

Gallery Plugin

<https://www.dokuwiki.org/plugin:gallery>

Syntax

A basic gallery can be added by selecting a [namespace](#) like this:

```
{{gallery>:namespace}}
```

All image files in the selected namespace will be added to the image gallery. Don't forget the ":" in front of the namespace.

Instead of using a whole namespace of images, you can also specify a single image - this makes more sense when combined with the lightbox mode (see below).

```
{{gallery>:namespace:someimage.jpg}}
```

The created gallery can be aligned by using white space (defaults to centered):

```
{{gallery> namespace}} (right aligned)  
{{gallery>namespace }} (left aligned)  
{{gallery> namespace }} (centered)
```

Instead of a namespace, you can also give an HTTP(s) URL to any [Media RSS](#) or ATOM feed with enclosures (as produced by most photo sharing sites like Flickr). The images will then be pulled from that feed instead:

```
{{gallery>http://www.23hq.com/rss/schabloni}}
```

Note: since the question mark is used to separate the parameters (see next section) the URL can not contain any question mark. To use such a feed URL with the gallery plugin, just use one of the many short URL services like <http://bit.ly>.

E.g. instead of

`http://api.flickr.com/services/feeds/photos_public.gne?id=22019303@N00&lang=en-us&format=rss_200` use a shortened URL like `http://bit.ly/HurZM`.

Additionally, to have thumbnail creation correctly working you need to set [fetchsize](#) big enough to get the remote images downloaded.

Parameters

A number of parameters can be set by appending them with ? character to the namespace or image. Each parameter needs to be separated with a & character. Defaults for all parameters can be set in the config manager. If a parameter is enabled by default it can be disabled in the syntax by prefixing

it with the syllable no. E.g. the parameter cache is usually enabled and can be disabled using the keyword nocache. Below is a list of all recognized parameters

Parameter	Default	Description
<number>x<number>	120x120	Sets the size for thumbnails. Unless the crop option is set, this is a boundary box into which the thumbnail will be fitted, maintaining the correct aspect ratio.
<number>X<number>	800X600	Sets the size for the linked images in direct mode. This is a boundary box into which the image will be fitted, maintaining the correct aspect ratio. Note the uppercase X.
<number>	5	The number images per row in the gallery table. If you specify a 0 no table is used instead all thumbnails are added in a sequence.
=<number>	=0	Limits the output to the given number of images. 0 means all.
+<number>	+0	Skip the first number of images. Useful with the option above.
~<number>	~0	Add a pagination for the thumbnails displaying the number of given thumbnails per page. 0 disables pagination. Pagination is added through JavaScript - when no JavaScript is available all thumbnails are displayed
cache	enabled	Usually the output of the created gallery is cached. When the images in your selected namespace change, you have to manually force an update of the gallery page's cache . To disable the cache for the page showing the gallery, set nocache as option.
crop	disabled	Make thumbnails the exact given thumbnail size big, cropping when needed.
direct	disabled	Link thumbnails with the bigger sized images not with their detail page
lightbox	disabled	Show images in a fancy JavaScript modal browsing window, see below for details. Setting this option automatically implies the direct mode
reverse	disabled	Reverse the order of the displayed images
recursive	enabled	Find images in the given namespace and all sub namespaces
random	disabled	Sort images randomly. You might want to use nocache as well
modsort	disabled	Sort images by file modification date
datesort	disabled	Sort images by EXIF date
titlesort	disabled	Sort images by EXIF title
showname	disabled	Show filename below thumbnails
showtitle	disabled	Show the EXIF tag Headline as title below thumbnails
anything containing a *	jpg,gif,png images	This can be used to filter the list of files found in the given namespace. * work as simple wildcard symbol.

Example

```
{{gallery>images:vacation?image_*.jpg&80x80&crop&lightbox}}
```

This displays all images beginning with image_ and ending in .jpg from the namespace

images:vacation. Thumbnails are cropped to 80×80 pixels and images will be opened in lightbox mode.

Example

```
{{gallery>?crop&lightbox}}
```

This displays all images in the current namespace using 2 parameters. Parameterlist begins with ? additional ones are concatenated with &.

About the Lightbox mode

This mode will open the clicked picture inside the current browser window without leaving the current page.

