

## Contents

1. Category:Training .....	2
2. Manual:Category .....	2
3. Manual:Namespace .....	6
4. Manual:Quality management .....	10
5. Manual:Subpage .....	18
6. Manual:Template .....	22
7. UserManual:Category .....	24
8. UserManual:Namespace .....	28
9. UserManual:Quality management .....	32
10. UserManual:Subpage .....	40
11. UserManual:Template .....	44
12. UserManual:Wikitext .....	46
13. Wikitext .....	50

## Category:Training

### Pages in category "Training"

---

The following 12 pages are in this category, out of 12 total.

#### C

- [Manual:Category](#)
- [UserManual:Category](#)

#### N

- [Manual:Namespace](#)
- [UserManual:Namespace](#)

#### Q

- [Manual:Quality management](#)
- [UserManual:Quality management](#)

#### S

- [Manual:Subpage](#)
- [UserManual:Subpage](#)

#### T

- [Manual:Template](#)
- [UserManual:Template](#)

#### W

- [Wikitext](#)
- [UserManual:Wikitext](#)

## Categories

A [quality version](#) of this page, [approved](#) on *27 April 2021*, was based off this revision.

You can assign several categories to a page. Categories can be organized as top categories with subcategories. Therefore, it is possible to create a tree structure and thus to build a hierarchical structure.

## Contents

1 What are categories? .....	4
2 Characteristics .....	4
3 Create and assign categories .....	4
3.1 In the title section of the page .....	4
3.2 With Visual Editor .....	5
3.3 In Source code .....	5
4 Links to category pages .....	5
5 Displays of categories .....	5
6 Category manager .....	5
7 Related info .....	6

## What are categories?

---

Categories work like keywords or tags. Adding categories to a page makes it easier to find the page in the wiki. All pages of the same category are automatically shown on the corresponding category page. For each category that has not been assigned before, the wiki will automatically create a redlink. Once the page has been created by clicking on the redlink, all pages and subcategories are automatically listed on the category page.

Categories are usually created by users. However, in special cases, MediaWiki also assigns auto-generated categories (for example, pages with broken links).

## Characteristics

---

- **Tagging:** Categories tag pages.
- **Multiple categories per page:** A page can be assigned to several categories.
- **Main categories:** Categories can be organized under main categories.
- **Independent of namespaces:** Categories are not contained in namespaces, but are shared wiki-wide.

## Create and assign categories

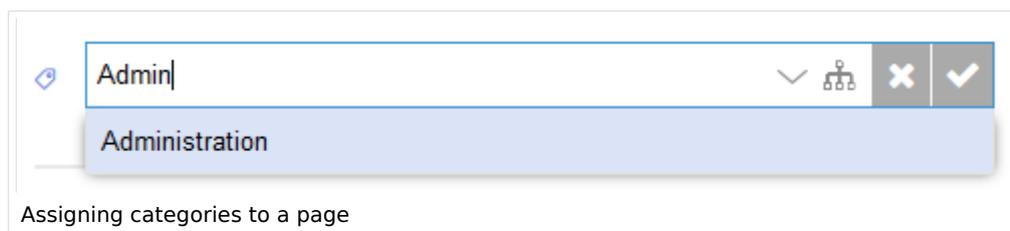
---

To create a category or assign a page to an existing category, there are multiple options:

1. In the title area (view mode)
2. Via the options menu in the visual editor (visual editing mode)
3. With Wikitext (source editing mode)
4. In the "Page Information" fyout of the quality management tools

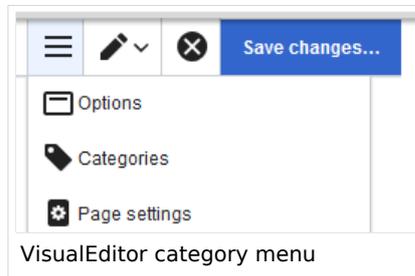
### In the title section of the page

The [title section](#) contains the category assignment field. Any user with edit rights can click on the edit link to add or remove categories from the page. To view all existing categories with subcategories, you can click on the tree icon. Once you are done, the categories are confirmed with the check mark.



## With Visual Editor

In the page options menu of [VisualEditor](#), you can access a Categories menu item. Here you can add and remove category assignments from the page and also add a sortkey to the category.



## In Source code

To add a category in source code, the namespace Category is added before the page name.

```
[[Category:Sales]]
```

**Note:** On category pages, umlauts are generally sorted all the way to the back, e.g., "Märchen" after "Muh". You can therefore use the variable `{{DEFAULTSORT:sortkey}}` to specify an alternative title to be used for sorting (e.g., `{{DEFAULTSORT:Maerchen}}`).

## Links to category pages

---

Links to a category require a colon **before** the namespace.

Example: `[[[:category:new_page]]]`

If this colon is missing, the article will not be linked to the category page, but the article will be assigned to the corresponding category. `[[category:new_page]]`

## Displays of categories

---

You can view a list of all wiki categories on the page [Special:Categories](#).

On the page *Category:category\_name*, you will find a list of all the pages and media associated with a particular category.

If you have accessed a previously unavailable (redlinked) category page, you can add a category description to the page. That way, users understand the purpose of the category.

## Category manager

---

The easiest way to manage categories and sub-categories is with the Category manager page (`Special:Category_manager`). You can navigate to the [Category manager](#) via Global actions.

## Related info

---

- [Category Manager](#)
- <https://www.mediawiki.org/wiki/Help:Categories>

## Manual:Namespace

A [quality version](#) of this page, [approved](#) on *5 March 2021*, was based off this revision.

### Contents

1	<a href="#">What is a namespace?</a>	7
2	<a href="#">Characteristics of a namespace</a>	7
3	<a href="#">Reasons for using namespaces</a>	7
4	<a href="#">BlueSpice standard namespaces</a>	8
5	<a href="#">Accessing pages in a namespace</a>	9
6	<a href="#">Creating a new namespace</a>	9
7	<a href="#">Organizing content without namespaces</a>	10
8	<a href="#">Viewing all pages of a namespace</a>	10
9	<a href="#">Related info</a>	10

## What is a namespace?

---

A **namespace** defines an area in a wiki. You can recognize a namespace by the namespace prefix in the page title:

```
<namespace>:Page title
```

In a wiki, you will find predefined namespaces. In addition, wiki administrators can create custom namespaces as needed.

There are certain aspects of namespaces that are important to consider before creating a namespace. It is therefore important to understand the characteristics of namespaces and the reasons for using a custom namespace.

