

BookStack



- [Attachments](#)
- [Content Overview](#)
- [Drawings & Diagrams](#)
- [Getting Around and Finding Content](#)
- [Markdown Editor](#)
- [Organising Content](#)
- [Page Templates](#)
- [Reusing Page Content](#)
- [Roles and Permissions](#)
- [Searching Content](#)
- [Content Tags](#)
- [Default Page Editor](#)

Attachments



Within BookStack you can attach files & links to pages, so they can be referenced within content and shown available for download. Access to attachments is controlled via view permissions to the page they're uploaded to, so users can only access attachments for pages they can view in the system.

File attachments are really just there for storage and access convenience. BookStack does not do anything special with attachments like scan & index their contents, since documentation content in BookStack is designed & intended to be within the text of pages.

Creating & Uploading Attachments

Attachments can be uploaded via the sidebar when in the page editor. The attachments section of the sidebar is represented with a paper-clip icon and, when open, looks like this:

[ View of the BookStack page editor, with the sidebar open on an "Attachments" view, with the buttons "Upload file" and "Attach Link", along with two listed attachment boxes below labelled as a link with names "kitty-cat.jpg" and "Secret Location"]

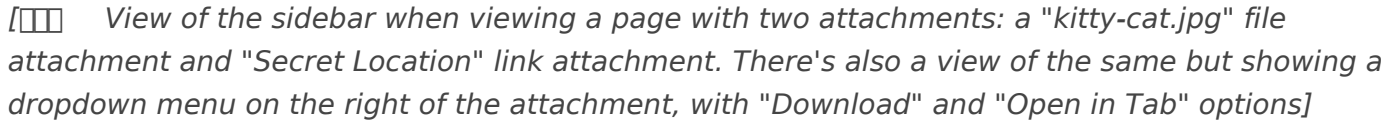
With this open there are options to "Upload File" or "Attach Link". You can also just drag and drop a file into that zone around the buttons to quickly upload a file from your device. Uploading files is the most common use-case but you can attach a link to the page to act as a dynamic, permission-controlled, reference/URL you can use & share.

Below the buttons is where any existing uploaded attachments can be found. You can use the handles on the left to reorder them if needed. There are buttons on each to delete or edit the attachment. Via editing you can alter the attachment name in addition to the underlying file or link, while the original attachment URL will remain the same. There's also a button to insert the attachment into the page, as referenced in the "Using Attachments in Pages" section below.

Keep in mind, attachments are strongly coupled to the pages they're uploaded to. Access to an attachment is controlled via the ability to see the owning page. If a page is deleted, the attachments uploaded to it will be deleted too!

Accessing Attachments

One uploaded, attachments will show in the sidebar when viewing a page. You can click on an attachment to open it up. Link attachments will open in a new tab. File attachments will download, or you can view them in a new tab via the dropdown menu on the right of the attachment:

 *View of the sidebar when viewing a page with two attachments: a "kitty-cat.jpg" file attachment and "Secret Location" link attachment. There's also a view of the same but showing a dropdown menu on the right of the attachment, with "Download" and "Open in Tab" options]*

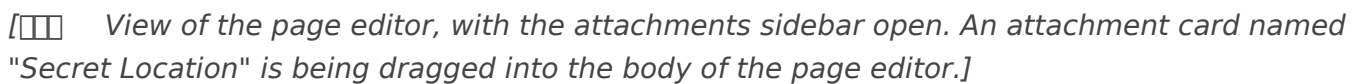
Alternatively you can hold down Ctrl (or Cmd on Mac) and then click on it directly to open in a new tab. Note that attachments in new tabs may still download if the file format is not deemed safe to show via a new tab, or if that's the browser's default behaviour.

Using Attachments in Pages

While attachments are listed in the sidebar when viewing a page, as shown in the section above, you can link to attachments within the page itself which can be a common need when wanting to add context or reference the attachment in a particular part of your documentation content. There are a few ways to do this.

The first method is to simply use the "link" icon button on the attachment in the sidebar. Clicking this will insert a link into the page, at the current cursor position, with the correct attachment URL & name.

The second method is via drag & drop. You can drag the attachment from the sidebar using its handle or name, and then drop it into the editor which will insert an attachment link at that location.

 *View of the page editor, with the attachments sidebar open. An attachment card named "Secret Location" is being dragged into the body of the page editor.]*

The third is just using normal browser and editor functionality. You can right-click and copy the URL for the attachment, then insert a link as normal into the page using that attachment URL you just copied.

Technical Details

When you upload & download attachments, BookStack will attempt to stream files where possible so that the application can serve large files quickly without needing to use up system memory or increase the PHP memory limit. If your instance is using S3-based storage, then attachments are

streamed from BookStack to server but not from S3 (or similar) to BookStack at this time, so it may be better to store and serve attachments direct from storage if possible in those large-file circumstances.

Even with streaming, you can often run into server-side (and proxy) upload/download/timeout limits. Our [admin documentation on file uploads](#) may help if you run into these.

