

# 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 „Edit this 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 <sub>subscript</sub> and <sup>superscript</sup>, 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](http://www.google.com) - You can set the link text as well: [This Link points to google](#). Email addresses like this one: [andi@splitbrain.org](mailto: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 [[syntax#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](#). However, there will still be a JavaScript warning about trying to open a Windows Share. To remove this warning (for all users), put the following line in `conf/userscript.js`:

```
LANG.nosmblinks = '';
```

## 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|{{wiki: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 <sup>1)</sup> 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:

# Media Files

You can include external and internal  images, videos and audio files with curly brackets. Optionally you can specify the size of them.



Real size:



Resize to given width:



Resize to given width and height<sup>2)</sup>:

Resized external image: 

```

Real size:                {{wiki:dokuwiki-128.png}}
Resize to given width:    {{wiki:dokuwiki-128.png?50}}
Resize to given width and height: {{wiki: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 }}
{{wiki: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}}
```

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

### Supported Media Formats

DokuWiki can embed the following media formats directly.

Image	gif, jpg, png
Video	webm, ogv, mp4
Audio	ogg, mp3, wav
Flash	swf

If you specify a filename that is not a supported media format, then it will be displayed as a link instead.

## Fallback Formats

Unfortunately not all browsers understand all video and audio formats. To mitigate the problem, you can upload your file in different formats for maximum browser compatibility.

For example consider this embedded mp4 video:

```
{{video.mp4|A funny video}}
```

When you upload a `video.webm` and `video.ogv` next to the referenced `video.mp4`, DokuWiki will automatically add them as alternatives so that one of the three files is understood by your browser.

Additionally DokuWiki supports a „poster“ image which will be shown before the video has started. That image needs to have the same filename as the video and be either a jpg or png file. In the example above a `video.jpg` file would work.

## 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
xxxxxxxxxxxxx	xxxxxxxxxxxxx	xxxxxxxxxxxxx

This is how it looks in the source:

```

^           Table with alignment           ^^^
|           right|           center   |left   |
|left        |           right|           center |
| xxxxxxxxxxxxx | xxxxxxxxxxxxx | xxxxxxxxxxxxx |

```

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 `<nowiki>` 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 `<code>` or `<file>`.

```

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 uses the same code and file blocks described in the previous section, but this time the name of the language syntax to be highlighted is included inside the tag, e.g. `<code java>` or `<file java>`.

```
/**
 * The HelloWorldApp class implements an application that
 * simply displays "Hello World!" to the standard output.
 */
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!"); //Display the string.
    }
}
```

The following language strings are currently recognized: *4cs, 6502acme, 6502kickass, 6502tasm, 68000devpac, abap, actionscript-french, actionscript, actionscript3, ada, algol68, apache, applescript, asm, asp, autoconf, autohotkey, autoit, avisynth, awk, bascomavr, bash, basic4gl, bf, bibtex, blitzbasic, bnf, boo, c, c\_loadrunner, c\_mac, caddcl, cadlisp, cfdg, cfm, chaiscript, cil, clojure, cmake, cobol, coffeescript, cpp, cpp-qt, csharp, css, cuesheet, d, dcs, delphi, diff, div, dos, dot, e, epc, ecmascript, eiffel, email, erlang, euphoria, f1, falcon, fo, fortran, freebasic, fsharp, gambas, genero, genie, gdb, glsl, gml, gnuplot, go, groovy, gettext, gwbasic, haskell, higest, hq9plus, html, html5, icon, idl, ini, inno, intercal, io, j, java5, java, javascript, jquery, kixtart, klonec, klonecpp, latex, lb, lisp, llvm, locobasic, logtalk, lolcode, lotusformulas, lotusscript, lscript, lsl2, lua, m68k, magiksf, make, mapbasic, matlab, mirc, modula2, modula3, mmix, mpasm, mxml, mysql, newlisp, nsis, oberon2, objc, object, ocaml-brief, ocaml, oobas, oracle8, oracle11, oxygene, oz, pascal, pcre, perl, perl6, per, pf, php-brief, php, pike, pic16, pixelbender, pli, plsql, postgresql, povray, powerbuilder, powershell, proftpd, progress, prolog, properties, providex, purebasic, pycon, python, q, qbasic, rails, rebol, reg, robots, rpmspec, rsplus, ruby, sas, scala, scheme, scilab, sdlbasic, smalltalk, smarty, sql, systemverilog, tcl, teraterm, text, thinbasic, tsql, typoscript, unicon, uscript, vala, vbnet, vb, verilog, vhdl, vim, visualfoxpro, visualprolog, whitespace, winbatch, whois, xbasic, xml, xorg\_conf, xpp, yaml, z80, zxbasic*

## Downloadable Code Blocks

When you use the `<code>` or `<file>` syntax as above, you might want to make the shown code available for download as well. You can do 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: `<code - myfile.foo>`.

## Embedding HTML and PHP

You can embed raw HTML or PHP code into your documents by using the `<html>` or `<php>` 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>
```

```
<html> This is some inline HTML </html> <HTML> <p style=„border:2px dashed red;“>And this is
some block HTML</p> </HTML>
```

