

# PDF look and feel

Last Modified on 12/05/2024 1:57 pm EST

Style your individual article, full PDF, and custom PDF downloads to force page breaks, display content only in PDFs (such as banner images), and display clickable links.

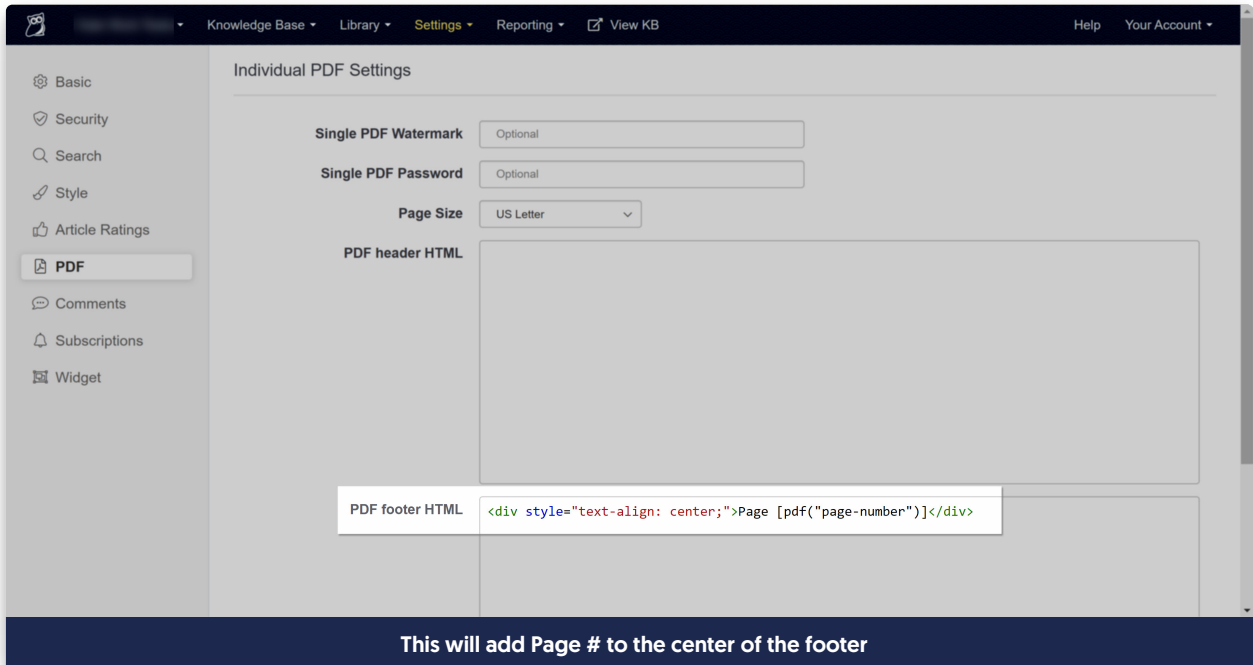
## Add page numbers to PDFs

You can add page numbers to the header or footer of all PDF exports in KnowledgeOwl, though the exact setup varies a bit based on the type of PDF. All methods use this merge code: `[pdf("page-number")]`

### Individual article PDFs

To add page numbers to your individual article PDFs:

1. Go to **Settings > PDF**.
2. Add PDF Header HTML or PDF Footer HTML that includes the `[pdf("page-number")]` merge code. Here, I've added a centered div that will create "Page [number]":



The screenshot shows the 'Individual PDF Settings' page in the KnowledgeOwl admin interface. The left sidebar contains navigation options: Basic, Security, Search, Style, Article Ratings, PDF (selected), Comments, Subscriptions, and Widget. The main content area has the following settings:

- Single PDF Watermark: Optional
- Single PDF Password: Optional
- Page Size: US Letter
- PDF header HTML: (Empty text area)
- PDF footer HTML: `<div style="text-align: center;">Page [pdf("page-number")]</div>`

A dark blue banner at the bottom of the settings area contains the text: "This will add Page # to the center of the footer".