When attachments are served inline (opened in new tab or with an `open=true` query parameter) BookStack will attempt to sniff the mime-type of the attachment content, then serve the file with that mime-type to allow in-browser viewing if supported, and if the detected mime-type is deemed safe (For example, HTML won't be served inline). This is done via an internal allow-list of mime-type values.

Content Overview



The principles of storing information within BookStack is based on the ideas of a normal stack of books. Just like normal books, BookStack books can contain chapters and pages. You start off by creating a book which acts as the highest level of categorisation. Ideally you'd have separate books for separate topics.

Within a book you can directly create pages or you can first create chapters. Chapters provide an additional level of page grouping to keep pages organised but are optional. All the information you write is held within pages. Although books and chapters do not hold information they can be given a short description to assist with searching and visibility.

Once you start to stack-up books you can start to use Bookshelves to organise your Books. Bookshelves can contain multiple books and a single book could be placed on multiple Bookshelves.

Default Colour Coding

Bookshelves, Books, chapters and pages have set colour coding in BookStack to ensure they are easily identifiable. The below examples show the default color scheme. Note that this may be overridden by your administrator.

Books Chapters Pages Draft Pages Bookshelves

Starting Out

When you start out with your new BookStack instance you can organise things in two ways; You can plan out your book/chapter/page structure or you can let things grow naturally over time. If you know or already have the content which will be going into BookStack then it's probably best to plan early otherwise, if you're starting from scratch, you'll want to let everything find its own place.

If you decide to grow naturally then you'll likely start with a book for each major category and add pages directly into those books. As you start getting a lot of pages in each book you'll start to use chapters to group those pages. Once a chapter gets too large you may find it better to split it out into its own book. Within BookStack you can easily move chapters and pages between books and chapters so you shouldn't worry about having to move things around in the future.

Drawings & Diagrams

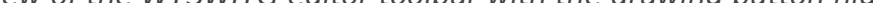
“ ” BookStack

BookStack has built-in support for creating and editing drawings via integrating with [diagrams.net](#). This integration provides very powerful drawing/diagram editing capabilities for a range of use-cases. By default BookStack will use the online version of diagrams.net but this is [configurable by an admin](#).

Inserting, Editing & Managing Drawings

WYSIWYG Editor

When using the WYSIWYG editor, drawings can be created & inserted into the page via a button in the editor toolbar. The button is located in an overflow menu that's within the same section as tables and links:

 View of the WYSWIYG editor toolbar with the drawing button highlighted]

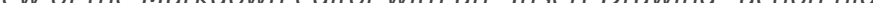
To open an existing drawing for editing, simply double click on the drawing within the editor. Alternatively you can press the toolbar drawing button while a drawing is selected. When you edit an existing drawing and save, the new version is saved as a separate image file by BookStack, instead of overwriting existing drawings. This is to allow proper revision history with old drawings intact.

Existing drawing files can be re-used or deleted via the drawing manager. This can be accessed via the dropdown that sits next to the editor drawing toolbar button:

[ Drawing managed popup view showing a grid of drawing thumbnails]

Markdown Editor

Within the markdown editor you can insert a drawing using the action found above the markdown content:

 View of the Markdown editor with an "Insert Drawing" action highlighted]

To open an existing drawing for editing, just double click the image within the preview window and the existing drawing should be opened for editing. When you edit an existing drawing and save, the new version is saved as a separate image file by BookStack, instead of overwriting existing drawings. This is to allow proper revision history with old drawings intact.

Drawing Image File Storage & Format

Drawings files created via the diagrams.net integration are saved to your own BookStack system, using the same storage mechanism as configured for normal images.

For portability & compatibility, drawings are saved in PNG file format, with the original diagrams.net drawing data embedded within. Since the drawing data is embedded with the PNG image file, you can continue editing such drawings in any diagrams.net instance just by dragging-in or importing the image file.

Getting Around and Finding Content

 BookStack 

There are many ways to get around and find content within your BookStack instance outside of following the core hierarchy. Below is an overview of some common options.

Book Sidebar

The book sidebar is visible when within a book, and it reflects the full ordered hierarchy of the book you are within. If you are within a chapter, that chapter will be expanded by default otherwise they will be collapsed, and expandable with a click.

 *View of the book sidebar showing child chapters and pages*

Next & Previous Links

When viewing a page or chapter, you'll find links below the main content that allow you to navigate between the next & previous content within the current book. These links will reflect sibling content, or the next item within the parent level if no adjacent content exists.

 *View of the next & previous actions when viewing a page*

Breadcrumbs

As you dive into your BookStack content you'll see breadcrumbs above the main content area to reflect the parent path of the current item. These will begin with the "Books" page unless you've traveled through the "Shelves" view in which case this would begin with the "Shelves" view and parent shelf.

 *View of BookStack items, in a hierarchy, with arrows in between*

You can click on breadcrumb items to navigate directly to those. You can also click the arrows between items to search and jump to orthogonal content within the structure hierarchy.

[ Breadcrumb view with an arrow clicked, showing a popover element listing BookStack pages and chapters]

Search

A major way to find and jump to content is by using search. There's the global search input in the header, where you can search across all content in your instance:

[ View of the header search bar with a search started and relevant items listed]

When you start a global search, you'll get live feedback showing a preview of search results. You can press enter or "View All" to view all relevant results on the search page. Within the book and chapter views, you'll also find search inputs allowing you to search just within that book or chapter.