You can close the picture view by clicking the X button in the upper right corner. You can move to the next or previous image by using the arrow buttons in the lower bar. You can also use the keyboard shortcuts listed below for navigation and closing. Mobile users can swipe to navigate and use the back button to close the gallery.

The following keys can be used to navigate:

Key	Action
→	next image
←	previous image
ESC	close the image view

Manipulate EXIF Tags

If you want to show a title below the image using the parameter `showtitle` it may appear that the title shown is not as you expected (e.g. `einstein.jpg` instead of `Albert Einstein`).

If you want to adapt the image's subtitle shown in the gallery, you have to manipulate the EXIF Tag called `Headline`. This can be done with the [fullscreen media manager](#). Alternatives include [ExifTool](#) for which even a [Windows GUI](#) exists to adapt the EXIF data to your needs. On *Windows*, the small picture viewer [IrfanView](#) can also be used to change only IPTC values.

Note that the EXIF-tags used for display can be changed. See [EXIF and IPTC metadata](#) for details. By default, the following tags are used:

- `IPTC:Headline` for the title
- `IPTC:Caption-Abstract` for the description below the picture in lightbox-mode
- `IPTC:By-line` for the photographer name
- `IPTC:CopyrightNotice` for the copyright
- `IPTC:Keywords` for keyword list

Known Limitations and Caveats

Uploading Images

Uploading images is beyond the scope of this plugin. Do not request any features regarding this.

- Use the [media manager popup](#) or [fullscreen media manager](#) to upload multiple image at once with compatible browsers, otherwise one by one.
- Use the [archiveupload](#) plugin to upload multiple images in a Zip file
- Upload the files manually via FTP to the `data/media` directory. Keep in mind that image names need to be valid [pagenames](#), all lowercase, no spaces or special chars!
- Images previously rotated 90° and saved in that orientation no longer appear that way when used in dokuwiki.

Caching

The gallery output is cached by default. When you add pictures later, they may not show up in the gallery: add `&purge=true` to the end of the URL to clear the cache. See [caching](#) for details.

Optionally use the `nocache` parameter of the plugin (not recommended).

EXIF Data Problems

Problems with accessing [EXIF or IPTC](#) data in the images, should be reported as DokuWiki bugs and not for this plugin. All EXIF handling is in DokuWiki core. Currently EXIF Data is expected in UTF-8 encoding.

Lightbox Problems

When the lightbox mode doesn't work and instead images are simply opened in the same window, the JavaScript was not correctly loaded. This is most likely a Browser-Cache issue. Simply follow the steps described for fixing a similar problem with the [toolbar](#). Also make sure you don't have any conflicting plugin installed. You should **not** install any additional lightbox plugin.

Images are not Resized

There are different reasons why thumbnails are not created:

- libGD extension is missing ⇒ Install the extension or configure DokuWiki to use [imagemagick](#)
- libGD extension is installed, but the source image size + overhead is larger than `memory_limit` set in `php.ini` ⇒ Lower the source image size or increase the `memory_limit`

MediaRSS feed displays "nothing found"

When no images from your feed are shown, be sure you don't have a question mark in your URL. Use an URL shortener as suggested above. Also be sure your feed URL (before shortening) starts with `http://` or `https://` and not with `feed://`. The latter is just a renamed HTTP link - simply rename it back.

External Images are not Resized

As written above, you need to increase the `fetchsize` config option. Also make sure LibGD or ImageMagick are installed.

Video Share Plugin

<https://www.dokuwiki.org/plugin:vshare>

Usage/Syntax

The basic syntax looks like this: `{{videosite>videoid?parameter1¶meter2|title}}`

- Where *videosite* is one of the identifiers listed in [Supported Services](#) chapter
- and *videoid* is the identifier of the video at the respective site
- The *parameters* are optional. You start these with a `?` and separate more of them by a `&`. Look in [Parameters](#) chapter
- The title is optional as well. Look in [Examples](#) chapter
- The video can be aligned by adding spaces on the left or right inside the curly brackets (like in the image syntax). Look in [Examples](#) chapter

A toolbar button pops up a prompt where you can simply paste the full URL to the page of the video you want to embed. The plugin will then try to figure out the video ID by itself