PHP example:


```
<php>
echo 'The PHP version: ';
echo phpversion();
echo ' (generated inline HTML)';
</php>
<PHP>
echo '<table class="inline"><tr><td>The same, but inside a block level
element:</td>';
echo '<td>'.phpversion().'</td>';
echo '</tr></table>';
</PHP>
```

```
<php> echo 'The PHP version: '; echo phpversion(); echo ' (inline HTML)'; </php> <PHP> echo
'<table class=„inline“><tr><td>The same, but inside a block level element:</td>'; echo
'<td>'.phpversion().'</td>'; echo '</tr></table>'; </PHP>
```

**Please Note:** HTML and PHP embedding is disabled by default in the configuration. If disabled, the code is displayed instead of executed.

## RSS/ATOM Feed Aggregation

[DokuWiki](#) can integrate data from external XML feeds. For parsing the XML feeds, [SimplePie](#) is used. All formats understood by SimplePie can be used in DokuWiki as well. You can influence the rendering by multiple additional space separated parameters:

Parameter	Description
any number	will be used as maximum number items to show, defaults to 8
reverse	display the last items in the feed first
author	show item authors names
date	show item dates
description	show the item description. If  HTML is disabled all tags will be stripped
n[dhm]	refresh period, where d=days, h=hours, m=minutes. (e.g. 12h = 12 hours).

The refresh period defaults to 4 hours. Any value below 10 minutes will be treated as 10 minutes. [DokuWiki](#) will generally try to supply a cached version of a page, obviously this is inappropriate when the page contains dynamic external content. The parameter tells [DokuWiki](#) to re-render the page if it is more than *refresh period* since the page was last rendered.

### Example:

```
{{rss>http://slashdot.org/index.rss 5 author date 1h }}
```

- [High-Severity Vulnerability In Linux Caused By a Single Errant Character](#) von BeauHD (09.06.2026 20:00)
- [EU Says Decision Not to Launch Siri AI in Europe Is Apple's Alone](#) von BeauHD (09.06.2026 19:00)
- [Meta Will Use Your Activity On Other Websites To Personalize Your Feeds](#) von BeauHD (09.06.2026 18:00)
- [Microsoft Hacked To Deliver Malware To Claude and Gemini Users](#) von BeauHD (09.06.2026 17:00)
- [NHS Prescribes Half a Million Copilot Licenses For Its Paperwork Headache](#) von BeauHD (09.06.2026 16:00)

## Control Macros

Some syntax influences how DokuWiki renders a page without creating any output it self. The following control macros are available:

Macro	Description
~~NOTOC~~	If this macro is found on the page, no table of contents will be created
~~NOCACHE~~	DokuWiki caches all output by default. Sometimes this might not be wanted (eg. when the <php> syntax above is used), adding this macro will force DokuWiki to rerender a page on every call

## Syntax Plugins

DokuWiki's syntax can be extended by [Plugins](#). How the installed plugins are used is described on their appropriate description pages. The following syntax plugins are available in this particular DokuWiki installation:

