Customizing the SmallBiz Template

This document is intended to get you started with the SmallBiz template system from JustDreamweaver.com.

SmallBiz is a very flexible template that allows you create many different website looks by changing as little code as possible.

The template uses the 960 grid system, which makes CSS layouts and columns easy to manipulate through the CSS classes. It also takes advantage of the popular jQuery JavaScript library to add in interactive tabbed and accordion content areas.

SmallBiz includes many pre-styled elements that can be added to your HTML pages simply by following the instructions in this manual. We've strived to make SmallBiz the easiest template EVER for launching a custom W3C valid website.

Getting Started

You will need a few things before you get started working with the SmallBiz template.

First, you will need a hosting plan. We highly recommend the following web hosting companies (in no particular order) for reliability and affordability:

- BlueHost
- HostGator
- InMotion Hosting
- HostMonster

Next, you will need a domain name for your new website. Some companies charge as much as $35 a year for domain name registration, but do not buy through these companies. You can get a domain name for only $7.49 through GoDaddy ONLY if you use the link below (normally they're over $11.00):

Get $7.49 .com Domains through this link

Once you have your domain name you will need to point your DNS to your hosting account. You should receive detailed information on the name servers used by your hosting company when you sign up.

Setting up the SmallBiz Template

First, create a folder on your computer where you will build out your new site.

Save the SmallBiz zip file in that folder and extract the contents of the zip file directly into the folder you created. The index.html file should be located in this folder when the files have been extracted.

If you are using Dreamweaver to edit the template, use the HTML files that are unzipped when you extract the file. The template is designed by default to work as a Dreamweaver template and nothing more is required.

If you are using a different HTML editor other than Dreamweaver, you will want to use the files located in the zip file titled 'Non-template Version for other HTML editors'. Those files do not have the Dreamweaver template code included and the navigation is not set up as a library item.

Dreamweaver users can delete the 'Non-template Version for other HTML editors.zip' file. There is no need to keep that file in your site or upload to your server.

Setting up your site in Dreamweaver

If you are using Dreamweaver to build your site, create a new site using Dreamweaver's Site > New Site menu.

Give the site a name, and in the 'Local root folder' browse to the folder you created to hold the site. If you extracted the zip file correctly, you will also be able to browse to the images folder in your main site folder to set in the Dreamweaver site setup options.

You can set the rest of the site options such as HTTP address, scripting language and testing server options as needed.

Once you click OK, you should be able to see your SmallBiz template files in the file inspector panel.

In order to use the niche header images with your template, you will need to unzip the header images zip file into the /images/ folder of your site.

You can now begin creating your site by editing the index.html file or creating your HTML pages from the templates as described in the next section.

Why isn't the index.html file attached to a template?

The index.html file is the home page of your site. We do not create a template for that page because in most cases going to be the only page on your site using the slider AND the full-width page content.

The navigation is a library item, so updates to the navigation will update in your index.html file, but home pages MOST of the time are slightly different in layout and/or content structure than the sub-pages of your site. Hence the lack of a home page template.

You can use any of the other templates for you home page if you wish, simply use File > New > Page from template, and choose the template you wish to use. Save your new document as index.html to overwrite the current home page. Your home page will now update whenever you make a change to the attached template file.

Dreamweaver Template Basics

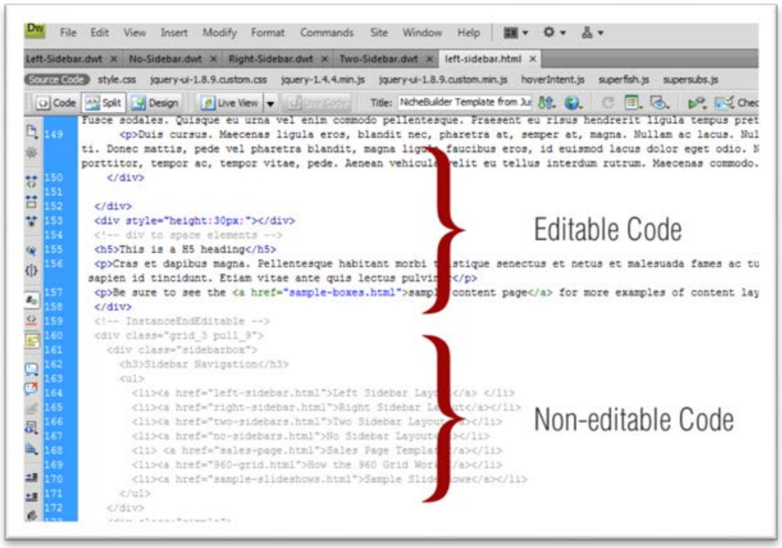
If you've used Dreamweaver templates before, you can probably skip this section as this is basic information about how the templates work.

A Dreamweaver template is a separate file with the .dwt extension and is located in the Templates folder of the SmallBiz site.

Templates allow you to create multiple HTML pages based upon a template, and when you make a code change in the template .dwt file, the changes are made in every HTML file that is linked to that template.

Within each template file, there are code regions that you can edit and code regions that are locked and uneditable.

The screen shot to the left shows the appearance difference between code that is editable and code that is NOT editable in Dreamweaver's Code View.



```
Fusce sodales. Quisque eu urna vel enim commodo pellentesque. Praesent eu risus hendrerit ligula tempus pret
      <p>Duis cursus. Maecenas ligula eros, blandit nec, pharetra at, semper at, magna. Nullam ac lacus. Nul
ti. Donec mattis, pede vel pharetra blandit, magna lig   faucibus eros, id euismod lacus dolor eget odio. N
porttitor, tempor ac, tempor vitae, pede. Aenean vehicul  velit eu tellus interdum rutrum. Maecenas commodo.
    </div>

  </div>
  <div style="height:30px;"></div>
  <!-- div to space elements -->
  <h5>This is a H5 heading</h5>
  <p>Cras et dapibus magna. Pellentesque habitant morbi t  stique senectus et netus et malesuada fames ac tu
 sapien id tincidunt. Etiam vitae ante quis lectus pulvi  </p>
  <p>Be sure to see the <a href="sample-boxes.html">sampl  content page</a> for more examples of content lay
  </div>
  <!-- InstanceEndEditable -->
  <div class="grid_3 pull_9">
    <div class="sidebarbox">
      <h3>Sidebar Navigation</h3>
      <ul>
        <li><a href="left-sidebar.html">Left Sidebar Lay  </a> </li>
        <li><a href="right-sidebar.html">Right Sidebar La  ut</a></li>
        <li><a href="two-sidebars.html">Two Sidebar Layout  </a></li>
        <li><a href="no-sidebars.html">No Sidebar Layout  </a></li>
        <li> <a href="sales-page.html">Sales Page Templat  /a></li>
        <li><a href="960-grid.html">How the 960 Grid Wor  /a></li>
        <li><a href="sample-slideshows.html">Sample Slide  ows</a></li>
      </ul>
    </div>
```

In Design View, your cursor will turn into a circle with a diagonal line through it when you hover your mouse over an uneditable region to indicate that area of the HTML page is only editable in the .dwt file.

