

More Template Modifications

1. moving your template to your main public_html folder

In order for the template you select to be your homepage, you'll need to move all of the files and folders from the template folder and into the main public_html folder.

First, delete default.html and reclaim_logo.svg from the main public_html folder.

Next, delete the entire "story" folder.

Then, open your template folder on your desktop, select all of the files and folders, and drag them into the public_html folder. Note that you should not upload the folder itself, just the files and folders inside of it.

Now when you go to your main URL (e.g., <http://sorapure.net>), the index.html from the template should load.

2. images used as backgrounds

Sometimes in a template you'll see an image being used but you won't see that image being referenced in the HTML. This happens a lot in the top banner on the homepage. Instead, the image is referenced in the CSS as a background image for a section of the page.

To find the image, open the main CSS file in Brackets or another text editor and do a search (Edit > Find, or Command > F) for "jpg" or "png". That will find all of the images referenced in the CSS file; there won't be many and they'll likely all be background images.

The advantage of using an image as a background for a section is that you can put text and even other images on top of it in the HTML file.

3. link placeholder

Most links in your template file will use the placeholder # -- like this,

```
<a href="#">home</a>
<a href="#">bio</a>
```

You just need to replace the # with the actual URL where you want the link to go to. Keep in mind that for links to pages in your own website, you typically just provide the filename – like this,

```
<a href="index.html">home</a>
<a href="bio.html">bio</a>
```

For links to pages on the Internet beyond your site, you need to provide the full URL, starting with https:// -- like this,

```
<a href="https://www.ucsb.edu">UCSB</a>
```

4. image paths

Sometimes in the HTML or CSS code you'll see this: "../" It means "go up one folder". For instance, in a CSS file you might see

```
header {  
    background: white url("../images/banner.jpg") no-repeat;  
}
```

This means to go up one folder from where the CSS file is, and then to go into the folder called "images" and get the file called "banner.jpg".

Anytime you use an image in your HTML or CSS, you need to specify the path to that image. For instance, there's a big difference between

```
 and 
```

In the first case, the file "me.jpg" is in the same folder as the HTML file that references it; in the second case "me.jpg" is inside a folder called "images".

5. setting the favicon

The favicon is the little icon that appears in the left corner of the tab for your website. It's a nice place to put a variation of your logo or some other small, simple icon that relates to your site identity. The favicon file name should be "favicon.ico".

You can create your favicon in a couple of different ways:

1. create a very small square version of your logo:
In Photoshop or Pixlr or another image editing program, open your logo file; reduce the size to something like 32px x 32px; delete the background and save the file as "favicon.ico".
2. find or create a favicon online:
There are sites with premade favicons and sites where you can create your own favicon (e.g., [Free Favicon gallery](#); [How to design a favicon](#))

Some templates come with a favicon and others don't. If your template comes with a favicon, then you can simply delete the "favicon.ico" file there and substitute your own "favicon.ico" file.

If your template doesn't come with a favicon, you can simply put the file "favicon.ico" in your main folder (e.g., in the same folder as your "index.html" file). Most browsers will see the file and automatically place it on the tab of your website. While this method will work, it's technically not correct.

The correct method for including a favicon is to upload your “favicon.ico” file to your “images” folder (or wherever your other images are) and then include the following code in the <head></head> of your index.html file:

```
<link rel="shortcut icon" href="/images/favicon.ico" type="image/x-icon" />
```

These last two items will only apply to some of you, so I won't cover them in class. But work through the directions and don't hesitate to contact me if you have any questions or run into problems.

6. setting up the contact form using Formspree (easy and free)

1. Set up a free account at [Formspree.io](https://formspree.io).
2. On the Your Forms page, select **+New Form**.
3. You can specify a form name or just accept the default; make sure the email where the form will send to is correct. Select **Create Form**.
4. On the next page, copy the code for “Your form’s end point”; it’ll look something like this:

<https://formspree.io/f/xwkyrgrd>

5. In Brackets, open the HTML file with your form. Fill in the code you copied above in the <form> opening tag, like this:

```
<form method="post" action=" https://formspree.io/f/xwkyrgrd">
```

6. Make sure that every section of the form as a **name** attribute. If you’re using a HTML5 UP template, this should already be done for you.
7. Save and upload the HTML file with the form.
8. Fill out the form on your live site and click Send or Submit.
9. You’ll be taken to a page in Formspree that asks you to confirm your email.
10. Log in to your email account and confirm your email.
11. You’ll be taken to a page that says Email Confirmed. Your form is up and running!

7. uploading web-licensed fonts and using @font-face

Web designers today most often use Google fonts (<https://fonts.google.com>) and choose from among the 900+ fonts that Google offers. However, if you want to use a font that isn’t available from Google—for example, one that you’ve selected in your brand style guide—here’s how to do it.

First, make sure that your font is licensed for use on the Web; licensing information is typically available when you download or buy the font.

Then use Font Squirrel (<https://www.fontsquirrel.com/tools/webfont-generator>) to generate both the files and the code for your font. Following the instructions there, upload your font and download the “kit”, a zipped folder. Unzip the folder.

Take the two font files—with the extensions .woff and .woff2—and include them in the folder for your template. You can put them in the main folder or in a separate folder for fonts.

Open the CSS file that comes with the kit, copy the @font-face code there, and paste it in the main CSS file for your template. It usually goes at the very beginning of the CSS file, at the top.

For a font named “charlotte”, the code will look something like this:

```
@font-face {
  font-family: "charlotte";
  src: url("charlotte-webfont.woff2") format("woff2"),
       url("charlotte -webfont.woff") format("woff");
}
```

If you put the font files in a folder, you’ll have to change the path; for instance, if you put the font files in a folder called “fonts”, here’s what the code would look like:

```
@font-face {
  font-family: "charlotte";
  src: url("fonts/ charlotte -webfont.woff2") format("woff2"),
       url("fonts/ charlotte -webfont.woff") format("woff");
}
```

Whenever you want to use this font, just use the font-family name. For instance, if you want your h1 to be in the “charlotte” font, your code would say this:

```
h1 {
  font-family: "charlotte";
}
```

More detailed instructions and explanations can be found [The Essential Guide to @font-face. https://www.webfx.com/blog/web-design/font-face-guide/](https://www.webfx.com/blog/web-design/font-face-guide/).