# HOW TO USE THIS TEMPLATE

## List of included files:

1\_column\_module.html

1\_column\_module\_with\_aligned\_table.html

2\_column\_module.html

3\_column\_module\_for\_images.html

Button.html

indented\_bulleted\_list.html

responsive\_footer\_links.html

Template\_base.html

## The code itself:

All files included in this template are coded in a method that is a hybrid of responsive and fluid layouts. This is most evident by the Outlook conditional content and the use of inline-block displayed divs and list items. Do not modify the conditional content (other than background color in the 3 column layout) or the display values. Each file has an explanation at the top of how the layout maybe changed safely.

## How To Start:

Begin with the Template\_base.html file. It has the basic shell of the email, including the header with styles, the body wrapper (some email clients do not respect styles on the body, so a table is utilized to mimic the body), and a centered email wrapper.

## Structuring The Insides:

You will build your email in the same manner as you read; right to left, top to bottom. Think of your email in terms of horizontal sections that are stacked vertically. You will build your email by stacking the selected 1, 2, or 3 column layout code on top of each other, in between the comments <!-- Begin modular layout --> and <!-- End modular layout --> in the Template\_base.html file. Be sure to copy all the code in the selected column file, including the beginning and ending Outlook conditional content. An example is as follows:

<!-- Start modular layout -->

<!--[if (gte mso 9)|(IE)]>

<table width="600" align="center" cellpadding="0" cellspacing="0" border="0">

<tr>

<td>

<![endif]-->

ONE COLUMN LAYOUT UNCONDITIONAL CODE

<!--[if (gte mso 9)|(IE)]>

</td>

</tr>

</table>

<![endif]-->

<!--[if (gte mso 9)|(IE)]>

<table width="600" align="center" cellpadding="0" cellspacing="0" border="0">

<tr>

<td>

<![endif]-->

TWO COLUMN LAYOUT UNCONDITIONAL CODE

<!--[if (gte mso 9)|(IE)]>

</td>

</tr>

</table>

<![endif]-->

<!-- End modular layout -->

Note that when your email is viewed on mobile, all right-hand columns will slide underneath the left-hand ones. Your mobile layout of a 1 column on top of a 2 column will look like the following in mobile:

|  |
| --- |
| 1 COLUMN LAYOUT – COLUMN 1 |
| 2 COLUMN LAYOUT – COLUMN 1 |
| 2 COLUMN LAYOUT – COLUMN 2 |

## Header CSS:

DO NOT add any CSS to the header that is not meant for the mobile view using media queries. ALL styles for the desktop view MUST be inline.

Some of the column files have styles at the top. Once you have copied the body code into the template\_base, copy the css contained in the style tags on the column files and paste it UNDER THE 480 MEDIA QUERY in the template\_base file. Without the column specific CSS, the layouts will not function correctly.

## Guidelines for modifying/adding CSS:

As a reminder, the header is for media query CSS ONLY. All styles that are not meant for a media query must be inline.

Apply all content styles to the tds, not to the tables. CSS applied to tables should be limited to defining width, border, and background-color, and zeroing padding and margin.

Avoid header and paragraph tags, as these elements are rendered differently across email clients. Create paragraphs with two break tags at the end of the last sentence. Create headers using span tags.

The html font tag should only be used in cases where an imported font is not working on a link. A font tag should never have to be used outside of this case. If a span tag is not producing the selected font, create a new table to house the content to which the special font should be applied.

## Using non-column files:

Button.html – This file includes code for a centered HTML button. HTML buttons are preferable to image buttons because the CTA image can still be read with image-blocking on.

indented\_bulleted\_list.html – List items in a traditional unordered or ordered list do not render the same across email clients. Therefore, you should use a table structure to mimic a list. This file contains that structure.

responsive\_footer\_links.html – Always keep in mind thumb space when developing for mobile. This file contains a helpful way to change a horizontal list of links into a horizontal one with plenty of white space. A link in its own, separate td is not an alternative suggested method, as that way is not fluid and will not look attractive in Gmail app and Gmail & Yahoo below 570px width.

## Useful links:

### Good primer on how to structure content within this template:

Professional Email Design by Jason Rodriguez – Ask your manager for copy if one is not already printed and available.

### How to hybrid code:

<http://labs.actionrocket.co/the-hybrid-coding-approach>

<http://labs.actionrocket.co/the-hybrid-coding-approach-2>

### What CSS is supported in which email client:

<https://www.campaignmonitor.com/css/>

### Creating a centered, responsive design without media queries: <https://www.campaignmonitor.com/blog/post/4240/creating-a-centred-responsive-design-without-media-queries>