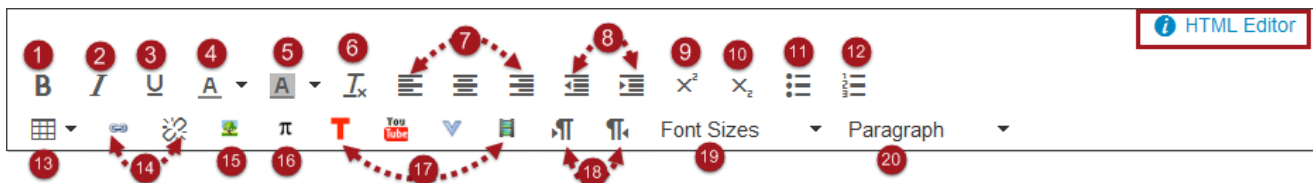


Instructional Designers: Rich Content Editor HTML Cheatsheet

Posted by [Susan Nugent](#) May 4, 2015

I created a HTML cheatsheet in Word last year for a workshop on Canvas pages. I needed to updated it so I thought I would share it here. Please note this cheatsheet is subject to change with Canvas updates.

Below is an image of the toolbar highlighted with numbers of each command. While it is not necessary to know HTML ([HyperText Markup Language](#)) and CSS ([Cascading Style Sheets](#)) to use the rich content editor in Canvas it sure helps to understand the basics if you need to fix something in code view. Each numbered command has a code example with some tips on using in the HTML Editor. Use the HTML editor to switch to code view so you can edit the code. Please note in the rich content editor there are only certain [HTML elements \(Tags\) that are allowed](#) and any elements added that are not allowed will be stripped out of the page when you save the page.



1-Bold

Code example:

```
<strong>Some Text</strong>
```

Notes:

The Strong element is used bold text. It is generally not recommended to use the strong element to create page headings. Use the actual heading elements to create this type of structure. See number 20 below for details on why.

2-Italics

Code Example:

```
<em>Some Text</em>
```

Notes:

The em element is used to italicize text. Generally speaking it probably should be used sparingly on webpages because the text can be hard to read on monitor.

3-Underline

Code Example:

```
<u>Some Text</u>
```

Notes:

This element can be used to emphasize text; however, on webpages underlined text is often confused with hyperlink text. I generally don't recommend using this element.

4-Text Color

Code Example:

```
<span style="color: #ff0000;">Some Text</span>
```

Notes:

This command creates a span element and inline CSS (the style attribute) to create the colored text. The style attribute can be applied to any text element such as paragraphs and headers. In the rich content editor toolbar there are only about 40 colors to choose from; however, in the code view you can change the color to any color you want by altering the hex color code. Two of my favorites place to find hex color codes are <http://www.colourlovers.com/colors> and [Adobe Color CC](#).

5-Background Color

Code Example:

```
<span style="background-color: #ff9900;">content</span>
```

Notes:

This command uses the span element and inline CSS (the style attribute) to create the background color. This should be used cautiously with text. If the background color and text color do not have enough contrast

between them, the text can be hard to read. In the example below the text is hard to read. This can be especially hard on color blind people or people like me who are losing their sight to old age. For further reading, view this Smashing Magazine article, [Design Accessibly, See Differently: Color Contrast Tips And Tools](#). On a side note, the Jive editor does not have background color element in the toolbar and does strip it when you try to add in code view so I had to use an image for this example.

Be careful how you color and highlight text. It may become unreadable

6-Clear Formatting

Notes:

This option is handy for getting rid of the extra HTML code that sometimes comes over when you copy and paste text from other locations such as from Word or other websites. It is important to note that this option works with most HTML elements but doesn't seem to work with the background element (see number 5 above). You can go to code view to remove the span element. If you are designer, I recommend using a text editor that has a good find and replace command to remove any extra HTML and CSS code before moving the text to Canvas. I use Dreamweaver's find and replace for this type of task a lot and it saves me quite a bit of time.

7-Text Alignment

Code Examples:

There are three alignment options and all of these are attributes that can be applied to headings and paragraph elements. The left alignment is the default in the editor. **Note:** Only use center and right alignment for headers or short lines of text. It is generally not recommended to use the center or right alignment for longer lines of text because the text is hard to read.

```
<p style="text-align: left;">Paragraph of text</p>
<p style="text-align: center;">Paragraph of text</p>
<p style="text-align: right;">Paragraph of text</p>
```

Notes:

For further reading I recommend the WebAIM articles, [Writing Clearly and Simply](#) and [Text/Typographical Layout](#).