- [Bookcreator Plugin 2024-02-07](#) von [Gerrit Uitslag \(previous: Luigi Micco\)](#)  
Allow to make a book (PDF, ODT, HTML or text) from selected pages
- [Bootstrap Wrapper Plugin 2022-09-22](#) von [Giuseppe Di Terlizzi](#)  
Bootstrap Wrapper for DokuWiki
- [CSV Plugin 2025-06-04](#) von [Andreas Gohr](#)  
Displays a CSV file, or inline CSV data, as a table
- [Dw2Pdf plugin 2026-01-08](#) von [Andreas Gohr and Luigi Micco](#)  
DokuWiki to PDF converter
- [folded plugin 2023-07-22](#) von [Michael Hamann, Fabian van-de-I\\_Isle, Christopher Smith, Esther Brunner](#)  
Foldable page sections
- [Graphviz Plugin 2025-08-27](#) von [Andreas Gohr](#)  
Create graph images from Graphviz dot language
- [Icons Plugin 2024-10-30](#) von [Giuseppe Di Terlizzi](#)  
Embed icons in Dokuwiki
- [Indexmenu Plugin 2024-01-05](#) von [Samuele Tognini, Gerrit Uitslag](#)  
Show a customizable and sortable index for a namespace
- [Info Plugin 2020-06-04](#) von [Andreas Gohr](#)  
Displays information about various DokuWiki internals
- [Math Plugin 2024-05-23](#) von [Christopher Smith](#)  
Add mathematical formulae to dokuwiki. Syntax: `<m size>math formulae</m>`
- [Open Document Plugin 2023-03-03](#) von [Andreas Gohr, Aurelien Bompard, Florian Lamml, LarsDW223](#)  
Export the current Wiki page to a OpenOffice ODT file
- [structgantt plugin 2024-07-17](#) von [Andreas Gohr](#)  
Display struct data as Gantt
- [struct plugin 2025-09-24](#) von [Andreas Gohr, Michael Große, Anna Dabrowska](#)  
Add and query additional structured page data
- [Table Calculations Plugin 09-02-17](#) von [Gryaznov Sergey](#)  
Enables Excel style formulas in table syntax
- [Wrap Plugin 2023-08-13](#) von [Anika Henke](#)  
Universal plugin which combines functionalities of many other plugins. Wrap wiki text inside containers (divs or spans) and give them a class (choose from a variety of preset classes), a width and/or a language with its associated text direction.
- [Yearbox Plugin 2021-03-19](#) von [Michael Große \(formerly: Symon Bent\)](#)  
Builds a complete HTML calendar for the given year, with a link to create a new page for each day of the year. Existing pages will have a tooltip showing a preview of the page content.

## Installierte Plugins

Das Notes Plugin ist installiert.


```
<note>
So sieht eine normal Notiz aus
</note>

<note important>
Das hier ist eine wichtige Notiz ;- )
</note>

<note tip>
Hier ist ein Tipp ...
</note>

<note warning>
Und so kann eine Warnung aussehen
</note>
```

und hier das Ergebnis : `<note> So sieht eine normal Notiz aus </note>`

`<note important> Das hier ist eine wichtige Notiz  </note>`

`<note tip> Hier ist ein Tipp ... </note>`

`<note warning> Und so kann eine Warnung aussehen </note>`

## ToDo Plugin

### Usage and examples

There is an additional instruction page at [http://www.eibler.at/dokuwiki-todo/index\\_en.php](http://www.eibler.at/dokuwiki-todo/index_en.php) in [german](#) and [english](#) which describes the installation, the settings and the DokuWiki [SearchPattern Plugin](#) suggestion.

#### Creating tasks

After installing the plugin, simply enter this information on the page you wish to create the todo list:

```
<todo>Title of todo Action</todo>
```

Each `<todo>` tag will create one checkbox for the action.

To create a checkbox that is checked by default, create your todo like this:

```
<todo #>Title of todo Action</todo>
```

Using bulleted lists (\*) is recommended, as it will allow you to more easily make a 'list' out of your items to do.

### Assign tasks to persons

To assign users use the @ symbol like this:

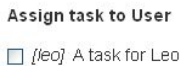
```
<todo @leo>Task for leo</todo>
<todo @leo #>Completed task for leo</todo>
```

The name of the user can use:

- the following characters: a-z, A-Z, 0-9, .-\_
- No (white)spaces are allowed!