## Characteristics of a namespace

---

- **Rights management** : Each namespace can contain its own user rights/roles assignments. However, we recommend flat user hierarchies for wikis.
- **Search function** : Search can be limited to individual namespaces.
- **Design** : Namespaces can be styled differently.
- **Flat structure** :Namespaces cannot contain other namespaces
- **One namespace per page** : The same page name in a different namespace creates a new page.
- **Special namespaces:**
  - Main (Pages) - No namespace prefix in page titles
  - Special pages (Special)
  - Images (File)
  - Documents (Media)
  - Discussions (Talk)

## Reasons for using namespaces

---

- **Rights management:** To assign different user permissions to separate content. For example, a company's HR department can use its own namespace to grant only Human Resources personnel access to the pages in that namespace. Here, rights management is the main reason for creating namespaces. However, flat hierarchies are recommended for wikis.
- **Search function:** To exclude pages and files from the [search](#).
- **Design:** To format pages in different wiki areas differently.

We often use the example of cupboards to make clear the principle of namespaces. There are shelves and everyone can look at them and work on something. There are glass cupboards with a lock that you can look into, but only some people have the key to change the contents. Then there are closed wooden cabinets with locks and only selected people can look inside and work in them. And each of these cupboards can look differently.

**Note:** While users cannot view the pages or files in a namespace for which they have no access rights, they will still see the pages and files listed on the wiki's special pages. They will also see the page names in the search results if the namespace has not been excluded from the search.

## BlueSpice standard namespaces

BlueSpice uses, among others, the 15 standard and the two so-called pseudo namespaces from MediaWiki:

ID	Namespace	Function
	Media	Pseudo-namespace for uploaded images and files. Will be replaced by path of the files.
	special	Pseudo-namespace for special pages.
0	(pages)	These are given without a prefix.
1	Talk	talk page. Is created in addition to each article (also in the following namespaces) to speak about its contents. Additional tab (+).
2	User	User page. Will be created for every logged in user.
3	User talk	Cf. talk
4	[Wiki-Name]	Information about the wiki or the current project. Here: BlueSpice.
5	[Wiki-Name] _Talk	Cf. talk
6	Image	Description of a picture and other files. Is automatically supplemented with version information and information about the occurrence of the file
7	File_Talk	Cf. talk
8	MediaWiki	system messages. A complete list can be found under Special: AllMessages. Can only be changed by registered users.
9	MediaWiki_Talk	Cf. talk
10	Template	Templates can be integrated into other pages.
11	Template_talk	Cf. talk
12	Help	Help pages describe how to perform actions in the wiki. No clear separation from the [wiki name] namespace.

ID	Namespace	Function
13	Help_talk	Cf. talk
14	Category	Pages can be assigned to categories, which are then accessible in this namespace.
15	Category_talk	Cf. talk

**Standard namespace:** A page is assigned to the "Main" namespace, as long as no other namespace is specified.

**Pseudo namespaces:** The two pseudo namespaces (Media, Special) are used like normal namespaces, but you cannot edit them or add pages to them.

Most pages in the namespaces are created automatically:

- Special pages are generated by the system during installation.
- Media and image pages are created during file upload.
- User pages are created whenever a user acts on the wiki.
- Discussion pages are also created with the corresponding pages.

This leaves only the default namespaces (*Main*), *[wiki name]*, *Template*, *Help* and *Category* namespaces where pages can be freely created.

**Note:** In the default MediaWiki, the names of these areas are "hard-coded". This means that they have been set in the source code of the MediaWiki and cannot be changed by users.

## Accessing pages in a namespace

---

To access or create a page in a specific namespace, its name is simply placed in front of the page title and separated by a colon:

```
[[Setup:Installation|]]
```

In the example, the pipe symbol (straight line) was also added. It has the effect that the namespace won't be displayed for the link label.

If you provide a namespace that is not known to the wiki, the page is created in the standard namespace Main. The portion of the reference specified as the namespace is then added to the title of the item instead and is not recognized as a namespace label.

## Creating a new namespace

---

You can create additional namespaces by using the Namespace manager under Global actions > Management > [Namespace manager](#). The menu item "Category manager" loads the page `Special:NamespaceManager`.

## Organizing content without namespaces

---

If you simply want to group pages in your wiki without the need for controlling access rights, you can use the syntax of the namespaces to **simulate a namespace in the namespace Main**. Simply prefix the pages that belong to a group with a name and a colon. If no namespace with this name exists, the prefix simply becomes part of the page name.

That way, it is possible to use pages with the same name in the main namespace and thus let different groups work on content "sets". Since these own "namespace" prefixes are part of the page name, you can now display all pages in a simulated namespace by searching for the prefix in the search field.

Since you can also organize pages with categories and subpages, you should start by planning how you would like to organize the content in your wiki.

After installing your wiki, you can visualize this with the BlueSpice draw.io Plugin directly in your wiki and discuss it with your team.

## Viewing all pages of a namespace

---

You can view all pages in a namespace:

1. Go to the Special page "All Pages"
2. Leave the input field empty
3. Select a namespace from the Namespace drop-down menu
4. Click "Go"

## Related info

---

- [Namespace CSS manager](#)
- [Namespace manager](#)

## Manual:Quality management

A [quality version](#) of this page, [approved](#) on *27 April 2021*, was based off this revision.

### Contents

<a href="#">1 The importance of content quality</a>	12
<a href="#">2 Page information</a>	12
<a href="#">3 Page assignments</a>	13
<a href="#">3.1 Characteristics</a>	13

4	Reminder	13
4.1	Characteristics	13
5	Expiry	13
5.1	Characteristics	13
6	Review	14
6.1	Characteristics of the review feature	14
7	Acceptance	14
7.1	Characteristics of the acceptance feature	14
8	Permissions overview	15
9	Related info	18

## The importance of content quality

---

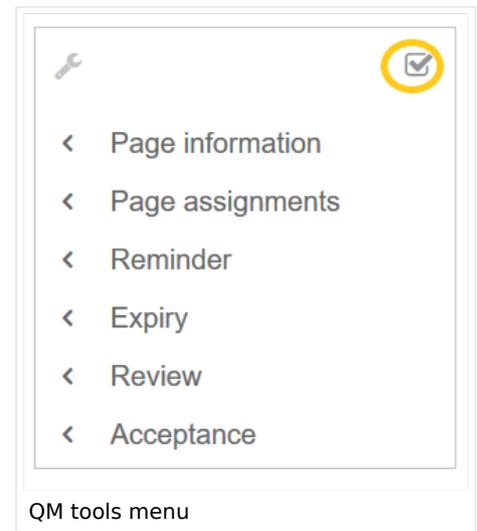
For many companies, the quality of the information they produce and share is of great importance. At the same time, they want to capture all the knowledge of their employees without creating any unnecessary barriers.

In BlueSpice, many users can contribute their knowledge. Ideally, anyone can contribute information right away and specialists can then review and approve the content. This ensures that the quality of the content meets the standards of the company. You should always weigh the benefits and drawbacks of individual quality management features so that you don't create unnecessary roadblocks for users who are eager to contribute.

To access these features, you open the QM tools (with the checkbox icon) in the Page tools panel.