Find more information on our ["Searching Content"](#) page.

Activity Lists

Many views, like that for a book or on the homepage, will show a list of "Recent Activity", reflecting recent actions that have been made relevant to the current context. On these, you can click the name of the item within the listed activity to jump to that item.

[ A list of user activity showing linked items names]

Recently Updated List

On the homepage you can find a "Recently Updated" list, showing the latest pages to have been created or updated. You can press "View All" to view a larger paginated list.

[ A card section with title "Recently Updates Pages" with a list of pages below]

Favourites

One way to easily recall content is by adding it to your favourites. You can choose to favourite any shelf, book, chapter or page to add it to your personal favourite collection. You can then easily find your favourites either as a list on the homepage or via the "My Favourites" link in the header profile dropdown.

[ A view of the "My Favourites" page, with the header profile dropdown menu showing containing a link to "My Favourites"]

Markdown Editor



If you prefer to write in a simpler format, a Markdown editor is available in BookStack. Markdown compatibility generally follows that of [CommonMark](#), along with a few extensions including tables and task-lists. This is an alternative to the default [WYSIWYG Editor](#).

Switching to the Markdown Editor

The default editor option, for when creating a new page, can be configured by an admin within the "Settings > Customization" area of BookStack.

You can change to the markdown editor at a page-level while within the page editor. Simply click the draft status, located above the page name input, and you'll be presented a drop-down with options which include those to change the editor. If you don't see these options, you may not have the relevant "Change page editor" system permission on one of your assigned system roles.

[ *Editor Switch Menu Options*]

For switching to Markdown, from the WYSIWYG editor, you'll see a couple of options:

- **Clean Content** - This is a system-cleaned markdown output, which is much nicer but has potential for formatting loss and potential functionality breaks (Things depending on HTML attributes/IDs for example).
- **Stable Content** - This retains existing HTML content in Markdown to avoid any potential functionality breakages or loss of formatting.

When you choose either option, there is risk of losing certain content details or formatting. BookStack will warn you of this when switching.

Editor Shortcuts

The following shortcuts are available in the Markdown Editor:

Shortcut (Windows & Linux/Mac)	Description
Ctrl+S / Cmd+S	Save Draft

Shortcut (Windows & Linux/Mac)	Description
Ctrl+Enter / Cmd+Enter	Save Page & Continue
Ctrl+1 / Cmd+1 Ctrl+2 / Cmd+2 Ctrl+3 / Cmd+3 Ctrl+4 / Cmd+4	Header Large (h2) Header Medium (h3) Header Small (h4) Header Tiny (h5)
Ctrl+5 / Cmd+5 Ctrl+D / Cmd+D	Normal Paragraph
Ctrl+6 / Cmd+6 Ctrl+Q / Cmd+Q	Blockquote
Ctrl+7 / Cmd+7 Ctrl+E / Cmd+E	Code Block
Ctrl+8 / Cmd+8 Ctrl+Shift+E / Cmd+Shift+E	Inline Code
Ctrl+9 / Cmd+9	Callout <i>(Keep pressing to toggle through styles)</i>
Ctrl+0 / Cmd+0 Ctrl+P / Cmd+P	Ordered List Bullet List
Ctrl+K / Cmd+K	Insert Link
Ctrl+Shift+K / Cmd+Shift+K	Link to BookStack content
Ctrl+Shift+I / Cmd+Shift+I	Insert Image via URL

Organising Content



Once your BookStack instance starts to grow you will find that you may want to re-organise your content. Within BookStack there are two options for moving content around; Either you can move pages and chapters individually or you can sort entire books.

Note that to move content you will need to have 'edit' permission for both the content being moved and the parent it's being moved into.

Moving Single Pages & Chapters

Pages and chapters can be moved directly to a new chapter or book. To move a chapter or page in this way, go to a page or chapter and select 'Move' in the actions list:

Page Move Action

Clicking the 'Move' action will take you to a screen where you can select a new location for your chapter or page. Here you can search for a particular book or chapter using the search bar at the top of the selection screen. Once you select a new parent for your chapter or page press 'Move Page' or 'Move Chapter' and your chapter or page will be moved to the new chapter or book. If you move a chapter all child pages will remain in that chapter and any related activity will now show up under the new parent book.

Sorting Books

The 'Book Sort' interface allows you to move multiple pages and chapters with ease in a simple drag and drop interface. To sort the content of a Book content, open the Book and find the "Sort" option within the actions list.



Initially, just the book you came from will show on the left. You can add extra books into the sort interface by selecting them on the right. Here you can simply drag and drop chapters and pages around and also between different books. Once you have organised your content press 'Save' and all included books will be re-organised.

Books & Shelves

A bookshelf can contain multiple books and a book can be on multiple shelves. To add an existing book to a shelf start by opening the shelf you wish to move the book to and then press "Edit". Under the "Add books to this shelf" section find & select the book that you wish to move into the shelf you are currently editing and then choose "Save Shelf". Within that same interface you can alternatively drag books between the two lists, and you can drag books around in the left-hand list to dictate the default ordering of books within that shelf.