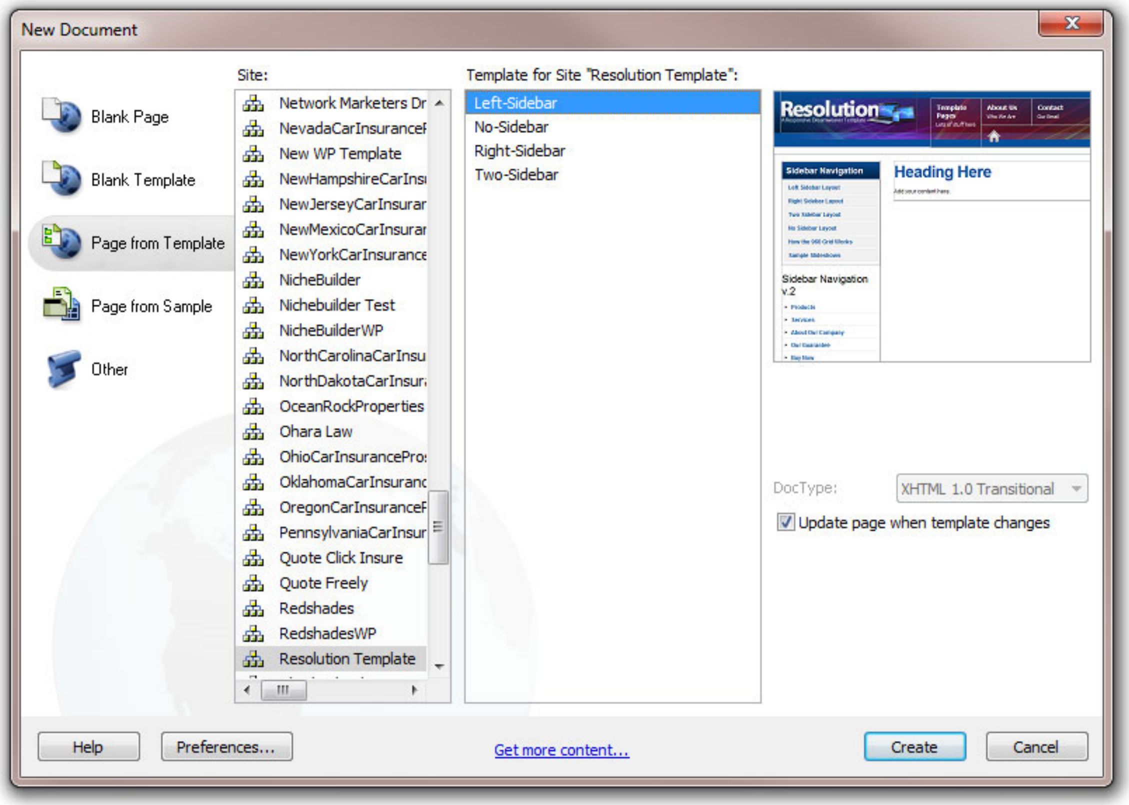
The SmallBiz template uses editable template regions only in the title/meta tags, head and content region of each template. This means that changes to the header, sidebar(s) and footer must either be done in the template .dwt file, or you must add an editable region to the template in order to edit that section of the site in an HTML file.

Custom scripts and CSS can be inserted into the 'head' editable region of any page that requires them. This is a good place for any javascript and analytics code you might be using on your site.

Creating New HTML Files from Templates

When you add new pages to your website, you will want to link them to one of the four included templates.

To create a new page, go to the main Dreamweaver menu and select File > New and you will be presented with a dialog like this one:



DocType: XHTML 1.0 Transitional

[✓] Update page when template changes

Choose the 'Page from Template' option and you will see the four templates that are built into SmallBiz. Choose whichever template you want your new page to use and click the 'Create' button.

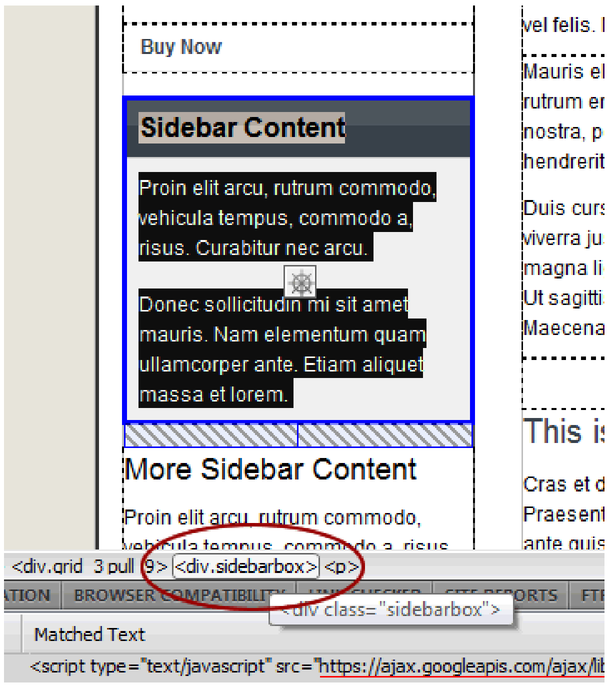
You will then have an untitled HTML file that is based upon your chosen template. You will need to save the file in order to preview it in a browser.

You will be able to edit the main content area of your new HTML file, but all other areas of the file will not be editable because they are locked by the template .dwt file.

When you make a change to one of the locked regions in the template .dwt file, all HTML pages linked to that template will be updated with your changes.

You can “release” areas from the template and make them editable on each HTML page by adding editable template regions.

Adding Editable Template Regions


Buy Now
Sidebar Content
Proin elit arcu, rutrum commodo, vehicula tempus, commodo a, risus. Curabitur nec arcu.
Donec sollicitudin mi sit amet mauris. Nam elementum quam ullamcorper ante. Etiam aliquet massa et lorem.
More Sidebar Content
Proin elit arcu, rutrum commodo,
<div.grid_3.pull_9> <div.sidebarbox> <p>
BROWSER COMPATIBILITY
<div class="sidebarbox">
Matched Text

```
<script type="text/javascript" src="https://ajax.googleapis.com/ajax/lib
```

Making sections of your template editable is very simple. Here's an example of how to make one of the sidebar content boxes editable.