For quality assurance purposes, each content page has the following built-in tools:

- Page information
- Page assignments
- Reminder (BlueSpice pro)
- Expiry (BlueSpice pro)
- Review (BlueSpice pro)
- Acceptance (BlueSpice pro - if FlaggedRevs is activated for the namespace)



## Page information

---

The page information gives you some very interesting statistics about the page. Based on this information, you can decide what quality management measures are necessary to improve the page.

The information that is listed on this page is:

- time of last edit and last editor
- number of views
- templates used on this page
- similar pages
- categories
- readers of this page
- authors of this page

## Page assignments

---

Page assignments are a quick way of giving one or more users informal responsibility and ownership of a page. They are very similar to a [watchlist](#). A user can add a page to their watchlist themselves, but assignments are made by other users.

Page assignments do not change the behavior of the page or grant special rights on a page. For example, the page does not get locked for editing when it is assigned to another user. It is rather a simple mechanism to have users "take a look" at a page. If the feature *Read confirmation* is activated for a namespace, assigned users have to confirm with a checkbox on the page that they have read the page.

### Characteristics

- **Notifications** are sent to assigned users when their page assignment is changed. (This can be deactivated in the user preferences.)
- **Transparency:** Assigned users are listed in the QM tools panel of the page
- **Assignment management:** ([Special:ManagePageAssignments](#))
- **Quick access** from the user profile menu
- **Read confirmation:** page shows a read confirmation checkbox for assigned users (optional)

[More about page assignments](#)

## Reminder

---

A user can create a reminder for a page by selecting a reminder date and by adding an optional comment. Reminders can also be created for other users by administrators and are a quick and informal way of setting due dates for page-related tasks.

### Characteristics

- The user with the reminder receives a notification. (This can be deactivated in the user preferences.)
- Users with reminders are listed on the reminders for the page in the QM tools.
- The page is listed on the Reminders special page under "Global actions" ([Special:Reminder](#)).
- The page is listed on the user's Reminders page, which is easy to access from the user profile menu.

## Expiry

---

If a page contains content with a known lifecycle, the page can be set to expire on a particular date. On the expiration date, a watermark "Outdated" is added to the page if the corresponding option is set in the [ConfigManager](#). In addition, a reminder can be set so that the user will receive a notification when the page expires. The expiration can be deleted from the page at any time.

### Characteristics

- **Watermark** on page background and in PDF (can be configured)
- **Management list:** *Special:Expiry*
- **Reminder** possible
- **Duration:** Expiry can be removed at any time

## Review

---

When a more formal review process is necessary, users can create a review workflow. The review workflow has an end date and is the only quality management tool that can lock the page for editing during the review phase. Each user casts a vote whether the page is acceptable or not.

The page is listed on the special page "Review" under "Global actions" ( `Special:Review` ) and on the reviewers' personal Review page.

### Characteristics of the review feature

- **One active review process** per page
- **Types:** Sequential or parallel review
- **Acceptance or rejection** by participants
- **Page is locked** for editing (Exception: depending on the configuration, reviewers can get edit rights)
- **Automatic restart** possible
- **Revisions-based:** Entries in the version history
- **Acceptance:** can be combined with the Acceptance feature

## Acceptance

---

If the pages in a namespace should have a published and a draft version, the setting `FlaggedRevs` needs to be activated for that namespace. This activates the *Acceptance* feature of the quality management tools.

All changes to a page are collected in a draft version until any user with review rights accepts the page. The pages in this namespace are in draft mode until a user with the right to publish content accepts the page.

There is a global option in the ConfigManager to define which user groups are allowed to read unpublished content. By default, all logged-in users can view drafts.

### Characteristics of the acceptance feature

- **Revisions-based:** First draft, draft, accepted
- **External files:** "Freezes" embedded media and templates
- **Automatic acceptance:** Often the result of a review process
- **Permissions:** Read permissions can (cautiously) be limited

## Permissions overview

This permissions table shows which [roles](#) have the necessary right to perform different quality assurance actions. The name of the right that grants a certain permission is shown in parentheses:

QA action	
user (read)	
editor	
reviewer	
admin	
intenance-admin	
bot	
Can create and manage <b>page assignments</b> for any user <i>(pageassignments)</i>	
x	
x	
x	
Can <b>be assigned</b> to a page <i>(pageassigneable)</i>	
x	
Can create and	

manage <b>reminders</b> for any user <i>(remindereeditall)</i>	
	x
	x
Can create and manage their own <b>reminders</b> <i>(read)</i>	
	x
Can set and manage page <b>expiration</b> dates <i>(expirearticle)</i>	
	x
	x
	x
Can create and manage <b>review</b> <b>workflows</b> <i>(workflowedit)</i>	
	x
	x
	x

x
Can be assigned to a <b>review workflow</b> <i>(read)</i>
x
Can view the <b>review workflow</b> on the page <i>(workflowview)</i>
x
x
x
x
x
Can view the complete <b>list of reviews</b> <i>(workflowlist)</i>
x
x
x
x
Can <b>approve</b> a page <i>(review)</i>

x
x
x
Own changes are <b>automatically approved</b> changes <i>(autoreview)</i>
x
x
Rollbacks are <b>automatically approved</b> <i>(autoreviewrestore)</i>
x
x

## Related info

---

- [Working with drafts \(Flagged revisions\)](#)
- [The review workflow](#)

## Manual:Subpage

A [quality version](#) of this page, [approved](#) on *9 September 2021*, was based off this revision.

## Contents

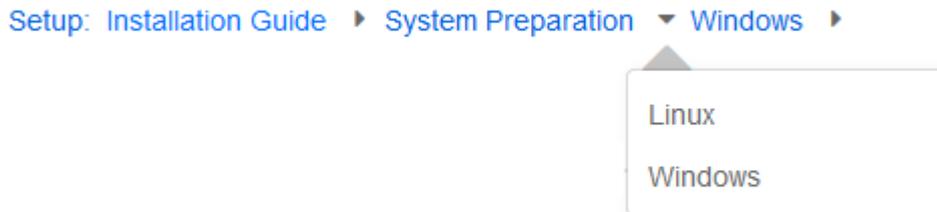
1	What is a subpage? .....	20
2	Characteristics of a subpage .....	20
3	Creating a subpage .....	20
4	Reasons for using subpages .....	21
5	Creating a list of subpages .....	21
6	Related info .....	22

## What is a subpage?

---

To structure wiki content, you can create subpages for a page. This is useful if you have a larger topic and you want to break this topic down into manageable chunks of information. For example, if you have a page about a car, you could have subpages about major car components such as Brakes and Transmission. Subpages can have subpages of their own.

Subpages are automatically displayed in the breadcrumb navigation of the page:



## Characteristics of a subpage

---

- **Syntax-based:** A slash in the page title creates a subpage (e.g., `Installation Guide/System Preparation`).
- **Navigation:** A subpage creates navigation points (breadcrumb).
- **Hierarchy:** Enables "strongly associated" page hierarchies.
- **Language versions:** Management of language versions of a page.