[] *Adding or removing books on a shelf*

Keep in mind that a book can be on multiple shelves, so adding a book to a new shelf won't cause it to be removed from any others. If you want to remove a book from a shelf you'll need to go into each shelf in the same manner as above but select, or drag them out, from the "Books on this shelf" list before saving.

Page Templates

“  BookStack 

Within BookStack you can mark a page as a template so that its content can easily be reused when editing and creating pages. This can be super useful when you need to create many pages following a similar format.

Creating a Page Template

A page template in BookStack is simply a normal page that has been marked as a template. Start by creating/editing the page you want to use as a template. Within the editor sidebar you should see a templates tab. Within this tab you should find a "Page is a template" checkbox. Simply check that option and save the page to make the page become a template. If you don't see that checkbox, you likely lack the "Manage page templates" role system permission.

 *Page edit view showing the templates tab and mentioned checkbox*

Note: Visibility of page templates are controlled via visibility of the page itself. If a user lacks permission to view the page marked as a template, it will not show as an available template for them.

Using Page Templates

Page templates can be accessed when creating or editing a page. Within the editor sidebar you should see a templates tab. If you have templates available they will be listed in this area. A search bar will also show for easy searching of templates.

There are multiple ways to add a template into a page:

- Select the template box itself to replace the editor content with contents of the template.
- Select the up arrow of the template box to prepend its contents to the editor.
- Select the down arrow of the template box to append its contents to the editor.
- Drag the template box into the editor to insert its contents into the dragged location.

 *Dragging a template into the page editor*

Keep in mind that using a template simply copies its content at time-of-use, it is not a "live" connection to the template page, and updating the template page won't affect pages that have previously used it as a template.

Default Page Templates

If you'd like to always start a new page with a specific template, this can be configured at a per-book level using the "Default Page Template" option when editing a book:

 *Dragging a template into the page editor*

With this option set, the selected template will be used to pre-fill the contents of any new page created within that book. Like with other template usages, access to the template will be controlled by permissions. If a user creates a page with the book, but does not have access to view the template page, it will not be used to pre-fill the contents of the new page.

Reusing Page Content

“ BookStack ████████████████████

Within BookStack you may find that you want to include the same block of content in multiple places. You could copy and paste that content into multiple pages but then, if that content needs to be changed, you'd have to update it multiple times across different pages. By using include tags you can include entire other pages and also include single blocks of content from other pages.

Considerations

Here are some things to consider when reusing page content via include tags:

- Permissions are enforced upon includes, so content will only be included from pages that the current user has permission to view. Includes are not obvious nor transparent when viewing a page so it could be easy for a user to externally share content, containing permission-controlled included content, with a non-privileged user without thinking about what permission controls may be in use.
- Includes work with up to three levels of nesting, limited to ensure system performance and to prevent recursion issues.
- Included content won't be indexed as searchable content for pages they're included into.
- Included content won't be shown within content preview text or snippets.

Include Tags

To include the content of a page within another you can use the following syntax:

`{{@page_id}}` or `{{@page_id#content_id}}`

Here are some examples of this in use:

- “ BookStack ████████████████████

Within BookStack you can attach files & links to pages, so they can be referenced within content and shown available for download. Access to attachments is controlled via view permissions to the page they're uploaded to, so users can only access attachments for pages they can view in the system.

File attachments are really just there for storage and access convenience. BookStack does not do anything special with attachments like scan & index their contents, since documentation content in BookStack is designed & intended to be within the text of pages.

Creating & Uploading Attachments

Attachments can be uploaded via the sidebar when in the page editor. The attachments section of the sidebar is represented with a paper-clip icon and, when open, looks like this:

 *View of the BookStack page editor, with the sidebar open on an "Attachments" view, with the buttons "Upload file" and "Attach Link", along with two listed attachment boxes below labelled as a link with names "kitty-cat.jpg" and "Secret Location"]*

With this open there are options to "Upload File" or "Attach Link". You can also just drag and drop a file into that zone around the buttons to quickly upload a file from your device. Uploading files is the most common use-case but you can attach a link to the page to act as a dynamic, permission-controlled, reference/URL you can use & share.

Below the buttons is where any existing uploaded attachments can be found. You can use the handles on the left to reorder them if needed. There are buttons on each to delete or edit the attachment. Via editing you can alter the attachment name in addition to the underlying file or link, while the original attachment URL will remain the same. There's also a button to insert the attachment into the page, as referenced in the "Using Attachments in Pages" section below.

Keep in mind, attachments are strongly coupled to the pages they're uploaded to. Access to an attachment is controlled via the ability to see the owning page. If a page is deleted, the attachments uploaded to it will be deleted too!