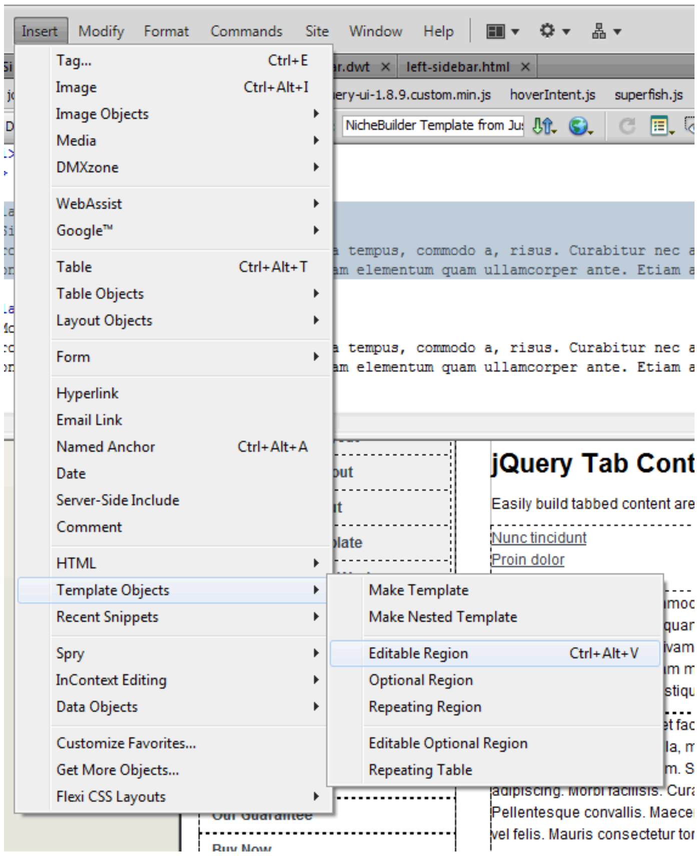
First, you start by selecting the content box in design view. Just click inside the box and look in your status bar (circled in red in the example).

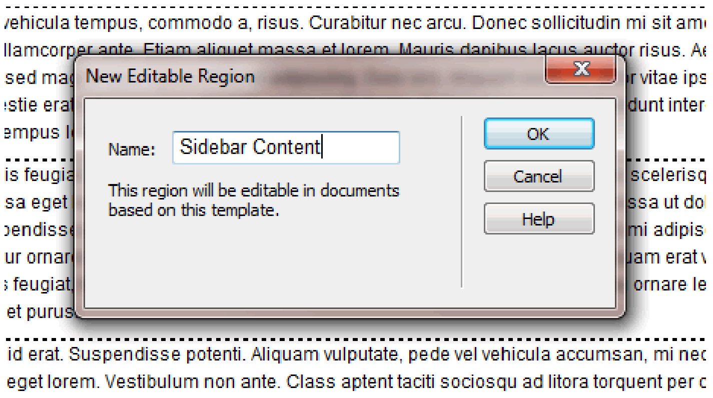
If you then click on the div.sidebarbox text in the status bar below the Design View window, you can select the entire div that you want to make editable.

Once you click on the div.sidebarbox, the div will highlight in blue as shown in the example.

The next step is to insert an editable region around the div you just selected in your layout. In your Dreamweaver main menu, choose Insert > Template Objects > Editable Region as shown in this image.

You will then be presented with a dialog box with which you can name your editable region to refer to it later.

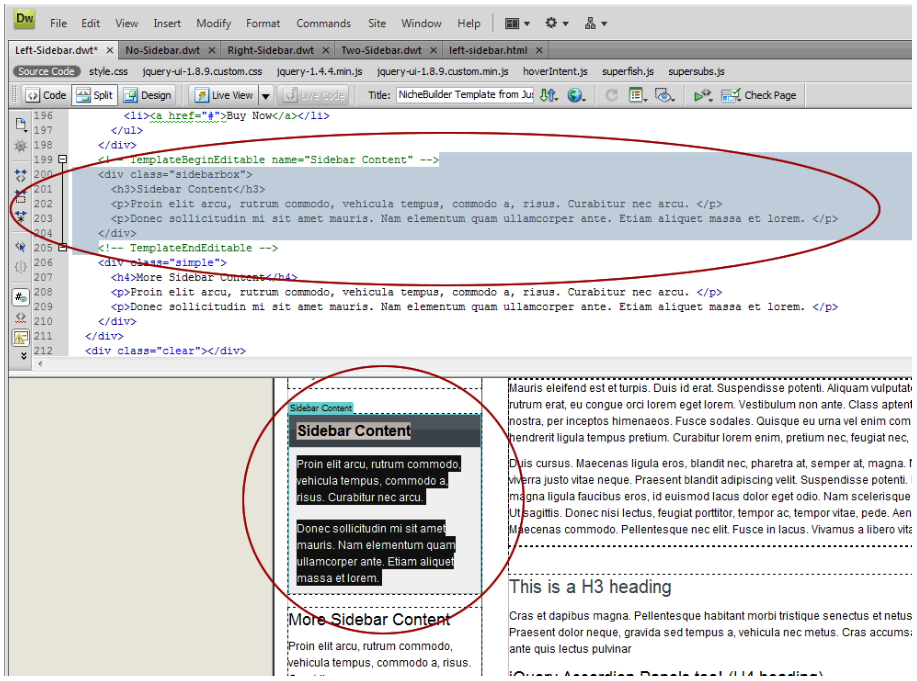




Name: Sidebar Content

This region will be editable in documents based on this template.

Once you assign a name and click the OK button, Dreamweaver will insert your new editable region around the selected element, in this case the div.sidebarbox. The image below shows the new code and visual indicators added by the new editable region.


File Edit View Insert Modify Format Commands Site Window Help
Left-Sidebar.dwt* No-Sidebar.dwt Right-Sidebar.dwt Two-Sidebar.dwt left-sidebar.html
Source Code style.css jquery-ui-1.8.9.custom.css jquery-1.4.4.min.js jquery-ui-1.8.9.custom.min.js hoverIntent.js superfish.js supersubs.js
Code Split Design Live View Live Code Title: NicheBuilder Template from Ju Check Page
Sidebar Content

```
        <li><a href="#">Buy Now</a></li>
      </ul>
    </div>
    <!-- TemplateBeginEditable name="Sidebar Content" -->
    <div class="sidebarbox">
      <h3>Sidebar Content</h3>
      <p>Proin elit arcu, rutrum commodo, vehicula tempus, commodo a, risus. Curabitur nec arcu. </p>
      <p>Donec sollicitudin mi sit amet mauris. Nam elementum quam ullamcorper ante. Etiam aliquet massa et lorem. </p>
    </div>
    <!-- TemplateEndEditable -->
    <div class="simple">
      <h4>More Sidebar Content</h4>
      <p>Proin elit arcu, rutrum commodo, vehicula tempus, commodo a, risus. Curabitur nec arcu. </p>
      <p>Donec sollicitudin mi sit amet mauris. Nam elementum quam ullamcorper ante. Etiam aliquet massa et lorem. </p>
    </div>
  </div>
  <div class="clear"></div>
```

You will now be able to edit that sidebar content area independently on every HTML page that is linked to that template.