### Unterseiten-Syntax

empty app.diagrams.net chart

## Creating a subpage

---

To create a subpage, it is simply appended to the page name with a slash without spaces. Subpages can in turn have subpages:

```
Installation Guide/System Preparation
Installation Guide/System Preparation/Windows
Installation Guide/System Preparation Windows/IIS
```

**Note:** Since MediaWiki interprets slashes as subpages, you should not use slashes in page names.

## Reasons for using subpages

---

Typical applications for subpages are:

- **To create modular pages:** For a larger topic, you can create a main page and link it to individual detail pages. The advantage is that the detail pages can then be specifically linked from other pages.
- **To maintain archives:** If the discussions on the user discussion page become too long, file them in a subpage.
- **To translate a page:** If the content of a page should be available in other languages. You can create a subpage in German for an English page, for example.
- **To document a template:** If you have a very complex template, you can document its functionality in subpages of the template.

## Creating a list of subpages

---

To list the subpages of the current page, add the following tag to the source text of the page:

```
{{Special:PrefixIndex/{{FULLPAGENAME}}/}}
```

Example:

To display the subpages of the page *VisualEditor* in the namespace *Manual* without showing the namespace or the main page name in the results and without including redirected pages, the syntax is:

```
{{Special:PrefixIndex/Manual:VisualEditor/ |hideredirects=1 |stripprefix=1}}
```

Output:

- [Files](#)
- [Images](#)
- [Tables](#)
- [Tables/Examples](#)
- [Version comparison](#)

The correct display of the subpages is only visible in read-mode after saving the page. The list is displayed automatically in a multi-column layout and cannot be adjusted.

For more flexibility in the output of subpages, you can work with the [extension DPL 3](#) which is included in BlueSpice.

## Related info

---

- <https://www.mediawiki.org/wiki/Help:Subpages>
- [Creating page lists](#)

## Manual:Template

A [quality version](#) of this page, [approved](#) on *23 November 2020*, was based off this revision.

### Contents

1	What is a template? .....	23
2	Characteristics of a template .....	23
3	Creating and including a template .....	23
4	Editing a template .....	24
5	Built-in variables .....	24
6	Further notes .....	24
7	Related info .....	24

## What is a template?

---

A template is a type of reusable content that can be integrated into other pages of the wiki. It can be content-based (text template) or structure-based (layout template) or a combination of both.

A simple example is a notification box that looks identical on all pages, but contains different content on each page where it is used (layout template).

**Tip:** Message goes here

A template can also be used as the source for a [page template](#).

## Characteristics of a template

---

- allows inserting reusable content with a placeholder
- is usually located in the namespace "Template"
- can have parameters
- often contains logic
- used to maintain a "look and feel"

[

## Creating and including a template

---

A new template is created like a regular page, but in the namespace **Template**.

**Example:** [Simple template for a notification box](#)

A template is then included in a page by putting the name of the template (i.e., the title of the template page without the namespace prefix "Template:") in two curly brackets:

```
{{NotificationBox}}
```

**Note:** Template names are case-sensitive.

## Editing a template

---

Changes to a template affect all pages that include this template. Before existing templates are changed, it is recommended that you check which pages use the template. To find these pages, you can go to Page tools > Toolbox > What links here.

## Built-in variables

---

Each MediaWiki has a number of [built-in variables](#) that are not pages in the "template:" namespace, but are still used as templates.

For example, the variable `{{CURRENTTIME}}` returns the the current UTC time.

## Further notes

---

- The use of a template is basically a [transclusion](#). It is also possible to use pages as templates that are not in the *Template* namespace. However, this is to be done wisely, since some functions require that the pages are in the template namespace.
- Templates can also be used to define the output of [semantic query](#). For more information, go to [Semantic MediaWiki](#). (This is an example where the pages must necessarily be in the template namespace.)
- The complete list of all pages of the wiki that are located in this namespace can be found on the page Special:Templates.

## Related info

---

- [meta.wikimedia.org/wiki/Help:Template](https://meta.wikimedia.org/wiki/Help:Template)

## 4.1 Categories

You can assign several categories to a page. Categories can be organized as top categories with subcategories. Therefore, it is possible to create a tree structure and thus to build a hierarchical structure.

### Contents

<a href="#">1 What are categories?</a>	26
<a href="#">2 Characteristics</a>	26
<a href="#">3 Create and assign categories</a>	26
<a href="#">3.1 In the title section of the page</a>	26

3.2 With Visual Editor .....	27
3.3 In Source code .....	27
4 Links to category pages .....	27
5 Displays of categories .....	27
6 Category manager .....	27
7 Related info .....	28

## What are categories?

---

Categories work like keywords or tags. Adding categories to a page makes it easier to find the page in the wiki. All pages of the same category are automatically shown on the corresponding category page. For each category that has not been assigned before, the wiki will automatically create a redlink. Once the page has been created by clicking on the redlink, all pages and subcategories are automatically listed on the category page.

Categories are usually created by users. However, in special cases, MediaWiki also assigns auto-generated categories (for example, pages with broken links).

## Characteristics

---

- **Tagging:** Categories tag pages.
- **Multiple categories per page:** A page can be assigned to several categories.
- **Main categories:** Categories can be organized under main categories.
- **Independent of namespaces:** Categories are not contained in namespaces, but are shared wiki-wide.

## Create and assign categories

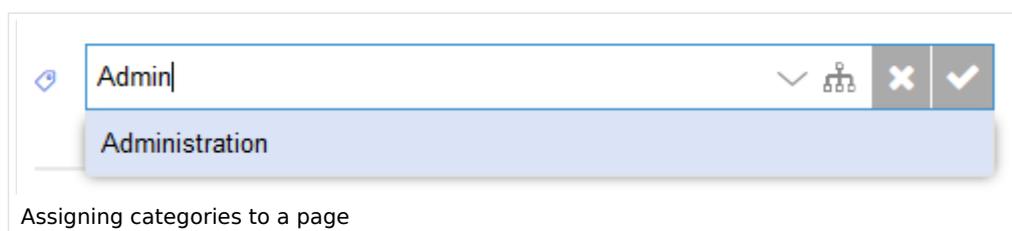
---

To create a category or assign a page to an existing category, there are multiple options:

1. In the title area (view mode)
2. Via the options menu in the visual editor (visual editing mode)
3. With Wikitext (source editing mode)
4. In the "Page Information" fyout of the quality management tools

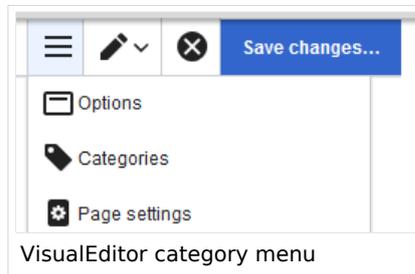
### In the title section of the page

The [title section](#) contains the category assignment field. Any user with edit rights can click on the edit link to add or remove categories from the page. To view all existing categories with subcategories, you can click on the tree icon. Once you are done, the categories are confirmed with the check mark.