Parameters

- The *size* parameter is the best supported parameter. You can either give it in the form:
 - *widthxheight* like `500x300`
 - or use the keywords `small`, `medium` or `large`
- Some other additional parameters are supported (depending on video service) as well:
 - Start point of video: `&start=<seconds>` (YouTube, DailyMotion) or `&initial_time=<seconds>` (TwitchTV)
 - End point of video segment: `&end=<seconds>` (YouTube)
 - Auto play: `&autoplay=1` (Vimeo, YouTube, DailyMotion, Bambuser) or `&ap=1` (MetaCafe)
 - Do not show related videos when playback ends: `&rel=0` (YouTube)
 - Start slide: `&startSlide=5` (SlideShare)

Examples

Display a YouTube Video:

```
{{youtube>L-WM8YxwqEU}}
```

Show a larger player:

```
{{youtube>L-WM8YxwqEU?large}}
```

Right-align the player:

```
{{ youtube>L-WM8YxwqEU}}
```

Show a small, centered player:

```
{{ youtube>L-WM8YxwqEU?small }}
```

Show a small, centered player with a title (look for right space!):

```
{{ youtube>L-WM8YxwqEU?small |Some funny video}}
```

Some other additional parameters are supported (depending on video service) as well:

```
{{youtube>L-WM8YxwqEU?small&start=30&end=45|A random segment of 15 seconds}}
```

Supported Services

Copy paste the video url in the toolbar pop-up prompt to generate the syntax

Identifier	Website	Supported parameters
5min	5min	
archiveorg	Archive.org	
bambuser	Bambuser	autoplay
bliptv	Blip.TV	
break	Break	
clipfish	Clipfish	
dailymotion	Daily Motion	start
gtrailers	GameTrailers	
metacafe	MetaCafe	ap
msoffice	Microsoft Office Training	
mypacetv	MySpaceTV	
rcmovie	RCMovie	
scivee	Scivee.tv	
twitchtv	Twitch.tv	chapter_id, initial_time
slideshare	Slideshare	startSlide

Identifier	Website	Supported parameters
ustream	Ustream.TV	
veoh	Veoh	
viddler	Viddler	offsetTime
vimeo	Vimeo	autoplay
youtube	YouTube	start, end, rel, autoplay

Additional sites can be added to the `sites.conf` file in the plugin directory. Video ID recognition patterns for the toolbar button are configured in `sites.js`

Wrap Plugin

One plugin to rule them all

This plugin gives you the ability to wrap wiki text inside containers (divs or spans) and give them

1. a certain class (with loads of useful preset classes)
2. a width
3. a language with its associated text direction

It potentially replaces a lot of other plugins and is IMHO the better alternative for many.

It fully replaces: [class](#), [clearfloat](#), [div_span_shorthand](#), [divalign2](#), [divalign](#), [emphasis](#), [hide](#), [important_paragraph](#), [importanttext](#), [lang](#), [ltr](#), [noprint](#), [pagebreak](#), [side_note](#), [tip](#), [wpre](#)

It partly replaces: [box](#), [button](#), [color](#), [columns](#), [fontcolor](#), [fontfamily](#), [fontsize2](#), [fontsize](#), [highlight](#), [layout](#), [note](#), [styler](#), [tab](#), [tablewidth](#), [typography](#)

Installation

Search and install the plugin using the [Extension Manager](#). Refer to [Plugins](#) on how to install plugins manually.

Syntax

Basic Syntax:

```
<WRAP classes #id width :language>
"big" content
</WRAP>

**or**
<block classes #id width :language>
```

```
"big" content
</block>

or
<div classes #id width :language>
"big" content
</div>
```

An uppercase **<WRAP>** (or alternatively **<block>** or **<div>**) creates a **div** and should be used for **“big”** containers, **surrounding** paragraphs, lists, tables, etc.

```
<wrap classes #id width :language>"small" content</wrap>

or
<inline classes #id width :language>"small" content</inline>

or
<span classes #id width :language>"small" content</span>
```

A lowercase **<wrap>** (or alternatively **<inline>** or ****) creates a **span** and should be used for **“small”** containers, **inside** paragraphs, lists, tables, etc.

Since version 2013-06-13 there is also a shorthand syntax (for wraps without content):

```
<WRAP classes #id /> or <block classes #id /> or <div classes #id />
```

and

```
<wrap classes #id /> or <inline classes #id /> or <span classes #id />
```

⚠ Please note, some things **won't work with spans: alignments** (including alignments generated by changing the text direction), **multi-columns** and **widths** if the according wrap isn't floated as well.