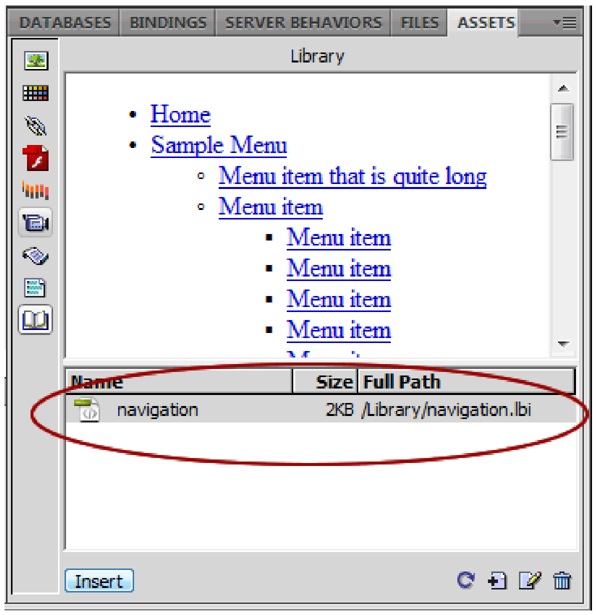
You can add new editable template regions anywhere you want page-level control over content. If you anticipate the content being identical site-wide, then you can leave the region editable only in the main template.

Customizing the Main Navigation

The SmallBiz template uses Dreamweaver's Library feature for the main navigation menu. Library items are similar to templates, in that when you make a change in the main library asset file, it updates the item anywhere you have used it in your site.

Creating the menu as a library item makes it easy to include the identical menu in all four of the SmallBiz templates (left sidebar, right sidebar, no sidebar and dual sidebars). Since all four of the built-in .dwt files use the same library item, updating the code in the navigation.lbi file (Library/navigation.lbi) updates all instances of the library item across your entire site.

The navigation menu is accessible from the Assets tab panel in Dreamweaver. If you do not see your assets panel, go to the main Dreamweaver menu and select Window > Assets to open the panel.



SmallBiz comes by default with sample navigation that demonstrates how to create multi-level drop-down menus. You will need to change the menu to meet your needs for your new site.

To edit the navigation.lbi file, click to your Assets panel and at the bottom of the panel, click the third icon to the left which looks like a piece of paper and a pencil. If you hover your arrow over the icon, it will popup the word 'Edit'. Click on that icon.

Your navigation.lbi file will now open in the main Dreamweaver workspace so you can edit the file.

If you wish to remove all the sample menu code and start with your own basic menu, delete everything you see in Code View and replace with the code below:

```
<ul class="sf-menu mobile-hidden">
   <li><a href="../index.html">Home</a></li>
   <li><a href="#">About Us</a></li>
   <li> <a href="#">Contact</a></li>
 </ul>
```

(Note that the # symbol is used to create an anchor link to the current page you're on. It doesn't actually link anywhere, it's just used to trigger the hover state and drop-down menus.)

You will need to change the href links to point to your own pages in your site. Remember that the navigation.lbi file is in the Library folder, so you will need to preface your links with a relative "folder up" format like this:

```
<li><a href="../aboutus.html">About Us</a> </li>
```

Depending on your site structure, you may also want to use absolute links in the menu like this:

```
<li><a href="/aboutus.html">About Us</a> </li>
```

If you have issues with your template correctly rewriting your menu links, the guaranteed way to insert your links so they work is to use the full URL to each page like this:

```
<li><a href="http://www.yourdomain.com/aboutus.html">About Us</a> </li>
```

Obviously you will want to change the reference to yourdomain.com to your own actual domain name.

Adding Secondary Navigation Text

The menu in SmallBiz features a second line of text where you can give additional insight into what the link is for.

The second line of text is created simply by wrapping the text within a <span class="desc"> tag inside the <a> tag for each link.

Here’s an example:

```
<ul class="sf-menu mobile-hidden">
   <li><a href="#">About Us<span class="desc">Our Staff</span></a></li>
   <li> <a href="#">Contact Us<span class="desc">Get in Touch</span></a></li>
 </ul>
```

This code will create two menu items each with a second line of text below the main menu link.

The home icon that is shown in the template files is created by adding this additional <li> element to the list:

```
<li><a href="../index.html"><img src="../images/home-icon.png" alt="Home"/></a></li>
```

Adding Drop-down Menus to your Navigation

The way to add drop-down menus to your main navigation bar is to embed a new unordered list within the list item tag (<li>) where you want the drop-down menu to appear. Here’s an example:

You have multiple contact methods and you want a drop-down menu from the original Contact menu item:

```
<li><a href="#">Contact Us<span class="desc">Get in Touch</span></a>
		<ul>
			<li><a href="#">Email</a></li>
			<li><a href="#">Facebook</a></li>
			<li><a href="#">Phone or Fax</a></li>
		</ul>
</li>
```

If you copy the code above and replace the original line of code for the Contact menu link, you will now have a drop-down menu that contains the three contact method links.

To create a fly-out menu from one of the drop-down options, you do the same process again within the drop-down list items:

```
<li><a href="#">Contact Us<span class="desc">Get in Touch</span></a>
        <ul>
                <li><a href="#">Email</a></li>
                <li><a href="#">Social Media</a>
                        <ul>
                                <li><a href="#">LinkedIn</a></li>
                                <li><a href="#">Facebook</a></li>
                                <li><a href="#">Twitter</a></li>
                        </ul>
                </li>
                <li><a href="#">Phone or Fax</a></li>
        </ul>
</li>
```

This code would then produce a drop-down menu with the Email, Social Media and Phone or Fax options, and the Social Media option would then have a sub-menu with LinkedIn, Facebook and Twitter as a fly-out sub-menu as shown below:

Customizing the Main Navigation Javascript

The main navigation uses a great menu script called Superfish, which is a jQuery implementation of the Suckerfish CSS menu code.