8- Outdent/Indent

Code Example:

What this option does depends on element it is applied to in the code. See examples below.

When the indent option is applied to paragraph element the style attribute is applied to the paragraph element with padding of 30 pixels.

```
<p style="padding-left: 30px;">Some text</p>
```

When the indent is applied to an unordered or ordered list a new nested list is created.

```
<ul>
<li>Some Text</li>
  <ul>
    <li>Some indented text</li>
  </ul>
</ul>
```

Notes:

See 11 and 12 for more details on writing lists.

9-Superscript

Code Example:

```
H<sup>2</sup>0
```

Notes:

This element is used to display text and numbers above the baseline. Be sure to only select the text that should be superscript when applying this command in the editor. If you make mistake use the remove formatting command (no. 6) to remove it or switch to code view to fix any issues that you might not be able to fix with the toolbar.

10-Subscript

Code Example:

```
2<sub>4</sub>
```

Notes:

This element is used to display text or numbers below the baseline. The same applies as number 9 above.

11-Unordered List

Code Example:

```
<ul>
<li>List Item</li>
<li>List item</li>
<li>List Item</li>
</ul>
```

Notes:

Unordered lists are good for list of items where the sequence of the items does not matter. Lists can be nested using the indent option. I have found this to be tricky sometimes so I prefer to edit lists in code view. See example of nested list in number 8.

12-Order List

Code Example:

```
<ol>
<li>Do this first</li>
<li>Do this second</li>
<li>Do this third</li>
</ol>
```

Notes:

Order lists are good when you are giving students a set of an instructions for homework assignments. You can alter the list number to display letters if preferred. This must be done in code view. See example code below.

```
<ol type="a">
<li>Do this first</li>
```

```
<li>Do this second</li>
<li>Do this third</li>
</ol>
```

13-Table

Code Example:

Table code involves several different elements. See code example below.

```
<table border="0">
<caption>Caption</caption>
<tbody>
<tr>
<td>Row 1</td>
<td>28</td>
</tr>
<tr>
<td>Row 2</td>
<td>23</td>
</tr>
</tbody>
</table>
```

Notes:

The new toolbar has a much improved table editor now so you may not need to switch to code view that much now. I will note that tables should only be used for tabular data; however, the majority of people do not use them this way. This stems from some [bad web design hacks](#) from the late 90s which can still spark heated debate about their use in designing webpages. The key point to remember is that you want your pages to be accessible to all. For further reading, visit the WebAIM article, [Creating Accessible Tables](#).

14-Link

Code Example:

```
<a href="www.google.com">Go to Google</a>
```

Notes:

The color of the link is controlled by the CSS (Cascading Style Sheets) that is linked to the HTML document. See [Canvas Styleguide](#) for more details. For further reading, read the WebAIM article, [Links and Hypertext](#) and [Accessible CSS](#).

15-Picture

Code Example:

```

```

Notes:

Images can be pulled from the web or Canvas files. Images do have several attributes that can be applied to the img element. When you add or edit the image in the editor the dialog box has options for adding alternative text and changing the width and height attributes. For further reading, read the WebAIM article, [Accessible Images](#).

16-Symbol

Code Example:

```

```

Notes:

When this option is used in the editor the equation editor will display in a dialog box. You can use the editor options or write the equations in [LaTeX](#). In the rich content editor the equation will be rendered as an image with the LaTeX as alternative text.

17-Embedded Objects, Media Comment & Other LTI Tools

Code Example:

```
<iframe width="640" height="360"src="//www.youtube-nocookie.com/embed/WetLiIvTwZE?rel=0" frameborder="0" allowfullscreen></iframe>
```

Notes:

When using the [LTI](#) and [Media Comment](#) tools the content in most cases will be embedded objects. The main issue with some of the LTI Tools is unsecure content. Canvas is hosted on secure server and almost all browsers will now block unsecured embedded content on secure webpages. You can also embed objects

by pasting the embed code in code view. For details on how to embed documents view the [GoogleDocs](#) and [Microsoft Documents](#) tutorials.