Examples

The plugin comes with an example page, which should explain a lot and looks like this in the default template (see below).

Classes

The following classes are currently available:



class name	description/notes
columns	- similar to columns , side_note , styler , tip
column	same as left in LTR languages and same as right in RTL languages

class name	description/notes
left	same as <code>column</code> , will let you float your container on the left
right	will let the container float right
center	will position the container in the horizontal center of the page
col2..col5	will show the text in multiple columns determined by their amount (2, 3, 4 or 5), only works in modern browsers (no IE9 and below)
colsmall, colmedium, collarge	will also show the text in multiple columns but determined by their width (small, medium or large), only works in modern browsers (no IE9 and below)
<u>widths</u> - ⚠ experimental, might not work as expected, includes mobile support	
half	fits two columns in a row, should be used in pairs
third	fits three or two columns in a row, should be used in triplets or together with <code>twothirds</code>
twothirds	fits two columns in a row when used together with <code>third</code> , one 1/3 wide and another 2/3 wide
quarter	fits four columns in a row, should be used in quads
<u>alignments</u> - similar to <code>divalign</code>, <code>columns</code>, <code>styler</code> - ⚠ don't work with spans!	
leftalign	aligns text on the left
rightalign	aligns text on the right
centeralign	centers the text
justify	justifies the text
<u>boxes and notes</u> - similar to <code>box</code>, <code>note</code>, <code>tip</code>	
box	creates a box around the container (uses colours from <code>style.ini</code>)
info (was <code>information</code> in first version)	creates a blue box with an info icon
important	creates an orange box with an important icon
alert (⚠ was <code>warning</code> in previous versions)	creates a red box with an alert icon
tip	creates a yellow box with a tip icon
help	creates a violet box with a help icon
todo	creates a cyan box with an todo icon
download	creates a green box with a download icon
round	adds rounded corners to any container with a background colour or a border (only works in modern browsers, i.e. no IE)
danger	creates a red danger safety note
warning	creates an orange warning safety note
caution	creates a yellow caution safety note
notice	creates a blue notice safety note
safety	creates a green safety note
<u>marks</u> - similar to <code>emphasis</code>, <code>important_paragraph</code>, <code>importanttext</code>	
hi	marks text as highlighted
lo	marks text as less significant
em	marks text as especially emphasised
<u>miscellaneous</u>	
clear	similar to <code>clearfloat</code> , should preferably be used with <code>divs</code> , i.e. uppercase <code><WRAP>s</code>
tabs	if wrapped around a list of links, will show those as tabs

class name	description/notes
hide	hides the text per CSS (the text will still appear in the source code, in non-modern browsers and is searchable)
noprint	displays text on the screen, but not in print, similar to noprint
onlyprint	displays text only in print, but not on the screen
pagebreak	forces a new page in printouts (not visible on the screen), similar to pagebreak
nopagebreak	tries to avoid a pagebreak in printouts (not visible on the screen)
spoiler	shows white text on a white background, only to be revealed by highlighting it; similar to hide
button	when wrapped around a link, styles it like a button
tablewidth	sets widths of tables inside to whichever width the wrap gets, partly replaces tablewidth
indent	indents the text, could be used instead of tab
outdent	"outdents" the text, could partly be used instead of outdent
prewrap	wraps text inside pre-formatted code blocks, similar to wpre

Known restrictions

- WRAPs export to ODT format but not everything works 100%
- Round corners only work in modern browsers (no IE8 and below).
- Multiple columns only work in modern browsers (no IE9 and below).
- Width classes are experimental and only work in modern browsers (no IE8 and below).
- Normal DokuWiki Headlines used to not work and a work-around was added. Now that headlines do work, the work-around is not needed anymore but kept for backwards-compatibility. It was deprecated in version 2018-04-22 and disabled by default. They can be enabled by using the `emulatedHeadlines` [config option](#). The following syntax would then produce two different kinds of emulated headlines inside any wrap:
 - `/**_Big Underlined Headline_**/` (They will look a bit different in safety notes.)
 - `/**Small Headline**/`

You might need to adjust a few of the classes to your template's needs, especially `hi`, `lo` and `em`. If you have a dark or otherwise heavily coloured theme, please use the `darkTpl` [config option](#).