If you look in the <head> of any of your pages, you will see this code:

```
<script type="text/javascript">  // This script is for the navigation menu
   $(document).ready(function(){
      $("ul.sf-menu").supersubs({
         minWidth:    12,   // minimum width of sub-menus in em units
         maxWidth:    27,   // maximum width of sub-menus in em units
         extraWidth:  1     // extra width can ensure lines don't sometimes turn over
                            // due to slight rounding differences and font-family
      }).superfish();  // call supersubs first, then superfish, so that subs are
                       // not display:none when measuring. Call before initialising
                       // containing tabs for same reason.
   });
</script>
```

This is the javascript that makes the menus work. The menus also use a supersubs function which automatically resizes the width of your drop-down menus based on their content. The minWidth and maxWidth values control the minimum and maximum widths of your drop-down menus. So if you want a wider menu or narrower menu, you can control that by changing those values.

The javascript can easily be changed with additional script parameters that control the delay, speed, animation, shadows, arrows and more.

View additional menu options here:

http://users.tpg.com.au/j_birch/plugins/superfish/#options

More information about the Superfish menus can be found here:

http://users.tpg.com.au/j_birch/plugins/superfish/

Customizing the Sidebar Navigation

The SmallBiz template comes with three styles of sidebar navigation.

The first style is enclosed in a box with a light grey background. The code for that style of navigation looks like this:

```
<div class="sidebarbox">
     <h3>Sidebar Navigation</h3>
     <ul>
       <li><a href="left-sidebar.html">Left Sidebar Layout</a> </li>
       <li><a href="right-sidebar.html">Right Sidebar Layout</a></li>
       <li><a href="two-sidebars.html">Two Sidebar Layout</a></li>
     </ul>
   </div>
```

To change the navigation items, simply add or remove list items. Each item must be enclosed in opening and closing li tags.

The second style of sidebar navigation doesn't have an enclosing box and has a white background. The code for that style looks like this:

```
<div class="simple">
     <h4>Sidebar Navigation v.2</h4>
     <ul>
       <li><a href="#">Products</a> </li>
       <li><a href="#">Services</a></li>
       <li><a href="#">About Our Company</a></li>
       <li><a href="#">Our Guarantee</a></li>
       <li><a href="#">Buy Now</a></li>
     </ul>
   </div>
```

The format is very similar to the other style, but the title uses an H4 tag rather than an H3 tag, and the parent container div uses the "simple" class rather than the "sidebarbox" class. As with the other style, any added items must be wrapped in li tags.

The third style of sidebar navigation uses icons on the left of each menu link. The code for the icon menu looks like this:

```
<div class="iconmenu">
     <ul>
       <li><a href="#"><img src="images/icons/stroke_general_shopping_cart_48.png" alt="" width="30"
height="30">Products<span>Get the goods here</span></a> </li>
       <li><a href="#"><img src="images/icons/stroke_accounting_send_box_48.png" alt="" width="30"
height="30">Services<span>How can we help you?</span></a></li>
       <li><a href="#"><img src="images/icons/stroke_general_bulb_48.png" alt="" width="30"
height="30">About Us<span>Who we are</span></a></li>
       <li><a class="lastitem" href="#"><img src="images/icons/stroke_networking_messenger_48.png"
alt="" width="30" height="30">Contact<span>Give us a shout</span></a></li>
     </ul>
   </div>
```

Each icon can be changed simply by linking to a new icon from the /images/icons/ folder included in the template. You can change the size of the icon by changing the height and width parameters for each image.

The “lastitem” class that is applied to the bottom menu item simply removes the bottom border from that item.

The icons included with SmallBiz are professionally designed premium icons created by IconShock as part of their Stroke collection. You can view the entire collection here:

Visit IconShock Website

Customizing the Sidebar Social Media Icons

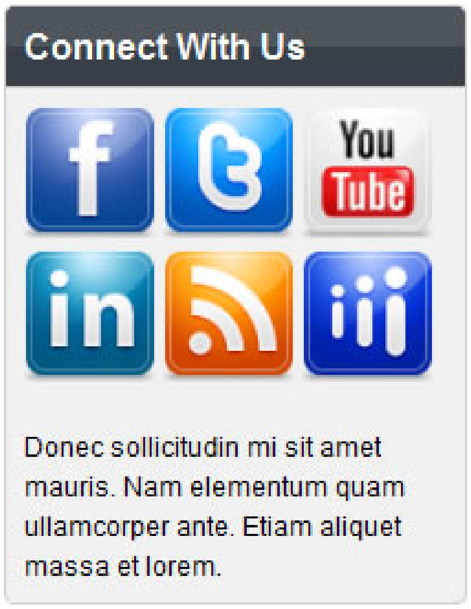
SmallBiz includes 8 icons that you can insert and link to your favorite social media profiles. The template examples include Facebook, Twitter and Youtube icons, but you can easily add more or remove icons as needed.

You can control the size of the icons simply by modifying the width property in the HTML code for each image. The template example used 60 pixel width images to fit them three across in the sidebars:

```
<a href="#"><img src="images/socialmedia/Twitter.png" alt="" width="60" /></a>
```

You will need to add your profile link to the href parameter for your icons to link properly, and you can also enter the alt text if you wish.

Two rows of icons can be displayed like this:

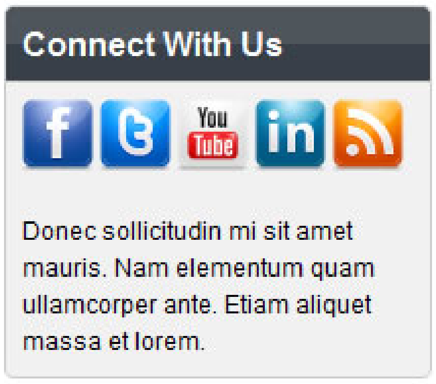


Donec sollicitudin mi sit amet mauris. Nam elementum quam ullamcorper ante. Etiam aliquet massa et lorem.

By using code like this:

```
<p><a href="#"><img src="images/socialmedia/Facebook.png" alt="" width="60" /></a>
    <a href="#"><img src="images/socialmedia/Twitter.png" alt="" width="60" /></a>
    <a href="#"><img src="images/socialmedia/Youtube.png" alt="" width="60" /></a>
    <a href="#"><img src="images/socialmedia/Linkedin.png" alt="" width="60" /></a>
    <a href="#"><img src="images/socialmedia/RSS.png" alt="" width="60" /></a>
    <a href="#"><img src="images/socialmedia/Myspace.png" alt="" width="60" /></a>
</p>
```

Create a single line of smaller icons like this:



Donec sollicitudin mi sit amet mauris. Nam elementum quam ullamcorper ante. Etiam aliquet massa et lorem.

