

Wiki テキスト内で HTML を使用する

#!html [Wiki プロセッサ](#) の機能によって、Trac では Wiki コンテキスト内での HTML 利用が可能です。

しかし、HTML は well-formed でなければいけないという制約があります。とりわけ、1 つ目の #!html ブロックに開始タグを挿入し、その後普通の wiki テキストを続けて、2 つ目の #!html ブロックに 1 つ目のブロックに挿入した開始タグに対応する終了タグを挿入することはできません。

幸いなことに、<div> や やもっと複雑なスタイルを使用して 任意の Wiki テキストを含んだ表を作成する場合は、強力な代替手段があります：表専用の #!div, #!span, #!table, #!tr, #!td, #!th ブロックを使用することです。

これらの Wiki プロセッサはビルトインされているので、追加で他のパッケージをインストールする必要はありません。

#!html の使い方

テキストブロックが HTML として取り扱われるよう、Wiki エンジンに html プロセッサを使用するという情報を与えます。

Wiki マークアップ	表示
<pre> {{{ #!html <h1 style="text-align: right; color: blue">HTML ■■■■</h1> }}}</pre>	HTML テスト

Trac が HTML コードを表示する前にサニタイズすることに注意して下さい。つまり Javascript のイベントハンドラのような潜在的に危険なコードを使おうとしても、出力から削除されるということです。

0.11 以降では、フィルタリングは Genshi が行いますので、出力は well-formed な HTML の断片でなければなりません。上記イントロダクションで記述したとおり、<div> を開く HTML ブロックと閉じる HTML ブロックの 2 つを使って Wiki テキストを囲むことができなくなります。<div> タグ内部に含まれている Wiki テキストをラップする新しい方法として、#!div Wiki プロセッサを使用する方法があります。

#!div and #!span ブロックの使い方

Wiki マークアップ	表示
<pre> {{{ #!div class="important" style="border: 2pt solid; text-align: center" ■■■ Trac ■■■■ 0.11 ■■■■ '■■■' ■■■■■■ }}}</pre> <pre> {{{ #!div class="wikipage" style="border: 1pt dotted" `wikipage` ■■■■■ (class ■■■■■■■■■■■■■■■■■) }}}</pre> <pre> {{{ #!div class="wikipage compact " style="border: 1pt solid" ■■■ class ■■■ (`compact` ■ `wikipage`) }}}</pre> <pre> {{{ #!div class="compact" style="border: 1pt dotted" `compact` ■■ }}}</pre> <pre> {{{ #!div class="" style="border: 1pt solid" class ■■■■■■■■■■■■■■■■■ (class ■■■■■■■■■■■■■■■■■ //■■■■■//) }}}</pre>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">これは Trac バージョン 0.11 における 唯一 の方法です。</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">wikipage でのみ使用可 (class 属性を指定しないのと同じです)</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">複数の class を指定 (compact と wikipage)</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">compact のみ</div> <div style="border: 1px solid black; padding: 5px;">class 属性の値を指定しない (class 属性そのものを指定しないのとは 異なります)</div>

Note: #!div ブロックの内容に 1 つもしくは複数の段落が含まれる場合、上下にマージンが追加されます。これは、上記の例にある通り、上下のパディングをもたらします。コンテンツの上下のマージンブロックの class 属性に compact を追加して下さい。wikipage と compact の他に予め定義された値は、important です。これを指定すると段落を目立たせることができます。例えば、その他の CSS のクラスは、site/style.css ファイル経由で定義することができます。 [サイトの外観](#) を参照して下さい。

span については、むしろマクロ呼び出しのシンタックスを使用すべきです:

Wiki マークアップ
<pre>Hello [[span(''WORLD'' (click [#anchor here]), style=color: green; font-size: 120%, id=anchor)]]!</pre>
表示
Hello WORLD (click here)!

#!td とその他の表関連のプロセッサの使い方

実際に、それぞれのテーブル行やヘッダのセルを作成するためには、`#!td` や `#!th` がメインのプロセッサになります。その他、`#!table` や `#!tr` プロセッサは `#!td` や `#!th`

プロセッサがこれらのプロセッサの役割を自動的に行ってしまうため、テーブル構造を紹介するためには必要ではありません。行セパレータ `|` を新しい行を始める際に必要に応じて使うことができます。しかし場合によっては、`#!tr` ブロックを使用の方が好ましいかもしれません。`#!tr` ブロックの方がより正式なグルーピングを提供することができ、より多くのインデントレベルを提供できる可能性があります。`#!table` や `#!tr` プロセッサを使用する主な目的は、`style` や `valign` などで HTML の属性を指定できることです。

Wiki マークアップ	表示
-------------	----

表の表示方法を自由に設定することができるようになります。特に、表、行、セルのいずれにも罫線を設定しない場合、いたるところで `style="border: no"` を指定するよりこの方法を使用したほうが、同じ効果を得るのにより効果的な方法となります。

Wiki マークアップ

表示

```

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

0	1	2
10	20	30
11	22	33
	数字	

他のクラスを代替手段として指定することもできます (site/style.css であなた自身のスタイルシートを定義できることを忘れないで下さい)

Wiki マークアップ

表示

```

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

0	1	2
10	20	30
11	22	33
	数字	

HTML コメント

HTML コメントは html プロセッサの出力から除外されます。HTML コメントを wiki ページに追加するために、`htmlcomment` プロセッサを使用して下さい (0.12 以降)。例えば、下記のようなコードブロックになります:

Wiki マークアップ

```

{{{
#!htmlcomment
■■■■■■ HTML ■■■■■■
<tags> ■ &entities; ■■■■■■
}}}
```

表示

```

<!--
■■■■■■ HTML ■■■■■■
<tags> ■ &entities; ■■■■■■
-->
```

Note: "--" という文字の並びは、HTML コメントでは使用できません。レンダリング時にエラーが発生します。

より詳しい情報

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

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