The classes are easily adjustable and extensible. Any wishes are welcome.

Widths

You can set any valid widths on any uppercase `<WRAP>` container: `%`, `px`, `em`, `rem`, `ex`, `ch`, `vw`, `vh`, `pt`, `pc`, `cm`, `mm`, `in`. Just set the width before or after or with the classes, e.g.

```
<WRAP someclass 50% anotherclass>...
```

All except percentages will be reduced to have the maximum width available on smaller screens.

You can also use the width keywords `half`, `third`, `twothirds` and `quarter`. To work correctly they need another wrap around them. E.g.

```
<WRAP group>
```

```
<WRAP half column>...</WRAP>
<WRAP half column>...</WRAP>
</WRAP>
```

will result in two columns next to each other, which will wrap underneath each other on smaller screens and mobile devices.

Languages and Text Directions

You can change the language and the direction of a container by simply adding a colon followed by the language code, like this:

```
<wrap :en>This text is explicitly marked as English.</wrap>
```

The text direction (`rtl`, right to left or `ltr`, left to right) will get inserted automatically and is solely dependent on the language. The list of currently supported languages is taken from: http://meta.wikimedia.org/wiki/Template:List_of_language_names_ordered_by_code

If you like to mark a text with a different text direction than the default one, you should use divs, i.e. uppercase `<WRAP>`s. Otherwise the text alignment won't change as well.

This makes it a better replacement of `ltr` (and `lang`).

Demo

You can see a demo of the plugin on demo.selfthinker.org.

“Examples” (demo) in Russian (for v2011-05-15). [Source](#).

Configuration options

Option	Description	Default value
<code>noPrefix</code>	Which (comma separated) class names should be excluded from being prefixed with “wrap_” (* and ? wildcards allowed)	tabs, group
<code>restrictedClasses</code>	Restrict usage of plugin to these (comma separated) classes (* and ? wildcards allowed)	(empty)
<code>restrictionType</code>	Restriction type, specifies if classes above shall be included or excluded	0
<code>syntaxDiv</code>	Which syntax should be used in the toolbar picker for block wraps?	WRAP (other choices: div, block)
<code>syntaxSpan</code>	Which syntax should be used in the toolbar picker for inline wraps?	wrap (other choices: span, inline)
<code>darkTpl</code>	Optimise colours for dark templates?	0
<code>emulatedHeadlines</code>	Use emulated headings? (deprecated)	0

ODT Support

 **Fix Me!** There have been more updates to the Wrap as well as the ODT plugin, so more stuff works. The below should be updated with what works and what doesn't.

Since version 2015-06-13 the Wrap plugin supports exporting most of its functionality/styling to ODT when using at least version 2015-06-29 of the [ODT Plugin](#). By default, Wrap syntax will be exported to ODT using 'print' CSS styles. This means the exported Wrap elements will look the same when printing a wiki page. If you want to have the ODT exported Wrap elements look like displayed in the browser (i.e. with 'screen' CSS styles), then use the following ODT plugin configuration settings:

- add wrap to the 'usestyles' config setting
- set the 'media_sel' setting to 'screen'

If you prefer a user defined CSS style for the Wrap ODT export, then simply place a file 'odt.css' into the Wrap plugin folder with your own CSS code (and set config setting 'media_sel' to 'print').

Here is what is currently **not** supported:

- columns: left/right/center/column is partly supported; they are positioned correctly, but content is not floating around them
- widths are not supported except % and half/third/quarter
- boxes and notes: hardly any formatting inside them is supported, therefore emulated headings also don't work
- tabs will just show a list of links
- noprint
- nopagebreak
- onlyprint only works on boxes
- languages are set correctly but do not seem to affect text alignment
- shorthand syntax
- Not supported because not relevant in ODT: clear, prewrap

Toolbar picker



The wrap picker in the editing toolbar adds the most common wrap syntaxes.

- “columns” creates a set of half columns
- “simple centered box” creates a standard box (60% wide, centered)
- “info, tip, important, alert, help, download, todo box” creates specifically themed boxes (also 60% wide, centered)
- “clear floats” creates a `<WRAP clear/>`
- “especially emphasised, highlighted, less significant” creates the respective marks

Extend with custom styles

If you like to add your own classes and styles to the plugin, you can simply add the styles for your

class preceded by "wrap_" to your [user styles](#). Please note, any classes need to be **lower case**.

