

Using HTML in Wiki Text

Trac supports the display of HTML in any wiki context, by using the `#!html` [WikiProcessor](#).

However, this HTML has to be well-formed. In particular, you can't insert a start tag in an `#!html` block, resume normal wiki text and insert the corresponding end tag in a second `#!html` block.

Fortunately, for creating styled `<div>`s, ``s or even complex tables containing arbitrary Wiki text, there is a powerful alternative: `#!div`, `#!span` and `#!table`, `#!tr`, `#!td` and `#!th` blocks. Those Wiki processors are built-in and do not require additional packages to be installed.

How to use `#!html`

To inform the wiki engine that a block of text should be treated as HTML, use the *html* processor:

Wiki Markup	Display
<pre>{{{ #!html <h1 style="text-align: right; color: blue">HTML Test</h1> }}}</pre>	HTML Test

Note that Trac sanitizes your HTML code before displaying it. That means that potentially dangerous constructs, such as Javascript event handlers, will be removed from the output.

The filtering is done by [Genshi](#) and the output will be a well-formed fragment of HTML. This means that you can no longer use two HTML blocks, one for opening a `<div>` and another for closing it, in order to wrap arbitrary wiki text. The new way to wrap any wiki content inside a `<div>` is to use the `#!div` Wiki processor.

How to use `#!div` and `#!span`

Wiki Markup	Display
<pre>{{{ #!div class="important" **important** is a predefined class. }}}</pre>	important is a predefined class.
<pre>{{{ #!div style="border: 1pt dotted; margin: 1em" **wikipage** is another predefined class that will be used when no class is specified. }}}</pre>	wikipage is another predefined class that will be used when no class is specified.
<pre>{{{ #!div class="compact" style="border: 1pt dotted; margin: 1em" **compact** is another predefined class reducing the padding within the `<code><div></code>` to a minimum. }}}</pre>	compact is another predefined class reducing the padding within the <code><div></code> to a minimum.
<pre>{{{ #!div class="wikipage compact" style="border: 1pt dotted" Classes can be combined (here **wikipage** and **compact** which results in this case in reduced //vertical//</pre>	Classes can be combined (here wikipage and compact) which results in this case in reduced <i>vertical</i> padding but there's still some horizontal space for coping with headings. Explicitly specifying no classes is <i>not</i> the same as specifying no class

Wiki Markup

```
padding but there's still some horizontal space for copy attribute, as this will remove the
with headings.
}}}
```

```
{{{
#!div class="" style="border: 1pt dotted; margin: 1em"
Explicitly specifying no classes is //not// the same
as specifying no class attribute, as this will remove
the //wiki page// default class.
}}}
```

Display

attribute, as this will remove the
wiki page default class.

Note that the contents of a `#!div` block are contained in one or more paragraphs, which have a non-zero top and bottom margin. This leads to the top and bottom padding in the example above. To remove the top and bottom margin of the content, add the `compact` class to the `#!div`. Another predefined class besides `wiki page` and `compact` is `important`, which can be used to make a paragraph stand out. Extra CSS classes can be defined via the `site/style.css` file for example, see [TracInterfaceCustomization#SiteAppearance](#).

For spans, you should use the Macro call syntax:

Wiki Markup

```
Hello
[[span('WORLD' (click [#anchor here]), style=color: green; font-size: 120%, id=anchor)]]!
```

Display

Hello **WORLD (click here)!**

How to use `#!td` and other table related processors

The `#!td` or `#!th` processors should be used to create table data and table header cells, respectively. The other processors `#!table` and `#!tr` are not required for introducing a table structure, as `#!td` and `#!th` will do this automatically. The `|` – row separator can be used to start a new row when needed, but some may prefer to use a `#!tr` block for that, as this introduces a more formal grouping and offers the possibility to use an extra level of indentation. The main purpose of the `#!table` and `#!tr` is to give the possibility to specify HTML attributes, like *style* or *valign* to these elements.

Wiki Markup

```
Simple 2x2 table with rich content:
{{{#!th align=left
- Left
- Header
}}}
{{{#!th align=left
- Right
- Header
}}}
|-----
{{{#!td style="background: #ffd"
- Left
- Content
}}}
{{{#!td style="vertical-align: top"
!RightContent
}}}
|-----
```

Display

Simple 2x2 table with rich content:

• Left	• Right
• Header	• Header

• Left	RightContent
• Content	

... and this can be mixed with pipe-based cells

Pick the style the more appropriate to your content

Wiki Markup

```
|| ... and this can be mixed||\
||with pipe-based cells ||
{{{#!td colspan=2
Pick the style the more appropriate
to your content
```

```
See WikiFormatting#Tables for details
on the pipe-based table syntax.
}}}
```

If one needs to add some attributes to the table itself...

```
{{{
#!table style="border:none;text-align:center;margin:auto"
  {{{#!tr =====
    {{{#!th style="border: none"
      Left header
    }}}
    {{{#!th style="border: none"
      Right header
    }}}
  }}}
  {{{#!tr ==== style="border: 1px dotted grey"
    {{{#!td style="border: none"
      1.1
    }}}
    {{{#!td style="border: none"
      1.2
    }}}
  }}}
  {{{#!tr =====
    {{{#!td style="border: none"
      2.1
    }}}
    {{{#!td
      2.2
    }}}
  }}}
}}}
```

Display

- Left
- Header
- Right
- Header

See [WikiFormatting#Tables](#) for details on the pipe-based table syntax.

If one needs to add some attributes to the table itself...

Left header Right header

1.1 1.2

2.1 2.2

Note that by default tables are assigned the "wiki" CSS class, which gives a distinctive look to the header cells and a default border to the table and cells, as can be seen for the tables on this page. By removing this class (`#!table class=""`), one regains complete control on the table presentation. In particular, neither the table nor the rows nor the cells will have a border, so this is a more effective way to get such an effect rather than having to specify a `style="border: no"` parameter everywhere.

Wiki Markup

Display

```
{{{#!table class=""
|| 0|| 1|| 2||
|| 10|| 20|| 30||
|| 11|| 22|| 33||
|||||= numbers =||
}}}
```

numbers

Other classes can be specified as alternatives (remember that you can define your own in [site/style.css](#)).

Wiki Markup	Display
<code>{{{#!table class="listing" 2</code>	
<code> 0 1 2 </code>	
<code> 10 20 30 </code>	10 20 30
<code> 11 22 33 </code>	11 22 33
<code> = numbers = </code>	
<code>}}}</code>	numbers

HTML comments

HTML comments are stripped from the output of the `html` processor. To add an HTML comment to a wiki page, use the `htmlcomment` processor, available since Trac 0.12:

Wiki Markup	Display
<code>{{{</code>	
<code>#!htmlcomment</code>	
<code>This block is translated to an HTML comment.</code>	
<code>It can contain <tags> and &entities; that will not be escaped in the output.</code>	
<code>}}}</code>	
<code><!--</code>	
<code>This block is translated to an HTML comment.</code>	
<code>It can contain <tags> and &entities; that will not be escaped in the output.</code>	
<code>--></code>	

Please note that the character sequence `--` is not allowed in HTML comments, and will generate a rendering error.

More Information

- <http://www.w3.org/> -- World Wide Web Consortium
- <http://www.w3.org/MarkUp/> -- HTML Markup Home Page

See also: [WikiProcessors](#), [WikiFormatting](#), [WikiRestructuredText](#)