By using code like this:

```
<p><a href="#"><img src="images/socialmedia/Facebook.png" alt="" width="35" /></a>
    <a href="#"><img src="images/socialmedia/Twitter.png" alt="" width="35" /></a>
    <a href="#"><img src="images/socialmedia/Youtube.png" alt="" width="35" /></a>
    <a href="#"><img src="images/socialmedia/Linkedin.png" alt="" width="35" /></a>
    <a href="#"><img src="images/socialmedia/RSS.png" alt="" width="35" /></a>
</p>
```

Simply adjust the width of each icon until the all your icons fit on one line in your sidebar.

You can use these icons anywhere on your site, they are not restricted to just your sidebar.

Customizing the Sidebar Newsletter Form

The sidebar form can be used for any kind of form submission such as newsletters subscriptions, E-mail list opt in, information requests, etc.

The form does not actually process the submission, as you will need to “attach” the form to your processor by adding a form action URL.

The form action by default looks like this:

```
<form id="sidebaroptin" method="post" action="">
```

Using Aweber forms as an example, you would add their form processor URL like this:

```
<form id="sidebaroptin" method="post" action="www.aweber.com/scripts/addlead.pl">
```

If you are using your own form processor script, you simply link to it with the form action.

If you are using a third party autoresponder service such as Aweber, Constant Contact, or Vertical Response to build an email list, you will need to change the ‘name’ parameters on the form fields to match what your form processor is expecting with the form is submitted.

The easiest way to do this is to create an actual form on your autoresponder’s website that is linked to your list.

When you complete the form, get the HTML code for the form. You will need to do three things:

1. First, find the action URL in the form and copy it over to your sidebar form’s action parameter. This is the URL that actually handles the form data that is submitted.
2. Copy all hidden fields from your autoresponder form to your sidebar form. Aweber, for example, may include these hidden fields in their form:

   ```
   <input type="hidden" name="listname" value="[your listname here]" />
   <input type="hidden" name="redirect"
   value="http://www.example.com/thankyou.htm" />
   <input type="hidden" name="meta_adtracking" value="custom form" />
   <input type="hidden" name="meta_message" value="1" />
   <input type="hidden" name="meta_required" value="name,email" />
   <input type="hidden" name="meta_forward_vars" value="1" />
   ```

3. Copy the form inputs from the autoresponder form to your sidebar form. In our Aweber example, our form inputs might look like this:

   ```
   <input type="text" name="name" value="" />
   and
   <input type="text" name="email" value="" />
   ```

   By copying the form inputs, you are making sure the ‘name’ parameters are identical to what your form processor is expecting in the form data.

   Alternatively, you can simply change the ‘name’ values in the default form code in the SmallBiz template to match the input names in your autoresponder form.

Once you have the form action, the hidden fields, and the proper name values on your form inputs, you should be able to enter data in your form and have it submit properly to your autoresponder or script.

Customizing your Layout with the 960 Grid System

The 960 grid system is an excellent framework for designing table-less CSS websites. You can read more about it at http://960.gs

The SmallBiz template uses the 960 grid in a 12 column format. So if you can imagine having 12 invisible columns down your page, each 50 pixels wide with a 30 pixel "gutter" or gap in between each column.

The CSS code for the SmallBiz template already has every possible column configuration pre-coded, so you just need to know how to implement the columns within your pages.

The SmallBiz template consists of four individual sections:

1. Header
2. Navigation
3. Content
4. Footer

Each of these sections is created by using the 'container' class like this:

```
<div id="header" class="container_12">
<div id="content" class="container_12">
<div id="footer" class="container_12">
```

The container_12 class sets the width of the div at 960 pixels and centers it in the browser window. The container_12 class is required on every parent container used in the template.

The 960 grid system uses a grid_X nomenclature, which simply means each div class is named grid_1 or grid_2 or grid_12. The number indicates how many of the columns your div will span.

So for example, since we are using a 12 column grid in SmallBiz, if you see code like this:

```
<div class="grid_12">
```

That simply means that div will span 12 columns which is the full width of the page. You would find that code INSIDE one of the container_12 divs listed above.

If you want to equal columns, you would use code like this:

```
<div class="grid_6">
   <p>Cras et dapibus magna. Pellentesque habitant morbi tristique senectus et netus et malesuada
fames ac turpis egestas. Praesent dolor neque, gravida sed tempus a, vehicula nec metus. Cras
accumsan ultricies sapien id tincidunt. Etiam vitae ante quis lectus pulvinar</p>
```

```
 </div>
 <div class="grid_6">
   <p>Cras et dapibus magna. Pellentesque habitant morbi tristique senectus et netus et malesuada
fames ac turpis egestas. Praesent dolor neque, gravida sed tempus a, vehicula nec metus. Cras
accumsan ultricies sapien id tincidunt. Etiam vitae ante quis lectus pulvinar</p>
 </div>
```

The grid_6 class is one-half the width across the 12 columns, so that code would result in 2 equal columns, each one-half the width of the page.

There is a page included in the SmallBiz template that has many more examples of 960 grid system layouts you can use. See the 960-grid.html file located in the root folder of the template. Another included file, styled-content.html, gives you more examples of how to use the 960 grid system.

Customizing the jQuery Slideshow

The jQuery slideshow script that is built into SmallBiz is a full-featured content rotation plugin that can rotate through any HTML, not just images.

There are many settings and additional features that are not used in the SmallBiz implementation, but you can read more about the script here: http://jquery.malsup.com/cycle/

The slideshow script is loaded with this line of code found in the <head>:

```
<script type="text/javascript" src="js/jquery.cycle.all.min.js"></script>
```

If you are not using the slideshow on your pages, you can remove that line in order to avoid loading the script and speed up your page load time.

Once the slideshow script has been loaded, you can call the cycle function like this in the <head>:

```
<script type="text/javascript">
        jQuery(document).ready(function() {
                jQuery('.slideshow').cycle({
                fx:  'fade',
                timeout: 5000,
                speed:  1000,
                easing: 'easeInOutQuad',
                pause: 1,
                pauseOnPagerHover: 1,
                sync: 1,
                autostop: 1,
                autostopCount: 3,
                pager: '#slidenav'
                });
        })
</script>
```

All available options and default values for the jQuery Cycle plugin can be found here: http://jquery.malsup.com/cycle/options.html

Example slideshow function options

fx

The available options for the fx parameter are: blindX, blindY, blindZ, cover, curtainX, curtainY, fade, fadeZoom, growX, growY, scrollUp, scrollDown, scrollLeft, scrollRight, scrollHorz, scrollVert, shuffle, slideX, slideY, toss, turnUp, turnDown, turnLeft, turnRight, uncover, wipe, zoom

The fx option controls how your slider area transitions between slides.

timeout

The timeout option controls how long each slide will show (in milliseconds). The larger the number, the longer each slide will show.

speed

Speed controls how fast the transition is between slides. This will control how fast the fx option completes.

easing

Easing can be thought of as the effect before and after each transition. Available options are easeInOutQuad, easeInOutCubic, easeInOutQuart, easeInOutQuint, easeInOutSine, easeInOutExpo, easeInOutCirc, easeInOutElastic, easeInOutBack, and easeInOutBounce.

