HTML Email Coding Tips For Web Designers

In response to some of our other articles on HTML emails I've heard readers say things like "I'm not a complete noob. I can code web pages with a text editor, I'm darn good with CSS, I've got my own server, and I've even setup my own opt-in database. But why am I not able to design a sharp-looking HTML email? Any tips for the more advanced user?" If you're a web designer and geek entrepreneur, and you know your stuff, I'll spare you all the WYSIWYGs and "Hello World" examples, and give you a quick, no-nonsense "Quick Start Guide" to HTML Email. HTML Email should be sent in "multipart-alternative" (or MIME) format. That means the email is embedded with both the plain-text version, and the HTML version, of the message. When people can't (or won't) view HTML email, the plain-text version displays instead. That's the primary reason to use a system like Follow Up Autoresponder which has you create both versions when you create a message. The fact of the matter is that products such as Follow Up Autoresponder are essential to your success because even if you figure out how to deliver in multipart alternative MIME format, you still need open, click, and bounce handling scripts, plus CRON jobs that remove false email accounts or innactive subscribers. Another point to consider is that Image files don't get sent along with HTML email as attachments. You host images on your server, then code the HTML in your email to point to them with absolute paths. In other words, you code something like this: **** Rather than like this: **** or this: ****. Notice how I included the whole web address in the first example. What you are doing is referencing the image on your server rather than by the image name or the relative directory they are located in because there is actually no relationship between the images and the email itself, they are simply being referenced. Design for the preview frame of email applications. That means you've got around 500 to 600 pixels max for your email designs. It's not likely that recipients are going to view your message in full screen "to appreciate all its glory," so keep those dimensions in mind. Most cool stuff won't work. With web pages, you are able to use CSS to position things, and break up content to enhance appearance. That won't work in email applications. Cast aside fancy DIVs and XML. Dust off those tables and font tags, because those are the most reliable tools for HTML email coding. Fancy CSS (particularly DIVs and positioning) won't work dependably. Testing is different with HTML email. When you code a web page, you test it in IE, Firefox, and Safari, right? And you sort of learned to "leave a little wiggle room" so that your designs function across all browsers. Well, when you code an HTML email, you must test it in Outlook 2003, Outlook Express, Mozilla Thunderbird, AOL, Yahoo!Mail, Google's Gmail, Hotmail, the list goes on. And just like a web browser they all display HTML email a bit differenty. When I say, "Keep your HTML email simple," I really mean it! You will save yourself the time of having to test in all these different applications if you follow the steps in our article 10 HTML Email Coding Tips. There isn't a whole lot of CSS support when coding HTML emails. You will be able to use a little CSS, such as for basic fonts and colors. But use it sparingly, and design things to fail gracefully. Use inline CSS, not a linked styelsheet. Embed CSS below the Inline CSS is safest, but if you have to embed your CSS, embed it below the BODY tag because browser-based email applications (like Gmail, Hotmail, Yahoo!Mail, etc) strip out your HTML, HEAD, and BODY tags so they won't interfere with their own web page. Make sure to Indent the CSS and don't start with periods. Don't ever have any lines in your email or HTML code start with a period. Some email servers render lines that begin with periods as the, "end of message" command, so your email will get stripped anywhere below that line. So be extra careful in your embedded CSS definitions. Add spaces in front of any lines of your CSS that start with a period. Anti-virus programs are killjoys. JavaScript, ActiveX, Flash, and embedded movie files won't work. The reason for this is because more or less everybody has anti-virus applications that block any of those from running. So the best thing to do is simply resist the tempation and refrain from using any elements of Flash or Javascript entirely. Remember how I said that most browser based email applications (like HotMail) strip out your BODY tag? Well that's important to consider when assigning background colors to the body of your email messages. They won't always work depending on what application the recipient is using. The most dependable way of applying a background color to your email is to wrap it all inside a big, 100% wide table, and give the table cell a bgcolor. Don't forget your plain-text version. When web designers create HTML email, they often neglect the plain-text versions, but plain-text emails are very important. If you ignore them, some spam filters will even think you're a sloppy coder, and throw your mail into the junk folder. Some tips for plain-text emails: You get absolutely no formatting. This is what you would call "super old school" but hey, we're all geeks here! Wrap them at about 60 or 70 characters. Use a plain-text editor (not Microsoft Word), and type a letter (I like the letter "W") across the top of the document about 60 times. That's your ruler. When you type your message and reach the end of the ruler, hit enter. With Follow Up Autoresponder you won't actually have to measure out the width because the program does this for you, how convenient! One big thing to remember is that in plain text emails your links to other pages need to simply be written out as the absolute path. So instead of www.othersite.com you would have to write http://www.othersite.com. And for obvious reasons you wouldn't put the href tag around your links in a plain text email either. You need a good balance of "graphics vs. text." Otherwise, spam filters will get you. Don't send one ginourmous graphic. Balance it out with some copy, particularly your company contact information and an unsubscribe link. Don't write like a spammer. Watch those exclamation points and don't make it sound like you're yelling at your recipients. Also be sure to close all those table cells and font tags so that your code will be clean. Some spam filters will punish you for using FrontPage (or other WYSIWYG tools) to code your HTML email. So strip out all those useless META tags that those applications produce. This can all be avoided by using the WYSIWYG view within Follow Up Autoresponder as well as most other autoresponder programs which don't include all the extra code. Here are a few things that every email campaign should have...

         - An instantly recognizable from-name and reply-to address. Use your company name, for example. Then, don't change it, because recipients might have whitelisted you.   - A good, descriptive subject line that's not spammy. You've got a blink of an eye to convince recipients that your message isn't junk.

        - A one-click unsubscribe link with a description saying something like "click here to stop receiving these messages".   - Your "offline" contact information, including your physical mailing address, phone number, etc. It will help people remember who you are because subscribers tend to forget who you are if you don't mail out often.

About the Author

Source: http://www.millionairegeeks.com