## With Visual Editor

In the page options menu of [VisualEditor](#), you can access a Categories menu item. Here you can add and remove category assignments from the page and also add a sortkey to the category.



## In Source code

To add a category in source code, the namespace Category is added before the page name.

```
[[Category:Sales]]
```

**Note:** On category pages, umlauts are generally sorted all the way to the back, e.g., "Märchen" after "Muh". You can therefore use the variable `{{DEFAULTSORT:sortkey}}` to specify an alternative title to be used for sorting (e.g., `{{DEFAULTSORT:Maerchen}}`).

## Links to category pages

---

Links to a category require a colon **before** the namespace.

Example: 

```
[[[:category:new_page]]]
```

If this colon is missing, the article will not be linked to the category page, but the article will be assigned to the corresponding category. `[[category:new_page]]`

## Displays of categories

---

You can view a list of all wiki categories on the page [Special:Categories](#).

On the page *Category:category\_name*, you will find a list of all the pages and media associated with a particular category.

If you have accessed a previously unavailable (redlinked) category page, you can add a category description to the page. That way, users understand the purpose of the category.

## Category manager

---

The easiest way to manage categories and sub-categories is with the Category manager page ([Special:Category\\_manager](#)). You can navigate to the [Category manager](#) via Global actions.

## Related info

---

- [Category Manager](#)
- <https://www.mediawiki.org/wiki/Help:Categories>

## 3.2 UserManual:Namespace

### Contents

1 What is a namespace? .....	29
2 Characteristics of a namespace .....	29
3 Reasons for using namespaces .....	29
4 BlueSpice standard namespaces .....	30
5 Accessing pages in a namespace .....	31
6 Creating a new namespace .....	31
7 Organizing content without namespaces .....	32
8 Viewing all pages of a namespace .....	32
9 Related info .....	32

## What is a namespace?

---

A **namespace** defines an area in a wiki. You can recognize a namespace by the namespace prefix in the page title:

```
<namespace>:Page title
```

In a wiki, you will find predefined namespaces. In addition, wiki administrators can create custom namespaces as needed.

There are certain aspects of namespaces that are important to consider before creating a namespace. It is therefore important to understand the characteristics of namespaces and the reasons for using a custom namespace.

## Characteristics of a namespace

---

- **Rights management** : Each namespace can contain its own user rights/roles assignments. However, we recommend flat user hierarchies for wikis.
- **Search function** : Search can be limited to individual namespaces.
- **Design** : Namespaces can be styled differently.
- **Flat structure** :Namespaces cannot contain other namespaces
- **One namespace per page** : The same page name in a different namespace creates a new page.
- **Special namespaces:**
  - Main (Pages) - No namespace prefix in page titles
  - Special pages (Special)
  - Images (File)
  - Documents (Media)
  - Discussions (Talk)

## Reasons for using namespaces

---

- **Rights management:** To assign different user permissions to separate content. For example, a company's HR department can use its own namespace to grant only Human Resources personnel access to the pages in that namespace. Here, rights management is the main reason for creating namespaces. However, flat hierarchies are recommended for wikis.
- **Search function:** To exclude pages and files from the [search](#).
- **Design:** To format pages in different wiki areas differently.

We often use the example of cupboards to make clear the principle of namespaces. There are shelves and everyone can look at them and work on something. There are glass cupboards with a lock that you can look into, but only some people have the key to change the contents. Then there are closed wooden cabinets with locks and only selected people can look inside and work in them. And each of these cupboards can look differently.

**Note:** While users cannot view the pages or files in a namespace for which they have no access rights, they will still see the pages and files listed on the wiki's special pages. They will also see the page names in the search results if the namespace has not been excluded from the search.

## BlueSpice standard namespaces

BlueSpice uses, among others, the 15 standard and the two so-called pseudo namespaces from MediaWiki:

ID	Namespace	Function
	Media	Pseudo-namespace for uploaded images and files. Will be replaced by path of the files.
	special	Pseudo-namespace for special pages.
0	(pages)	These are given without a prefix.
1	Talk	talk page. Is created in addition to each article (also in the following namespaces) to speak about its contents. Additional tab (+).
2	User	User page. Will be created for every logged in user.
3	User talk	Cf. talk
4	[Wiki-Name]	Information about the wiki or the current project. Here: BlueSpice.
5	[Wiki-Name] _Talk	Cf. talk
6	Image	Description of a picture and other files. Is automatically supplemented with version information and information about the occurrence of the file
7	File_Talk	Cf. talk
8	MediaWiki	system messages. A complete list can be found under Special: AllMessages. Can only be changed by registered users.
9	MediaWiki_Talk	Cf. talk
10	Template	Templates can be integrated into other pages.
11	Template_talk	Cf. talk
12	Help	Help pages describe how to perform actions in the wiki. No clear separation from the [wiki name] namespace.

ID	Namespace	Function
13	Help_talk	Cf. talk
14	Category	Pages can be assigned to categories, which are then accessible in this namespace.
15	Category_talk	Cf. talk

**Standard namespace:** A page is assigned to the "Main" namespace, as long as no other namespace is specified.

**Pseudo namespaces:** The two pseudo namespaces (Media, Special) are used like normal namespaces, but you cannot edit them or add pages to them.

Most pages in the namespaces are created automatically:

- Special pages are generated by the system during installation.
- Media and image pages are created during file upload.
- User pages are created whenever a user acts on the wiki.
- Discussion pages are also created with the corresponding pages.

This leaves only the default namespaces (*Main*), *[wiki name]*, *Template*, *Help* and *Category* namespaces where pages can be freely created.

**Note:** In the default MediaWiki, the names of these areas are "hard-coded". This means that they have been set in the source code of the MediaWiki and cannot be changed by users.

## Accessing pages in a namespace

---

To access or create a page in a specific namespace, its name is simply placed in front of the page title and separated by a colon:

```
[[Setup:Installation|]]
```

In the example, the pipe symbol (straight line) was also added. It has the effect that the namespace won't be displayed for the link label.

If you provide a namespace that is not known to the wiki, the page is created in the standard namespace Main. The portion of the reference specified as the namespace is then added to the title of the item instead and is not recognized as a namespace label.

## Creating a new namespace

---

You can create additional namespaces by using the Namespace manager under Global actions > Management > [Namespace manager](#). The menu item "Category manager" loads the page `Special:NamespaceManager`.

## Organizing content without namespaces

---

If you simply want to group pages in your wiki without the need for controlling access rights, you can use the syntax of the namespaces to **simulate a namespace in the namespace Main**. Simply prefix the pages that belong to a group with a name and a colon. If no namespace with this name exists, the prefix simply becomes part of the page name.