Each easing effect results in a little bit different entry and exit for your slides.

pause

The pause option controls whether or not the slides stop when the mouse is positioned over the slider area. If set to 1, the slideshow will stop while the mouse is hovered over the slider. If set to 0, the slideshow will not pause.

pauseOnPagerHover

If you are using the slider navigation buttons that show up as 1, 2, 3, etc below the slideshow, this option when set to 1 will pause the slideshow when your mouse hovers over one of the navigation links. If set to 0, the slideshow will not pause.

sync

The sync option controls whether or not your transitions occur at the same time. If set to 1, your slides will overlap during transition. If set to 0, the first slide transition will complete before the next slide transition starts.

autostop

The autostop option forces your slideshow to stop on the slide you specify. Setting the option to 1 uses the autostopCount parameter. In the example, it is set to stop on slide #3. This works well if you have content in a particular slide that you want the user to act on, such as a form or clickable link.

autostopCount

This option works with the autostop option. If autostop is set to 1, set the autostopCount option to the slide you want the slideshow to end on.

pager

This option tells the script what div ID the navigation buttons will be displayed in. If you do not want pager navigation to show, simply remove this option. You can also remove the pager div from the template code is present.

Example slider HTML markup

This is the sample slider as used on the home page of the SmallBiz template:

```
<div id="slidewrap" class="full_width">
<div class="container_12 slidebackground rounded shadow">
  <div class="slideshow slide930">
    <div class="slideshowitem"><img src="images/home-slideshow1.jpg" alt="" /></div>
    <div class="slideshowitem"><img src="images/home-slideshow2.jpg" alt="" /></div>
    <div class="slideshowitem"><img src="images/home-slideshow3.jpg" alt="" /></div>
  </div>
  <div id="slidenav"></div>
</div>
```

First, the “slidewrap” div creates a box to hold the slider. It spans the full with of the browser window, so you can apply full-width backgrounds to the wrap as shown on the slider example page included in the template.

Next, you have a 960 grid system container which sets the width of the slider to 960 pixels and centers it on the page:

```
<div class="container_12 slidebackground rounded shadow">
```

The extra three classes apply styling to the div included rounding the corners and applying a slight CSS box shadow. The ‘slidebackground’ class sets the background color of the inner slider container (960px wide), adds a small border, fixes the height and width to a set pixel value, and adds positioning and padding around the edges.

Next, a div is added with two classes, slideshow and slide930. The slideshow class is required for the script to function, and the slide930 class is added to control the height of the slide container and hide overflow slides. By default the slider height is 250px, so if you require a shorter or taller slider you can change that CSS properly value.

Next, there are three div elements with the slideshowitem class. Anything you place inside the slideshowitem divs will be rotated in your slide show. To add slides, you simply add another slideshowitem div and place your content HTML inside. To remove slides, you remove the slideshowitem divs.

The inner slideshow divs DO NOT have to have the class of slideshowitem. This is strickly a descriptive class to help you when creating your own site. The slideshowitem divs do have the CSS height and width properties applied which match the size of the parent slide930 div height and width.

The slideshow script will rotate through ANY repeating elements and they don’t even have to be divs. Try adding three paragraphs instead of the slideshowitem divs and you’ll have a slideshow that rotates your paragraphs.

The last part of the slider code is the optional slider navigation holder div with the class of slidenav:

```
<div id="slidenav"></div>
```

If you have the pager option in your script options, the option value must match the ID of the div you want your navigation to appear in. If you are not using the pager option, you can remove the slidenav div from the HTML markup.

The sample images used in SmallBiz are included in the template files, so you can use those as a guide if you wish.

Customizing the jQuery Tabbed Boxes

The SmallBiz template comes with jQuery tabbed boxes built in. The boxes are part of the jQuery UI, which is loaded separately from jQuery in the <head> of the templates.

NOTE: If you are not using tabs or accordions on your pages, you can remove the links to the jQuery UI style sheet and javascript in your templates or individual pages and have faster page load times. Remove these lines from the <head> of your templates or pages:

```
<link href="js/jquery-ui-1.8.9.custom.css" rel="stylesheet" type="text/css" />

<script src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.8.23/jquery-ui.min.js"></script>
```

The tabbed content areas require two things to operate. First, you need to have the proper javascript loaded to make the tabs work. Second, you need to have the proper div structure around the tabs so the javascript can work properly.

The javascript is loaded automatically in the SmallBiz.js file that is linked in the <head> of each page. The code in that file that triggers the tabs is this:

```
$(function() {
                $("#tabs").tabs();
        });
```

This code simply tells jQuery to find a div with an ID of “tabs” and apply the tab function to the contents of that div.

So, we obviously need to place some code on our page with an ID of “tabs”. The ID can be ANYTHING, not just tabs, you just need to match up the ID called in the javascript with the ID used on your tabbed div container.

Our code might look like this:

```
<div id="tabs">
  <ul>
    <li><a href="#tabs-1">Nunc tincidunt</a></li>
    <li><a href="#tabs-2">Proin dolor</a></li>
    <li><a href="#tabs-3">Aenean lacinia</a></li>
  </ul>
  <div id="tabs-1">
    <p>Proin elit arcu, rutrum commodo, vehicula tempus, commodo a, risus..</p>
  </div>
  <div id="tabs-2">
    <p>Morbi tincidunt, dui sit amet facilisis feugiat, odio metus gravida ante, ut pharetra massa metus
id nunc.</ p>
  </div>
  <div id="tabs-3">
    <p>Mauris eleifend est et turpis. Duis id erat. Suspendisse potenti..</p>
  </div>
 </div>
```

The tabs code starts by identifying the ID of the container div, in this case “tabs”.

The tab links are created by the unordered list items, each having a distinct link to one of three separate tab containers, identified by ID’s of tabs-1, tabs-2 and tabs-3.

That’s all the code needed to create the tabs, jQuery and javascript handle the styling and function of the tabs once the page is loaded in a javascript-enabled browser.

You can add new tabs by adding a new list item to the unordered list with a unique href identifier. Then add a new div with an ID the same as your href code in the unordered list.

Multiple jQuery tabs on the same page

You can easily have multiple tabbed sections on the same page. All you need to do is have a unique ID for each one like shown.

