

Wiki Tutorial and Guidelines

Namespaces

Please get acquainted with the concepts behind namespaces in DokuWiki:

<https://www.dokuwiki.org/namespaces>

Plain Articles

When you are creating just a *single* page then it is OK to use one of our existing namespaces:

- [howtos](#):
The logical place for all your tutorials
- [security](#):
Documentation about how to deal with security issues on Linux (not particularly just Slackware); HOWTO's about securing your Linux computer and applications; how to safely surf the Internet; etcetera.
- [slackware](#):
Articles directly related to the distro
- [wiki:user](#):
Where every Wiki user can dress up his own "home page"
- Or any of the localized namespaces ([nl](#): [de](#): [fr](#): etcetera) which in turn contain the same sub-namespaces as mentioned above

Complex Articles

If you add new tutorials etc which are split into *multiple articles, contain screenshots etc*, it is advisable that you put these in their own sub-namespace. Otherwise you will flood the current namespace and maintaining uploaded files will become a mess for the users and admins alike.

When you add a link inside an existing page which points to your new page in its own sub-namespace, you have to add your own namespace in the link too.

You do this by using [W colons](#), and if you start your "namespacing" with a dot '.' you make the namespace relative to the current pages namespace, which is standard.



Imagine that you are editing the main HOWTOs page and want to add your own HOWTO called "icecast streams" and you want to use multiple connected pages with screenshots. You would normally add a link to it in the [HOWTOS main page](#) by using the pagename ".icecast_streams:start".

In this Wiki, the [HOWTOs index](#) page is automatically populated by using the tag "howtos" so use your tags wisely.

Such a name (inside a normal internal link) would put the page under the `howtos:icecast_streams:` namespace. You can then put all your uploaded files in that same namespace.



Note that I replaced the space in “icecast streams” with an underscore. Prevent spaces in the pagenames and namespaces!

Please try to maintain a clean namespace structure, it will be very hard to fix all the links if we have to make corrections to page locations and namespaces.

Syntax and Formatting

First make yourself familiar with the [DokuWiki syntax](#). It is not the same as MediaWiki syntax.

Try to keep the page formatting as clean as possible:

- A page's main title heading should be in **H1**, the headings under that **H2**, headings under **H2** should be **H3** etcetera.
- Don't use horizontal rulers unless necessary (for instance they are not necessary if there is a heading right above or beneath where you want to mark a “new section”).
- Surround text strings containing example commands, filenames and computer text with a “code text” markup. For example: `man rc.inet1.conf`. There is a “TT” button in the button bar above which will insert exactly that markup.

Here is an example of what a small page could look like, using namespaces for sub pages etc:

```
===== This is the page =====
```

```
You would write a description of your article here.
Remember that you can use sub-pages (sub-namespaces), where appropriate.
```

```
===== The First Section =====
```

```
Some bla bla
```

```
[[.subpage:start | My in depth article]]
```

```
===== The Second Section =====
```

```
More text
```

```
[[.another_subpage:start | More information with screenshots]]
```

```
<!-- You can leave comments in your page - they will not be visible in the
resulting Wiki article -->
```

```
<!-- Tags are essential for creating a Table Of Contents (TOC) -->
{{tag>howtos software}}
```

Images

You might want to include rather big screenshots within your articles, but having them displayed full

size directly inside the article can make things look bad. Fortunately you can resize the image directly within the article. You can also set the alignment of the picture, some examples will show more than words:

Resize to given width:

```
{{wiki:dokuwiki-128.png?50}}
```

And left/center/right alignment like this:

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

See [DokuWiki information about images and other files](#) for more info.

Code

You can insert [code](#) using the code tag, like this:

```
<code c> float a; </code>
```

```
float a;  
float b;  
a = b * 10 + 2 * sin1(x);
```

Like this example shows, the readability is enhanced a lot if you apply code [syntax highlighting](#) when documenting shell scripts and such.

Spell Check

Please spell check your work, either with UK or USA English; many visitors do not have English as their primary language. Grammar checks in particular are hard for non-english authors, so do not hesitate to [contact one of our site Editors](#) to assist you in creating legible and fluent english texts.