Accessing Attachments

One uploaded, attachments will show in the sidebar when viewing a page. You can click on an attachment to open it up. Link attachments will open in a new tab. File attachments will download, or you can view them in a new tab via the dropdown menu on the right of the attachment:

 *View of the sidebar when viewing a page with two attachments: a "kitty-cat.jpg" file attachment and "Secret Location" link attachment. There's also a view of the same but showing a dropdown menu on the right of the attachment, with "Download" and "Open in Tab" options]*

Alternatively you can hold down Ctrl (or Cmd on Mac) and then click on it directly to open in a new tab. Note that attachments in new tabs may still download if the file format is not deemed safe to show via a new tab, or if that's the browser's default behaviour.

Using Attachments in Pages

While attachments are listed in the sidebar when viewing a page, as shown in the section above, you can link to attachments within the page itself which can be a common need when wanting to add context or reference the attachment in a particular part of your documentation content. There are a few ways to do this.

The first method is to simply use the "link" icon button on the attachment in the sidebar. Clicking this will insert a link into the page, at the current cursor position, with the correct attachment URL & name.

The second method is via drag & drop. You can drag the attachment from the sidebar using its handle or name, and then drop it into the editor which will insert an attachment link at that location.

[ View of the page editor, with the attachments sidebar open. An attachment card named "Secret Location" is being dragged into the body of the page editor.]

The third is just using normal browser and editor functionality. You can right-click and copy the URL for the attachment, then insert a link as normal into the page using that attachment URL you just copied.

Technical Details

When you upload & download attachments, BookStack will attempt to stream files where possible so that the application can serve large files quickly without needing to use up system memory or increase the PHP memory limit. If your instance is using S3-based storage, then attachments are streamed from BookStack to server but not from S3 (or similar) to BookStack at this time, so it may be better to store and serve attachments direct from storage if possible in those large-file circumstances.

Even with streaming, you can often run into server-side (and proxy)

upload/download/timeout limits. Our [admin documentation on file uploads](#) may help if you run into these.

When attachments are served inline (opened in new tab or with an `open=true` query parameter) BookStack will attempt to sniff the mime-type of the attachment content, then serve the file with that mime-type to allow in-browser viewing if supported, and if the detected mime-type is deemed safe (For example, HTML won't be served inline). This is done via an internal allow-list of mime-type values.

- Includes all the content from the page with an id of '5'.
- - Includes the content within the element of id `bkmrk-copyright-year` in the page with an id of '10'.

You simply enter that in the page editor and then, when the page is viewed, the content will be dynamically be fetched. This means you can update the referenced content and it will show the same changes on any page the content is included within.

Easily Grabbing the Syntax

Finding the id of a page can be difficult so a quick and easy way to find include tags is built into BookStack. View a page with some content you want to re-use. Use your mouse to select some content and you will see a popup appear. By default this contains a direct link to that content. If you click the link icon on the left it will switch to show you the include tag for that content. Simply copy that and place it in another page. Remove the `#` and content id to include the whole content of the page instead of just a single block.

Forcing Content IDs

When including a specific block of content the id of the block is used (part after the hash). By default these are generated by BookStack when you save a page. If you are using the WYSIWYG editor the id will stay the same unless the block is removed or re-formatted. In the markdown editor it changes with every save based on the content so while it may remain the same there's a good chance an ID can change therefore it's recommended to force an ID if you're using the content for includes.

Note: All ID values must start with "bkmrk" to avoid being overwritten.

WYSIWYG Editor

In the WYSIWYG editor you can force an ID by going into the source code view and adding it manually:

[] *Forcing WYSIWYG Ids*

Markdown Editor

In the markdown editor you can simply insert HTML with an ID to ensure it does not change:

```
# Old Content
Copyright BookStack Enterprises 2020

# Revised with ID
<p id="bkmrk-include-copyright-text">Copyright BookStack Enterprises 2020</p>
```

Locating Include Tag Target

In some cases, you may come across an include tag and want to locate the page that the tag is targeting to include content from. You can do this by making use of BookStack page direct links,

which follow the format `<base_url>/link/<page_id>`. Take the numeric ID (Located after the `@` but before the `#` if existing) from the include tag then use it as the `<page_id>` in that described URL format and you'll be taken to the source page.

As an example if you had the include tag , you'd go to the URL path `/link/5`. A full example URL, using our demo instance, would be: <https://demo.bookstackapp.com/link/5>

Technical Include Logic

BookStack attempts to be smart with includes, to properly handle different content types and to be somewhat compliant with HTML rules. Below are some of the decisions BookStack will make when parsing includes:

- If targeting a specific section to include, BookStack will fetch and includes the inner contents of that section in most cases, apart from when the section needs to be used as a whole in which case the entire block will be used as a whole. Examples of this include tables, lists and code blocks.
- If including whole blocks, whole pages, or ranges of content, BookStack will attempt to structure the content to avoid nested paragraphs which are invalid in HTML and can cause issues when browsers attempt to automatically fix. This will be attempted in a location accurate manner if reasonable, although in some cases content may be moved to before/after the section it's included within.
- IDs of content sections will be de-duplicated after includes are parsed, which may result in existing page content IDs dynamically changing after page content load.

Roles and Permissions

“ ” BookStack

Within BookStack the abilities of a user are controlled by the roles assigned to them and the permissions provided to those roles. A user can be assigned multiple roles, in which case the permissions will stack and the user will receive any ability if any of the roles is provided that specific ability.