3. Be sure to **Save your changes**.

## Standard & Custom PDF exports

For both the Standard and Custom PDF exports, you don't need to add full HTML. You can just add the page number merge code directly to the relevant field:

1. Go to **Knowledge Base > Exports**.
2. Click on **Standard PDF** to edit the full knowledge base standard PDF, or **Custom PDFs** and then the gear icon next to the custom PDF you'd like to add page numbers to.
3. Add the `[pdf("page-number")]` merge code into the Content footer left, center, or right, depending on where you'd like it. Here, we've added it to the center to get the same treatment as in the individual PDFs above:

The screenshot shows a settings interface for PDF exports. It includes fields for 'Watermark text', 'Password', 'Table of contents title', 'Cover page title', and 'Custom cover page content'. Below these are three footer options: 'Content footer left', 'Content footer center', and 'Content footer right'. The 'Content footer center' field is populated with the merge code `Page [pdf("page-number")]`. A green button labeled 'Save and Generate PDF Export' is located at the bottom of the settings area. A dark blue banner at the very bottom of the image contains the text: 'This will add Page # to the center of the footer'.

4. Be sure to **Save and generate your PDF export**.

## Display content only in PDFs

There are two ways to customize the look and feel of individual article PDFs.

For general structure/layout of individual articles, go to **Settings > PDF** and adjust the settings there--you can add a watermark, password, header HTML, and footer HTML.

But if you want to show or hide content in PDFs, custom CSS is your friend.

The class "hg-pdf" is added to our PDF downloads. You can use this class to style elements differently in your PDF downloads. Add this class before the element you want to target, and you can create customized styles for your PDFs!

Here's how to add your logo to your PDFs:

1. Go to **Settings > Style**.
2. Below the preview pane, select **Custom HTML**.
3. Select **Article** from the Custom HTML dropdown.
4. Near the top of your article, wherever you'd like the logo to appear, add some HTML for your logo. We recommend using a div overall with a class of "pdf-logo", and putting the image inside that div, like this:

```
<div class="pdf-logo" align="center">  
    
</div>
```

5. If you use the above code as a sample, be sure to replace the img src in row 2 with a URL of a valid image file in your File Library!
6. Go to **Custom CSS**.
7. Copy the CSS below and paste it into your Custom CSS. The first CSS here will hide the PDF logo in the live knowledge base; the second will display it only in PDFs.

```
/* Hide PDF logo in live knowledge base */  
.pdf-logo {  
  display:none;  
}  
  
/* Display PDF logo in PDFs only */  
.hg-pdf .pdf-logo {  
  display:block;  
}
```

8. Be sure to **Save your changes**.



**Note:** Once you make changes to PDF styling, you'll only see these changes once PDFs are regenerated. For the Standard PDF and Custom PDF exports, you can generate these yourself. For individual article PDFs, save the article with a small change (such as some added text or a new paragraph break) to regenerate the PDF to see if you like your changes. Once you have the styling the way you want, [contact us](#) to regenerate all your article PDFs.