So @leo.eibler, @Leo\_Eibler will work but @Leo Eibler will not work!

```
=== Assign task to User ===
<todo @leo>A task for Leo</todo>
```



Multiple users can be assigned in one todo (separated by space character). This way a todo can show up in todo lists of different users.

```
<todo @leo @runout @somebody>Task for leo and runout and somebody</todo>
```

### Render username

Render the username as id, full name or not at all, allowed values: user or real or none

```
<todo @somebody username:real>Task with real username of somebody shown</todo>
```

### Start and Due date

Set a start and/or due date and get colored output if todo has started or is due

```
<todo start:YYYY-MM-DD>Text</todo>
<todo due:YYYY-MM-DD>Text</todo>
```

```
<todo start:YYYY-MM-DD due:YYYY-MM-DD>Text</todo>
```

Colors,... can be set in the style.css file in the plugin directory (lib/plugin/todo)

### Show/Hide Start/Due Date

Do (not) show the Start/Due date, allowed values: yes or no

```
<todo showdate:yes start:YYYY-MM-DD>Text</todo>
```

### Formatting

Formatting is not allowed inside of the `<todo></todo>` tags. However, if you wish for your text to be bold or have some other type of style (other than the default one provided), you can surround the todo tag with the formatting of your choice. The resulting action text should then be displayed with your choice of formatting when your changes have been saved.

### QuickButton

A quickbutton for inserting todo syntax is available in the editor toolbar. You can click the button (which is pictured as a checked-checkbox) or you can use the keyboard shortcut of [T].

## Lightweight Task Management System

The new version (since version 2014-03-17) of the todo plugin allows to create an easy to use task management system to collect all todos from all pages on a single page.

### First turn off caching

```
~~NOCACHE~~
```

Display all open and closed todos from all pages

```
~~TODOLIST~~
```

### Completion state

Filter the completion state, allowed values: yes or no

```
~~TODOLIST completed:no~~  
~~TODOLIST completed:yes~~
```

## Assignments

Filter the assignments, allowed values: yes, no, usernames in format@username or username, or placeholders @@USER@@ or @@MAIL@@ which are replaced by username or e-mail of current logged in user.

```
~~TODOLIST assigned:yes~~  
~~TODOLIST assigned>User1>User2~~  
~~TODOLIST assigned:@User1>@User2~~  
~~TODOLIST assigned:@@USER@@~~  
~~TODOLIST assigned:@@MAIL@@~~ finds tasks assigned using someones email.
```

## Show/Hide checkbox

Do (not) show the checkbox, allowed values: yes or no

```
~~TODOLIST checkbox:no~~
```

## Render username

Render the username as id, full name or not at all, allowed values: user or real or none

```
~~TODOLIST username:real~~
```

## Render header

Render the header as pageID, as the first header of the page or not at all, allowed values: id or firstheader or none

```
~~TODOLIST header:firstheader~~
```

## Show/Hide Start/Due Date

Do (not) show the Start/Due date, allowed values: yes or no

```
~~TODOLIST showdate:yes~~
```

## Completed Data Filter

Filter by date completed.

```
~~TODOLIST completedbefore:<date>~~
```

```
~~TODOLIST completedafter:<date>~~
```

## Start/Due Date Filter

Filter the TODOLIST by Start/Due Date. Absolute and relative Dates are possible. Not yet in the stable Version. Download: [ZIP by Markus](#)

```
~~TODOLIST startbefore:<date>~~
~~TODOLIST startafter:<date>~~
~~TODOLIST duebefore:<date>~~
~~TODOLIST dueafter:<date>~~
```

<date> can be:

```
Y-m-d like 2014-06-12
or a value like +10 or -10 or 0
```

invalid values will be substituted with todays date

## Combined options

```
~~TODOLIST assigned:@@USER@@ completed:yes ~~
~~TODOLIST assigned:Gerrit,leo completed:no ~~
```

Some default values for these options can be set in the Configuration Manager

## Filter the namespace of the task list

I've added the following option to the `~~TODOLIST~~` tag: ns:

If omitted it uses ns:all.

It can be used with the following parameters:

- all - displays todo's from everywhere (the current default behaviour)
- / - only displays todo's from pages containing in the ROOT namespace
- <string> - Where string can be any text. It checks the page-id if it starts with this string and then returns the pages with todo's that do; for example:

```
~~TODOLIST ns:todo~~
```

returns all todo's from the page todo in the root of the wiki, or from the namespaces starting with todo or from the page todo and the namespaces starting with todo.