Managing Roles

Roles can be created and edited by an admin user by navigating to "Settings > Roles" within the BookStack interface. A set of default roles are provided in a fresh BookStack instance for common use cases. When altering a role you'll be able to change the system permissions which provides certain system-wide abilities. In addition, you'll see a range of "Asset Permissions" which are the default abilities provided to members of that role for content within the system. These may be overridden by content-level permissions. Many of the asset permissions have an "Own" option. This provides the members of that role the ability only when they are the owner of an item in the system.

Assigning Roles

Roles can be assigned to a user, by an admin, via editing a user's profile in the "Settings > Users" area of the BookStack interface. Here multiple roles can be assigned to a user, in which case the permissions will stack and the user will receive any ability if any of the roles is provided that specific ability.

Keep in mind that certain mechanisms within BookStack do have the ability to alter the roles of a user. These include:

- The "Default user role after registration" option in the registration settings.
- LDAP group syncing.
- SAML2 group syncing.

Content Level Permissions

Permissions can be overridden at a Shelf, Book, Chapter or Page level where required. This can be done using the "Permissions" action when viewing such content, by any user that has a role permission to "Manage all/own permissions".

[] *A view of the page permissions screen showing custom permissions being selected]*

Within the permissions view you can use the "Override permissions for role" dropdown to select a role to set permissions for. When saved, these custom permissions will override the default permissions provided by roles for the configured content item.

An "Everyone Else" option is available to allow permission override/control for any roles that are not specifically overridden. By default, "Inherit defaults" will be selected for this item, which means that normal role permissions are used unless overridden. Deselecting this checkbox will use the permissions configured, in the same row, instead.

Content level permissions within BookStack have the following behavior:

- Custom permissions applied to books or chapters will auto-cascade to all child content within unless that content has its custom permissions.
- Custom permissions applied to shelves **will not** auto-cascade, due to the many-to-many relation to books, but they can be copied to all child books in a single action where required.
- The default admin role always retains all permissions for ultimate control, and this role will not be configurable in permission views.

When custom content level permissions are active and affecting the currently viewed item, you'll see an indicator within the details sidebar section. If you have permission to edit the active content level permissions, this acts as a link that will take you to the relevant permissions view, even if applied to a parent chapter or book.

[] *Content level permissions indicator for a page showing "Book Permissions Active"]*

Advanced Permission Logic

With all the different permission options in BookStack, the scenarios and logic can become quite complex as these combine & stack. To help make sense of permission control interplay, below are some general high-level rules that BookStack follows. Within the below, "content permissions" refers to those set directly on shelves, books, chapters, and pages.

There are three main levels of permission, which have different levels of specificity.

From least specific to most specific:

1. Role permissions (Edited via the "Roles" interface).
2. Content "Everyone Else" permissions (Edited via the "Permissions" view for an item).
3. Role-specific content permissions (Also edited via the "Permissions" view for an item).

With those levels in mind:

- Most specific permission application (as above) take priority and can deny less specific permissions.
- Parent role-specific content permissions, that may be inherited ("Everyone Else" Inheriting checkbox active), are considered to essentially be applied on the item they are inherited to unless a lower level has its own permission rule for that specific role.
- Where both grant and deny exist at the same specificity, we side towards the grant.

Searching Content

“ ” BookStack

The ability to search your documentation is vital to day-to-day use. There are a few locations within BookStack where you can search for your content. Below is a list of search functions within BookStack:

- **Header Search Bar** - The search bar/link in the header of every page allows you to search from anywhere. This search is a global search which will look across all books, chapters and pages in your system. After performing a search in this box you'll be led to a search page that includes options and features that can help you build a more advanced search.
- **Book/Chapter Search Bar** - When viewing a book or chapter a search bar can be found in the top of the left sidebar. These searches will look across all child items.
- **Move & Link Selection** - When choosing to move a page/chapter or when selecting a page/chapter/book to link to within the editor the most popular items are shown but you also have the ability to search.

Advanced Search Syntax

All of the above search locations within BookStack share the ability to use advanced search syntax. An easy way to see this syntax in action is to use the global search in BookStack then play with the search filters which will update the search term with the below syntax. Below are details of the different types of syntax that can be used:

Search Type	Syntax	Examples	Description
Normal Searches	<term_a> <term_b>	london meeting	Normal word searches across the name and description or body of your content. When mulitple terms are searched only one term has to match your content but content containing both terms will be higher in the results.
Exact Searches	"<term>"	"london meeting"	Exact matches will require that the whole string within quotes exists in your content in exactly the same format. Use this if you're looking for an exact phrase containing or if you need to search for a term with spaces in.

Tag Searches	[<name>] [<operator><value>] [<name><operator><value>]	[location] [=london] [location=london] [attendees>5]	Tag searches allow you to find pages which have specific tags applied. You can search by tag name, by tag value or by both name and value. When searching by tag value an operator must be used to define the match type. You can use =, !=, <, >, <=, >= or like as operators. When using the like operator you can use % symbols to represent wildcards in your search.
Filter Searches	{<filter_name>} {<filter_name>:<filter_value>}	See below	Filters perform additional advanced functionality to make your searches even more powerfull. Some filters take values but some don't need to. See below for a full list of filters available.