## Snippet to force page breaks in PDFs

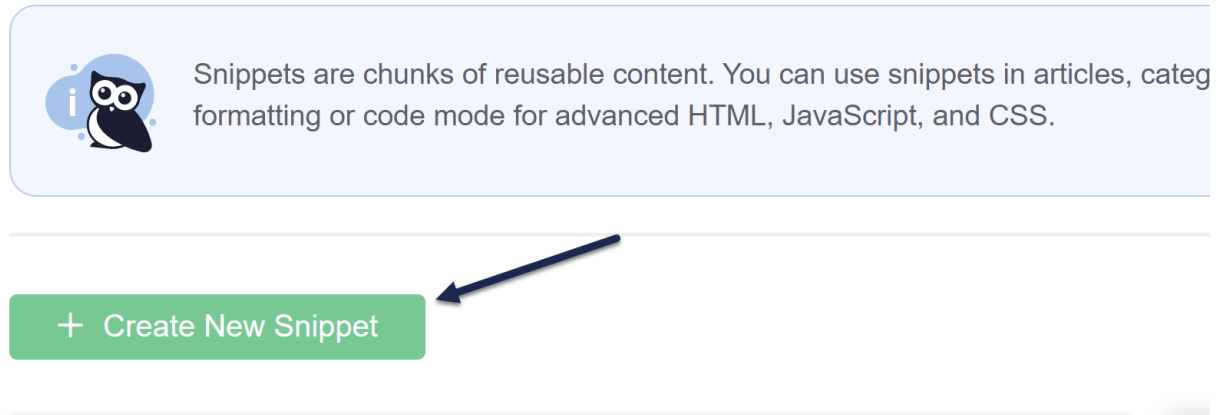
For individual article PDFs as well as the standard and custom PDF export options, sometimes a page break comes between two chunks of content you'd like to stay together (such as a numbered step and the screenshot that goes with that numbered step).

If you'd like to be able to enforce a page break before or after certain portions of an article, you can create a snippet that will force a PDF page break.

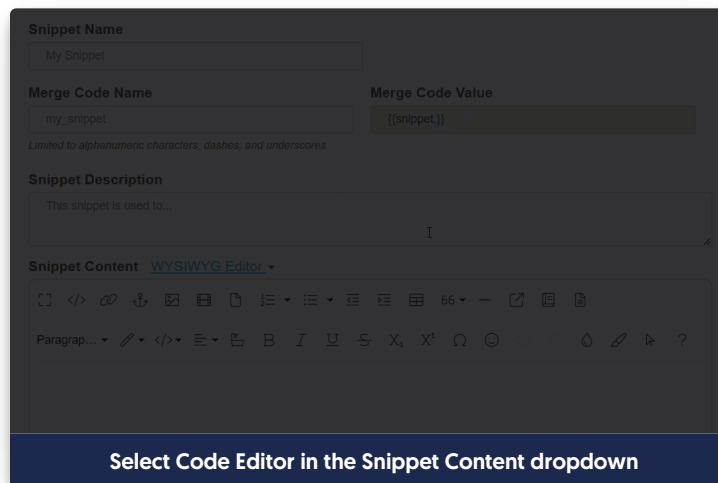
To do so:

1. Go to **Library > Snippets**.
2. Select the **+ Create New Snippet** button.

## Snippets



3. This will open a new page where you can define your snippet.
4. Be sure to give it a **Snippet Name**. Here, we use "PDF Page Break After."
5. The merge code will be automatically generated based on the snippet name.
6. Add a **Snippet Description** to explain what this snippet does. For example: "Add this snippet anywhere in individual articles where you'd like to force a page break in the PDF exports."
7. In the **Snippet Editor**, be sure **Code Editor** is selected from the dropdown.



8. Copy the code below and paste it into the Code Editor. This code creates an empty paragraph with CSS to

force a page break after it:

```
<p style="page-break-after:always;"></p>
```

9. Click the **Create** button. Your completed snippet should look something like this:

The screenshot shows the 'Snippets' editor in KnowledgeOwl. The interface includes a left sidebar with 'Files', 'Snippets', 'Synonyms', and 'Tags'. The main area contains fields for 'Snippet Name' (PDF Page Break After), 'Merge Code Name' (PDFPageBreakAfter), and 'Merge Code Value' ({{snippet.PDFPageBreakAfter}}). A 'Snippet Description' field contains the text: 'Add this snippet anywhere in individual articles where you'd like to force a page break in the PDF exports.' Below this is a 'Snippet Content' field with a 'Code Editor' button and the code: `<p style="page-break-after:always;"></p>`. On the right, there are 'Save Changes' and 'Back' buttons, and a 'Visibility' section with options for 'Hide from PDFs', 'Include snippet content in article blurbs', and 'Restrict to Reader Groups' (with 'None / Show to all readers' selected).