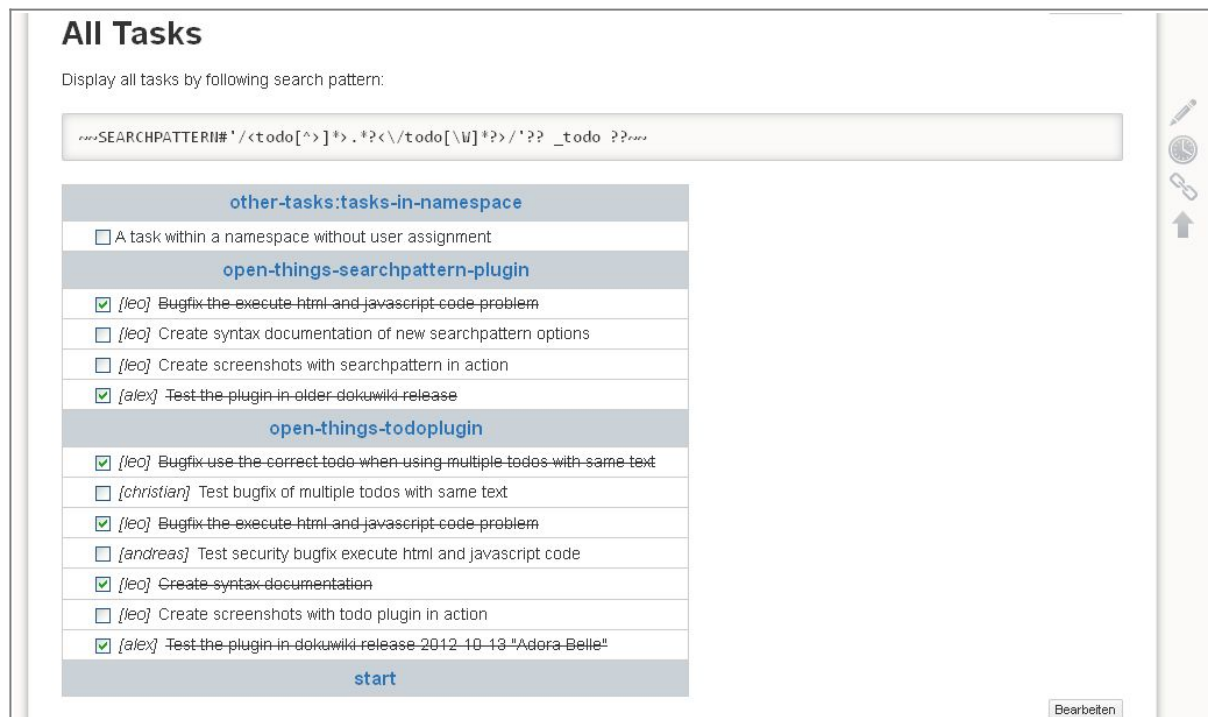
```
~~TODOLIST ns:todo:~~
```

returns all todo's from the namespace todo.

~~TODOLIST ns:todo:r~~

returns all todo's from the namespaces or pages under todo, starting with an r.

Example of 'all todos'



## Plugin Settings

### Allow actions to also link to pages with the same name?

Default: Off

This option (when enabled) will allow your actions to also link to a page with the same name. This way, if you have important details about an action, you can place them in this link.

### What namespace should your actions be created in (".:" = Current NS, Blank = Root NS)

Default: [Empty]

This option allows you specify where actions links should be placed. Using „.:" will cause the actions to link to whatever namespace you are currently in. Leaving it blank or typing „:" will cause all actions to link to the root namespace. Using something like „.:actions:" will cause the actions to link to whatever the current namespace is, plus an „actions" namespace. (Added with version 9-9-09)

### Should the actions have strike-through applied when checked?

Default: On

This option (when enabled) will place a strike-through in the actions that have been completed. When

disabled, an action can be checked, but the text will not have strike-through applied.

### **If AllowLinks is disabled, should clicking the actions' text mark the action complete?**

Default: On

When this option is enabled, and the links option is disabled, then clicking on the text of an action will mark that action as being complete. (Added with version 10-14-09)

1)

This is a footnote

2)

when the aspect ratio of the given width and height doesn't match that of the image, it will be cropped to the new ratio before resizing

From:

<https://dokuwiki.brauns.de/> - **Brauns Control GmbH**

Permanent link:

<https://dokuwiki.brauns.de/doku.php?id=wiki:syntax&rev=1434710897>

Last update: **05.06.2020 12:59**