

Math in HTML

You can put math in your web pages & email (& elsewhere, if you want to use HTML as "cross-media"), without resorting to ugly or oversized pictures, attachments, obscure TeX code, or things like MathML that are hard to edit by hand & few browsers support yet.

MathML will support fancier equations (in browsers, @ least). **MathJax** does it now, as javascript. To get any of the fancier stuff (big integral signs, etc.) to work, you need new fonts. For now, these either have to be installed (most readers won't) or repeatedly downloaded by the browser (slowing things down; not enormously, but even a few seconds is considered slow by today's standards). This is the best way to go if you need sophisticated math, but simpler stuff (symbols, sub/superscripts) can do without it, & is easier to proofread (mostly WYSIWYG, not TeX).

Unicode

It mostly involves just using more characters in the usual fonts. Unicode fonts are 16-bit (2^{16} characters), and so have a lot more than your standard 8-bit ASCII fonts. In particular, they include almost all the Greek and math(s) symbols you need for science: e.g.,

Greek: ΓΔΘΛΞΠΣΥΦΨΩαβγδεζηθικλμνξπρστυφχψωθϖ

Accented, e.g.: čç

Combining accents: àáâãäåǎǏǐǒǓǔǖǗ

Arrows: ←↑→↓↔↕↔↔⇒⇈⇉

Math: ħħ□∂∇∏Σ∫∫√∞±∓·×∈⊂~≈≅≈≠≡≤≥≦≧≪≫⊕⊗½¼¾

Misc: ⟨⟩†‡★☺☻

Large delimiters (missing on some computers & mobile devices): (\) (etc.)

Actually, Unicode is mostly about languages, but you get the math for free. You may have to use an appropriate font, that has all the characters, especially the combining accents; e.g., Lucida Grande (Mac)/Lucida Sans Unicode (MS Windows) seems to have almost everything.

HTML also has the <sub> & <sup> tags. Between using the right characters, and subscripts & superscripts, you have almost everything you need for basic math.

Reading

That means you can read such "letters" directly in various programs, like:

Web

You need the tag

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

right after the <head> tag at the top of your web page. (Alternatively, you can type in character codes for each symbol, but this page is about avoiding that: see below.)

Email

Modern email programs also can read HTML, i.e., just about anything you can put on a web page (e.g., look at some of the spam you've been getting). If you write email in languages other than English, you should already know this. (Yes, Greek is used for more than just math.)

[my recommendations for mailers](#)

TeX

You can use Unicode characters in TeX source files with [Xe\(La\)TeX](#) or [Lua\(La\)TeX](#). Then you can write macros to define those characters, Greek & math (e.g., `\def \alpha{\alpha}`), after making its `\catcode` active (in that example, `\catcode"03B1=13`). Unfortunately, this won't work for accents because accents are typed before a character to be accented in TeX, but after in Unicode. (In TeXShop, add the line `"%!TEX encoding = UTF-8 Unicode"` (no quotes) @ the top to be able to save in Unicode if it isn't your default.)

Tricks

Most of the following tricks are rather messy, so not the kind of stuff you'd want to do regularly. (@ this point you might consider the available alternatives mentioned @ the beginning.)

There are some things that aren't individual symbols, accents, subscripts or superscripts. Also, some combining accents appear too low, so $\acute{\alpha}$ (α -dot) is legible but $\dot{\beta}$ (β -dot) isn't. There are several ways to put things on top of each other, for fractions and mathematical accents, but among different browsers most work nonuniformly (because of different ideas on how spacing should be implemented), or not at all (because they are recent standards). The best way would be `"display:inline-block"`, which works poorly on Internet Explorer (for Windows) and even worse on Firefox, or `"inline-table"`, which doesn't exist on Explorer. Another way would be to use `"position:absolute"`, whose exact position varies among browsers, and is particularly far from the rest for Opera. `"Line-height"` also varies among browsers, and sometimes makes pieces of things disappear.

Tables

So the best results seem to be to use `"table"` with `"display:inline"` (for use in a paragraph), and `"border:top"` (or bottom) for fractions. (Explorer also needs the table border color set somewhere.) For vertical positioning in paragraphs, `"vertical-align:text-bottom"` works for single-row tables (which works best for accents), and `"position:relative"` for sums with delimiters, but for double-row tables (like fractions) every browser has its own ideas for positioning. (Maybe fractions don't belong inline anyway. For displayed equations, you can always put the rest into the table.)