That way, it is possible to use pages with the same name in the main namespace and thus let different groups work on content "sets". Since these own "namespace" prefixes are part of the page name, you can now display all pages in a simulated namespace by searching for the prefix in the search field.

Since you can also organize pages with categories and subpages, you should start by planning how you would like to organize the content in your wiki.

After installing your wiki, you can visualize this with the BlueSpice draw.io Plugin directly in your wiki and discuss it with your team.

## Viewing all pages of a namespace

---

You can view all pages in a namespace:

1. Go to the Special page "All Pages"
2. Leave the input field empty
3. Select a namespace from the Namespace drop-down menu
4. Click "Go"

## Related info

---

- [Namespace CSS manager](#)
- [Namespace manager](#)

## 7 UserManual:Quality management

### Contents

<a href="#">1 The importance of content quality</a>	34
<a href="#">2 Page information</a>	34
<a href="#">3 Page assignments</a>	35
<a href="#">3.1 Characteristics</a>	35

4	Reminder	35
4.1	Characteristics	35
5	Expiry	35
5.1	Characteristics	35
6	Review	36
6.1	Characteristics of the review feature	36
7	Acceptance	36
7.1	Characteristics of the acceptance feature	36
8	Permissions overview	37
9	Related info	40

## The importance of content quality

---

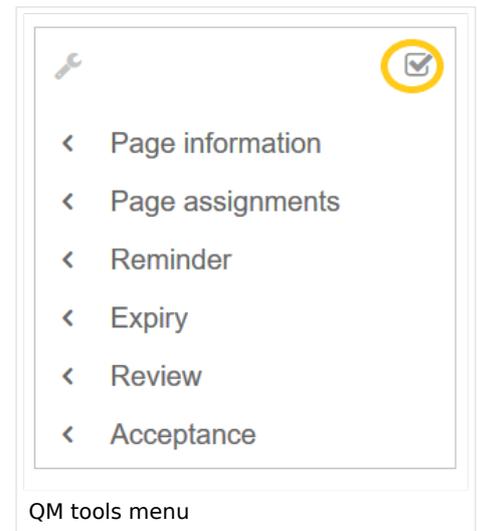
For many companies, the quality of the information they produce and share is of great importance. At the same time, they want to capture all the knowledge of their employees without creating any unnecessary barriers.

In BlueSpice, many users can contribute their knowledge. Ideally, anyone can contribute information right away and specialists can then review and approve the content. This ensures that the quality of the content meets the standards of the company. You should always weigh the benefits and drawbacks of individual quality management features so that you don't create unnecessary roadblocks for users who are eager to contribute.

To access these features, you open the QM tools (with the checkbox icon) in the Page tools panel.

For quality assurance purposes, each content page has the following built-in tools:

- Page information
- Page assignments
- Reminder (BlueSpice pro)
- Expiry (BlueSpice pro)
- Review (BlueSpice pro)
- Acceptance (BlueSpice pro - if FlaggedRevs is activated for the namespace)



## Page information

---

The page information gives you some very interesting statistics about the page. Based on this information, you can decide what quality management measures are necessary to improve the page.

The information that is listed on this page is:

- time of last edit and last editor
- number of views
- templates used on this page
- similar pages
- categories
- readers of this page
- authors of this page

## Page assignments

---

Page assignments are a quick way of giving one or more users informal responsibility and ownership of a page. They are very similar to a [watchlist](#). A user can add a page to their watchlist themselves, but assignments are made by other users.

Page assignments do not change the behavior of the page or grant special rights on a page. For example, the page does not get locked for editing when it is assigned to another user. It is rather a simple mechanism to have users "take a look" at a page. If the feature *Read confirmation* is activated for a namespace, assigned users have to confirm with a checkbox on the page that they have read the page.

### Characteristics

- **Notifications** are sent to assigned users when their page assignment is changed. (This can be deactivated in the user preferences.)
- **Transparency:** Assigned users are listed in the QM tools panel of the page
- **Assignment management:** ([Special:ManagePageAssignments](#))
- **Quick access** from the user profile menu
- **Read confirmation:** page shows a read confirmation checkbox for assigned users (optional)

[More about page assignments](#)

## Reminder

---

A user can create a reminder for a page by selecting a reminder date and by adding an optional comment. Reminders can also be created for other users by administrators and are a quick and informal way of setting due dates for page-related tasks.

### Characteristics

- The user with the reminder receives a notification. (This can be deactivated in the user preferences.)
- Users with reminders are listed on the reminders for the page in the QM tools.
- The page is listed on the Reminders special page under "Global actions" ([Special:Reminder](#)).
- The page is listed on the user's Reminders page, which is easy to access from the user profile menu.

## Expiry

---

If a page contains content with a known lifecycle, the page can be set to expire on a particular date. On the expiration date, a watermark "Outdated" is added to the page if the corresponding option is set in the [ConfigManager](#). In addition, a reminder can be set so that the user will receive a notification when the page expires. The expiration can be deleted from the page at any time.

### Characteristics

- **Watermark** on page background and in PDF (can be configured)
- **Management list:** *Special:Expiry*
- **Reminder** possible
- **Duration:** Expiry can be removed at any time

## Review

---

When a more formal review process is necessary, users can create a review workflow. The review workflow has an end date and is the only quality management tool that can lock the page for editing during the review phase. Each user casts a vote whether the page is acceptable or not.

The page is listed on the special page "Review" under "Global actions" ( `Special:Review` ) and on the reviewers' personal Review page.

### Characteristics of the review feature

- **One active review process** per page
- **Types:** Sequential or parallel review
- **Acceptance or rejection** by participants
- **Page is locked** for editing (Exception: depending on the configuration, reviewers can get edit rights)
- **Automatic restart** possible
- **Revisions-based:** Entries in the version history
- **Acceptance:** can be combined with the Acceptance feature

## Acceptance

---

If the pages in a namespace should have a published and a draft version, the setting `FlaggedRevs` needs to be activated for that namespace. This activates the *Acceptance* feature of the quality management tools.

All changes to a page are collected in a draft version until any user with review rights accepts the page. The pages in this namespace are in draft mode until a user with the right to publish content accepts the page.

There is a global option in the ConfigManager to define which user groups are allowed to read unpublished content. By default, all logged-in users can view drafts.