Text Template

You will notice that whenever you create a new page, you will already have some text pre-filled. This text comes from one of our templates- every namespace has a slightly different template text. The most important aspect of the template is that it adds default [tags](#) to your page. These tags will allow us to populate our Table Of Contents automatically.

An example of a template text:

```
<!-- Add your text below. We strongly advise to start with a Headline (see
```

button bar above). -->

```

<!-- Please do not add anything below, except additional tags.-->
<!-- However we request that you remove the tag-word "template" below.
Otherwise your page will not show up in the Table of Contents -->

<!-- Do not remove this line and the tag definition below. Thanks!
slackdocs@-->
{{tag>howtos template}}

```

We advise to add a tag '*author_yourname*' with of course 'yourname' being replaced by your wiki username. The use of this tag facilitates automated generation of a page list for which you are the author (or one of the authors).

As an example, embedding '`{{topic>author_alienbob nodate&desc&sort&table&tags}}`' in a page will automatically generate an overview of articles written by AlienBOB.

Use of Discussion Pages

It is a good idea to use the “discussion” pages (linked as a tab at the top of each article) to discuss the content of the article, point out errors, clarify doubts, bring something to the attention of the author, and/or propose any major changes.

If you want to add a comment to a discussion page, take a few moments to think about what you want to write *before* clicking the **edit this page** tab. Try to keep the time during which you keep the page locked as short as possible, so that you are not an obstacle to other people who also want to comment.

When you decide to add a comment to a discussion page, there are some guidelines you should follow in order to keep discussions legible and orderly:

- Sign all your comments by clicking the **insert signature** button on the top WYSIWYG editor toolbar or by using `[ALT]+[SHIFT]+[Y]` keyboard shortcut
- Use a **Level 2 Headline** to add a completely new comment by clicking the blue **H** on the top WYSIWYG editor toolbar or by using the `[ALT]+[SHIFT]+[2]` keyboard shortcut
- Add all comments at the bottom of the page, unless you are replying. In that case, add your comments directly underneath the post you are replying to. Use the > characters to thread your discussions, adding as many as needed. For example:

```

===== Topic =====

Original Comment
> The first reply
>> The second reply
>>>The third reply

```

Will look somewhat like this:

FAQ sections

It may be a good idea to split the [FAQ](#) into sections once there are more topics to cover. — *Harishankar* 20

There seems to be quite a few topics covered in this [FAQ](#) now. These are the logical sections that come to

- Getting Slackware (questions 1 - 6)
- Installation & Support (questions 7 - 12)
- Slackware Philosophy (questions 13 - 17)

Does anyone have any suggestions? — *Gerardo Zamudio* 2012/08/26 14:54

Good idea. I'll add sections. If anybody wants to tweak them, feel free. — *Harishankar* 2012/08/26 19:09

I went ahead and moved the headers up one level so they're a bit bigger. [FAQ](#) sections are still prominent sections. The table of contents looks pretty organized this way, too. Any thoughts? — *Gerardo Zamudio*

Headings serve a purpose for document structure. I suggest we keep the correct heading levels viz. h1 f and so on. Edited to add: Your heading levels look fine to me. — *Harishankar* 2012/08/27 01:44

You can subscribe to the talk page's changes if you want to be notified of replies by clicking on the **manage subscriptions** tab while on the talk page.

Language Considerations

In English, Please

English is the primary language of the Slackware Documentation Project. This seems only natural since Slackware Linux has an English-only installer and English-only documentation. The admin team wants to prevent this project from getting stuck in [a Babylonian mess](#).

So How Are non-English Contributions Being Handled?

Please check [Creating a Page in Your Own Language](#) in the Translation Guidelines.

[slackdocs](#), [tutorial](#), [needs attention](#)

From: <https://docs.slackware.com/> - **SlackDocs**

Permanent link: <https://docs.slackware.com/slackdocs:tutorial>

Last update: **2023/11/30 18:41 (UTC)**