E.g. if you need a <WRAP myclass>, you edit (or create if it doesn't exist) your conf/userstyle.css and add your .wrap_myclass{} with its style definitions to it. (If necessary, edit conf/userprint.css¹ for the print view, conf/userrtl.css² for RTL languages and conf/userall.css³ for all styles as well.)

User permissions for every file used must be similar to original DokuWiki files.

Since version 2010-03-14 you have the possibility to exclude certain class names from being prefixed with "wrap_". Just add a comma separated list of class names into the config option "noPrefix" in the configuration manager.

PageList Plugin

<https://www.dokuwiki.org/plugin:pagelist>

Description

The Pagelist Plugin takes a list of wiki pages and provides a nicely formatted table with information about them. The plug-in has a number of flags that can be used to control the information and format of the page list. The user can provide a list of specific page references as can some popular helper plugins such the [Blog](#), [Discussion](#), [Editor](#), [Tag](#), [Task](#) and [Dir](#) plugins.

Syntax

Just wrap a regular unordered list of internal links with the <pagelist> tag. You may provide specific internal page references or have plug-ins supply them as in the example below:

```
<pagelist&[flags]>
* [[:::blog:|Blog Plugin]]
* [[:::discussion:|Discussion Plugin]]
* [[:::editor:|Editor Plugin]]
* [[:::tag:|Tag Plugin]]
* [[:::wrap|Wrap Plugin|This is shown in the description cell]]
</pagelist>
```

[flags] flags can be used to alter the appearance of the pagelist, [flags](#) optional

Flags

Setting	Default	Alternative
style	default table with horizontal lines	table, list or simplelist standard DokuWiki table or list style

Setting	Default		Alternative	
showheader	noheader	hide the heading row of the pagelist table	header	show the header
showdate	date	show the creation or last modification date	nodate	hide the date
showuser	user	show creator or contributors	nouser	hide the user
showdesc	nodesc	hide the description	desc	show the description (from metadata)
showcomments	nocomments	hide the number of comments	comments	show the number of comments (if Discussion Plugin is installed)
showtags	notags	hide the tags	tags	show the tags (if Tag Plugin is installed)
showfirsthl	firsthl	show the first headline	nofirsthl	show the page name
rsort/sort	nosort	no sortation of pages	rsort/sort	sorts the pages (reverse) alphabetically by pagename
showdiff	nodiff	no displaying of differences column	showdiff	displays the differences column with the diff icon linking to the corresponding diff page for each row
image	noimage	show image of the page	image	needs Pageimage Plugin installed. Can either be defined on page or image with same name as page will be used

Example

```
<pagelist&header&comments>
  //an unordered list of pages to display//
</pagelist>
```

In the example above, pagelist will display information about the provided pages in a table with a header line and a comments column (if the [Discussion Plugin](#) is installed). The user (or a plugin) must supply the specific pages to display in the list.

Configuration

The plugin can be configured using the DokuWiki configuration manager available in the admin menu. The settings also apply to plugins which use the helper component of the pagelist plugin, like for example the archive component of the [blog](#) plugin.

style	List style (default, table, table/list, simplelist)
showheader	Show table header
showdate	Shows/hides the date column (hide, creation date, modification date)
showuser	Shows/hides the user column (hide, creator, contributors)

showdesc	Shows/hides a short description taken from the first paragraph of a page (hide, max. 160 characters, max. 500 characters)
showcomments	Shows/hides comments of a page (requires the discussion plugin)
showlinkbacks	Shows/hides linkbacks of a page (requires the linkback plugin)
showtags	Shows/hides tags of a page (requires the tag plugin)
sort	Sorts the pages alphabetically by pagename
showdiff	Displays a differences column with the diff icon linking to the corresponding diff page for each row
showimage	Shows/hides the image column (requires the Pageimage Plugin)

Helper Plugin

You can easily use the functionality of the Pagelist Plugin in your own plugins. Here is a basic code snippet:

```
$pages = array(
    array('id' => 'wiki:dokuwiki'),
    array('id' => 'wiki:syntax'),
);
$pagelist =& plugin_load('helper', 'pagelist');
if (!$pagelist) return false; // failed to load plugin
$pagelist->startList();
foreach ($pages as $page){
    $pagelist->addPage($page);
}
$renderer->doc .= $pagelist->finishList();
```