### Characteristics of the acceptance feature

- **Revisions-based:** First draft, draft, accepted
- **External files:** "Freezes" embedded media and templates
- **Automatic acceptance:** Often the result of a review process
- **Permissions:** Read permissions can (cautiously) be limited

## Permissions overview

This permissions table shows which [roles](#) have the necessary right to perform different quality assurance actions. The name of the right that grants a certain permission is shown in parentheses:

QA action	
user (read)	
editor	
reviewer	
admin	
intenance-admin	
bot	
Can create and manage <b>page assignments</b> for any user <i>(pageassignments)</i>	
x	
x	
x	
Can <b>be assigned</b> to a page <i>(pageassigneable)</i>	
x	
Can create and	

manage <b>reminders</b> for any user <i>(remindereeditall)</i>	
	x
	x
Can create and manage their own <b>reminders</b> <i>(read)</i>	
	x
Can set and manage page <b>expiration</b> dates <i>(expirearticle)</i>	
	x
	x
	x
Can create and manage <b>review</b> <b>workflows</b> <i>(workflowedit)</i>	
	x
	x
	x

x
Can be assigned to a <b>review workflow</b> <i>(read)</i>
x
Can view the <b>review workflow</b> on the page <i>(workflowview)</i>
x
x
x
x
x
Can view the complete <b>list of reviews</b> <i>(workflowlist)</i>
x
x
x
x
Can <b>approve</b> a page <i>(review)</i>

x
x
x
Own changes are <b>automatically approved</b> changes <i>(autoreview)</i>
x
x
Rollbacks are <b>automatically approved</b> <i>(autoreviewrestore)</i>
x
x

## Related info

---

- [Working with drafts \(Flagged revisions\)](#)
- [The review workflow](#)

## 4.2 UserManual:Subpage

## Contents

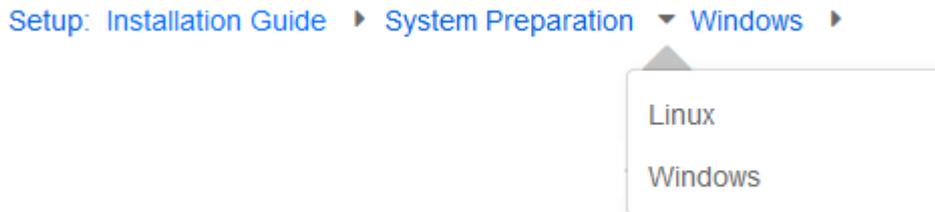
1 What is a subpage? .....	42
2 Characteristics of a subpage .....	42
3 Creating a subpage .....	42
4 Reasons for using subpages .....	43
5 Creating a list of subpages .....	43
6 Related info .....	44

## What is a subpage?

---

To structure wiki content, you can create subpages for a page. This is useful if you have a larger topic and you want to break this topic down into manageable chunks of information. For example, if you have a page about a car, you could have subpages about major car components such as Brakes and Transmission. Subpages can have subpages of their own.

Subpages are automatically displayed in the breadcrumb navigation of the page:



## Characteristics of a subpage

---

- **Syntax-based:** A slash in the page title creates a subpage (e.g., `Installation Guide/System Preparation`).
- **Navigation:** A subpage creates navigation points (breadcrumb).
- **Hierarchy:** Enables "strongly associated" page hierarchies.
- **Language versions:** Management of language versions of a page.

### Unterseiten-Syntax

empty app.diagrams.net chart

## Creating a subpage

---

To create a subpage, it is simply appended to the page name with a slash without spaces. Subpages can in turn have subpages:

```
Installation Guide/System Preparation
Installation Guide/System Preparation/Windows
Installation Guide/System Preparation Windows/IIS
```

**Note:** Since MediaWiki interprets slashes as subpages, you should not use slashes in page names.

## Reasons for using subpages

---

Typical applications for subpages are:

- **To create modular pages:** For a larger topic, you can create a main page and link it to individual detail pages. The advantage is that the detail pages can then be specifically linked from other pages.
- **To maintain archives:** If the discussions on the user discussion page become too long, file them in a subpage.
- **To translate a page:** If the content of a page should be available in other languages. You can create a subpage in German for an English page, for example.
- **To document a template:** If you have a very complex template, you can document its functionality in subpages of the template.

## Creating a list of subpages

---

To list the subpages of the current page, add the following tag to the source text of the page:

```
{{Special:PrefixIndex/{{FULLPAGENAME}}/}}
```

Example:

To display the subpages of the page *VisualEditor* in the namespace Manual without showing the namespace or the main page name in the results and without including redirected pages, the syntax is:

```
{{Special:PrefixIndex/Manual:VisualEditor/ |hideredirects=1 |stripprefix=1}}
```

Output:

- [Files](#)
- [Images](#)
- [Tables](#)
- [Tables/Examples](#)
- [Version comparison](#)

The correct display of the subpages is only visible in read-mode after saving the page. The list is displayed automatically in a multi-column layout and cannot be adjusted.

For more flexibility in the output of subpages, you can work with the [extension DPL 3](#) which is included in BlueSpice.

## Related info

---

- <https://www.mediawiki.org/wiki/Help:Subpages>
- [Creating page lists](#)

## 4.3 UserManual:Template

### Contents

1 What is a template? .....	45
2 Characteristics of a template .....	45
3 Creating and including a template .....	45
4 Editing a template .....	46
5 Built-in variables .....	46
6 Further notes .....	46
7 Related info .....	46

## What is a template?

---

A template is a type of reusable content that can be integrated into other pages of the wiki. It can be content-based (text template) or structure-based (layout template) or a combination of both.

A simple example is a notification box that looks identical on all pages, but contains different content on each page where it is used (layout template).

**Tip:** Message goes here

A template can also be used as the source for a [page template](#).

## Characteristics of a template

---

- allows inserting reusable content with a placeholder
- is usually located in the namespace "Template"
- can have parameters
- often contains logic
- used to maintain a "look and feel"

[

## Creating and including a template

---

A new template is created like a regular page, but in the namespace **Template**.

**Example:** [Simple template for a notification box](#)

A template is then included in a page by putting the name of the template (i.e., the title of the template page without the namespace prefix "Template:") in two curly brackets:

```
{{NotificationBox}}
```

**Note:** Template names are case-sensitive.

## Editing a template

---

Changes to a template affect all pages that include this template. Before existing templates are changed, it is recommended that you check which pages use the template. To find these pages, you can go to Page tools > Toolbox > What links here.

## Built-in variables

---

Each MediaWiki has a number of **built-in variables** that are not pages in the "template:" namespace, but are still used as templates.

For example, the variable `{{CURRENTTIME}}` returns the the current UTC time.

## Further notes

---

- The use of a template is basically a **transclusion**. It is also possible to use pages as templates that are not in the *Template* namespace. However, this is to be done wisely, since some functions require that the pages are in the template namespace.
- Templates can also be used to define the output of **semantic query**. For more information, go to **Semantic MediaWiki**. (This is an example where the pages must necessarily be in the template namespace.)
- The complete list of all pages of the wiki that are located in this namespace can be found on the page Special:Templates.

## Related info

---

- [meta.wikimedia.org/wiki/Help:Template](https://meta.wikimedia.org/wiki/Help:Template)

## 3.3 UserManual:Wikitext

Wikitext is a simplified markup language that is used to format and structure text and other data.

While HTML is the most common markup language for web pages, Wikitext is used in wiki systems. Like HTML, it uses tags to mark parts of text or other data.

