Formatting Syntax

[DokuWiki](#) supports some simple markup language, which tries to make the datafiles to be as readable as possible. This page contains all possible syntax you may use when editing the pages. Simply have a look at the source of this page by pressing the *Edit this page* button at the top or bottom of the page. If you want to try something, just use the [playground](#) page. The simpler markup is easily accessible via [quickbuttons](#), too.

Basic Text Formatting

DokuWiki supports **bold**, *italic*, *underlined* and `monospaced` texts. Of course you can ***combine*** all these.

```
DokuWiki supports **bold**, //italic//, __underlined__ and ''monospaced''
texts.
Of course you can **__//''combine''//__** all these.
```

You can use $_{\text{subscript}}$ and $^{\text{superscript}}$, too.

```
You can use <sub>subscript</sub> and <sup>superscript</sup>, too.
```

You can mark something as ~~deleted~~ as well.

```
You can mark something as <del>deleted</del> as well.
```

Paragraphs are created from blank lines. If you want to **force a newline** without a paragraph, you can use two backslashes followed by a whitespace or the end of line.

This is some text with some linebreaks
Note that the two backslashes are only recognized at the end of a line
or followed by
a whitespace \\this happens without it.

```
This is some text with some linebreaks\ Note that the
two backslashes are only recognized at the end of a line\
or followed by\ a whitespace \this happens without it.
```

You should use forced newlines only if really needed.

Links

DokuWiki supports multiple ways of creating links.

External

External links are recognized automagically: http://www.google.com or simply www.google.com - You can set the link text as well: [This Link points to google](#). Email addresses like this one: andi@splitbrain.org are recognized, too.

```
DokuWiki supports multiple ways of creating links. External links are
recognized
automagically: http://www.google.com or simply www.google.com - You can set
link text as well: [[http://www.google.com|This Link points to google]].
Email
addresses like this one: <andi@splitbrain.org> are recognized, too.
```

Internal

Internal links are created by using square brackets. You can either just give a [pagename](#) or use an additional [link text](#).

```
Internal links are created by using square brackets. You can either just give
a [[pagename]] or use an additional [[pagename|link text]].
```

[Wiki pagenames](#) are converted to lowercase automatically, special characters are not allowed.

You can use [namespaces](#) by using a colon in the pagename.

```
You can use [[some:namespaces]] by using a colon in the pagename.
```

For details about namespaces see [namespaces](#).

Linking to a specific section is possible, too. Just add the section name behind a hash character as known from HTML. This links to [this Section](#).

```
This links to [[справка#internal|this Section]].
```

Notes:

- Links to [existing pages](#) are shown in a different style from [nonexisting](#) ones.
- DokuWiki does not use [CamelCase](#) to automatically create links by default, but this behavior can be enabled in the [config](#) file. Hint: If DokuWiki is a link, then it's enabled.
- When a section's heading is changed, its bookmark changes, too. So don't rely on section linking too much.

Interwiki

DokuWiki supports [Interwiki](#) links. These are quick links to other Wikis. For example this is a link to Wikipedia's page about Wikis: [Wiki](#).

```
DokuWiki supports [[doku>Interwiki]] links. These are quick links to other
Wikis.
For example this is a link to Wikipedia's page about Wikis: [[wp>Wiki]].
```

Windows Shares

Windows shares like [this](#) are recognized, too. Please note that these only make sense in a homogeneous user group like a corporate [Intranet](#).

```
Windows Shares like [[\server\share|this]] are recognized, too.
```

Notes:

- For security reasons direct browsing of windows shares only works in Microsoft Internet Explorer per default (and only in the «local zone»).
- For Mozilla and Firefox it can be enabled through different workaround mentioned in the [Mozilla Knowledge Base](#).

Image Links

You can also use an image to link to another internal or external page by combining the syntax for links and [images](#) (see below) like this:

```
[[http://www.php.net|{{dokuwiki-128.png}}]]
```


Please note: The image formatting is the only formatting syntax accepted in link names.

The whole [image](#) and [link](#) syntax is supported (including image resizing, internal and external images and URLs and interwiki links).

Footnotes

You can add footnotes [1] by using double parentheses.

```
You can add footnotes ((This is a footnote)) by using double parentheses.
```

Sectioning

You can use up to five different levels of headlines to structure your content. If you have more than three headlines, a table of contents is generated automatically – this can be disabled by including the string ~~NOTOC~~ in the document.

Headline Level 3

Headline Level 4

Headline Level 5

```
==== Headline Level 3 ====
=== Headline Level 4 ===
== Headline Level 5 ==
```

By using four or more dashes, you can make a horizontal line:

Images and Other Files

You can include external and internal [images](#) with curly brackets. Optionally you can specify the size of them.