18-Text Direction

Code Example:

```
<p dir="rtl">Some text</p>
```

Notes:

This attribute is essential for setting how script languages (Arabic, Hebrew etc) will display and be read on the webpage. For more details, go to the WC3 article, [Structural markup and right-to-left text in HTML](#)

19-Font Sizes

Code Example:

```
<span style="font-size: x-large;">some text</span>
```

Notes:

Uses the span element and inline CSS (the style attribute) to create the larger text. It is generally not recommended to use this option to create headings. See number 20 below on why.

20-Paragraph and Header Elements

Code Example:

Paragraph

```
<p>Some body text</p>
```

Heading 1

```
<h1>Some Header text</h1>
```

Preformatting

```
<pre>Some Text that will display as you type it</pre>
```

Notes:

The paragraph element surrounds each paragraph of text you create when you use the return/enter key on the computer. In the editor you can specify some text to be headers. The headings are typically larger text that create structure for your pages and makes it easier to read especially for people who view your content in screenreader and on mobile devices. Unstyled paragraphs and headings will have default padding and margins applied by the browser. In Canvas paragraph and headings are styled with the linked CSS documents. You can applied style attribute in paragraph and heading elements to change the font and margins if desired to have a different look that the default editor settings (see examples in 5, 7, 8, and 19). This will create inline CSS which trumps the linked CSS. I generally don't recommend doing that for all your pages because you must edit each element on the page to make this change. That is too much work. A better option would be to get your IT people at your institution to setup [KennethWare](#) so you can create pages with consistent look and feel. I am working with ours to hopefully get this setup for our instance of Canvas. The important key here is that paragraphs and headings are considered structural elements in HTML and are essential to making your pages accessible to all. For further reading, visit the WebAIM articles, [Semantic Structure](#) and [Designing for Screen Reader Compatibility](#). I also recommend viewing the recording of the CanvasLIVE webinar [Mobile Series: Just-in-Time Design](#) and joining the [Canvas Mobile Users](#)

Additional Resources

- [Learn the Basics of HTML](#) - via the W3C website
- [HTML Dog Tutorials](#) - easy to understand tutorial for beginners
- [Sitepoint HTML Reference](#) - a comprehensive guide to all the HTML elements

629 Views Tags: [rce](#), [css](#), [code](#), [code-snippets](#), [html](#), [rich content editor](#), [text](#)



[John Sills](#)

May 7, 2015 8:24 AM

EXCELLENT resource thank you for sharing. This will be invaluable!



[Mary Wiseman \(Designer\)](#)

May 7, 2015 7:13 AM

Thank you for creating this Susan...it will be on my bookmark bar.



[Stefanie Sanders](#) in response to [Tom Gibbons](#) on page 10

May 6, 2015 6:36 AM

Thank you! This one goes straight into my bookmarks.

"The second rule of learning HTML: **Steal.**" LOLOL.



[Gerol Petruzella](#) in response to [Tom Gibbons](#) on page 10

May 6, 2015 4:44 AM

[Tom Gibbons](#) , who cares about mess when there's frogs? :-D Thanks for sharing this.



[Tom Gibbons](#)

May 5, 2015 5:44 PM

Lovely. Following the organizational structure of the RCE is kind of genius.

My HTML module is a bit of a wreck in terms of organization, but in the interest of sharing, there may be things in it that are useful: [What is HTML: Advanced Canvas Trainings](#)



[Gerol Petruzella](#)

May 5, 2015 10:47 AM

What a great resource, [Susan](#)! Many thanks for sharing.



[Stefanie Sanders](#)

May 5, 2015 6:58 AM

Awesome, [Susan](#)! This is unbelievably helpful. I'm going to add a link to this document to any and every blog that even *mentions* HTML!



[Catherine Coltman](#)

May 5, 2015 2:42 AM

Thank you for sharing.



[Ryan Seilhamer](#)

May 4, 2015 5:48 PM

Wow, so very helpful! Thanks for sharing!



[Chris Hofer](#)

May 4, 2015 8:49 AM

Nice! This is something I think I'll post for our faculty at some point on our Canvas Dashboard. Bookmarking it now!



[Kona Jones](#)

May 4, 2015 8:47 AM

This is really useful [Susan](#)! Thanks for sharing!!

