



Snippet best practices

Last modified on 07/16/2026 4:19 pm EDT

Snippets can be incredibly powerful. Here are some general best practices for making good use of them.

The most common use cases for snippets are:

- Code or CSS
- In-line text
- Reusable content

Below, we've captured some general guidance and best practices for each use case, as well as some examples to help guide you in your snippet journey. [Contact us](#) if you have questions about anything not detailed below!

Code or CSS

One of the best use cases for snippets is to add HTML, code, CSS, or any combination of those.

Why you might want to do this:

- The snippet editor will support complex HTML that the article editor strips out on-save.
- You control where snippets are added, so it's a great way to add functionality or styling to a very specific, manually-selected subset of your documentation.
- Snippet [references](#) are tracked, so you can see at a glance where this code or CSS has been used, which makes auditing much faster.

For straight HTML, you can use the **Code Editor** for total control, or use the **WYSIWYG editor** to construct the HTML you need.

Best practices for adding code or CSS to snippets:

1. In the **Snippet Content** section, be sure that **Code Editor** is selected, not **WYSIWYG** editor.
2. Don't ever toggle back to the **WYSIWYG** Editor view, as that toggle applies the same extra-vigilant HTML parsing that the article editor does.
3. To add scripts, be sure you include opening and closing `<script>` tags.
4. To add code from other systems for embeds, etc., copy whatever code they give you exactly and paste it in. You don't need to add `<script>` tags if they aren't already in the embed.
5. To add CSS styles, be sure you include opening and closing `<style>` tags.

Here are some examples of those best practices at work:

Script and style

Some snippets contain both scripts that do things and a style section to do CSS, like the snippet created in [Add search to the glossary page](#).

HTML and script

We have a snippet to [create a Subscribe button](#) for our custom content categories. This includes HTML to add the button merge code in a paragraph as well as a script to insert the button where we want it.

Third-party script and style

Some third-party tools will provide specially-formatted HTML to add functionality. The regular article editor will generally strip these out because they aren't part of the HTML spec. Using a snippet for that special script is great, and you can still tuck a style section in for some extra styling of that element or others.

For example, we embed [our Zapier integration](#) in our documentation. Zapier provided us with a pre-built script for this, but we also use some CSS to hide the article title. So we use Zapier's code and also include a style section to add our custom CSS.

In-line text

To use a snippet to add in-line text (such as a text variable you want to use various places), follow these general best practices:

1. In the **Snippet Content** section, be sure that **Code Editor** is selected, not WYSIWYG editor.
2. Add your in-line text directly into that Code Editor, with no paragraph or other tags around it.
3. Don't ever toggle back to the WYSIWYG Editor view, as that toggle will apply some auto-formatting like the article editor, and your text will get wrapped in a paragraph.

Example

For example, we have a set limit on the size of files you can upload to KnowledgeOwl. Since that limit might change at some point in the future, we use a snippet to provide the text, so if we update it, we only have to update the snippet and every article referencing it will update too. The Snippet Content for this is in Code Editor and just entered directly with no HTML:

□

Reusable content

The last major use case for snippets is that they can contain content you might want to reuse. Snippets can have reader group restrictions different from the article where they're embedded, so this can sometimes be a way to show content to some folks but not others.

Best practices for using snippets for reusable content include:

1. Be sure a [topic article](#) wouldn't be a better choice.
2. Make sure it's content you don't expect to turn up in search; snippet content is NOT indexed for search (topic article content is, though).
3. We recommend using the WYSIWYG Editor for the **Snippet Content** to be sure the content looks and is laid out exactly how you want it.

The best use cases for this include things like:

- Callouts or alerts that you want to standardize/reuse and don't need indexed for search
 - Copyright statements or other disclaimers
-