Real size:

Resize to given width:

Resize to given width and height[2]:

Resized external image: ×

```
Real size:                    {{dokuwiki-128.png}}
Resize to given width:        {{dokuwiki-128.png?50}}
Resize to given width and height: {{dokuwiki-128.png?200x50}}
Resized external image:
{{http://de3.php.net/images/php.gif?200x50}}
```

By using left or right whitespaces you can choose the alignment.

```
{{ wiki:dokuwiki-128.png}}
{{dokuwiki-128.png }}
{{ wiki:dokuwiki-128.png }}
```

Of course, you can add a title (displayed as a tooltip by most browsers), too.


```
{{ wiki:dokuwiki-128.png |This is the caption}}
```

If you specify a filename (external or internal) that is not an image (`gif, jpeg, png`), then it will be displayed as a link instead.

For linking an image to another page see [Image Links](#) above.

Lists

Dokuwiki supports ordered and unordered lists. To create a list item, indent your text by two spaces and use a * for unordered lists or a - for ordered ones.

- This is a list
- The second item

- ○ You may have different levels
- Another item

1. The same list but ordered
2. Another item
   1. Just use indention for deeper levels
3. That's it

```
  * This is a list
  * The second item
    * You may have different levels
  * Another item

  - The same list but ordered
  - Another item
    - Just use indention for deeper levels
  - That's it
```

Also take a look at the **FAQ on list items**.

Text Conversions

DokuWiki can convert certain pre-defined characters or strings into images or other text or HTML.

The text to image conversion is mainly done for smileys. And the text to HTML conversion is used for typography replacements, but can be configured to use other HTML as well.

Text to Image Conversions

DokuWiki converts commonly used emoticons to their graphical equivalents. Those Smileys and other images can be configured and extended. Here is an overview of Smileys included in DokuWiki:

- 😎 8-)
- 😲 8-O
- 🙁 :-(
- 🙂 :-)
- 😃 =)
- 😕 :-/
- 😕 :-\
- 😃 :-?
- 😄 :-D
- 😛 :-P
- 😮 :-O
- 😣 :-X
- 😐 :-|
- 😉 ;-)
- 😁 ^_^
- ❓ :?:
- ❗ :!:
- 😆 LOL
- 🔧Fix Me! FIXME

- 🗑 **Delete!** DELETEME

Text to HTML Conversions

Typography: [dokuwiki](#) can convert simple text characters to their typographically correct entities. Here is an example of recognized characters.

→ ← ↔ ⇒ ⇐ ⇔ » « – — 640×480 © ™ ® «He thought 'It's a man's world'...»

```
-> <- <-> => <= <=> >> << -- --- 640x480 (c) (tm) (r)
"He thought 'It's a man's world'..."
```

The same can be done to produce any kind of HTML, it just needs to be added to the [pattern file](#).

There are three exceptions which do not come from that pattern file: multiplication entity (640×480), 'single' and «double quotes». They can be turned off through a [config option](#).

Quoting

Some times you want to mark some text to show it's a reply or comment. You can use the following syntax:

```
I think we should do it

> No we shouldn't

>> Well, I say we should

> Really?

>> Yes!

>>> Then lets do it!
```

I think we should do it

> No we shouldn't

>> Well, I say we should

> Really?

>> Yes!

>>> Then lets do it!

Tables

DokuWiki supports a simple syntax to create tables.

Heading 1	Heading 2	Heading 3
Row 1 Col 1	Row 1 Col 2	Row 1 Col 3
Row 2 Col 1	some colspan (note the double pipe)	
Row 3 Col 1	Row 3 Col 2	Row 3 Col 3

Table rows have to start and end with a | for normal rows or a ^ for headers.

```
^ Heading 1      ^ Heading 2       ^ Heading 3          ^
| Row 1 Col 1    | Row 1 Col 2     | Row 1 Col 3        |
| Row 2 Col 1    | some colspan (note the double pipe) ||
| Row 3 Col 1    | Row 3 Col 2     | Row 3 Col 3        |
```

To connect cells horizontally, just make the next cell completely empty as shown above. Be sure to have always the same amount of cell separators!

Vertical tableheaders are possible, too.

	Heading 1	Heading 2
Heading 3	Row 1 Col 2	Row 1 Col 3
Heading 4	no colspan this time	
Heading 5	Row 2 Col 2	Row 2 Col 3

As you can see, it's the cell separator before a cell which decides about the formatting:

```
|                ^ Heading 1         ^ Heading 2          ^
^ Heading 3      | Row 1 Col 2        | Row 1 Col 3        |
^ Heading 4      | no colspan this time |                  |
^ Heading 5      | Row 2 Col 2        | Row 2 Col 3        |
```

You can have rowspans (vertically connected cells) by adding ::: into the cells below the one to which they should connect.

Heading 1	Heading 2	Heading 3
Row 1 Col 1		Row 1 Col 3
Row 2 Col 1	this cell spans vertically	Row 2 Col 3
Row 3 Col 1		Row 2 Col 3

Apart from the rowspan syntax those cells should not contain anything else.

```
^ Heading 1      ^ Heading 2                 ^ Heading 3          ^
| Row 1 Col 1    | this cell spans vertically | Row 1 Col 3        |
| Row 2 Col 1    | :::                        | Row 2 Col 3        |
| Row 3 Col 1    | :::                        | Row 2 Col 3        |
```

You can align the table contents, too. Just add at least two whitespaces at the opposite end of your text: Add two spaces on the left to align right, two spaces on the right to align left and two spaces at

least at both ends for centered text.

