. Try and minimize the number of selectors entered and avoid general selectors that match multiple elements (ie: P, H2, PRE etc) when possible.
<code>levels: integer</code>	A positive integer that sets the number of "levels" the user can increase/decrease the text size by. Once the user goes pass that level, no further changes are made. Note: Each level of change is around 1.2x the previous font size in either direction. Lets say you've defined "increase and "decrease" controls on your page: 
	When "levels" is set to 3, that means the two controls will cease to work further when they have been clicked on 3 times.
<code>persist: "string"</code>	Specify whether changes to the text size should be persisted for the duration of the browser session: enter "session" or "none".
<code>animate: milliseconds</code>	Sets the duration of the animation in milliseconds when the font is resized. To disable animation all together, set it to 0.

► Two ways to define your text size controls

Once you've initialized an instance of Fluid Text Resizer on the page, next comes the fun part- defining the controls for it.

To better explain things, lets create a new instance of Fluid Text Resizer that changes the text size of specific DIVs on the page, and illustrate the two ways you can create controls for it.

```
<script type="text/javascript">

var mytextsizer2=new fluidtextresizer({
  controldiv: "mycontentcontrol", //id of special div containing your resize controls. Enter "" if not defined
  targets: [".mycontent"], //target elements to resize text within: ["selector1", "selector2", etc]
  levels: 2, //number of levels users can magnify (or shrink) text
  persist: "session", //enter "session" or "none"
  animate: 200 //animation duration of text resize. Enter 0 to disable
})

</script>
```

Sample P with Class="mycontent":

In typography, a font is traditionally defined as a complete character set of a single size and style of a particular typeface. For example, the set of all characters for 9-point Bulmer italic is a font, and the 10-point size would be a separate font, as would the 9-point upright. After the introduction of computer fonts based on fully scalable outlines, a broader definition evolved. Font is no longer size-specific, but still refers to a single style. Bulmer regular, Bulmer italic, Bulmer bold and Bulmer bold italic are four fonts, but one typeface.

So the script is initialized. Lets run down the two ways we can define the controls for it.

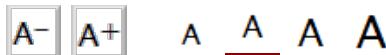
1) Define the controls inside a special DIV, with their href attribute set to special values

The first way to define your controls is to create a blank DIV anywhere on your page with a unique ID, and define your controls inside this DIV. The advantage of this method of defining your controls is that the script will automatically select the control that corresponds to the current text size when the page loads or as you toggle it:

```
<div id="mycontentcontrol" class="controlstyle">
Increase/Decrease controls: <a href="#smaller"></a> &ampnbsp<a href="#bigger"></a><br /><br />

Font levels controls: <a href="#fontsize-1"></a> <a href="#fontsize0"></a> <a href="#fontsize1"></a> <a href="#fontsize2"></a>
</div>
```

Demo controls:



Set this DIV's ID to the value specified inside the "controlsdiv" option of the script, in this case, "mycontentcontrol". Then, define arbitrary links inside it with special href attribute values to give these links special meaning. The 3 values supported are:

- **#smaller**: A link carrying this href value will shrink the text size by 1 level each time it's clicked on.
- **#bigger**: A link carrying this href value will enlarge the text size by 1 level each time it's clicked on.
- **#fontsizeX**: A link carrying this href value will set the text size to the level as defined by "x". "X" should either be positive or negative number, or 0 to set the text size to its default size. Some examples: "#fontsize-3", "#fontsize0", "#fontsize2".

In the case of **#fontsizeX**, as mentioned, a link with href set to "#fontsize0" would cause the text size to reset back to the default when clicked on, "#fontsize2" enlarge the text size by two levels etc. The link control that corresponds to the current text size will automatically be selected, by giving it a CSS class of "**selectedcontrol**". You should style this class in the HEAD section of your page as desired. Here we're doing so with a thick dark red bottom border:

```
.controlstyle a{ /*links inside DIV sizecontroldiv*/
outline:none;
}

.controlstyle a img{ /*image links inside DIV sizecontroldiv*/
border-width:0;
}

.controlstyle a.selectedcontrol img{ /*selected control's image*/
border-bottom:4px solid darkred;
}
```

Note: When using the value "#fontsizeX" inside links, "x" should not exceed the value set by the "levels" option within `fluidtextresizer()`. If this option is set to 3 for example, links with the values "#fontsize4" or "#fontsize-4" will have no effect when clicked on, as both exceed 3 as absolute numbers.

2) Define arbitrary links anywhere on your page that act as controls

You can also define arbitrary links anywhere on your page to manipulate the text size. Links defined in this manner however do not get automatically selected based on the current text size. It simply provides another way to implement the controls for your text resizer that can be placed in multiple locations on the page, or called dynamically inside a script. The JavaScript method to invoke is:

```
scriptinstance.setFontLevel(keyword)
```

Scriptinstance should be the variable name assigned to your script instance when you invoked `fluidtextresizer()`. Keyword should either be the string "smaller", "bigger", or an integer representing the level you want to change the text size to (0=reset font size to default level). With that said, here are some sample controls that make use of this method to toggle the text size of our demo paragraph:

Sample P with Class="mycontent":

In typography, a font is traditionally defined as a complete character set of a single size and style of a particular typeface. For example, the set of all characters for 9-point Bulmer italic is a font, and the 10-point size would be a separate font, as would the 9-point upright. After the introduction of computer fonts based on fully scalable outlines, a broader definition evolved. Font is no longer size-specific, but still refers to a single style. Bulmer regular, Bulmer italic, Bulmer bold and Bulmer bold italic are four fonts, but one typeface.

```
<a href="javascript:mytextsizer2.setFontLevel('smaller')"></a> &nbsp; <a
href="javascript:mytextsizer2.setFontLevel('bigger')"></a><br />

<a href="javascript:mytextsizer2.setFontLevel(-1)">Font Level -1</a> |
<a href="javascript:mytextsizer2.setFontLevel(0)">Default Level</a> |
<a href="javascript:mytextsizer2.setFontLevel(1)">Font Level 1</a> |
<a href="javascript:mytextsizer2.setFontLevel(2)">Font Level 2</a>
```

Demo controls:

[A-](#) [A+](#)

[Font Level -1](#) | [Default Level](#) | [Font Level 1](#) | [Font Level 2](#)

Notice the keyword "mytextsizer2" used above, which is just the [variable](#) name we assigned to the script instance for the paragraph above that we wish to toggle the text for.

[Resize Photos on Mac](#)

Reduce File Size up to 5 Times. Download JPEG Mini for OS X Free!

www.jpegmini.com/Mac

AdChoices ▾



Copyright 2012 (c) Dynamic Drive. All rights reserved. [Please see TOS](#) before using any of our scripts.