Available Filters

Filters are set advanced search features that can be used in your search term. The below table shows all the filters available in BookStack and how they can be used.

Syntax	Examples	Description
Date Filters		
{updated_after:<date>}	{updated_after:2016-12-30}	Adds the condition that the content must have been last updated after the given date. The date should be in the format YYYY-MM-DD
{updated_before:<date>}	{updated_before:2016-12-30}	Adds the condition that the content must have been last updated before the given date. The date should be in the format YYYY-MM-DD
{created_after:<date>}	{created_after:2016-12-30}	Adds the condition that the content must have been created after the given date. The date should be in the format YYYY-MM-DD
{created_before:<date>}	{created_before:2016-12-30}	Adds the condition that the content must have been created before the given date. The date should be in the format YYYY-MM-DD
User Filters		
{updated_by:<user_slug me>}	{updated_by:barry} {updated_by:me}	Adds the condition that the content must have been last updated by the user of the given slug. If 'me' is used in place of a slug then it will find content that was last updated by the current logged-in user.
{created_by:<user_slug me>}	{created_by:barry} {created_by:me}	Adds the condition that the content must have been created by the user of the given slug. If 'me' is used in place of a slug then it will find content that was created by the current logged-in user.
{owned_by:<user_slug me>}	{owned_by:barry} {owned_by:me}	Adds the condition that the content must have be actively owned by the user of the given slug. If 'me' is used in place of a slug then it will find content that is owned by the current logged-in user.

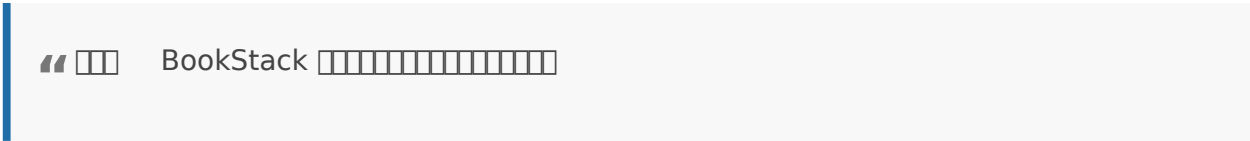
Content Filters		
{in_name:<search>}	{in_name:London Meetings} {in_name:Meetings}	Will require the content to have the given <search> term in the name rather than the name or content body.
{in_body:<search>}	{in_body:London Meetings} {in_body:Meetings}	Will require the content to have the given <search> term in the body rather than both the name or content body.
Option Filters		
{is_restricted}	{is_restricted}	Will require the content to have content-level permissions active. Does not return items with only inherited asset permissions.
{viewed_by_me}	{viewed_by_me}	Will require the content to have been viewed by the current user at least once.
{not_viewed_by_me}	{not_viewed_by_me}	Will not return any content that has been viewed by the current user.
{type:<content_types>}	{type:page chapter book} {type:page chapter} {type:book}	Restricts the types of content that will be in the search results. Use of this will depend on the type of search. For example, in a chapter search only pages are shown so this has no effect.

Search Examples

Below are some examples of using the above syntax and filters with descriptions:

- "my cat" {viewed_by_me} {updated_after:2017-01-24}
 - "my cat" - Search for content containing the exact phrase 'my cat'
 - {viewed_by_me} - that has been viewed by me
 - {updated_after:2017-01-24} - and was last updated after the 24th of Jan 2017.
- textbook discussion [meeting] {type:page} {created_by:me}
 - textbook discussion - Search content for the words textbook or discussion
 - [meeting] - only show content that has a meeting tag applied
 - {type:page} - only show pages, hide chapters and books
 - {created_by:me} - that was created by me.
- {type:book|chapter} {created_by:me} {created_after:2016-08-12} {created_before:2017-02-18}
 - {type:book|chapter} - Search all books and chapters
 - {created_by:me} - that were created by me
 - {created_after:2016-08-12} - after the 12th of Aug 2016
 - {created_before:2017-02-18} - but before the 18th of Feb 2017

Content Tags



While BookStack has a hierarchy at the center of the platform to organise content, you can sometimes desire an extra dimension of categorisation for content. Tags in BookStack offer this in a flexible manner, where a tag can represent a category and value across all levels & parts of the hierarchy.

Tag Format

Tags in BookStack have a name, but can also optionally have a value. This allows detailed & categorised levels of tagging where required for a mixed range of use-cases.

[Screenshot] Three tags applied to content. A "Reviewed At" tag with value "20240611". A "Admin Books" tag with no value. A "Category" tag with value "Team planning"]

For example, you could use a tag named `Critical` with no value to mark important high priority pages. Alternatively, if you have a range of priorities, you could use a tag named `Priority` with a value of `Critical`. Using both a name and value here allows you to identify any content with a Priority set, in addition to allowing identification of just `Critical` content.

This can also come in useful when you want to use tags for metadata. For example, you may want to record content review dates, which you could achieve through a `Reviewed At` tag with value representing the last review date.