Table with alignment		
right	center	left
left	right	center
xxxxxxxxxxxx	xxxxxxxxxxxx	xxxxxxxxxxxx

This is how it looks in the source:

```
^          Table with alignment          ^^^
|         right|    center    |left           |
|left          |        right|    center      |
| xxxxxxxxxxxx | xxxxxxxxxxxx | xxxxxxxxxxxx |
```

Note: Vertical alignment is not supported.

No Formatting

If you need to display text exactly like it is typed (without any formatting), enclose the area either with tags or even simpler, with double percent signs %%.

This is some text which contains addresses like this: http://www.splitbrain.org and **formatting**, but nothing is done with it. The same is true for //__this__ text// with a smiley ;-).

```
<nowiki>
This is some text which contains addresses like this:
http://www.splitbrain.org and **formatting**, but nothing is done with it.
</nowiki>
The same is true for %%//__this__ text// with a smiley ;-)%%.
```

Code Blocks

You can include code blocks into your documents by either indenting them by at least two spaces (like used for the previous examples) or by using the tags or .

```
This is text is indented by two spaces.
```

```
This is preformatted code all spaces are preserved: like              <-this
```

```
This is pretty much the same, but you could use it to show that you quoted a
file.
```

Those blocks were created by this source:

```
  This is text is indented by two spaces.
```

```
<code>
This is preformatted code all spaces are preserved: like              <-this
</code>
```

```
<file>
This is pretty much the same, but you could use it to show that you quoted a
file.
</file>
```

Syntax Highlighting

[dokuwiki](#) can highlight sourcecode, which makes it easier to read. It uses the [GeSHi](#) Generic Syntax Highlighter – so any language supported by GeSHi is supported. The syntax is the same like in the code and file blocks in the previous section, but this time the name of the used language is inserted inside the tag. Eg. or .

```
/**
 * The HelloWorldApp class implements an application that
 * simply displays "Hello World!" to the standard output.
 */
class HelloWorldApp &#123;
    public static void main&#40;<a
href="http://www.google.com/search?hl=en&q=allinurl%3Adocs.oracle.com+javase+
docs+api+string">String</a>&#91;&#93; args&#41; &#123;
        <a
href="http://www.google.com/search?hl=en&q=allinurl%3Adocs.oracle.com+javase+
docs+api+system">System</a>.out.println&#40;"Hello World!"&#41;; //Display
the string.
    &#125;
&#125;
```

The following language strings are currently recognized: *abap, actionscript-french, actionscript, actionscript3, ada, apache, applescript, asm, asp, autoit, avisynth, bash, basic4gl, bf, bibtex, blitzbasic, bnf, boo, c, c_mac, caddcl, cadlisp, cfdg, cfm, cil, cmake, cobol, cpp, cpp-qt, csharp, css, d, dcs, delphi, diff, div, dos, dot, eiffel, email, erlang, fo, fortran, freebasic, genero, glsl, gml, gnuplot, groovy, gettext, haskell, hq9plus, html, idl, ini, inno, intercal, io, java5, java, javascript, kixtart, klonec, klonecpp, latex, lisp, locobasic, lolcode, lotusformulas, lotusscript, lscript, lsl2, lua, m68k, make, matlab, mirc, modula3, mpasm, mxml, mysql, nsis, oberon2, objc, ocaml-brief, ocaml, oobas, oracle8, oracle11, pascal, perl, per, php-brief, php, pic16, pixelbender, plsql, povray, powershell, progress, prolog, properties, providex, python, qbasic, rails, rebol, reg, robots, ruby, sas, scala, scheme, scilab, sdlbasic, smalltalk, smarty, sql, tcl, teraterm, text, thinbasic, tsql, typoscript, vbnet, vb, verilog, vhdl, vim, visualfoxpro, visualprolog, whitespace, winbatch, whois, xml, xorg_conf, xpp, z80*

Downloadable Code Blocks

When you use the or syntax as above, you might want to make the shown code available for download as well. You can to this by specifying a file name after language code like this:

```
<file php myexample.php>
<?php echo "hello world!"; ?>
</file>
```

myexample.php

```
<?php echo "hello world!"; ?>
```

If you don't want any highlighting but want a downloadable file, specify a dash (-) as the language code: .

Embedding HTML and PHP

You can embed raw HTML or PHP code into your documents by using the or tags. (Use uppercase tags if you need to enclose block level elements.)

HTML example:

```
<html>
This is some <span style="color:red;font-size:150%;">inline HTML</span>
</html>
<HTML>
<p style="border:2px dashed red;">And this is some block HTML</p>
</HTML>
```

This is some <span style="color:red;font-size:150%;">inline HTML