### Contents

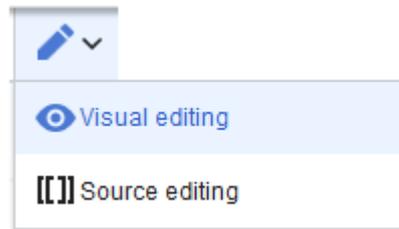
<a href="#">1 How do I use Wikitext?</a>	48
<a href="#">2 Examples of Wikitext</a>	48
<a href="#">2.1 Text formatting</a>	48
<a href="#">2.2 Structuring</a>	48

2.3 Links / Embedded files .....	49
2.4 Magic words .....	49
3 Related info .....	50

## How do I use Wikitext?

---

When you edit a page, you can switch between visual editing and source editing using the editor toolbar.



Wikitext can be entered in source editing mode. When you are in visual editing mode, VisualEditor inserts Wikitext "behind the scenes". If VisualEditor doesn't format elements on a page correctly, you can always switch to source editing to correct the issues directly in Wikitext.

## Examples of Wikitext

---

Here are some typical examples of Wikitext:

### Text formatting

- This is `'''bold'''`
- This is `''italic''`
- This is `<u>underlined</u>`

### Structuring

- This is a `==Heading==`
- This is an `*unordered list element`
- Table:

```
{| class="wikitable"
|-
!header 1
!header 2
|-
|row 1, cell 1
|row 1, cell 2
|-
|row 2, cell 1
|row 2, cell 2
|}
```

## Links / Embedded files

### Link to internal wiki page

```
[[Wikipage|Label]]
[[Pancake|Best Pancake]]
```

### Link to external website

```
[url label]
[https://wikipedia.org Wikipedia]
```

### 'Link with parameters

```
<span class="plainlinks">[{{fullurl:{{FULLPAGENAME}}|action=edit}} Edit this page]</span>
```

This link opens the referenced page in edit mode in a new tab.

## Embed image/file

- `[[Datei:image-or-pdf-name.png]]` - - - **Image/PDF is embedded on page**
  - the wiki checks, if a preview can be generated (e.g., for "png", "jpeg", "gif", "svg") and embeds the preview
  - otherwise, a link to the file description page is created
    - can be embedded: PDF, Tiff (with extension)
    - cannot be embedded: Microsoft Office- und Libre Office-Dateien; Shell-Skripte, ...
- `[[Media:image-or-pdf--name.png]]` - - - - **Links to image / PDF**
- `[[[:Datei:image-or-pdf--name.png]]` - - - **Links to file description page**

## Magic words

Type of Wikitext	Example
Special character	<code>&amp;atilde;</code> → ã
	<code>&amp;ccedil;</code> → ç
	UTF-8 is possible

Type of Wikitext	Example
Comment	<code>&lt;!-- unnoticed --&gt;</code>
No "wiki translation"	<code>&lt;nowiki&gt;</code> and <code>&lt;/pre&gt;</code>
Table of contents	<code>__TOC__</code> and <code>__NOTOC__</code>
Signature	<code>~~~~</code>
Redirect	<code>#REDIRECT [[targetpage]]</code>
Template	<code>{{template name}}</code>

## Related info

- [meta.wikimedia.org/wiki/Help:Wikitext\\_examples](https://meta.wikimedia.org/wiki/Help:Wikitext_examples)
- [Magic words](#)

## Wikitext

Wikitext is a simplified markup language that is used to format and structure text and other data.

While HTML is the most common markup language for web pages, Wikitext is used in wiki systems. Like HTML, it uses tags to mark parts of text or other data.

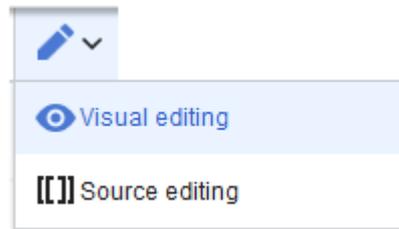
### Contents

<a href="#">1 How do I use Wikitext?</a>	51
<a href="#">2 Examples of Wikitext</a>	51
<a href="#">2.1 Text formatting</a>	51
<a href="#">2.2 Structuring</a>	51
<a href="#">2.3 Links / Embedded files</a>	52
<a href="#">2.4 Magic words</a>	52
<a href="#">3 Related info</a>	53

## How do I use Wikitext?

---

When you edit a page, you can switch between visual editing and source editing using the editor toolbar.



Wikitext can be entered in source editing mode. When you are in visual editing mode, VisualEditor inserts Wikitext "behind the scenes". If VisualEditor doesn't format elements on a page correctly, you can always switch to source editing to correct the issues directly in Wikitext.

## Examples of Wikitext

---

Here are some typical examples of Wikitext:

### Text formatting

- This is `'''bold'''`
- This is `''italic''`
- This is `<u>underlined</u>`

### Structuring

- This is a `==Heading==`
- This is an `*unordered list element`
- Table:

```
{| class="wikitable"
|-
!header 1
!header 2
|-
|row 1, cell 1
|row 1, cell 2
|-
|row 2, cell 1
|row 2, cell 2
|}
```

## Links / Embedded files

### Link to internal wiki page

```
[[Wikipage|Label]]
[[Pancake|Best Pancake]]
```

### Link to external website

```
[url label]
[https://wikipedia.org Wikipedia]
```

### 'Link with parameters

```
<span class="plainlinks">[{{fullurl:{{FULLPAGENAME}}|action=edit}} Edit this page]</span>
```

This link opens the referenced page in edit mode in a new tab.

## Embed image/file

- `[[Datei:image-or-pdf-name.png]]` - - - **Image/PDF is embedded on page**
  - the wiki checks, if a preview can be generated (e.g., for "png", "jpeg", "gif", "svg") and embeds the preview
  - otherwise, a link to the file description page is created
    - can be embedded: PDF, Tiff (with extension)
    - cannot be embedded: Microsoft Office- und Libre Office-Dateien; Shell-Skripte, ...
- `[[Media:image-or-pdf--name.png]]` - - - - **Links to image / PDF**
- `[[[:Datei:image-or-pdf--name.png]]` - - - **Links to file description page**

## Magic words

Type of Wikitext	Example
Special character	<code>&amp;atilde;</code> → ã
	<code>&amp;ccedil;</code> → ç
	UTF-8 is possible

Type of Wikitext	Example
Comment	<code>&lt;!-- unnoticed --&gt;</code>
No "wiki translation"	<code>&lt;nowiki&gt;</code> and <code>&lt;/pre&gt;</code>
Table of contents	<code>__TOC__</code> and <code>__NOTOC__</code>
Signature	<code>~~~~</code>
Redirect	<code>#REDIRECT [[targetpage]]</code>
Template	<code>{{template name}}</code>

## Related info

---

- [meta.wikimedia.org/wiki/Help:Wikitext\\_examples](https://meta.wikimedia.org/wiki/Help:Wikitext_examples)
- [Magic words](#)