10. Now that your snippet exists, use the **Insert Snippet** option in the editor to add the snippet into the place in any article where you'd like the page break to occur.



## Prevent Page Breaks in PDF Table Rows

If you have table rows that span multiple lines, by default, the PDF generator won't make any effort to keep those multi-line rows on the same page if they fall on a page break. This is most evident when you have table row headers:

interpretaris ut eum. Audire mediocritatem vix no. Sed exerci habemus no, minimum expetenda ut quo.		
	Lorem ipsum dolor sit amet, duo utamur copiosae	

2

Header 1	Header 2	Header 3
Minimum expetenda ut quo.	vituperata ne cum, id quo aliquando adolescens argumentum. Eum nemore sadipscing eu, gloriatur deterruisset ei sea, denique definitiones mei ne. Mel ea	Minimum expetenda ut quo.

Default table row behavior: page breaks can happen in the middle of tall row

You can add some Custom CSS to prevent this mid-row page break and force page breaks to occur between table rows:

1. Go to **Settings > Style**.
2. Copy the code below and paste it anywhere into the **Custom CSS** section below the preview pane.

```
/* Force PDF page breaks between rows, not mid-row */
.hg-pdf table tr {
  page-break-inside: avoid;
}
```

Your code should look like this:

Custom CSS
Custom HTML
Custom <head>

**Custom CSS**

```
/* Force PDF page breaks between rows, not mid-row */
.hg-pdf table tr {
  page-break-inside: avoid;
}
```

### 3. Save.

Once you re-save the article, your PDF should now break properly between the rows:

	uennitiones merne. Miera erat aperiam, falli interpretaris ut eum. Audire mediocritatem vix no. Sed exerci habemus no, minimum expetenda ut quo.	
--	---	--

2

---

Header 1	Header 2	Header 3
Lorem ipsum dolor sit amet, duo utamur copiosae torquatos ad, sale virtute		

The PDF page break now happens between rows, rather than mid-row.



As with all PDF style settings, by default this style change will only be applied to existing articles once they are resaved. [Contact us](#) to have all of your existing article PDFs regenerated with the new style.

To choose where to insert page breaks in an article, check out our [Snippet to Force Page Breaks in PDFs](#).

## Display full URL of links in PDFs

The PDF generator will create clickable hyperlinks for most hyperlinks. (It just doesn't handle anchor hyperlinks)

well.)

But if you're using PDFs as printed resources, you may want to also display the actual URL of the hyperlink. You can display something like:

Here's a regular hyperlink: KnowledgeOwl (<http://www.knowledgeowl.com>)

Sample displayed hyperlink

To create these parentheses with the full URL text automatically, you can add some CSS to your knowledge base and then regenerate PDFs:

1. Go to **Settings > Style**.
2. Below the preview pane, be sure **Custom CSS** is selected.
3. Copy the code below and paste it into the bottom of your Custom CSS.

```
.ko-pdf-clickable-link {display: initial;}
```

4. **Save your changes.**



**Note:** This Custom CSS will not generate clickable hyperlinks for same-page anchors.

Once you're done, your Style Settings should include this line:



Custom CSS

Custom HTML

Custom <head>

### Custom CSS

```
font-family: 'Source Sans Pro';
src: url('/app/image/id/5d3f365b6e121c233a3ff13c/n/sourcesanspro-bolditalic.ttf');
font-weight: bold;
font-style: italic;
}*/

.ko-homepage-top {
  background-image: linear-gradient(-90deg, red, yellow);
}

.ko-pdf-clickable-link {display: initial;} ←
```

-- Select a Merge Code --

Save

Sample Custom CSS for clickable hyperlinks in PDFs

Which will produce a treatment like this (though the exact color of your hyperlinks will vary based on your style settings!):

Here's a regular hyperlink: [KnowledgeOwl \(http://www.knowledgeowl.com\)](http://www.knowledgeowl.com)

Sample in-PDF clickable hyperlink