

Warning: Trying to access array offset on value of type null in
/var/www/vhosts/thecmsbcookbook.com/httpdocs/recipeDetail3.php on line 44

The CMS Builder Cookbook

[Home](#)

[Read Excerpts](#)

[View the
Table of
Contents](#)

[All
User Submitted
CMS Plugins](#)

[CMSB Resources](#)

[Contact Us](#)

USING @FONT-FACE TO REPLACE GOOGLE FONTS THAT DON'T WORK IN GOOGLE CHROME (OH, THE IRONY!) - Dec 29th, 2018

Using @font-face to replace Google Fonts that don't work in Google Chrome (Irony)

Surprising as it may seem, there are some fonts (Bubbler One for example) that don't display correctly in either Google Chrome or Safari on a windows platform. (The bottom half of the font is cut off)

I assume the folks at Google will fix this embarrassment at some point, but until then, you can fix the issue using the new CSS3 @font-face tag.

A little background... (If you know this already, you can skip this paragraph.)

When you use a specific font on a web site, if that font is not installed on the viewer's device, that font will be replaced by a default font that is installed on that device, and your page will not render as you intended. That's one of the reasons why Google developed Google Fonts in the first place.

Although I needed to implement the @font-face solution because a specific Google Font had a problem, you can use this approach to make sure that any (non-standard) font will render properly

Here's how:

- 1) Download the non working font(s) from <http://www.google.com/webfonts> to a temporary folder (on your desktop).
- 2) Upload your font(s) to the FONT SQUIRREL WEBFONT GENERATOR at <http://www.fontsquirrel.com/fontface/generator> . Leave the radio button at "Optimal", then check the Agreement checkbox, and download your "Kit". NOTE: This can take a while depending on the size of your kit and what's going on at Font Squirrel.
- 3) UnZip the kit. You'll see a stylesheet.css file, and a series of font files that may be in .eot, svg, ttf, and woff formats. Upload the font files to a directory on your server. (I created a 'fonts' directory in the site root.)

Open the stylesheet.css file and change the url paths to relative paths that are appropriate for your site. (Note: Firefox accepts only relative URLs as a security precaution)

Since I set up a 'fonts' directory in the site root, for this example, I would change:

```
@font-face {
  font-family: 'my-font-family';
  src: url('my-webfont.eot');
  src: url('my-webfont.eot?#iefix') format('embedded-opentype'),
       url('my-webfont.woff') format('woff'),
       url('my-webfont.ttf') format('truetype'),
       url('my-webfont.svg#my-font-family') format('svg');
}
```

To:

```
@font-face {
```

```

font-family: 'my-font-family';
src: url(../fonts/my-webfont.eot');
src: url(../fonts/my-webfont.eot?#iefix) format('embedded-opentype'),
    url(../fonts/my-webfont.woff) format('woff'),
    url(../fonts/my-webfont.ttf) format('truetype'),
    url(../fonts/my-webfont.svg#my-font-family) format('svg');
}

```

Remove the font-weight: normal; and font-style: normal; from the @font-face code, since you'll probably want to include these in your individual classes where appropriate.

Then add your classes to the stylesheet.css code:

For this example we'll set up a masthead, heading, and text class using the font family 'my-font-family' (don't repeat any class names that already exist) . NOTE: I added !important to the color: declaration, so that links using these classes would render in the declared color in some browsers.

I.E.:

```

.masthead {font-family:my-font-family;
font-size:1.6em;
color: #72433a; !important
font-weight: bold;
font-style: normal;
}

```

```

.heading {font-family:my-font-family;
font-size:1.5em;
color: #72433a; !important
font-weight: normal;
font-style: italic;
}

```

```

.text {font-family:my-font-family;
font-size:1.1em;
color: #72433a; !important
font-weight: normal;
font-style: normal;
}

```

4) If you already have a .css file attached to your web pages, you can copy all of the stylesheet.css code into that file, or you can attach the stylesheet.css file to your page.

That's it, now you can use these new classes in your code and they should render correctly in all CSS3 enabled browsers.

GOING FURTHER:

If you want to give your client the ability to change CSS parameters, take a look at the recipe called USING CMSB TO POPULATE A CSS STYLESHEET at: <http://www.thecmsbcookbook.com/recipeDetail.php?335>

If you are using Google Fonts and @font-face, and you want to give your client the ability to change CSS parameters for both, take a look at the recipe called GOOGLE FONTS AND CMSB at: <http://www.thecmsbcookbook.com/recipeDetail.php?418>

The materials on this web site have been created for use with CMS Builder content management software. CMS Builder software is published and licensed for use by InteractiveTools.com. Please contact [Interactive Tools](#) for information on the downloading of the software or the purchasing of licenses.

Terms of Service