First, add the new ID to the javascript (newtabs in this example):

```
$(function() {
		$("#tabs").tabs();
		$("#newtabs").tabs();
});
```

Then add your tabbed content like this:

```
<div id="newtabs">
  <ul>
    <li><a href="# newtabs -1">Nunc tincidunt</a></li>
    <li><a href="# newtabs -2">Proin dolor</a></li>
    <li><a href="# newtabs -3">Aenean lacinia</a></li>
  </ul>
  <div id=" newtabs -1">
    <p>Proin elit arcu, rutrum commodo, vehicula tempus, commodo a, risus..</p>
  </div>
  <div id=" newtabs -2">
    <p>Morbi tincidunt, dui sit amet facilisis feugiat, odio metus gravida ante, ut pharetra massa metus
id nunc.</ p>
  </div>
  <div id=" newtabs 3">
    <p>Mauris eleifend est et turpis. Duis id erat. Suspendisse potenti..</p>
  </div>
 </div>
```

That’s all there is to it. You will now have two separate tabbed content sections on the same page.

For more information and configuration options for the tabs, please see the following:

http://jqueryui.com/demos/tabs/

http://docs.jquery.com/UI/Tabs

Customizing the jQuery Accordion

NOTE: If you are not using tabs or accordions on your pages, you can remove the links to the jQuery UI style sheet and javascript in your templates or individual pages and have faster page load times. Remove these lines from the <head> of your templates or pages:

```
<link href="js/jquery-ui-1.8.9.custom.css" rel="stylesheet" type="text/css" />

<script src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.8.23/jquery-ui.min.js"></script>
```

The jQuery accordion is very similar to the tabs. First you have the javascript that implements the accordion, which is also located in the SmallBiz.js file:

```
$("#accordion").accordion({
                    autoHeight: false,
                    collapsible: true,
                    icons: {'header': 'ui-icon-plus', 'headerSelected': 'ui-icon-minus' }
          });
```

Then you have the on-page code that is structured specifically for the accordion script:

```
<div id="accordion">
  <h3><a href="#">Section 1</a></h3>
  <div>
    <p> Sed non urna. Donec et ante. Phasellus eu ligula. Vestibulum sit amet
      purus. </p>
  </div>
  <h3><a href="#">Section 2</a></h3>
  <div>
    <p> Sed non urna. Donec et ante. Phasellus eu ligula. Vestibulum sit amet
      purus. </p>
  </div>
  <h3><a href="#">Section 3</a></h3>
  <div>
    <p> Sed non urna. Donec et ante. Phasellus eu ligula. Vestibulum sit amet
      purus. </p>
  </div>
 </div>
```

The script ID must match the container div ID, in this case #accordion in the script and id=”accordion” in the page code. (# means ID in jQuery)

The accordion does not use the initial unordered list like the tabs, but rather builds the clickable title into each accordion section as an H3 tag.

So each section of the accordion has these components:

- H3 title (this is clickable to expand and contract the panel)
- Opening <div>
- Content
- Closing </div>

To add a new panel, you need all of these elements. The content can be any HTML code you wish to be enclosed by the panel.

As with the tabs, you can use multiple accordions on each page. You just need to assign a unique identifier in the javacript, and match it to the same unique ID in the body code.

For more information and options for the accordion, please see:

http://jqueryui.com/demos/accordion/

http://docs.jquery.com/UI/Accordion

Custom Page Content Styles

The SmallBiz template comes with many content styles built-in including many content boxes, styled lists and images.

To explore the types of styles you can use on your pages, please see the included styled-content.html file in the root template directory. To use any of the included elements, simply copy the code from the styled-content.html page to your own site page.

Using Google Web Fonts

SmallBiz uses the Google Web Fonts API to style some of the Heading tags such as H1, H2, etc.

All that is needed to use Google fonts is a link to the font CSS file in the head of your page, and the proper font name in the CSS selector for the element you wish to style using the font.

SmallBiz uses the PT Sans Narrow font, but you can change to any other available font family. If you look in the head links for any template page, you will see this:

```
<link href='http://fonts.googleapis.com/css?family=PT+Sans+Narrow:400,700' rel='stylesheet'
type='text/css'>
```

That is the link to the PT Sans Narrow font on the Google network. The font is applied in the style.css file anywhere you see the 'PT Sans Narrow' font family. The file includes both the 'normal' (400) and 'bold' (700) font weights.

To use a different font or add additional web fonts to your site, first browse the available fonts here: http://www.google.com/webfonts

Each font has a 'Quick-use' link where you can copy the proper link to add to the head of your site, and also the proper CSS font family name to use in your CSS file.

Fonts may have multiple styles and weights, so make sure you select which variations you want to enable on your site. There is also a page load speed indicator that will give you an idea of how much 'overhead' you are adding to your site with each file load.

If you want to replace the PT Sans Narrow font, simply copy the new font link and replace the current font link in your files. Then search the style.css file for each instance of PT Sans Narrow and replace it with your new font family name as shown in the Google font quick use page.

After saving your files, refreshing your site in a browser will show the new font.

Just a note, if you are developing locally the fonts will not show on Internet Explorer. Only after uploading your site to a live server and previewing from that location will the fonts display in IE.

Mobile Responsiveness

SmallBiz is built using CSS3 @media code that serves different styles when it detects a narrower browser width.

Users who visit your site on either a portrait tablet or smaller desktop monitor will have slightly larger font sizes and the layout switches to fluid rather than fixed width. Sidebars are still visible down to a width of 640 pixels on the viewing browser.

Users who visit on a tabloid oriented tablet with a width exceeding 960 pixels will see the normal site with no modifications.

For small handheld devices such as most smart phones, the sidebars are not visible and the content spans the full width of the mobile browser. The header navigation is replaced by a selectable drop-down menu place directly below the logo.

The mobile menu is created via javascript and it automatically generates the menu based on the menu items on the site. There is no need to create a separate menu for mobile devices.

Menu items that have a hash symbol "#" href value (i.e. to trigger a drop-down menu) are removed from the menu entirely and only pages with valid href URL's are added to the mobile menu.

SmallBiz also scales images on small devices to not exceed the width of the browser. So even if you have larger images in the content of your site, SmallBiz will scale those images down as the browser shrinks.

The large slider below the header is removed from the page on any browser less than 960 pixels in width. This is due to the inability of the slider javascript to detect image sizes that are scaled after page load via CSS.

To make changes to the @media CSS code, simply go to the end of the style.css file and look for the two @media sections. The first is for small browser widths up to 640px and the second is for widths from 641px to 959px.

The second @media code is more involved due to changing from a fixed width to a fluid width layout. The 960 grid classes must change from pixel values to percentages, hence the need for more code.