This seems to work pretty well for the latest versions of all still-supported major browsers: Safari (& other WebKit-based browsers like OmniWeb, Shiira, iCab, etc.), Firefox (& other Gecko-based browsers like Camino, Mozilla Seamonkey, ...), Opera, and [Internet Explorer](#) (on Windows). (Even looks good on [Konqueror](#).)

For example, for fractions,

How much is

```
<table style="display:inline; text-align: center; vertical-align:middle">
<tr><td> $\alpha+\beta$ </td></tr>
<tr><td style="border-top:solid 2px"> $4\pi$ </td></tr>
</table> ?
```

gives

How much is $\frac{\alpha+\beta}{4\pi}$?

while for accents and delimiters,

```
a<table style="display:inline; text-align: center;
vertical-align:text-bottom" cellspacing=0 cellpadding=0><tr><td>
<small style="position:relative; bottom:-1.4ex"> $\rightarrow$ </small><br>
x</td></tr></table><sup>2</sup>
+
a<table style="display:inline; text-align: center;
vertical-align:text-bottom" cellspacing=0 cellpadding=0><tr><td>
<span style="font-size:22px; position:relative; bottom:-1.6ex">'</span>
<br>
 $\beta$ </td></tr></table><sup>2</sup>
+
<table style="display:inline; text-align: center; position:relative;
bottom:-2.3ex; line-height: 100%"><tr>
<td><small> $\infty$ </small><br> $\sum$ <br><small>n=1</small></td>
</tr></table>x = 5
```

gives

$$a\vec{x}^2 + a\dot{\beta}^2 + \sum_{n=1}^{\infty} x = 5$$

Other

Combining accents don't work so well for tall letters, so you might want to use overlines instead of bars: e.g.,

```
<span style="text-decoration:overline"> $\Lambda$ </span>
```

gives

$\overline{\Lambda}$

vs. $\bar{\Lambda}$ (cf. \bar{x} and \overline{x}). Overlines are also good for completing square roots: e.g.,

```
 $\sqrt{\text{span style="text-decoration:overline"}-(dX\wedge dX)^2\text{/span}}$ 
```

gives

$$\sqrt{\overline{-(dX\wedge dX)^2}}$$

Typing

Reading such characters is easier than getting them in there in the first place. For HTML, you can type names of the characters, like in tex, such as `α` for α . (See [common symbols](#).) But it gets worse for less common characters, like `∓` for \mp . (See this [detailed list](#).) Again, a good text-editing program can make it easier, by giving you a "palette" of such letters. For example, in Mac OS X, there is a Character Viewer available to any (Cocoa) program for such purposes. At worst, you can make a file containing all the common characters once the hard way, then cut and paste from then on.

To make it even easier yet, I made a ["keybord layout"](#) for Mac OS X that allows you to type all the Greek and more useful math characters directly from the keyboard using the Option key (e.g., Option-a for α). (Actually, I already made a [TrueType font](#) that lets me do that, but nobody can read that unless they have the font, and can read Mac OS Roman encoding, whereas anybody can read Unicode with the standard fonts.) It installs in `/Library/Keyboard Layouts` (or `~/Library/...`). I created it with [this web program](#). (See this [Apple technical note](#) for more info.) There is also the program [Ukelele](#).

Something similar can be done for XWindows, by running the command `xmodmap` (by hand or in your `.xinitrc`) on a file created by `xmodmap -pke` and edited according to codes given in (on my machine) `/usr/X11R6/include/X11/keysymdef.h`, but it doesn't seem possible to get quite all the Unicode characters I would want (e.g., $\leftrightarrow \oplus \otimes \mp \langle \rangle \leftarrow \Sigma \prod \in \notin \ll \gg$). You can also install the program [xkeycaps](#), which does the same with a nice GUI.

For MS Windows there is [Microsoft Keyboard Layout Creator](#). See also this [list of links](#) for other tools for all operating systems.

For Mac OS X Snow Leopard another alternative is to use the menu item `Edit>Substitutions>Text Replacement`, after making some definitions in `System Preferences>Language & Text>Text`. Then you can make your own abbreviation for a symbol, & it's automatically replaced. (E.g. `\a` might get converted into α .) For other systems there exist programs to do the same.

EXIT

This page should look like [this](#).