Since release 2017-08-24 the function `startList()` has got an optional parameter to specify a CSS class for adding it to the class of the table element:

```
$pagelist->startList('myclass');
```

newpagetemplate plugin

<https://www.dokuwiki.org/plugin:newpagetemplate>

Description

This plugin loads into the edit window a template specified in the `$_REQUEST` "newpagetemplate" parameter. Effectively, this means that you can invoke a template, similar to the namespace template pages, either in the URL or in a POST form coded to create a new page.

The [openas](#) plugin and the [addnewpage](#) plugin provide techniques which support the newpagetemplate plugin. The [openas](#) plugin includes a simple method of creating a form designed for use with the newpagetemplate plugin, making it possible for each user to assign unique values

to the template's variables.

Usage

If a new page is created with the URL:

```
doku.php?id=:mynewpage&do=edit&rev=&newpagetemplate=:pagetemplates:yourtemplate&newpagevars=@HI@,Howdy!
```

then the `:pagetemplates:yourtemplate` page is pasted into the edit window. If the "standardreplace" configuration option is true, then the substitutions noted below are performed. If the "userreplace" configuration option is true, then you can also replace `@HI@` with `Howdy!`, as this substitution was passed in `newpagevars`. The format for `newpagevars` is:

```
newpagevars=key,value;key2,value2;key3,value3;
```

See [configuration_options](#) and [substitutions](#) for a description of the standard and user replacements.

Example of syntax for a template page

If you create a template page `yourtemplate` with this markup in `:pagetemplates`:

```
This page is: @ID@  
@HI@ @NAME@
```

And if you paste the above URL into your browser's location bar, you will get a page in your browser that looks something like this:

```
This page is: mynewpage  
  
Howdy! Joe
```

`@ID@` is one of the 'standardreplace' macros and `@HI@` is a 'userreplace' macro.

How to create a link using wiki syntax

Is it possible to use standard wiki-link syntax for this? For eg. like:

```
[[:mynewpage?do=edit&rev=&newpagetemplate=:pagetemplates:homepagetemplate&newpagevars=@HI@,Howdy!|New page with template]]
```

Yes, you can do this. But you will have to substitute some url-encoded characters. In particular, the comma must be represented as `%2c`. So this would give you:

```
&newpagevars=@HI@%2CHowdy
```

You may find with use that there are others.

Substitutions

As with the `_template.txt` [namespace templates](#), the following variables are replaced in your template.

@ID@	full ID of the page
@NS@	namespace of the page
@PAGE@	page name (ID without namespace and underscores replaced by spaces)
@!PAGE@	same as above but with the first character uppercased
@!!PAGE@	same as above but with the first character of all words uppercased
@!PAGE!@	same as above but with all characters uppercased
@FILE@	page name (ID without namespace, underscores kept as is)
@!FILE@	same as above but with the first character uppercased
@!FILE!@	same as above but with all characters uppercased
@USER@	ID of user who is creating the page
@NAME@	name of user who is creating the page
@MAIL@	mail address of user who is creating the page
@DATE@	date and time when edit session started
@DATE@	date and time when edit session started
%a %d-%m-%y etc.	e.g. Thu 06-12-12. Strftime placeholders are replaced by page creation time

Configuration Options

userreplace	Replace user defined macros as explained above in usage
standardreplace	Replace standard macros as listed above in substitutions
prettytitles	Replace underscores with spaces in the following standard macros @!PAGE@, @!!PAGE@, @!PAGE!@
<code>skip_unset_macros</code>	Remove macros which have no substitution values, so that they don't appear in the newly created page.

The prettytitles patch was contributed by Matthias Bannach.

- 1) [conf/printstyle.css](#) in Anteater
- 2) [conf/rtlstyle.css](#) in Anteater
- 3) [conf/allstyle.css](#) in Anteater

From: <https://www.ff6hacking.com/wiki/> - **ff6hacking.com** wiki

Permanent link: <https://www.ff6hacking.com/wiki/doku.php?id=wiki:explugins>

Last update: **2019/02/12 09:52**



