

## Wiki Processors

Processors are [WikiMacros](#) designed to provide alternative markup formats for the [Wiki engine](#). Processors can be thought of as *macro functions to process user-edited text*.

Wiki processors can be used in any Wiki text throughout Trac, for various different purposes, like:

- [syntax highlighting](#) or for rendering text verbatim,
- rendering [Wiki markup inside a context](#), like inside `<div>` blocks or `<span>` or within `<td>` or `<th>` table cells,
- using an alternative markup syntax, like [raw HTML](#) and [Restructured Text](#), or [textile](#)

## Using Processors

To use a processor on a block of text, first delimit the lines using a Wiki *code block*:

```
{{{
The lines
that should be processed...
}}}
```

Immediately after the `{{{` or on the line just below, add `#!` followed by the *processor name*.

```
{{{
#!processorname
The lines
that should be processed...
}}}
```

This is the "shebang" notation, familiar to most UNIX users.

Besides their content, some Wiki processors can also accept *parameters*, which are then given as `key=value` pairs after the processor name, on the same line. If `value` has to contain space, as it's often the case for the `style` parameter, a quoted string can be used (`key="value with space"`).

As some processors are meant to process Wiki markup, it's quite possible to *nest* processor blocks. You may want to indent the content of nested blocks for increased clarity, this extra indentation will be ignored when processing the content.

## Examples

Wiki Markup	Display
<b>Example 1:</b> Inserting raw HTML	
<pre>{{{ #!html &lt;hl style="color: grey"&gt;This is raw HTML&lt;/hl&gt; }}}</pre>	This is raw HTML
<b>Example 2:</b> Highlighted Python code in a <code>&lt;div&gt;</code> block with custom style	
<pre>{{{#!div style="background: #ffd; border: 3px ridge"  This is an example of embedded "code" block:  {{{ #!python def hello():     return 'world' }}} }}}</pre>	<div>This is an example of embedded "code" block:</div> <pre>def hello():     return 'world'</pre>
<b>Example 3:</b> Searching tickets from a wiki page, by keywords.	
<pre>{{{ #!html &lt;form action="/query" method="get"&gt;&lt;div&gt; &lt;input type="text" name="keywords" value="" size="30"/&gt; &lt;input type="submit" value="Search by Keywords"/&gt; &lt;!-- To control what fields show up use hidden fields &lt;input type="hidden" name="col" value="id"/&gt; &lt;input type="hidden" name="col" value="summary"/&gt; &lt;input type="hidden" name="col" value="status"/&gt; &lt;input type="hidden" name="col" value="milestone"/&gt; &lt;input type="hidden" name="col" value="version"/&gt; &lt;input type="hidden" name="col" value="owner"/&gt; &lt;input type="hidden" name="col" value="priority"/&gt; &lt;input type="hidden" name="col" value="component"/&gt; --&gt; &lt;/div&gt;&lt;/form&gt; }}}</pre>	

## Available Processors

The following processors are included in the Trac distribution:

<code>#!default</code>	Present the text verbatim in a preformatted text block. This is the same as specifying <i>no</i> processor name (and no <code>#!</code> )
<code>#!comment</code>	Do not process the text in this section (i.e. contents exist only in the plain text - not in the rendered page).
<code>#!rtl</code>	Introduce a Right-To-Left block with appropriate CSS direction and styling ( <i>since 0.12.2</i> )
<b>HTML related</b>	
<code>#!html</code>	Insert custom HTML in a wiki page.
<code>#!htmlcomment</code>	Insert an HTML comment in a wiki page ( <i>since 0.12</i> ).
	Note that <code>#!html</code> blocks have to be <i>self-contained</i> , i.e. you can't start an HTML element in one block and close it later in a second block. Use the following processors for achieving a similar effect.
<code>#!div</code>	Wrap an arbitrary Wiki content inside a <code>&lt;div&gt;</code> element ( <i>since 0.11</i> ).
<code>#!span</code>	Wrap an arbitrary Wiki content inside a <code>&lt;span&gt;</code> element ( <i>since 0.11</i> ).
<code>#!td</code>	Wrap an arbitrary Wiki content inside a <code>&lt;td&gt;</code> element ( <i>since 0.12</i> )
<code>#!th</code>	Wrap an arbitrary Wiki content inside a <code>&lt;th&gt;</code> element ( <i>since 0.12</i> )
<code>#!tr</code>	Can optionally be used for wrapping <code>#!td</code> and <code>#!th</code> blocks, either for specifying row attributes or better visual grouping ( <i>since 0.12</i> )
<code>#!table</code>	Can optionally be used for wrapping <code>#!tr</code> , <code>#!td</code> and <code>#!th</code> blocks, for specifying table attributes. One current limitation however is that tables cannot be nested. ( <i>since 0.12</i> )
	See <a href="#">WikiHtml</a> for example usage and more details about these processors.
<b>Other Markups</b>	
<code>#!rst</code>	Trac support for Restructured Text. See <a href="#">WikiRestructuredText</a> .
<code>#!textile</code>	Supported if <a href="#">Textile</a> is installed. See <a href="#">a Textile reference</a> .
<b>Code Highlighting Support</b>	
<code>#!c</code> <code>#!cpp</code> (C++) <code>#!python</code> <code>#!perl</code> <code>#!ruby</code> <code>#!php</code> <code>#!asp</code> <code>#!java</code> <code>#!js</code> (Javascript) <code>#!sql</code> <code>#!xml</code> (XML or HTML) <code>#!sh</code> (Bourne/Bash shell) <code>etc.</code>	<p>Trac includes processors to provide inline syntax highlighting for source code in various languages.</p> <p>Trac relies on external software packages for syntax coloring, like <a href="#">Pygments</a>.</p> <p>See <a href="#">TracSyntaxColoring</a> for information about which languages are supported and how to enable support for more languages.</p>
<p>Using the MIME type as processor, it is possible to syntax-highlight the same languages that are supported when browsing source code.</p> 	

For more processor macros developed and/or contributed by users, visit:

- [■ProcessorBazaar](#)
- [■MacroBazaar](#)
- [■Trac Hacks](#) community site

Developing processors is no different from Wiki macros. In fact they work the same way, only the usage syntax differs. See [WikiMacros#DevelopingCustomMacros](#) for more information.

See also: [WikiMacros](#), [WikiHtml](#), [WikiRestructuredText](#), [TracSyntaxColoring](#), [WikiFormatting](#), [TracGuide](#)