Adding & Managing Tags

Tags can be added to any level of the content hierarchy (Shelves, books, chapters & pages). You can apply tags when editing any such item. For shelves, books & chapters, tags are found in a dropdown section on the edit view:

[Screenshot] View of editing tags for a Book, showing two tags, each as small cards with two edit zones within for tag name and value]

For pages, you can find tags in the sidebar toolbox when in the page editor:

[Screenshot] View of the page editor, with a "Page Tags" section open in the sidebar]

When editing a tag name or value BookStack will show a list of potential existing relevant options to help you quickly fill the input. Suggestions for values will be just those used before for the chosen tag name, rather than all values across all tags.

While editing, you're also able to re-order & remove tags as desired. Any changes to tags are saved alongside when you save the item they're applied to.

Tag Display & Overview

When applied, tags are shown in a few areas of BookStack alongside the content they're applied to:

- In the sidebar or info pane when viewing any tagged item.
- On items listed within search results.
- In API responses for items.

When viewing an item with tags applied, you can click the tag to start a tag search. Clicking the tag name will start a search for all content that also has a tag of that name. Clicking the value will start a search for other content that has that same tag name & value applied.

To view all tags being used, you can access an Tag overview page via the "View Tags" action found on the books or shelves index, or you can navigate straight to the `/tags` URL path. Here you'll see a list of all tag names used across content you have visibility of, alongside figures detailing their usage counts across each type of content.

 *View of a tags list, including a search box and sort options. It shows 4 tags as rows, and each row has a breakdown of usage.]*

You can click the tag name, or item counts, to start a relevant filtered search. Where tag values are used, you can also click the link on the right to access a list of all tag values for that tag name, with a similar breakdown as before.

Tags in the Search Index

When content is indexed for searching in BookStack, tags names & values are considered as part of the search engine, and are even given a boost above normal content. This means you can boost content in search results for specific search terms using tags where needed.

Tag Search Operations

Content can be queried in BookStack by tag name or value. This can be used in many areas of BookStack where searching is made available. As mentioned in the sections above, you can often initiate tag searches via clicking tags or statistics in the tag index. Otherwise you can directly use the following syntax:

- Search by tag name:
 - Syntax: [`<name>`]
 - Example: [`Location`]
- Search by tag value:
 - Syntax: [`<operator><value>`]
 - Example: [`=London`]
- Search by tag name & value:
 - Syntax: [`<name><operator><value>`]
 - Example: [`Location=London`]

The following are valid operators: `=`, `!=`, `<`, `>`, `<=`, `>=`, `like`. When using the `like` operator you can use `%` symbols in the search value to represent wildcards.

Using this syntax you can get quite inventive. As an example, if you were applying `Reviewed At` tags with values in a format like `20240627` (YYYYMMDD) you could search for all content with "Reviewed At" dates before 2024 using:

```
[Reviewed At<20240101]
```

See our [documentation on searching](#) for more advanced search syntax.

Tags for Customization

Tags can also be handy for customization in a range of ways, even if just to assist as extra metadata or identification points. For example, you could use the [logical theme system](#) to run a custom action on page creations, and within that change behaviour or action based upon whether the parent Book has a specific tag.

Additionally, BookStack will convert tags to classes within the HTML templates used when viewing content in BookStack, which can then be used for easy tag-controlled customization & styling.

Details of these tag classes can be found in our [Hacking BookStack documentation here](#).

Default Page Editor



The default editor in BookStack is a 'What You See Is What You Get' (WYSIWYG) editor. The usage of this editor is similar to typical word-processing applications such as Word or Google Docs. For those that would prefer it, there is an alternative [Markdown Editor](#) which can be set as default or switched to where permissions allow.

Editor Shortcuts

The following shortcuts are available in the WYSIWYG Editor:

Shortcut (Windows & Linux/Mac)	Description
Ctrl+S / Cmd+S	Save Draft
Ctrl+Enter / Cmd+Enter	Save Page & Continue
Ctrl+B / Cmd+B	Bold
Ctrl+I / Cmd+I	Italic
Ctrl+U / Cmd+U	Underline
Ctrl+1 / Cmd+1 Ctrl+2 / Cmd+2 Ctrl+3 / Cmd+3 Ctrl+4 / Cmd+4	Header Large Header Medium Header Small Header Tiny
Ctrl+5 / Cmd+5 Ctrl+D / Cmd+D	Normal Paragraph
Ctrl+6 / Cmd+6 Ctrl+Q / Cmd+Q	Blockquote
Ctrl+7 / Cmd+7 Ctrl+E / Cmd+E	Code Block
Ctrl+8 / Cmd+8 Ctrl+Shift+E / Cmd+Shift+E	Inline Code
Ctrl+9 / Cmd+9	Callout <i>(Keep pressing to toggle through styles)</i>
Ctrl+0 / Cmd+0 Ctrl+P / Cmd+P	Ordered List Bullet List
Ctrl+K / Cmd+K	Insert Link

Shortcut (Windows & Linux/Mac)	Description
Ctrl+Shift+K / Cmd+Shift+K	Link to BookStack content